



Etude algébrique des multitableaux : apport de l'algèbre tensorielle

A. Franc

► To cite this version:

A. Franc. Etude algébrique des multitableaux : apport de l'algèbre tensorielle. Sciences de l'environnement. Doctorat Statistique option : biostatistique, Université de Montpellier II, 1992. Français. NNT : . tel-02574527

HAL Id: tel-02574527

<https://hal.inrae.fr/tel-02574527>

Submitted on 3 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

92/0063

ACADEMIE DE MONTPELLIER

UNIVERSITE DE MONTPELLIER II
--SCIENCES ET TECHNIQUES DU LANGUEDOC--

THESE

Présentée à l'UNIVERSITE MONTPELLIER II - Sciences et Techniques du Languedoc
pour obtenir le DIPLOME DE DOCTORAT

SPECIALITE : STATISTIQUE

Formation Doctorale : Biostatistique

ETUDE ALGEBRIQUE DES MULTITABLEAUX : APPORTS DE L'ALGEBRE TENSORIELLE

par

A l a i n F R A N C

Soutenue le 29 juin 1992 devant le jury composé de :

MM. Francis CAILLIEZ

Bernard CHARLES

Jean-Baptiste DENIS

(Rapporteur)

Yves ESCOUFIER

Pieter KROONENBERG

(Rapporteur)

Jean Dominique LEBRETON

(Rapporteur)

Robert SABATIER

ACADEMIE DE MONTPELLIER

UNIVERSITE DE MONTPELLIER II
--SCIENCES ET TECHNIQUES DU LANGUEDOC--

THESE

Présentée à l'UNIVERSITE MONTPELLIER II - Sciences et Techniques du Languedoc
pour obtenir le DIPLOME DE DOCTORAT

SPECIALITE : STATISTIQUE

Formation Doctorale : Biostatistique

ETUDE ALGEBRIQUE DES MULTITABLEAUX : APPORTS DE L'ALGEBRE TENSORIELLE

par

A l a i n F R A N C

Soutenue le 29 juin 1992 devant le jury composé de :

MM. Francis CAILLIEZ

Bernard CHARLES

Jean-Baptiste DENIS

(Rapporteur)

Yves ESCOUFIER

Pieter KROONENBERG

(Rapporteur)

Jean Dominique LEBRETON

(Rapporteur)

Robert SABATIER

à Anne Marie

INTRODUCTION

PREREQUIS

RAPPEL : ANALYSE D'UNE MATRICE

APPLICATIONS LINEAIRES ASSOCIEES A UN TENSEUR

MODELE DE TUCKER

MODELE PARAFAC / CANDECOMP

ETUDE DU RANG

APPLICATIONS ET OUVERTURES

PROGRAMMES

ANNEXES

RESUME ANGLAIS

BIBLIOGRAPHIE

REMERCIEMENTS

A l'issue de ce travail, il m'est agréable de remercier ceux qui ont contribué à son achèvement.

Le Professeur J.B. KRUSKAL, qui, bien qu'à New York donc éloigné de l'encadrement direct de mon travail, a toujours été un exemple stimulant par la clarté de ses exposés, et par sa correspondance toujours encourageante depuis la rencontre MULTIWAY de Rome en 1986.

Francis CAILLIEZ, forestier qui s'est aventuré sur le terrain de l'analyse des données, dont les remarques pertinentes et parfois incisives sur l'utilisation de l'analyse des données en foresterie et écologie me seront d'un grand secours dans la mise au point des applications.

Bernard CHARLES, qui, par son ouverture aux applications des mathématiques, a apporté son regard d'algébriste.

Jean Baptiste DENIS, par la rigueur et l'exigence toujours amicale de ses critiques, a fortement contribué à la construction de mon travail. Il a été (et sera ?) un guide très éclairant pour les choix raisonnables à réaliser en termes de traitements numériques. Je lui suis très reconnaissant de cette présence et aide.

Yves ESCOUFFIER, dès le début, par ses discussions, encouragements, suggestions, a ouvert la voie d'une vision algébrique plus unitaire dans la grande diversité des travaux réalisés en analyses des multitableaux. Directeur au début de ce travail du laboratoire INRA/ENSAM/USTL de Biométrie de Montpellier, il m'y a accueilli, et fait partagé ses relations avec les équipes montpelliéraines.

Pieter KROONENBERG, par sa longue pratique des traitements des tableaux à trois entrées (et plus ...), m'a plusieurs fois orienté dans l'écheveau des différents modèles existants. C'est un grand honneur pour moi qu'il ait accepté d'être rapporteur, malgré l'épreuve que représente la lecture

approfondie de calculs dans une langue étrangère.

Jean Dominique LEBRETON, par la justesse de ses remarques, a su m'encourager à aller toujours plus loin, et fait miroiter des perspectives très stimulantes de développement de ces travaux algébriques. Je lui suis particulièrement reconnaissant d'avoir pour un temps délaissé l'observation des mouettes pour être rapporteur.

Robert SABATIER, dont l'enthousiasme communicatif et l'agilité pour jongler avec les schémas de dualité et les calculs matriciels (va-t-il développer un talent aussi fécond pour le calcul tensoriel ?) a parfaitement joué le rôle de locomotive.

Jos ten BERGE, pour l'organisation chaleureuse et rigoureuse d'un séminaire sur le rang des multitableaux à Groningue durant l'été 1991, auquel j'ai eu la chance de participer. Sans ce séminaire, le chapitre sur les rang n'aurait pas été écrit.

Que Marie José MONNET trouve ici l'expression de ma gratitude pour son travail inlassable et efficace concernant le rassemblement des articles et ouvrages de la bibliographie.

Enfin, last but not least, je n'oublie pas les nombreux collègues du CEMAGREF et de l'INRA, souvent plus forestiers que mathématiciens certes (mais pas systématiquement), dont les nombreuses discussions, parfois fort tard dans la nuit notamment à Nancy, ont été le terreau dont je me suis nourri pour proposer ces méthodes de traitement des données.

Puissent elles être utiles aux forestiers et écologues !

□

PLAN

INTRODUCTION

A - PREREQUIS

0. NOTATIONS
1. COMPLÉMENTS D'ALGÈBRE MULTILINÉAIRE
2. ANNEXE : VISUALISATION DES TENSEURS

B - RAPPEL : ANALYSE D'UNE MATRICE

1. QUELQUES REPÈRES D'HISTOIRE
2. LE THÉORÈME DE SCHMIDT-MIRSKY
3. PRÉSENTATION DE L'ACP

C - APPLICATIONS LINEAIRES ASSOCIEES A UN TENSEUR

1. ESPACES CONSIDÉRÉS ET NOTATIONS
2. APPLICATIONS LINÉAIRES ASSOCIÉES
3. ACP SUR LES ALAAP AVEC MÉTRIQUE NON TRIVIALE
4. ALAAP AVEC CONTRAINTES
5. ALA AUX TENSEURS D'ORDRE P

D - MODELE DE TUCKER

1. ORIGINE ET DÉVELOPPEMENTS DU MODÈLE DE TUCKER
- 2- MODÈLE DE TUCKER D'ORDRE 3
- 3 - MODÈLE DE TUCKER D'ORDRE 3 AVEC DES MÉTRIQUES PARTICULIÈRES
- 4 - MODÈLE DE TUCKER D'ORDRE 3 AVEC DES CONTRAINTES LINÉAIRES
- 5 - MODÈLE DE TUCKER D'ORDRE P
- 6 - EXEMPLE DU MODÈLE TUCKER-4
- 7 - MODÈLE DE TUCKER D'ORDRE P AVEC MÉTRIQUES

E - MODELE PARAFAC / CANDECOMP

1. ORIGINES ET DÉVELOPPEMENTS DU MODÈLE PARAFAC / CANDECOMP
2. PRÉSENTATION DU MODÈLE D'ORDRE 3
3. MÉTHODES GÉOMÉTRIQUES DE RÉOLUTION
4. RECHERCHE D'UNE SOLUTION APPROCHÉE
5. MODÈLE AVEC MÉTRIQUE SUR LES ESPACES
6. MODÈLE AVEC CONTRAINTES LINÉAIRES
7. MODÈLE D'ORDRE P
8. MODÈLE D'ORDRE P AVEC P GRAND

F - ETUDE DU RANG

1. QUELQUES REPÈRES D'HISTOIRE
2. MAJORANTS DU RANG MAXIMUM D'UN TENSEUR

G - APPLICATIONS ET OUVERTURES

1. ETUDE DE $\mathcal{A}_3(\mathbb{R}^2)$
2. RECHERCHE D'UN MODÈLE POLYNOMIAL MULTIPLICATIF
3. MODÈLES AVEC CONTRAINTES ALGÈBRIQUES

H - PROGRAMMES

ANNEXES

1. CALCULS DE DÉTERMINANTS
2. SYSTÈMES DE N ÉQUATIONS ALGÈBRIQUES À N INCONNUES

RESUME ANGLAIS

BIBLIOGRAPHIE

CONTENTS

INTRODUCTION

A - PRELIMINARIES

0. NOTATIONS

1. COMPLEMENTS OF MULTILINEAR ALGEBRA
2. APPENDIX : VISUALISATION OF TENSORS

B - AS AN EXAMPLE ALREADY KNOWN : MATRIX ANALYSIS

1. HISTORICAL BACKGROUND
2. SCHMIDT-MIRSKY THEOREM
3. PRINCIPAL COMPONENT ANALYSIS

C - LINEAR APPLICATIONS LINKED WITH A TENSOR

1. SPACES AND NOTATIONS
2. LINEAR APPLICATIONS
3. PCA OF LINKED LINEAR APPLICATIONS, WITH NON TRIVIAL METRICS
4. PCA OF LINKED LINEAR APPLICATIONS WITH LINEAR CONSTRAINTS
5. LINKED LINEAR APPLICATION WITH P-WAYS ARRAYS

D - TUCKER'S MODEL

1. ORIGIN AND DEVELOPMENT OF TUCKER'S MODEL
2. TUCKER'S MODEL ON 3-WAY ARRAY
3. TUCKER'S MODEL ON 3-WAY ARRAY WITH NON TRIVIAL METRICS
4. TUCKER'S MODEL ON 3-WAY ARRAY WITH LINEAR CONSTRAINTS
5. TUCKER'S MODEL ON P-WAY ARRAY

6. EXAMPLE OF TUCKER'S MODEL ON 4-WAY ARRAY
7. TUCKER'S MODEL ON P-WAY ARRAY WITH NON TRIVIAL METRICS

E - PARAFAC & CANDECOMP MODELS

1. ORIGIN AND DEVELOPMENT OF PARAFAC / CANDECOMP MODELS
2. MODELS FOR 3-WAY ARRAYS
3. GEOMETRICAL RESOLUTION METHODS
4. SEARCH FOR AN APPROXIMATION
5. MODELS WITH NON TRIVIAL METRICS
6. MODELS WITH LINEAR CONSTRAINTS
7. MODELS ON P-WAY ARRAYS

F - RANK OF A TENSOR

1. HISTORICAL BACKGROUND
2. MAJORANTS DU RANG MAXIMUM D'UN TENSEUR

G - APPLICATIONS AND PROSPECTIVE

1. STUDY OF $\mathcal{A}_3(\mathbb{R}^2)$
2. MULTIPLICATIVE POLYNOMIAL MODEL
3. MODELS WITH ALGEBRAIC CONSTRAINTS

H - PROGRAMS

ANNEXES

1. CALCULATION OF SOME DETERMINANTS
2. SYSTEMS OF N ALGEBRAIC EQUATIONS WITH N UNKNOWN

SUMMARY (IN ENGLISH)

BIBLIOGRAPHY

PLAN DETAILLE

INTRODUCTION	25
A - PREREQUIS	
0. NOTATIONS	31
1. CONVENTIONS	31
2. NOTATIONS UTILISEES	32
3. ESPACES CONSIDERES	34
4. ABREVIATIONS UTILISEES DANS LE TEXTE	35
1. COMPLÉMENTS D'ALGÈBRE MULTILINÉAIRE	37
1. QUELQUES REPERES D'HISTOIRE	38
2. COMMENT ABORDER LE PRODUIT TENSORIEL ?	42
1. Construction axiomatique	
2. Existence du produit tensoriel	
3. Extension aux modules	
4. Présentation retenue par différents auteurs	
3. PRODUIT TENSORIEL DE VECTEURS	47
1. Forme multilinéaire	
2. Produit tensoriel de p vecteurs	
4. ALGÈBRE TENSORIELLE	49
1. Définition	
2. Tenseur	
3. Produit tensoriel de tenseurs	
4. Structure d'algèbre pour l'opération \otimes	
5. Lemme fondamental de l'algèbre multilinéaire	
5. STRUCTURE EUCLIDIENNE CANONIQUE SUR L'ALGÈBRE TENSORIELLE	52
1. Produit scalaire sur l'algèbre tensorielle	
2. Structure hilbertienne de l'algèbre tensorielle	
6. PRODUIT INTERIEUR DANS L'ALGÈBRE TENSORIELLE	55

7. DECOMPOSITION D'UN TENSEUR SUR UNE BASE	57
1. Décomposition sur une base orthonormée	
2. Notion de base duale	
3. Décomposition sur une base quelconque	
8. ACTION DU GROUPE DES PERMUTATIONS SUR UN TENSEUR	59
1. Action sur un tenseur élémentaire décomposé	
2. Action sur un tenseur élémentaire	
3. Action sur un tenseur quelconque	
9. VARIETES DE SEGRE	61
1. Définition	
2. Treillis des sous variétés de Sègre	
3. Applications linéaires conservant les variétés de Sègre	
4. Variétés de Sègre de rang r	
ANNEXE : VISUALISATION DES TENSEURS	64
B - RAPPEL : ANALYSE D'UNE MATRICE	67
1. QUELQUES REPÈRES D'HISTOIRE	67
1. Rappel du cadre conceptuel de l'ACP	
2. La Décomposition en valeurs singulières d'une matrice	
3. Evolutions récentes	
4. L'ACP sous Double Contrainte Linéaire	
5. Analyse en Composantes Principales avec métriques	
2. LE THÉORÈME DE SCHMIDT-MIRSKY	73
1. LA DECOMPOSITION EN VALEURS SINGULIERES	73
2. LE THEOREME DE SCHMIDT-MIRSKY	73
1. Norme invariante par transformation unitaire	
2. Fonction symétrique de jauge	
3. Théorème de Mirsky	
4. Théorème de Schmidt-Mirsky	

3. PRÉSENTATION DE L'ACP	76
1. NOTATIONS ET CONVENTIONS	76
1. Notations et conventions sur les espaces vectoriels	
2. Métrique sur $\mathcal{M}(\mathcal{E}, \mathcal{F})$	
2. DUALITE DANS L'ESPACE $\mathcal{M}(\mathcal{E}, \mathcal{F})$	77
3. ANALYSE EN COMPOSANTES PRINCIPALES	78
1. Définition	
2. Différents cas particuliers	
3. Recherche des r premières composantes principales	
4. ACP SANS CONTRAINTES	79
1. Cas de la métrique unité	
2. Cas d'une métrique T_{MN}	
3. Isométrie définie par une métrique T_{MN}	
5. ACP AVEC CONTRAINTES LINEAIRES	85
1. Notations	
2. Rappels sur les projecteurs	
3. Définition	
4. Lemme	
5. Application	
6. ACP AVEC CONTRAINTES LINEAIRES : ACP-DCL	89
1. Notations	
2. Définition	
3. Lemme	
4. Résultat	
7. VERS LE MODELE DE TUCKER	90
1. Définition	
2. Remarque	
C - APPLICATIONS LINEAIRES ASSOCIEES A UN TENSEUR	93
1. ESPACES CONSIDÉRÉS ET NOTATIONS	93
2. APPLICATIONS LINÉAIRES ASSOCIÉES	94

1. RAPPEL	95
2. APPLICATIONS LINEAIRES ASSOCIEES	95
1. Définition	
2. Ecriture matricielle de U_{Λ}	
3. Ecriture matricielle de V_{Λ}	
4. Matrices associées	
3. APPLICATIONS LINEAIRES ASSOCIEES ET PERMUTATIONS DES INDICES	
1. Construction	97
2. Exemple	
3. Relations entre les ALAAP	
4. ACP SUR LES APPLICATIONS LINEAIRES ASSOCIEES AVEC PERMUTATIONS	
1. ACP sur les ALA	99
2. ACP indépendantes	
3. Expression analytique	
3. ACP SUR LES ALAAP AVEC MÉTRIQUE NON TRIVIALE	103
1. RAPPELS ET NOTATIONS	103
2. ACP DE U_{Λ}	101
3. GENERALISATION	105
4. ALAAP AVEC CONTRAINTES	106
1. Notation et définitions	
2. Lemme	
5. ALA AUX TENSEURS D'ORDRE P	108
1. RAPPELS ET NOTATIONS	108
2. APPLICATION LINEAIRE ASSOCIEE	109
1. Définition	
2. Ecriture matricielle de U_i	
3. APPLICATIONS LINEAIRES ASSOCIEES AVEC PERMUTATION DES INDICES	
1. Construction	110
2. Nombre d'ACP différentes sur les ALAAP	

D - MODELE DE TUCKER	117
1. ORIGINE ET DÉVELOPPEMENTS DU MODÈLE DE TUCKER	117
2- MODÈLE DE TUCKER D'ORDRE 3	122
1. SOUS ESPACES VECTORIELS DE $\varepsilon_1 \otimes \varepsilon_2 \otimes \varepsilon_3$	122
1. Treillis des SEV de $\varepsilon_1 \otimes \varepsilon_2 \otimes \varepsilon_3$	
2. Projections	
2. ACP-MT (ACP SELON UN MODELE DE TUCKER)	123
1. Présentation du problème	
2. Expression du problème	
3. Solution du problème	
4. Lemme	
5. Algorithme de résolution	
3 - MODÈLE DE TUCKER D'ORDRE 3 AVEC DES MÉTRIQUES PARTICULIÈRES	129
1. Rappel des notations et présentation du problème	
2. Expression du problème	
3. Solution du problème	
4. Lemme	
5. Deuxième solution du problème	
4 - MODÈLE DE TUCKER D'ORDRE 3 AVEC DES CONTRAINTES LINÉAIRES	134
1. Définition	
2. Lemme	
5 - MODÈLE DE TUCKER D'ORDRE P	136
1. ACP SELON UN MODELE DE TUCKER (ACPMT)	136
1. Expression du problème	
2. Lemme	
3. Remarque	
4. Solution du problème	

6 - EXEMPLE DU MODÈLE TUCKER-4	140
1. NOTATIONS	140
2. INITIALISATION	141
3. ITERATION	141
4. REMARQUE	144
7 - MODÈLE DE TUCKER D'ORDRE P AVEC MÉTRIQUES	145
1. Rappel des notations et présentation du problème	
2. Expression du problème	
3. Solution du problème	
E - MODELE PARAFAC / CANDECOMP	149
1. ORIGINES ET DÉVELOPPEMENTS DU MODÈLE	149
1. PROBLEME A RESOUDRE	149
2. ORIGINE DU MODELE PARAFAC/CANDECOMP	149
3. DISTINCTION ENTRE LE MODELE CANDECOMP ET LE MODELE DE TUCKER	151
2. PRÉSENTATION DU MODÈLE D'ORDRE 3	155
1. RANG D'UN TENSEUR	155
2. MODELE PARAFAC / CANDECOMP D'ORDRE 3	156
3. MÉTHODES GÉOMÉTRIQUES DE RÉOLUTION	157
1. PRESENTATION DES METHODES	157
2. PROJECTION SUR UNE VARIETE DE SEGRE	157
1. Expression algébrique	
2. Expression analytique	
3. PROJECTION SUR UN CONE	165
1. Remarque géométrique	
2. Expression algébrique	
3. Expression analytique	

4. RECHERCHE D'UNE SOLUTION APPROCHÉE	168
1. JUSTIFICATION D'UNE SOLUTION APPROCHÉE	168
2. REMARQUES SUR L'ALGORITHME PROPOSÉ	169
3. PRINCIPE DE L'ALGORITHME D'APPROXIMATION	170
4. ALGORITHME NUMÉRIQUE	172
1. Séquence proposée	
2. Choix d'un point initial	
5. ACCELERATION DE LA CONVERGENCE	173
1. Présentation des méthodes et choix	
2. Mise en oeuvre de l'algorithme de descente d'ordre n	
3. Résultat	
5. MODÈLE AVEC MÉTRIQUE SUR LES ESPACES	181
1. NOTATIONS ET RAPPELS	181
2. LE MODÈLE	181
3. L'ALGORITHME D'APPROXIMATION	182
1. Premier lemme	
2. Deuxième lemme	
3. Calcul de V	
4. DEUXIÈME CALCUL DE LA SOLUTION DU MODÈLE AVEC MÉTRIQUES	186
1. Définition de l'isométrie	
2. Solution du modèle Parafac/Candecomp	
3. Résultat	
4. Identité entre les deux solutions	
6 - MODÈLE AVEC CONTRAINTES LINÉAIRES	188
1. DÉFINITION	189
2. LEMME	187
3. APPLICATION	190

7 - MODÈLE D'ORDRE P	191
1. RAPPEL DES NOTATIONS	191
2. MODELE PARAFAC / CANDECOMP	191
1. Définition	
2. Algorithme	
3. Eléments pour une programmation	
3. MODELE AVEC METRIQUES	193
1. Rappels sur les métriques	
2. Le modèle	
3. L'algorithme d'approximation	
4. Deuxième calcul de la solution du modèle avec métriques	
4. MODELE AVEC CONTRAINTES LINEAIRES	198
1. Définition	
2. Lemme	
3. Application	
F - RANG D'UN TENSEUR D'ORDRE 3	203
1. QUELQUES REPÈRES D'HISTOIRE	203
1. RAPPEL DES ELEMNTS DU PROBLEME	203
1. Définition	
2. Applications du rang d'un tenseur	
3. Dépendance du rang selon le corps de base et extension aux modules	
4. Solution du problème pour $\mathcal{A}_3(\mathbb{R}^2 \times \mathbb{R}^n \times \mathbb{R}^n)$	
2. DEFINITIONS	210
1. Rang d'un tenseur	
2. Rang maximum d'un tenseur	
3. MINORANT DU RANG MAXIMAL D'UN TENSEUR	211
1. Théorème	
2. Généralisation	
3. Remarques	

4. Résultats publiés par Howell (1978)	
4. MAJORANTS DU RANG MAXIMAL D'UN TENSEUR PROPOSES DANS LA LITTEARTURE	213
1. Position du problème	
2. Résultats généraux	
3. Résultats particuliers	
4. Méthodes utilisées	
2. MAJORANTS DU RANG MAXIMUM D'UN TENSEUR	217
1. TROIS LEMMES IMPORTANTS	217
1. Premier lemme	
2. Deuxième lemme	
3. Troisième lemme	
2. PREMIER MAJORANT DU RMT	221
3. DEUXIEME MAJORANT DU RMT	221
4. TROISIEME MAJORANT DU RMT	225
G - APPLICATIONS ET OUVERTURES	231
1. ETUDE DE $\mathcal{A}_3(\mathbb{R}^2)$	231
1. NOTATIONS	231
2. RANG D'UN TENSEUR	231
1. Premier lemme	
2. Deuxième lemme	
3. APPROXIMATION D'UN TENSEUR	236
1. Premier calcul de la première composante principale	
2. Deuxième calcul de la première composante principale	
3. Application numérique	
2. MODÈLES MULTIPLICATIFS POLYNOMIAUX	249
1. Notations	249
2. Polynômes orthogonaux	250

3. Rang et ACP d'un polynôme à deux inconnues	250
4. Modèle multiplicatif : cas de deux variables	251
5. Modèle multiplicatif : cas de plusieurs variables	252
3. MODÈLES AVEC CONTRAINTES ALGÈBRIQUES	255
1. VARIETES A PARAMETRAGE POLYNOMIAL	256
1. Exemple et généralisation	
2. Définition	
3. Exemple	
4. Ecriture tensorielle d'une VPP	
5. Schéma de Horner pour le calcul d'un polynome à plusieurs indéterminées	
2. ACP D'UNE MATRICE AVEC CONTRAINTES ALGÈBRIQUES	261
1. Définition	
2. Solution sur un exemple	
3. Généralisation	
4. Prise en compte de plusieurs composantes	
3. EXTENSION AUX TABLEAUX A PLUSIEURS ENTREES	268
1. Analyse n-mode	
2. Modèle Parafac / Candecomp	
3. Modèle de TUCKER	
ANNEXE	270
H - PROGRAMMES DE FAISABILITE	273
ANNEXES	313
1. CALCULS DE DÉTERMINANTS	313
1. CALCUL D'UN DETERMINANT D'ORDRE 2N	313
1. Lemme	
2. Exemple	
3. Remarque	
2. CALCUL DU DETERMINANT DE SYLVESTER POUR N = 2	316

1. Résultat	
2. Démonstration	
3. CALCUL DE DETERMINANT DE SYLVESTER POUR $N = 4$	318
1. Notations	
2. Résultat	
3. Démonstration	
2. SYSTÈMES DE N ÉQUATIONS ALGÈBRIQUES À N INCONNUES	329
1. RACINES COMMUNES A DEUX POLYNOMES A UNE INCONNUE	329
1. Lemme de Sylvester	
2. Deuxième lemme	
2. RESOLUTION DES SYSTEMES ALGEBRIQUES DE N EQUATIONS A N INCONNUES	332
3. DETERMINANTS DE POLYNOMES	333
1. Polynômes à une inconnue	
2. Polynômes à plusieurs inconnues	
ENGLISH SUMMARY	339
BIBLIOGRAPHIE	351

INTRODUCTION

□

L'Analyse en Composantes Principales et les techniques connexes comptent parmi les techniques les plus utilisées en analyse des données. Selon la métrique choisie ou construite sur les différents espaces de travail, selon les contraintes et la nature des données, plusieurs dizaines de méthodes sont proposées : Analyse en Composantes Principales normée, Analyse Factorielle des Correspondances, Analyse Canonique, Analyse en Composantes Principales sur Variables Instrumentales, ... (voir par exemple ANDERSON, 1958 ; CAILLIEZ & PAGES, 1976 ; LEBART & Al., 1982).

Par rapport aux calculs, toutes se ramènent à la décomposition en valeurs singulières d'une matrice, retrouvée par ECKART & YOUNG (1936), et le coeur des algorithmes est en général une diagonalisation de matrice carrée réelle symétrique, problème bien étudié par les mathématiciens. Les algorithmes diffèrent, parfois notablement, par la préparation des données, le calcul de la matrice à diagonaliser, et la traduction des résultats.

Les analyses factorielles traitent de matrices, qui sont des tableaux à deux entrées. Or, en écologie par exemple, mais dans bien d'autres domaines d'application également, particulièrement la psychométrie (voir notamment KROONENBERG, 1983) on est souvent confronté à des tableaux à plus de deux entrées. Un exemple simple est fourni par une table de contingence de k facteurs qualitatifs à p modalités, qui est un tableau à k entrées de dimension p . Mais on pourrait en trouver facilement une grande multitude.

□

Deux directions ont, depuis une vingtaine d'années, jalonné l'étude de ces tableaux à plusieurs entrées :

□ ramener leur étude à celle d'un tableau à deux entrées, dont le choix est optimisé : Analyse Factorielle Multiple, par exemple (CAZES, 1980 & 1981)

□ créer des algorithmes de calcul spécifiques pour ces tableaux à plusieurs entrées : modèles PARAFAC (HARSHMANN, 1972), CANDECOMP (CARROLL & CHANG, 1972), de TUCKER (TUCKER, 1966), STATIS (L'Hermier des Plantes, 1976), par exemple.

Or, autant la décomposition en valeurs singulières d'une matrice permet une vision mathématique claire des algorithmes utilisés, autant l'étude des tableaux à trois entrées ou plus présente un paysage de méthodes plus dispersé, sans que des liens évidents puissent être tissées entre elles.

D'autre part, parmi les méthodes proposées dans la littérature, certaines sont spécifiques aux tableaux à trois entrées (STATIS, modèle TUCKAL-3, par exemple), d'autres permettent une extension simple aux tableaux à p entrées (CANDECOMP, Analyse 3-modes, par exemple). Pour certaines, la prise en compte de contraintes linéaires a été proposée (Modèle CANDELINC), et dans peu de cas une extension aux métriques non triviales sur les espaces a été proposée.

□

Dans ce travail, nous proposons comme thèse que l'algèbre tensorielle est un outil qui permet de réunir, comparer et compléter ces différentes méthodes, comme l'algèbre matricielle permet d'aborder en toute sécurité l'analyse factorielle sur matrices. Cette démarche est parfaitement naturelle, puisqu'un tenseur n'est autre qu'une forme multilinéaire, donc un tableau à plusieurs entrées.

Rappelons que l'algèbre tensorielle a déjà été utilisée ponctuellement par différents auteurs, à l'occasion d'un algorithme ou d'une méthode

particulière (BENZECRI, 1973 ; D'AUBIGNY & POLIT, 1989, en France par exemple, ou KROONENBERG, 1983 ; KAPTEYN & Al, 1986 sous forme de produit de Kronecker de deux ou plusieurs matrices). L'analyse spectrale d'un endomorphisme sous forme de somme de produits tensoriels est classique en algèbre (voir par exemple DEHEUVELS, 1981).

Parce que le mot tenseur peut rebuter le lecteur non mathématicien, alors que c'est un être mathématique aussi simple qu'une matrice, nous rappelons en préambule les notions de base d'algèbre tensorielle (Chapitre A).

Afin de familiariser le lecteur avec la manipulation des tenseurs, nous avons réécrit les principaux résultats de l'analyse factorielle d'une matrice avec les calculs découlant de l'algèbre tensorielle : une matrice est un tenseur particulier (Chapitre B).

Nous ne prétendons pas à l'examen exhaustif des méthodes publiées à ce jour, qui sont très nombreuses (se reporter à la bibliographie de KROONENBERG, 1983 pour un aperçu). Notre base de travail est composée des trois modèles les plus utilisés dans les applications :

- l'analyse 3 modes, puis n-modes (Chapitre C)
- le modèle de TUCKER (Chapitre D)
- le modèle PARAFAC et/ou CANDECOMP (Chapitre E)

□

Le résultat principal amené par cette démarche est alors triple :

◦ L'ACP 3-modes, le modèle de TUCKER, le modèle PARAFAC/CANDECOMP sont trois modèles différents, qui ne se ramènent pas l'un à l'autre.

- Chacun de ces modèles est complété sur les trois points suivants :
 - métriques particulières sur les espaces,
 - contraintes linéaires sur les composantes

→ Extension aux tableaux à n entrées.

• Concernant les algorithmes numériques, la prise en compte des métriques et des contraintes peut s'opérer par un prétraitement des données, donc chacune de ces méthodes, dans la diversité de ses mises en oeuvre, se ramène à un algorithme numérique d'optimisation.

□

L'ACP 3-modes (et n -modes), le modèle de TUCKER, le modèle Parafac/Candecomp sont trois méthodes différentes. Chacune peut être employée comme décomposition d'un tenseur, où comme meilleure approximation avec un nombre donné de composantes. Ces trois méthodes sont confondues dans l'ACP, dans le cas particulier des tenseurs d'ordre deux que sont les matrices.

L'ACP reçoit plusieurs généralisations différentes lorsque l'on passe de deux à trois entrées et plus pour les tableaux de données. Il n'y a pas une analyse factorielle des tableaux à trois entrées et plus, mais plusieurs.

En revanche, les méthodes utilisables pour les tableaux à quatre entrées et plus sont de simples extensions naturelles des méthodes qui ont été mises au point pour les tableaux de trois entrées (chapitres C5, D5 et E7). Le saut qualitatif se situe entre deux entrées et trois entrées ou plus.

L'algèbre tensorielle permet alors de compléter simplement ces méthodes pour des tableaux à n entrées, avec des métriques quelconques sur les espaces (mais définies par des matrices carrées symétriques définies positives), avec des contraintes linéaires sur les composantes.

La prise en compte de métriques peut se réaliser en utilisant une isométrie, et se ramener à l'étude d'un tableau image du tableau initial

par cette isométrie, muni de la métrique unité sur chacun des espaces (paragraphes B.4.3, C3, D3, E5).

La prise en compte de contraintes linéaires peut se réaliser en étudiant un tableau image du tableau initial des données par une projection (chapitres B.3.5, C4, D4, E6).

Chaque méthode se ramène à un algorithme donné d'approximation numérique, non linéaire, mais dont chaque étape est une technique éprouvée : régression linéaire pour le modèle PARAFAC, Analyse en Composantes Principales avec contraintes linéaires pour le modèle de TUCKER. Il existe encore de nombreux problèmes ouverts : vitesse de convergence, convergence vers un optimum global, optimisation entre le temps de calcul et l'espace indispensable pour les stockages temporaires de valeurs dans l'exécution de l'algorithme,

En revanche, de même que les analyses factorielles d'une matrice se ramènent à un algorithme d'optimisation et un seul qui est la recherche des valeurs et vecteurs propres d'une matrice réelle carrée symétrique (à construire à partir de la matrice des données initiales, des métriques, des contraintes), chacune des trois méthodes :

- ACP n-modes
- TUCKER
- Parafac / Candecomp

se ramène à un algorithme et un seul, sur un tenseur dont la construction est indiquée dans ce travail à partir du tenseur des données initiales, des métriques utilisées et des contraintes. Les recherches sur l'amélioration des algorithmes numériques peuvent donc se concentrer sur un algorithme et un seul pour chacune des trois méthodes, notamment l'optimisation de la vitesse de convergence.

□

Nous espérons avoir montré que le détour par l'algèbre tensorielle pour

étudier l'analyse factorielle des tableaux à plusieurs entrées, même s'il est également un plaisir esthétique, va largement au delà de la seule rigueur formelle : relier par les chemins de l'algèbre, connus depuis plusieurs décennies, les algorithmes numériques qui fleurissent dans le paysage très diversifié des applications permet d'en extraire ce qui nous semble le noyau dur, et proposer ainsi quelques directions privilégiées où pourraient se concentrer les efforts de recherches pour l'amélioration des ces algorithmes.

CHAPITRE A - 0

NOTATIONS

1 - CONVENTIONS

Les minuscules grecques	: $\alpha, \beta, \gamma, \dots$ représentent des scalaires
Les minuscules latines	: a, b, c, \dots représentent des vecteurs
Les majuscules latines	: A, B, C, \dots représentent des matrices
Les majuscules latines grasses	: $\mathbf{A}, \mathbf{B}, \mathbf{C}, \dots$ représentent des tenseurs
Les majuscules rondes	: $\mathcal{A}, \mathcal{B}, \mathcal{C}, \dots$ représentent des espaces vectoriels
Les majuscules grecques	: $\Gamma, \Omega, \Xi, \dots$ représentent des variétés

Certaines minuscules latines : i, j, \dots , notamment lorsqu'elles sont employées en indice, peuvent désigner des entiers de \mathbb{N} .

2 - NOTATIONS UTILISÉES

\approx	: est isomorphe à
$\langle ., . \rangle$: Produit scalaire
$\langle ., . \rangle_M$: Produit scalaire défini par la MSDP M
$ a_1, \dots, a_n $: Déterminant de la matrice dont les vecteurs colonnes sont a_1, \dots, a_n
$ M $: Déterminant de la matrice M
$\ .\ $: Norme associée au produit scalaire
\otimes	: Produit tensoriel
\cdot	: Produit intérieur
\times	: Produit cartésien
(a_1, \dots, a_n)	: Matrice de vecteurs colonnes a_1, \dots, a_n
$(\alpha_1, \dots, \alpha_n)$: Vecteur de K^n de coordonnées $\alpha_1, \dots, \alpha_n$.
(e_i^*)	: Base duale de la base (e_i)
$[]_{\vec{i}}$: Multi-indice $\vec{i} = \{i_1, \dots, i_p\}$
$\mathcal{A}(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$: Espace vectoriel des formes p-linéaires sur $\mathcal{E}_1 \times \dots \times \mathcal{E}_p$
$\mathcal{A}_p(\mathcal{E})$: Espace vectoriel des formes p-linéaires sur \mathcal{E}^p .
$\mathcal{A}(\mathcal{E})$: Algèbre tensorielle sur \mathcal{E} .
$\mathcal{A}(\mathcal{E}_1, \dots, \mathcal{E}_p)$: Algèbre tensorielle sur la famille $(\mathcal{E}_1, \dots, \mathcal{E}_p)$
$\text{Det}(a_1, \dots, a_n)$: Déterminant de la matrice dont les vecteurs colonnes sont a_1, \dots, a_n
$\text{Det}(M)$: Déterminant de la matrice M
$\text{Diag}(\lambda_1, \dots, \lambda_p)$: Matrice diagonale d'éléments diagonaux $\lambda_1, \dots, \lambda_p$
$\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$: Produit tensoriel des espaces $\mathcal{E}_i \approx \mathcal{A}(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$
$f_r(A)$: Matrice des r premières composantes principales de A
$\Gamma(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$: Variété de Sègre sur le produit $(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$
$\Gamma^r(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$: Variété de Sègre d'ordre r sur le produit $(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$
$\Gamma_p(\mathcal{E})$: Variété de Sègre sur \mathcal{E}^p

$\Gamma_p^r(\mathcal{E})$: Variété de Sègre d'ordre r sur \mathcal{E}^p
${}_1A$: Tenseur $\sigma_1(A)$, où σ_1 est une permutation
$\mathcal{L}(\mathcal{E})$: Espace vectoriel des endomorphismes de \mathcal{E}
$\mathcal{L}(\mathcal{E}, \mathcal{F})$: Espace vectoriel des applications linéaires de \mathcal{E} dans \mathcal{F}
M'	: Matrice transposée de M
$\mathcal{M}(\mathcal{E})$: Espace vectoriel des matrices opérant sur \mathcal{E} .
$\mathcal{M}(\mathcal{E}, \mathcal{F})$: Espace vectoriel des matrices de \mathcal{E} sur \mathcal{F} .
$\mathcal{M}(n, p)$: Espace des matrices à n lignes et p colonnes
$r(m, n, p)$: Rang maximal d'un tenseur de $\mathcal{A}_3(R^m \times R^n \times R^p)$
$r(T, A)$: Rang du tenseur T sur l'anneau A
$r(T, K)$: Rang du tenseur T sur le corps K
$\mathcal{S}_p(\mathcal{E})$: Espace vectoriel des formes p -linéaires symétriques sur \mathcal{E}^p
$\mathcal{S}\mathcal{A}(\mathcal{E})$: Algèbre tensorielle symétrique sur \mathcal{E}
\mathcal{S}_p	: Groupe des permutations de $\{1, \dots, p\} \subset N$
$\mathcal{S}(\mathcal{A}_p(\mathcal{E}))$: Groupe des permutations agissant sur $\mathcal{A}_p(\mathcal{E})$
$\mathcal{S}(\mathcal{A}\mathcal{T}(\mathcal{E}))$: Groupe des permutations agissant sur $\mathcal{A}\mathcal{T}(\mathcal{E})$
$\text{Tr}(M)$: Trace de la matrice M
U'	: Application linéaire transposée de l'application linéaire U

3 - ESPACES CONSIDÉRÉS

1. Espace simple

1 - \mathcal{E} est un espace vectoriel sur \mathbb{R} de dimension finie n .

2 - \mathcal{E} est un espace vectoriel euclidien muni d'un produit scalaire défini par une MSDP M , noté $\langle \dots \rangle$ et d'une base orthonormée (e_i) .

2. Famille d'espaces vectoriels

1 - Soit $(\mathcal{E}_i)_{i=1,p}$ une famille finie d'espaces vectoriels sur \mathbb{R} .

2 - La dimension de \mathcal{E}_i est n_i : $\dim(\mathcal{E}_i) = n_i$

3 - Chaque espace \mathcal{E}_i est muni d'un produit scalaire noté $\langle \dots \rangle$ ou $\langle \dots \rangle_i$ lorsqu'il y a ambiguïté sur l'espace dans lequel on travaille. Ce produit scalaire est défini par une MSDP M_i .

4 - Chaque espace \mathcal{E}_i est muni d'une base orthonormée $(e_{ij})_{j=1,n_i}$ choisie une fois pour toute tout au long de ce texte.

5 - L'espace $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ est muni canoniquement de la structure euclidienne définie par la métrique T :

$$\langle x_1 \otimes \dots \otimes x_p, y_1 \otimes \dots \otimes y_p \rangle_T = \langle x_1, M_1.y_1 \rangle \dots \langle x_p, M_p.y_p \rangle$$

6 - Chaque EV \mathcal{E}_i peut être muni d'un sous espace vectoriel \mathcal{F}_i de dimensions p_i .

4 - ABRÉVIATIONS UTILISÉES DANS LE TEXTE

AC	: Analyse Canonique
ACP	: Analyse en Composantes Principales
ACP-DCL	: ACP sous Double Contrainte Linéaire
ACP-MT	: ACP selon un Modèle de Tucker
ACPMI	: Analyse en Composantes Principales sur Variable Instrumentale
AFC	: Analyse Factorielle des Correspondances
ALA	: Application Linéaire Associée
ALAAP	: Application Linéaire Associée Avec Permutations
BOG	: Base OrthoGonale
BON	: Base OrthoNormée
CP	: Composantes Principales
DVS	: Décomposition en valeurs singulières
EVE	: Espace Vectoriel Euclidien
EVRE	: Espace Vectoriel Réel Euclidien
MPC	: Modèle Parafac / Candecomp
MMFI	: Modèle Multiplicatif à Facteurs Indépendants
MMP	: Modèle Multiplicatif Polynomial
MSDP	: Matrice Symétrique Définie Positive
MT	: Modèle de Tucker
NITO	: Norme Invariante par Transformation Orthogonale
NUI	: Norme Unitairement Invariante
PCP	: Premières Composantes Principales
PGVP	: Plus Grandes Valeurs Propres
SEV	: Sous Espace Vectoriel
SEVO	: Sous Espace Vectoriel Orthogonal
TED	: Tenseur Élémentaire Décomposé
VPP	: Variété à Paramétrage Polynomial

COMPLEMENTS D'ALGEBRE MULTILINEAIRE

L'algèbre linéaire, en y incluant l'étude des formes bilinéaires et la décomposition spectrale d'une matrice, est à la base de l'analyse factorielle des données. Un tableau à plusieurs entrées, comme objet mathématique, peut être considéré comme une forme multilinéaire. La notion de tenseur est à la base de l'étude de ces formes, aussi nous rappelons quelques résultats classiques qui seront utiles dans la suite.

Les éléments d'algèbre linéaire et multilinéaire utilisés sont développés par exemple dans BERTIN & BERTIN (1981) ; CHAMBADAL & OVAERT (1968) ; CHARLES & ALLOUCH (1984) ; DEHEUVELS (1981) ; KUROSH (1967) ; LANG (1984) ; POSTNIKOV (1981).

Les trois premiers paragraphes sont des rappels de définitions essentielles à la suite. Nous avons privilégié une approche pragmatique des tenseurs, qui sont présentés comme outils.

Le quatrième paragraphe rappelle l'extension naturelle de la métrique euclidienne à l'algèbre tensorielle. Cette extension est importante dans les applications : nous chercherons par exemple à approcher au mieux un tenseur (un tableau de données) par un tenseur choisi dans un sous ensemble fixé. On aura alors besoin d'une mesure de la distance entre tenseurs, qui sera fournie par une métrique euclidienne. Les métriques non euclidiennes ne sont pas abordées, bien que le théorème de SCHMIDT-MIRSKY (MIRSKY, 1963) permette d'étendre à une famille très large de métriques les résultats de l'analyse factorielle. Nous n'avons pas exploré cette voie pour les tenseurs.

Les résultats de calcul tensoriel algébrique présentés dans les trois paragraphes suivants seront utilisés sans relâche dans la suite du document.

C'est le produit intérieur, qui généralise l'action d'une matrice sur un vecteur, dont la présentation est inspirée de DIEUDONNE (1970) auquel nous avons emprunté le terme de produit intérieur. En calcul tensoriel, où sont distingués les tenseurs de type (p,q) , p fois covariant et q fois contravariant, cette opération est la contraction (parfois à une permutation près). Les règles algébriques simples liées au produit intérieur permettent de simplifier considérablement les notations, et de rendre ainsi plus accessibles les calculs.

De même, la décomposition d'un tenseur sur une base ou l'action d'un groupe de permutation sur un tenseur (qui est un cadre formel pour prendre en compte la permutation des indices) seront les outils de base de bien des algorithmes proposés dans la suite.

Ces rappels s'achèvent par une présentation des variétés de Sègre, classiques en géométrie projective et calcul différentiel extérieur. Leur utilisation directe n'est pas indispensable, mais le produit de Sègre peut être abordé comme une adaptation du produit tensoriel aux espaces projectifs. Or, une propriété centrale tout au long de ce document, qui est le rang d'un tenseur, est une propriété projective. Cette voie non plus n'a pas été explorée dans ce document.

1. QUELQUES REPÈRES D'HISTOIRE

L'algèbre linéaire et multilinéaire est un champ mathématique où utilisations, applications et dégagement des structures se sont développés parallèlement durant deux siècles, du début du XVIII^{ème} siècle au début du XX^{ème}. Et les premiers pas d'algèbre linéaire sont de l'algèbre multilinéaire, avec dès 1750 les travaux de MACLAURIN, CRAMER, VANDERMONDE, LAPLACE, notamment, sur l'étude des déterminants pour la résolution des systèmes d'équations linéaires. Ces travaux ont été formalisés par CAUCHY, qui proposa le nom de déterminant en 1812, et par FROBENIUS (1842).

A cette même époque, toutes les notions clés en algèbre linéaire émergent, mais sans qu'elles soient rassemblées en un tout cohérent : dépendance linéaire, transformations linéaires, valeurs et vecteurs propres, forme bilinéaire, dualité, orthogonalité

La notion de matrice a été introduite par SYLVESTER, et le calcul matriciel, développé également par CAYLEY, est un outil courant pour EISENSTEIN et HERMITE dès 1850. Durant la seconde moitié du XIX^{ème} siècle, SYLVESTER, JACOBI, WEIERSTRASS, KRONECKER, JORDAN, SMITH, notamment, étudient les formes réduites et la décomposition spectrale des matrices. L'ensemble de ces travaux sera codifié par FROBENIUS qui batira une théorie cohérente des matrices entre 1877 et 1880, avant que n'émerge la notion d'espace vectoriel.

Si l'addition des vecteurs du plan avait été introduite par GAUSS dans la représentation géométrique des nombres complexes, le calcul vectoriel a été développé dans la première moitié du XIX^{ème} siècle par GRASSMAN, MÖBIUS (élève de GAUSS à Göttingen) et HAMILTON. Le concept de vecteur est dû à CAYLEY et GRASSMANN, en 1843-1845. GRASSMANN s'est de plus très approché de la notion plus abstraite d'espace vectoriel, mais dans des oeuvres considérées comme obscures car méconnues à son époque, où il a notamment défini une opération qui se rattache au produit tensoriel.

En particulier pour répondre à une demande croissante issue de la physique mathématique, le calcul vectoriel dans les espaces de faible dimension, à la fin du XIX^{ème} siècle, évolue vers un foisonnement d'objets tels que vecteurs polaires, vecteur axial, affineurs, rotateurs, dyades, etc ... , sans qu'aucun formalisme ne vienne clarifier ces notions.

Le concept d'espace vectoriel sur le corps des réels a été introduit par PEANO en 1888, mais le concept actuel plus général d'espace vectoriel sur un corps quelconque l'a été par TOEPLITZ, élève de HILBERT, en 1909, à l'occasion de travaux sur l'application des espaces vectoriels à l'analyse ("Über die Auflösung unendlichvieler linearen Gleichungen mit unendlichvielen Unbekannten, *Rend. Circ. Mat. Palermo*, XXVIII, 88-96). De

façon générale, les besoins pressants d'analyse de la fin du XIX^{ème} siècle pour la résolution d'équations aux dérivées partielles où apparaissent des opérateurs linéaires (laplacien, etc ...) a été un formidable levier pour stimuler les travaux d'algèbre linéaire : applications et dégagement des structures se sont mutuellement entraînés.

L'étude approfondie des formes bilinéaires dès la première moitié du XIX^{ème} siècle conduira naturellement à l'étude des formes trilinéaires, puis multilinéaires, étudiées intensivement par CAYLEY, HERMITE et SYLVESTER dans leur théorie des invariants, et poursuivie par de nombreux algébristes de la seconde moitié du XIX^{ème} siècle. CAYLEY appelait "quantics" les formes multilinéaires. L'idée est d'étudier les fonctions algébriques de degré n à p inconnues, et leurs transformations lors de transformations linéaires des inconnues. Cette théorie purement algébrique s'est ensuite développée sous l'impulsion notamment de ARNOHOLD, CLEBSCH, GORDAN et HESSE en Allemagne, avec un formalisme très épuré et des règles de composition propres.

STRUIK (1984) reconnaît dans la concision des notations concernant les invariants proposées par ARNOHOLD et CLEBSCH l'embryon de la notion de tenseur, et, par la même, de l'algèbre linéaire et multilinéaire en tant que théorie construite.

De cette théorie datent également les termes de covariant et contravariant (selon la transformation de la forme), proposés par SYLVESTER, et repris en calcul tensoriel. Un état de cette théorie est fourni notamment dans GRACE & YOUNG, (1903).

En suivant une autre voie parallèle, l'étude des formes multilinéaires s'est également développée dans la deuxième moitié du XIX^{ème} siècle à l'occasion de travaux de géométrie différentielle, associés à des surfaces munis de métriques riemanniennes, où il faut citer notamment les noms de CHRISTOFFEL et LIPSCHITZ et BELTRAMI.

Mais c'est surtout l'école italienne de la fin du XIX^{ème} siècle qui formalisera ses résultats en jetant les bases du calcul tensoriel, sous le

nom de calcul différentiel absolu, à partir de 1880 à l'occasion de travaux sur la géométrie différentielle des surfaces, grâce aux travaux de BIANCHI, BELTRAMI et RICCI.

Selon LEVI-CIVITA(1926), le calcul tensoriel a été essentiellement conçu par Gregorio RICCI-CUBASTRO et ses élèves (dont LEVI-CIVITA), entre les années 1887 et 1896. Une première publication a été réalisée par RICCI dans le Bulletin des Sciences Mathématiques en 1892.

Le calcul tensoriel a été présenté de façon systématique pour la première fois par RICCI et LEVI-CIVITA en 1901 dans un mémoire fameux : "Méthodes de calcul différentiel absolu et leurs applications. *Math. Ann.* LIV 125-201", publié à l'invitation de F. KLEIN (LEVI-CIVITA, 1926), et qui déjà établissait le lien avec des problèmes de mécanique rationnelle et élasticité : les applications à la physique mathématique étaient au coeur des préoccupations des initiateurs.

Le nom de tenseur a été proposé indépendamment par le VOIGT, cristallographe allemand de Göttingen, en 1908, et trouve son origine en élasticité : un tenseur était en fait une matrice carrée symétrique dont on recherchait certains invariants (notamment les axes propres) utiles à la théorie de l'élasticité dans les cristaux, et servait d'abord à modéliser l'état des contraintes, ou tensions, dans un cristal (BRILLOUIN, 1936). Voigt a notamment présenté sa théorie en 1910 : *Lehrbuch der Kristall Physik*, Teubner, Berlin.

Le nom de calcul tensoriel a été introduit par Einstein en 1913, et son succès rapide est dû en grande partie à son utilisation dans la théorie de la relativité, par des auteurs aussi illustres que WEYL, LAUE et EDDINGTON, dès le début des années 20. Le premier manuel de calcul tensoriel semble être celui de JUVET : *Introduction au calcul tensoriel et au calcul différentiel absolu*, Blanchard, Paris, 1922, mais l'ouvrage qui formalisa la théorie est celui de SCHOUTEN : *Ricci-Calcul*, en 1924 (cité par STRUIK, 1984).

Bien que les notations et présentations aient évolué dans la suite

selon la maturation des techniques mathématiques employées, toutes les opérations sur les tenseurs présentées dans ce chapitre étaient connues des initiateurs, et peuvent se trouver dans EDDINGTON (1921), LEVI-CIVITA (1926), notamment l'addition, la multiplication (produit tensoriel), la contraction et la composition, qui correspond au produit intérieur. L'action du groupe de permutations sur un tenseur est notamment développé dans WEYL (1946).

Une évolution ultérieure de l'algèbre tensorielle, notamment l'étude des formes multilinéaires sur les anneaux (et non plus les corps), s'est développée à partir des années 1930, complique singulièrement la théorie du point de vue technique mathématique. Mais nous n'y ferons pas appel dans ce qui suit.

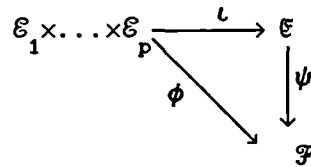
2. COMMENT ABORDER LE PRODUIT TENSORIEL ?

Les ouvrages déjà cités abordent le produit tensoriel et les tenseurs selon plusieurs perspectives, qui sont équivalentes, mais ne présentent pas les mêmes degrés d'abstraction.

1. Construction axiomatique

Comme l'algèbre linéaire est un corpus de connaissances bien établi, où des calculs souvent simples s'enchaînent de façon fluide et naturelle, l'idée générale est de linéariser les applications multilinéaires par la construction d'un espace \mathcal{E} à partir d'une famille (\mathcal{E}_i) d'espaces vectoriels sur un même corps K tel que toute application multilinéaire de (\mathcal{E}_i) dans un K -espace vectoriel \mathcal{F} soit équivalente à une application linéaire de \mathcal{E} dans \mathcal{F} . Sous l'axiomatisation parfois abstraite perce ainsi un réel souci d'efficacité au niveau des calculs, qui est la véritable difficulté pour aborder l'algèbre tensorielle.

Plus précisément, l'objectif est de construire un espace vectoriel \mathcal{E} et une injection canonique $\iota : (\mathcal{E}_1) \rightarrow \mathcal{E}$ tels que pour tout espace vectoriel \mathcal{F} et toute application multilinéaire $\phi : (\mathcal{E}_1) \rightarrow \mathcal{F}$, il existe une application linéaire ψ et une seule telle que le diagramme suivant soit commutatif :



c'est à dire que $\phi = \psi \circ \iota$.

La construction mène au produit tensoriel, à savoir :

- $\mathcal{E} = \mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$
- $\iota(x_1, \dots, x_p) = x_1 \otimes \dots \otimes x_p$.

Ainsi, pour toute application multilinéaire ϕ de (\mathcal{E}_1) dans \mathcal{F} , il existe une application linéaire L et une seule telle que :

$$\phi(x_1, \dots, x_p) = L(x_1 \otimes \dots \otimes x_p).$$

Cette propriété dite universelle du produit tensoriel sera très utilisée dans la suite du document, car elle présente un double avantage :

- c'est en fait un théorème d'existence
- elle permet le calcul effectif de ϕ en connaissant L sur une base.

Plus particulièrement, étant donné un p -uplet (a_1, \dots, a_p) de (\mathcal{E}_1) , l'application $\phi : (x_1, \dots, x_p) \rightarrow \langle a_1, x_1 \rangle \dots \langle a_p, x_p \rangle$ est une application p -linéaire. L'application L associée est $a_1 \otimes \dots \otimes a_p$, et, en identifiant ψ et ϕ par l'injection canonique ι , l'égalité suivante est vérifiée :

$$a_1 \otimes \dots \otimes a_p (x_1, \dots, x_p) = \langle a_1, x_1 \rangle \dots \langle a_p, x_p \rangle$$

qui peut être prise comme définition du produit tensoriel de vecteurs.

le choix précédent est souvent retenu dans les présentations du produit tensoriel orientées vers les applications, car il requiert moins

d'abstraction.

2. Existence du produit tensoriel

Montrons, sur l'exemple plus simple en écriture des formes bilinéaires sur $\mathcal{E} \times \mathcal{E}$, où \mathcal{E} est un espace vectoriel sur le corps K , comment la commutativité du diagramme précédent définit le produit tensoriel. Cette présentation est inspirée de CHAMBADAL & OVAERT (1968).

1 - Comme ϕ est bilinéaire :

$$\phi(\alpha x + \beta y, z) = \alpha \phi(x, z) + \beta \phi(y, z)$$

$$\phi(x, \lambda y + \mu z) = \lambda \phi(x, y) + \mu \phi(x, z)$$

2 - Comme $\phi = \psi \circ \iota$, et que ψ est linéaire :

$$\psi [\iota(\alpha x + \beta y, z) - \alpha \iota(x, z) - \beta \iota(y, z)] = 0$$

$$\psi [\iota(x, \lambda y + \mu z) - \lambda \iota(x, y) - \mu \iota(x, z)] = 0$$

3 - Soit \mathcal{H} le SEV de \mathcal{E} engendré par les éléments de la forme :

$$\iota(\alpha x + \beta y, z) - \alpha \iota(x, z) - \beta \iota(y, z)$$

ou de la forme :

$$\iota(x, \lambda y + \mu z) - \lambda \iota(x, y) - \mu \iota(x, z)$$

où $\alpha, \beta, \lambda, \mu$ décrivent K et x, y, z décrivent \mathcal{E} .

Alors, $\forall h \in \mathcal{H}, \psi(h) = 0$. Donc, $\mathcal{H} \subset \text{Ker}(\psi)$.

4 - Il est ensuite possible démontrer que l'ensemble $\mathcal{E} = K^{(\mathcal{E} \times \mathcal{E})} / \mathcal{H}$ convient.

3. Extension aux modules

En fait, dans la présentation axiomatique précédente, l'inversibilité des éléments κ du corps K n'est pas intervenue. Cette construction s'étend naturellement aux modules sur un anneau A commutatif unitaire.

Cependant, dans la suite du document, nous nous restreindrons aux produits tensoriels d'espaces vectoriels.

4. Présentation retenue par différents auteurs

BERTIN & BERTIN (1981) partent de la définition axiomatique générale suivante : si

- A est un anneau
- $\mathcal{M}_1, \dots, \mathcal{M}_p$ est une famille de modules sur A (on dit également A -modules)
- \mathcal{P} désigne le A -module somme directe des \mathcal{M}_i
- \mathcal{Q} désigne le A -module engendré par les p -uplets de la forme :

$$(m_1, \dots, m_{i-1}, m_i + m'_i, m_{i+1}, \dots, m_p) - (m_1, \dots, m_p) - (m_1, \dots, m'_i, \dots, m_p)$$

ou

$$(m_1, \dots, m_{i-1}, \alpha m_i, m_{i+1}, \dots, m_p) - \alpha(m_1, \dots, m_p)$$

Alors, on appelle produit tensoriel des A -modules \mathcal{M}_i et l'on note

$$\mathcal{M}_1 \otimes_A \dots \otimes_A \mathcal{M}_p$$

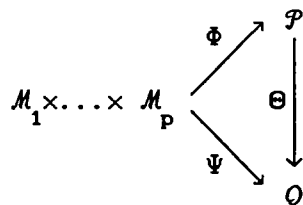
le module quotient \mathcal{P}/\mathcal{Q} .

De plus, la classe de (m_1, \dots, m_p) modulo \mathcal{Q} se note $m_1 \otimes \dots \otimes m_p$, et s'appelle produit tensoriel des éléments (m_i) .

Ensuite, en notant Φ l'application canonique $\mathcal{M}_1 \times \dots \times \mathcal{M}_p \longrightarrow \mathcal{M}_1 \otimes \dots \otimes \mathcal{M}_p$, il est possible de montrer que pour tout A -module \mathcal{N} et toute application linéaire $\Psi : \mathcal{M}_1 \otimes \dots \otimes \mathcal{M}_p \longrightarrow \mathcal{N}$, $\Psi \circ \Phi$ est p -linéaire. Réciproquement, pour toute application Θ p -linéaire de $\mathcal{M}_1 \times \dots \times \mathcal{M}_p$ dans \mathcal{N} , il existe une application linéaire $\Psi : \mathcal{M}_1 \otimes \dots \otimes \mathcal{M}_p \longrightarrow \mathcal{N}$ et une seule telle que $\Theta = \Psi \circ \Phi$.

Puis, l'ensemble des propriétés algébriques du produit tensoriel sont démontrées (associativité, distributivité, etc ...).

LANG (1984) utilise le langage des catégories, introduit par EILENBERG vers 1940. A partir de la définition des applications multilinéaires sur une famille de A -modules (\mathcal{M}_i) où A est un anneau commutatif : $\Phi : \mathcal{M}_1 \times \dots \times \mathcal{M}_p \longrightarrow \mathcal{N}$, il définit un morphisme Θ entre ces applications multilinéaires :



tel que le diagramme précédent soit commutatif. Un objet universel de cette catégorie est alors appelé produit tensoriel des modules (M_i) . Il faut ensuite prouver qu'un tel objet existe, et en construire un.

La construction proposée par LANG (1984) est celle du module \mathcal{P}/Q utilisé comme définition par BERTIN & BERTIN (1981). Les principales propriétés algébriques du produit tensoriel en sont déduites.

Puis LANG (1984) construit l'algèbre tensorielle d'un module M comme algèbre graduée somme directe des produits tensoriels $\mathcal{A}_r(M)$, puis le produit symétrique et le produit antisymétrique de deux tenseurs.

CHAMBADAL & OVAERT (1968) utilisent une construction quasi équivalente, à partir de la propriété universelle du produit tensoriel puis la construction de l'algèbre \mathcal{P}/Q , mais sur les espaces vectoriels (et non les modules).

CHARLES & ALLOUCH (1984) définissent les applications multilinéaires sur une famille de modules sur un anneau commutatif unitaire, puis directement le produit tensoriel de formes (a_1^*, \dots, a_p^*) par :

$$a_1^* \otimes \dots \otimes a_p^* (x_1, \dots, x_p) = \langle a_1^*, x_1 \rangle \dots \langle a_p^*, x_p \rangle$$

puis montrent que ces éléments engendrent le module $\mathcal{A}_p(M_1 \times \dots \times M_p)$ des formes multilinéaires sur $M_1 \times \dots \times M_p$ et en déduisent l'algèbre tensorielle.

DEHEUVELS (1981) définit par le même procédé le produit tensoriel de formes multilinéaires sur des espaces vectoriels, mais utilise la notion de propriété universelle pour la définition du produit extérieur (applications multilinéaires alternées).

3. PRODUIT TENSORIEL DE VECTEURS

Les définitions rappelées dans ce paragraphe sont classiques, et à la base de l'étude des tableaux à plusieurs entrées.

Soit $(\mathcal{E}_i)_{i=1,p}$ une famille d'espaces vectoriels euclidiens sur \mathbb{R} , dont le produit scalaire est noté $\langle \cdot, \cdot \rangle$.

1. FORME MULTILINEAIRE

■ Définition

□ On appelle forme p -linéaire sur le produit cartésien $\mathcal{E}_1 \times \dots \times \mathcal{E}_p$ une application réelle :

$$\begin{aligned} \varphi : \mathcal{E}_1 \times \dots \times \mathcal{E}_p &\longrightarrow \mathbb{R} \\ (x_1, \dots, x_p) &\longmapsto \varphi(x_1, \dots, x_p) \end{aligned}$$

qui est linéaire par rapport à chacun de ses composantes.

□ C'est à dire :

$\forall i \leq p, \forall \alpha, \beta \in \mathbb{R}, \forall x, y \in \mathcal{E}_i,$

$$\begin{aligned} \varphi(x_1, \dots, x_{i-1}, \alpha x + \beta y, x_{i+1}, \dots, x_p) &= \alpha \varphi(x_1, \dots, x_{i-1}, x, x_{i+1}, \dots, x_p) \\ &+ \beta \varphi(x_1, \dots, x_{i-1}, y, x_{i+1}, \dots, x_p) \end{aligned}$$

■ Exemple

Si \mathcal{E} est un EV de dimension n sur \mathbb{R} , la fonction Det qui à tout n -uplet (x_i) associe son déterminant $\text{Det}(x_1, \dots, x_n)$ est une forme n -linéaire sur \mathcal{E}^n .

■ Espace vectoriel des formes p -lineaires

L'ensemble des formes p -lineaires sur $\mathcal{E}_1 \times \dots \times \mathcal{E}_p$ forme un espace vectoriel de dimension $n_1 n_2 \dots n_p$ qui sera noté dans la suite :

$$\mathcal{A}(\mathcal{E}_1 \times \dots \times \mathcal{E}_p).$$

2. PRODUIT TENSORIEL DE P VECTEURS

Chaque espace \mathcal{E}_1 est identifié avec son dual \mathcal{E}_1^* .

Soit (a_i) un p-uplet (a_1, \dots, a_p) de vecteurs de $\mathcal{E}_1 \times \dots \times \mathcal{E}_p$.

■ Définition

On appelle produit tensoriel des vecteurs (a_i) , et l'on note $a_1 \otimes \dots \otimes a_p$ ou

$\bigotimes_{i=1}^p a_i$ la forme p-linéaire définie par :

$$\begin{aligned} \bigotimes_{i=1}^p a_i : \mathcal{E}_1 \times \dots \times \mathcal{E}_p &\longrightarrow \mathbb{R} \\ (x_1, \dots, x_p) &\longmapsto \langle a_1, x_1 \rangle \dots \langle a_p, x_p \rangle \end{aligned}$$

soit, de façon synthétique :

$$\bigotimes_{i=1}^p a_i (x_1, \dots, x_p) = \prod_{i=1}^p \langle a_i, x_i \rangle$$

■ Base de l'espace vectoriel des formes p-linéaires

Rappelons que $(e_{11}, \dots, e_{1n_1}) = (e_{ij})_{j=1, \dots, n_1}$ est une BON de \mathcal{E}_1 .

La famille $(e_{1j_1} \otimes \dots \otimes e_{pj_p})$ où j_1 parcourt $\{1, \dots, n_1\}$, ..., j_p parcourt $\{1, \dots, n_p\}$ est une BON de $\mathcal{A}(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$. C'est la BON canoniquement associée à la famille de bases $(e_{ij})_{ij}$.

Cette BON est notée de façon synthétique :

$$E_k = E_{j_1 \dots j_p} = \bigotimes_{i=1}^p e_{ij_i}$$

où k varie de 1 à $n_1 \cdot n_2 \dots n_p$.

4. ALGÈBRE TENSORIELLE

1. DEFINITION

■ Conventions

1 - Soit $(\mathcal{E}_i)_{i=1,p}$ une famille d'espaces vectoriels sur \mathbb{R} .

2 - Pour tout $m \in \mathbb{N}$, on considère l'ensemble des espaces vectoriels

$$\mathcal{X}_1 \times \dots \times \mathcal{X}_m$$

où \mathcal{X}_i est l'un quelconque des \mathcal{E}_j . Par exemple, si $p=2$ et $m=3$, ces espaces sont :

$$\begin{aligned} &\mathcal{E}_1 \times \mathcal{E}_1 \times \mathcal{E}_1 \\ &\mathcal{E}_1 \times \mathcal{E}_1 \times \mathcal{E}_2 \\ &\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_1 \\ &\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_2 \\ &\mathcal{E}_2 \times \mathcal{E}_1 \times \mathcal{E}_1 \\ &\mathcal{E}_2 \times \mathcal{E}_1 \times \mathcal{E}_2 \\ &\mathcal{E}_2 \times \mathcal{E}_2 \times \mathcal{E}_1 \\ &\mathcal{E}_2 \times \mathcal{E}_2 \times \mathcal{E}_2 . \end{aligned}$$

Il y a p^m tels espaces vectoriels.

3 - A chacun de ces espaces, on associe l'espace $\mathcal{A}(\mathcal{X}_1 \times \dots \times \mathcal{X}_m)$, espace vectoriel des formes p -linéaires sur $\mathcal{X}_1 \times \dots \times \mathcal{X}_m$.

■ Définition

On appelle alors algèbre tensorielle sur la famille $(\mathcal{E}_i)_{i=1,p}$ l'espace vectoriel somme directe des espaces $\mathcal{A}(\mathcal{X}_1 \times \dots \times \mathcal{X}_m)$ lorsque les \mathcal{X}_i parcourent les \mathcal{E}_j et m parcourt \mathbb{N} . On note $\mathcal{T}(\mathcal{E}_1, \dots, \mathcal{E}_p)$ ou $\mathcal{T}((\mathcal{E}_i)_{i=1,p})$ ou plus simplement $\mathcal{T}(\mathcal{E}_i)_i$ lorsqu'aucune ambiguïté n'est à craindre sur les \mathcal{E}_i cet espace :

$$\mathcal{T}(\mathcal{E}_1, \dots, \mathcal{E}_p) = \bigoplus_{\substack{i=0 \\ \mathcal{X}_1 \in (\mathcal{E}_1)_j}}^{\infty} \mathcal{A}(\mathcal{X}_1 \times \dots \times \mathcal{X}_m)$$

2. TENSEUR

■ Définition

On appelle tenseur sur la famille $(\mathcal{E}_1, \dots, \mathcal{E}_p)$ tout élément de $\mathcal{T}(\mathcal{E}_1, \dots, \mathcal{E}_p)$.

■ Rappel

Rappelons que $\mathcal{T}(\mathcal{E}_1, \dots, \mathcal{E}_p)$ est un espace vectoriel sur \mathbb{R} , et donc que si A et B sont deux tenseurs et λ et μ deux scalaires, les égalités suivantes sont vérifiées :

- $\lambda A + \mu B \in \mathcal{T}(\mathcal{E}_1, \dots, \mathcal{E}_p)$
- $\lambda(\mu A) = (\lambda\mu)A$
- $(\lambda + \mu)(A + B) = \lambda A + \lambda B + \mu A + \mu B$

■ Composantes d'un tenseur

On appelle composante d'un tenseur A sur l'espace $\mathcal{A}(\mathcal{X}_1 \times \dots \times \mathcal{X}_m)$ sa projection sur cet espace. Tout tenseur est évidemment somme de ses composantes. Par construction, seul un nombre fini de composantes sont non nulles.

■ Tenseur élémentaire

On appelle tenseur élémentaire tout tenseur ne possédant qu'une seule composante non nulle. Si cette composante est $\mathcal{E}_1 \times \dots \times \mathcal{E}_p$, le tenseur est dit élémentaire d'ordre p . C'est une forme p -linéaire sur $\mathcal{E}_1 \times \dots \times \mathcal{E}_p$.

■ Tenseur élémentaire décomposé

On appelle tenseur élémentaire décomposé sur $\mathcal{A}(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$, et on notera TED, un tenseur élémentaire qui s'écrit comme produit tensoriel de vecteurs des

\mathcal{E}_1 .

$$A = x_1 \otimes \dots \otimes x_p$$

où $x_i \in \mathcal{E}_1$.

3. PRODUIT TENSORIEL DE TENSEURS

■ Produit tensoriel de tenseurs élémentaires

Soit $A \in \mathcal{A}(\mathcal{X}_1 \times \dots \times \mathcal{X}_m)$ et $B \in \mathcal{A}(\mathcal{Y}_1 \times \dots \times \mathcal{Y}_q)$ deux tenseurs élémentaires. A est donc une forme m -linéaire sur $\mathcal{X}_1 \times \dots \times \mathcal{X}_m$ et B une forme q -linéaire sur $\mathcal{Y}_1 \times \dots \times \mathcal{Y}_q$. On appelle produit tensoriel des tenseurs A et B , et on note $C = A \otimes B$, la forme $m+q$ -linéaire :

$$\begin{aligned} A \otimes B : \mathcal{X}_1 \times \dots \times \mathcal{X}_m \times \mathcal{Y}_1 \times \dots \times \mathcal{Y}_q &: \longrightarrow \mathbb{R} \\ (x_1, \dots, x_m, y_1, \dots, y_q) &\longmapsto A(x_i)_i \cdot B(y_j)_j \end{aligned}$$

$$A \otimes B(x_1, \dots, x_m, y_1, \dots, y_q) = A(x_1, \dots, x_m) \cdot B(y_1, \dots, y_q)$$

■ Produit tensoriel de tenseurs

Soient A et B deux tenseurs de $\mathcal{A}(\mathcal{E}_1)_1$. Ces deux tenseurs sont somme de leurs composantes : $A = \sum_{i=0}^{\infty} A_i$; $B = \sum_{j=0}^{\infty} B_j$. Le produit tensoriel C de A et B est alors :

$$C = A \otimes B = \sum_{i,j=0}^{\infty} A_i \otimes B_j$$

4. STRUCTURE D'ALGÈBRE POUR L'OPERATION \otimes

■ Lemme

Le produit tensoriel est une opération interne à $\mathcal{A}(\mathcal{E}_1)_1$. Muni de cette opération, $\mathcal{A}((\mathcal{E}_1)_1)$ est une algèbre unitaire non commutative.

5. LEMME FONDAMENTAL DE L'ALGÈBRE MULTILINEAIRE

■ Enoncé

Soit (\mathcal{E}_i) une famille finie d'espaces vectoriels sur \mathbb{R} , et \mathcal{E} leur somme directe : $\mathcal{E} = \bigoplus_{i=1}^p \mathcal{E}_i$. Alors, il existe un isomorphisme canonique entre l'algèbre tensorielle $\mathcal{A}(\mathcal{E}_i)_i$ définie précédemment et $\mathcal{A}(\mathcal{E})$:

$$\mathcal{A} \left(\bigoplus_{i=1}^p \mathcal{E}_i \right) \cong \mathcal{A}(\mathcal{E})_i$$

■ Démonstration

Cet isomorphisme est une transcription directe de l'isomorphisme :

$$\begin{aligned} \mathcal{E} &\longrightarrow \bigoplus_{i=1}^p \mathcal{E}_i \\ x &\longmapsto (u_i)_i \\ \text{tel que : } x &= \sum_i u_i \end{aligned}$$

■ Conséquence

Ainsi, l'étude de $\mathcal{A}(\mathcal{E}_i)_i$ peut elle se ramener à l'étude de $\mathcal{A}(\mathcal{E})$.

On notera $\mathcal{A}_m(\mathcal{E})$ l'espace $\mathcal{A}(\mathcal{E}^m)$ des formes m -linéaires sur \mathcal{E} .

D'où :

$$\mathcal{A}(\mathcal{E}) = \bigoplus_{m=1}^{\infty} \mathcal{A}_m(\mathcal{E})$$

5. STRUCTURE EUCLIDIENNE CANONIQUE SUR L'ALGÈBRE TENSORIELLE

Dans ce paragraphe et la suite, nous étudions l'algèbre tensorielle $\mathcal{A}(\mathcal{E})$.

1. PRODUIT SCALAIRE SUR L'ALGÈBRE TENSORIELLE

■ Cas des tenseurs élémentaires décomposés

Soient A et B deux tenseurs décomposés de $\mathcal{A}_p(\mathcal{E})$:

$$A = \bigotimes_{i=1}^p a_i ; B = \bigotimes_{j=1}^p b_j .$$

Le produit scalaire de A et B est alors défini par :

$$\langle A, B \rangle = \prod_{i=1}^p \langle a_i, b_i \rangle$$

■ Cas des tenseurs élémentaires

Le produit scalaire de tenseurs décomposés est étendu par linéarité à $\mathcal{A}_p(\mathcal{E})$. En effet, tout tenseur élémentaire est combinaison linéaire de tenseurs élémentaires décomposés. On vérifie sans peine que le produit scalaire ainsi défini ne dépend pas de la combinaison choisie.

■ Cas des matrices

Les matrices sont des tenseurs particuliers : $\mathcal{M}_n(\mathbb{K}) \cong \mathcal{A}_2(\mathbb{K}^n)$.

Soient M et N deux matrices définies par :

$$M = \sum_{i,j=1}^n \mu_{ij} e_i \otimes e_j ; N = \sum_{i,j=1}^n \nu_{ij} e_i \otimes e_j$$

$$\begin{aligned} \text{Alors, } \langle M, N \rangle &= \left\langle \sum_{i,j=1}^n \mu_{ij} e_i \otimes e_j, \sum_{k,l=1}^n \nu_{kl} e_k \otimes e_l \right\rangle \\ &= \sum_{i,j=1}^n \sum_{k,l=1}^n \mu_{ij} \nu_{kl} \langle e_i \otimes e_j, e_k \otimes e_l \rangle \\ &= \sum_{i,j=1}^n \sum_{k,l=1}^n \mu_{ij} \nu_{kl} \langle e_i, e_k \rangle \langle e_j, e_l \rangle \\ &= \sum_{i,j=1}^n \mu_{ij} \nu_{ij} \\ &= \text{Tr}(N'M) \end{aligned}$$

■ Cas général

Soient A et B deux tenseurs de $\mathcal{A}(\mathcal{E})$, décomposés comme sommes de tenseurs élémentaires :

$$A = \sum_{i=0}^{\infty} A_i \quad ; \quad B = \sum_{j=0}^{\infty} B_j .$$

Le produit scalaire de A et B est alors :

$$\langle A, B \rangle = \sum_{i=0}^{\infty} \langle A_i, B_i \rangle$$

Cela revient à étendre par linéarité le produit scalaire sur les $\mathcal{A}_p(\mathcal{E})$ en précisant : $\mathcal{A}_p(\mathcal{E}) \perp \mathcal{A}_q(\mathcal{E})$ dès que $p \neq q$.

2. STRUCTURE HILBERTIENNE DE L'ALGÈBRE TENSORIELLE

■ Lemme

Munie du produit scalaire précédent, l'algèbre $\mathcal{A}(\mathcal{E})$ est préhilbertienne réelle. Ce n'est pas un espace hilbertien car ce n'est pas un espace complet.

■ Base orthonormée

Soit (e_i) une BON de \mathcal{E} . Une BON de $\mathcal{A}_p(\mathcal{E})$ est alors :

$$E_{\vec{j}} = E_{j_1 \dots j_p} = \bigotimes_{i=1}^p e_{i j_i}$$

où j_k varie de 1 à n dans N . Une BON de $\mathcal{A}(\mathcal{E})$ est la réunion des $E_{\vec{j}}$ précédentes lorsque p parcourt N .

■ Complétion topologique

On peut construire une algèbre $\overline{\mathcal{A}}$ hilbertienne contenant \mathcal{A} en complétant \mathcal{A} par les limites des suites de Cauchy. On obtient alors une algèbre de Banach.

Par exemple, si x^p désigne $x \otimes \dots \otimes x$ (p fois), et si $0 < \|x\| < 1$, $\sum_{i=0}^{\infty} x^i$ est une série convergente de limite T tel que $T \otimes (1-x) = 1$.

6. PRODUIT INTÉRIEUR DANS L'ALGÈBRE TENSORIELLE

Cette notion de produit intérieur, qui généralise le produit d'une matrice par un vecteur et peut être regardé comme une contraction après identification des espaces avec leurs duals, est centrale car elle permet une économie certaine dans les calculs par un allègement important dans les notations, qui deviennent synthétiques. La présentation est inspirée de DIEUDONNE (1970).

■ Définition

• Soit A un tenseur de $\mathcal{A}(\mathcal{E})$. On peut lui associer l'endomorphisme U_A de $\mathcal{L}(\mathcal{A}(\mathcal{E}))$ défini par :

$$U_A(X) = A \otimes X.$$

• On associe à U_A son endomorphisme dual U_A^* défini par :

$$\forall X, Y \in \mathcal{A}(\mathcal{E}), \quad \langle U_A(X), Y \rangle = \langle X, U_A^*(Y) \rangle$$

• On note alors $Y.A$ et on appelle produit intérieur de Y par A le tenseur $U_A^*(Y)$.

• Pour trois tenseurs quelconques X, Y, Z , on a l'égalité :

$\langle X, Y \otimes Z \rangle = \langle X.Y, Z \rangle$

■ Cas des tenseurs élémentaires

Soit A et Y deux tenseurs élémentaires. L'égalité précédente étant vérifiée pour tout X , on peut choisir X comme étant un tenseur élémentaire. Supposons donc :

$$A \in \mathcal{A}_p(\mathcal{E}) ; X \in \mathcal{A}_q(\mathcal{E}) ; Y \in \mathcal{A}_r(\mathcal{E})$$

$$\text{D'où : } A \otimes X \in \mathcal{A}_{p+q}(\mathcal{E}) .$$

Pour que le produit scalaire $\langle A \otimes X, Y \rangle$ soit non nul, il est nécessaire que $Y \in \mathcal{A}_{p+q}(\mathcal{E})$, soit $r = p + q$

D'où :

$$\left. \begin{array}{l} Y \in \mathcal{A}_{p+q}(\mathcal{E}) \\ A \in \mathcal{A}_p(\mathcal{E}) \end{array} \right\} \Longrightarrow Y.A \in \mathcal{A}_q(\mathcal{E})$$

■ Propriétés algébriques du produit intérieur

Le produit intérieur présente plusieurs propriétés algébriques élémentaires, mais utiles pour la simplification des calculs, qui sont présentées ci-après.

■ Le produit intérieur est distributif par rapport à l'addition.

$$\forall A, B, C \in \mathcal{A}(\mathcal{E}),$$

$$A.(B+C) = A.B + A.C$$

□ En effet :

$$\begin{aligned} \forall X \in \mathcal{A}(\mathcal{E}), \quad \langle A.(B+C), X \rangle &= \langle A, (B+C) \otimes X \rangle = \langle A, B \otimes X + C \otimes X \rangle = \langle A, B \otimes X \rangle + \langle A, C \otimes X \rangle \\ &= \langle A.B, X \rangle + \langle A.C, X \rangle = \langle A.B + A.C, X \rangle \quad \square \end{aligned}$$

■ $\forall A, B, C \in \mathcal{A}(\mathcal{E})$

$$A.(B \otimes C) = (A.B).C$$

□ En effet :

$$\forall X \in \mathcal{A}(\mathcal{E}), \langle A.(B \otimes C), X \rangle = \langle A, (B \otimes C) \otimes X \rangle = \langle A, B \otimes (C \otimes X) \rangle = \langle A.B, C \otimes X \rangle = \langle (A.B).C, X \rangle \quad \square$$

$$\blacksquare \forall p, r \in \mathbb{N} \text{ tq } r \leq p, \forall A \in \mathcal{A}_p(\mathcal{E}), \forall B \in \mathcal{A}_r(\mathcal{E}), \forall C \in \mathcal{A}_r(\mathcal{E})$$

$$(A \otimes B).C = (A.C) \otimes B$$

□ En effet :

$$\rightarrow \forall X \in \mathcal{A}(\mathcal{E}), \langle (A \otimes B).C, X \rangle = \langle A \otimes B, C \otimes X \rangle.$$

→ Soit (E_i) une base de $\mathcal{A}_r(\mathcal{E})$. On peut alors décomposer A sur cette base comme suit : $A = \sum_i E_i \otimes A_i$ [$A_i = A.E_i$].

$$\rightarrow \text{D'où : } \langle A \otimes B, C \otimes X \rangle = \left\langle \left[\sum_i E_i \otimes A_i \right] \otimes B, C \otimes X \right\rangle$$

$$= \sum_i \langle E_i \otimes A_i \otimes B, C \otimes X \rangle$$

$$= \sum_i \langle E_i, C \rangle \langle A_i \otimes B, X \rangle$$

$$= \sum_i \langle \langle E_i, C \rangle A_i \otimes B, X \rangle$$

$$= \sum_i \langle [(E_i \otimes A_i).C] \otimes B, X \rangle$$

$$= \langle [(\sum_i E_i \otimes A_i).C] \otimes B, X \rangle$$

$$= \langle (A.C) \otimes B, X \rangle \quad \square$$

$$\blacksquare \forall p \in \mathbb{N}, \forall A, B \in \mathcal{A}_p(\mathcal{E}), \quad A.B = \langle A, B \rangle$$

□ En effet :

$$\rightarrow A \in \mathcal{A}_p(\mathcal{E}) \text{ et } B \in \mathcal{A}_p(\mathcal{E}) \implies A.B \in \mathbb{R}$$

$$\rightarrow \forall \lambda \in \mathbb{R}, (A.B)\lambda = \langle A.B, \lambda \rangle = \langle A, B \otimes \lambda \rangle = \langle A, \lambda B \rangle = \lambda \langle A, B \rangle \quad \square$$

7. DÉCOMPOSITION D'UN TENSEUR SUR UNE BASE

1. DECOMPOSITION SUR UNE BASE ORTHONORMEE

Bien que la décomposition sur une BON soit un cas particulier de la décomposition sur une base quelconque présentée en 1.5.3., elle est présentée ici de façon indépendante car c'est une situation particulière très fréquente en pratique.

■ Lemme

Soit A un tenseur de $\mathcal{A}_p(\mathcal{E})$, où $p \in \mathbb{N}$. Soit $r \in \mathbb{N}$ tel que $r \leq p$. Soit $(E_i)_1$ une base orthonormée de $\mathcal{A}_r(\mathcal{E})$. Alors :

$$A = \sum_i E_i \otimes A \cdot E_i$$

■ Démonstration

→ Si $(F_j)_j$ est une BON de $\mathcal{A}_{p-r}(\mathcal{E})$, alors $(E_i \otimes F_j)_{i,j}$ est une BON de $\mathcal{A}_p(\mathcal{E})$.
 → Tout tenseur A de $\mathcal{A}_p(\mathcal{E})$ se décompose alors comme suit : $A = \sum_{i,j} \alpha_{ij} E_i \otimes F_j$.
 → Il vient alors : $A = \sum_{i,j} \alpha_{ij} E_i \otimes F_j = \sum_i E_i \otimes \left(\sum_j \alpha_{ij} F_j \right) = \sum_i E_i \otimes A_i$.
 → Soit donc $A = \sum_i E_i \otimes A_i$. Il vient : $\forall k \leq r \in \mathbb{N}, A \cdot E_k = \left(\sum_i E_i \otimes A_i \right) \cdot E_k = \sum_i \langle E_i, E_k \rangle A_i = A_k \quad \square$

2. NOTION DE BASE DUALE

■ Définition

Soit \mathcal{E} un espace vectoriel, et $(e_i)_1$ une base de \mathcal{E} . Alors, il existe une base $(e_i^*)_1$ et une seule telle que :

$$\langle e_i, e_j^* \rangle = \delta_{ij}$$

Cette base est appelée base duale de $(e_i)_i$.

■ Propriétés

Les deux propriétés suivantes sont immédiates :

$$\rightarrow (e_i^*)^* = e_i$$

$$\rightarrow \text{Si } (e_i)_i \text{ est une BON, } e_i^* = e_i.$$

3. DECOMPOSITION SUR UNE BASE QUELCONQUE

■ Lemme

Soit A un tenseur de $\mathcal{A}_p(\mathcal{E})$, où $p \in \mathbb{N}$. Soit $r \in \mathbb{N}$ tel que $r \leq p$. Soit $(E_i)_i$ une base quelconque de $\mathcal{A}_r(\mathcal{E})$. Alors :

$$A = \sum_i E_i \otimes A \cdot E_i^*$$

■ Démonstration

\rightarrow Si $(F_j)_j$ est une base de $\mathcal{A}_{p-r}(\mathcal{E})$, alors $(E_i \otimes F_j)_{i,j}$ est une base de $\mathcal{A}_p(\mathcal{E})$.

\rightarrow Tout tenseur A de $\mathcal{A}_p(\mathcal{E})$ se décompose alors comme suit : $A = \sum_{i,j} \alpha_{ij} E_i \otimes F_j$.

\rightarrow Il vient alors : $A = \sum_{i,j} \alpha_{ij} E_i \otimes F_j = \sum_i E_i \otimes (\sum_j \alpha_{ij} F_j) = \sum_i E_i \otimes A_i$.

\rightarrow Soit donc $A = \sum_i E_i \otimes A_i$. Il vient : $\forall j \leq r \in \mathbb{N}, A \cdot E_j^* = (\sum_i E_i \otimes A_i) \cdot E_j^* = \sum_i \langle E_i, E_j^* \rangle A_i = A_j \quad \square$

8. ACTION DU GROUPE DES PERMUTATIONS SUR UN TENSEUR

1. ACTION SUR UN TENSEUR ELEMENTAIRE DECOMPOSE

■ Définition

Soit $A = \bigotimes_{i=1}^p x_i$ un TED de $\mathcal{A}_p(\mathcal{E})$. Soit σ une permutation de \mathcal{S}_p . On définit alors l'action de σ sur A par :

$$\sigma \left[\bigotimes_{i=1}^p x_i \right] = \bigotimes_{i=1}^p x_{\sigma(i)}$$

■ Exemples

- Soit $p = 2$. Il vient : $\sigma(1)=2$ et $\sigma(2) = 1$, ce que nous convenons d'écrire : $\sigma(1,2) = (2,1)$. Si $A = x \otimes y$, alors, $\sigma(x \otimes y) = y \otimes x$.
- Soit $p = 3$, $\sigma(1,2,3) = (2,3,1)$. $\sigma(x \otimes y \otimes z) = y \otimes z \otimes x$.

2. ACTION SUR UN TENSEUR ELEMENTAIRE

■ Définition

Soit p un entier naturel : $p \in \mathbb{N}$, σ une permutation de \mathcal{S}_p , A un tenseur de $\mathcal{A}_p(\mathcal{E})$. Le tenseur $\sigma(A)$, transformé de A par l'action de σ , est alors défini

par : quel que soit le tenseur décomposé $X = \bigotimes_{i=1}^p x_i$ de $\mathcal{A}_p(\mathcal{E})$:

$$\langle \sigma(A) , \bigotimes_{i=1}^p x_i \rangle = \langle A , \sigma^{-1} \left[\bigotimes_{i=1}^p x_i \right] \rangle = \langle A , \bigotimes_{i=1}^p x_{\sigma^{-1}(i)} \rangle$$

■ Exemple

□ Choisissons $p = 2$. Il vient $\sigma(1,2) = (2,1)$ et $\sigma^{-1} = \sigma$. D'où :

$\forall M \in \mathcal{L}(\mathcal{E}), \forall x, y \in \mathcal{E}, \langle \sigma(M), x \otimes y \rangle = \langle M, y \otimes x \rangle = \langle M, y, x \rangle$. D'où : $\sigma(M) = M'$, ou M' désigne la matrice transposée de M □

■ Propriétés

□ Toute permutation est un endomorphisme de $\mathcal{L}(\mathcal{A}_p(\mathcal{E}))$.

$$\rightarrow \sigma(\lambda A + \mu B) = \lambda \sigma(A) + \mu \sigma(B)$$

→ Ce résultat est une conséquence immédiate de la définition.

$$\square \forall A, B \in \mathcal{A}_p(\mathcal{E})$$

$$\langle \sigma(A), B \rangle = \langle A, \sigma^{-1}(B) \rangle$$

→ En effet, B est une combinaison linéaire de tenseurs décomposés ; cette égalité est vérifiée pour tout tenseur B décomposé [définition de σ] ; et σ est un endomorphisme de $\mathcal{A}_p(\mathcal{E})$.

□ Toute permutation σ de \mathcal{G}_p est une isométrie sur $\mathcal{A}_p(\mathcal{E})$: $\forall A, B \in \mathcal{A}_p(\mathcal{E})$

$$\langle \sigma(A), \sigma(B) \rangle = \langle A, B \rangle$$

$$\|\sigma(A)\| = \|A\|$$

→ En effet : $\langle \sigma(A), \sigma(B) \rangle = \langle A, \sigma^{-1} \circ \sigma(B) \rangle = \langle A, B \rangle$ et $\|\sigma(A)\|^2 = \langle \sigma(A), \sigma(A) \rangle$

3. ACTION SUR UN TENSEUR QUELCONQUE

■ Soit A un tenseur quelconque de $\mathcal{A}(\mathcal{E})$. A s'écrit de façon unique sous la forme : $A = \sum_1 A_1$, où $A_1 \in \mathcal{A}_1(\mathcal{E})$ et où tous les A_1 sont nuls sauf un nombre fini d'entre eux.

■ On définit une permutation agissant sur A comme une collection $\sigma = (\sigma_1)_1$, où $\sigma_1 \in \mathcal{G}_1$ et où toutes les permutations σ_1 sont la permutation identité sauf un nombre fini d'entre elles.

■ Alors, par définition :

$$\sigma\left(\sum_1 A_1\right) = \sum_1 \sigma_1(A_1)$$

9. VARIÉTÉS DE SÈGRE

Rappelons que, dans $\mathcal{A}(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$, on appelle tenseur décomposé un tenseur T

de la forme :

$$T = x_1 \otimes \dots \otimes x_p \quad \text{où } x_i \in \mathcal{E}_i.$$

1. DEFINITION

■ Rappel : Variété de Segre en géométrie projective

Etant donnés deux espaces projectifs P_n et P_q de dimension n et q sur un corps K , on considère l'application qui, aux points (x_i) de P_n et (y_j) de P_q fait correspondre le point $z_{ij} = x_i y_j$ de $P_{(n+1)(q+1)-1}$. Cette application est appelée application de Sègre, et l'image de $P_n \times P_q$ est appelée variété de Sègre pour les dimensions n et q .

■ Application à l'algèbre tensorielle

On appelle variété de Sègre de $\mathcal{A}(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$ la variété constituée par l'ensemble des tenseurs élémentaires décomposés. On note $\Gamma(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$ cette variété.

Dans le cas où tous les \mathcal{E}_i sont égaux à un même espace \mathcal{E} , on note $\Gamma_p(\mathcal{E})$ la variété de Sègre, dite d'ordre p sur \mathcal{E} .

2. TREILLIS DES SOUS-VARIETES DE SEGRE

Soient $\mathcal{F}_1, \dots, \mathcal{F}_p$ p sous espaces vectoriels de $\mathcal{E}_1, \dots, \mathcal{E}_p$ respectivement. La variété $\Gamma(\mathcal{F}_1 \times \dots \times \mathcal{F}_p)$ est alors une sous variété de Sègre de $\Gamma(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$. L'ensemble de ces sous variétés, lorsque les \mathcal{F}_i parcourent l'ensemble des sous espaces vectoriels des \mathcal{E}_i , forme un treillis pour la relation d'inclusion.

3. APPLICATIONS LINEAIRES CONSERVANT LES VARIETES DE SEGRE

1 - Soit $T \in \mathcal{L}(\mathcal{A}_p(\mathcal{E}))$. On cherche les applications T précédentes conservant globalement les variétés de Sègre.

2 - Le résultat est immédiat à partir du cas $p = 2$. Nous avons alors :

$$T(x \otimes y) = (M.x) \otimes (N.y)$$

et, pour p quelconque :

$$T. \left[\bigoplus_{i=1}^p x_i \right] = \bigoplus_{i=1}^p [M_i.x_i]$$

3 - L'ensemble de ces applications linéaires forme un groupe, sous groupe de $GL(\mathcal{A}_p(\mathcal{E}))$. Si les applications M_i conservent respectivement les sous espaces \mathcal{F}_i de \mathcal{E}_i , alors, T conserve globalement $\Gamma(\mathcal{F}_1 \times \dots \times \mathcal{F}_p)$.

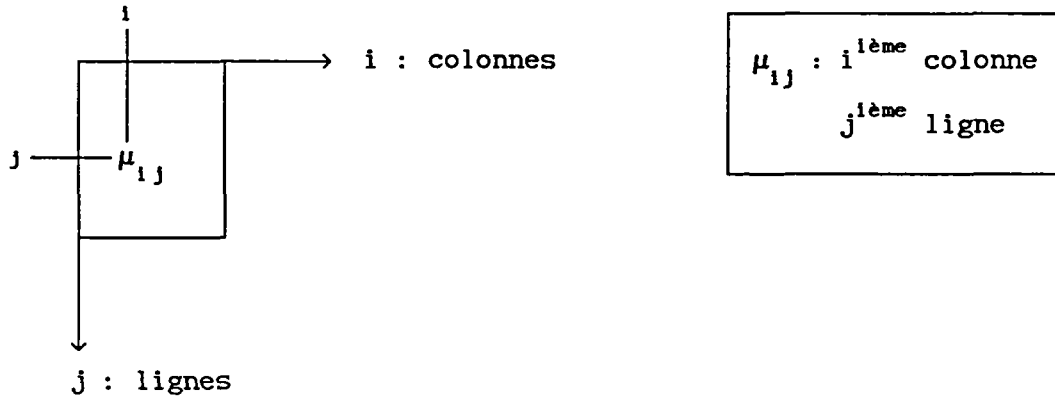
4 - Le groupe de ces applications intervient tout au long de l'analyse des espaces \mathcal{E}_i avec des métriques définies par des MSDP M_i .

4. VARIETES DE SEGRE DE RANG R

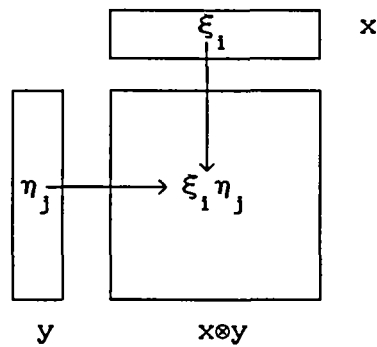
Dans ce qui suit, il sera largement fait appel, non seulement aux tenseurs décomposés de $\mathcal{A}_p(\mathcal{E})$, donc à la variété de Sègre $\Gamma_p(\mathcal{E})$, mais également aux tenseurs qui peuvent s'écrire comme somme de r tenseurs décomposés. Aussi, nous appelons "variété de Sègre de rang r ", et nous notons $\Gamma_p^r(\mathcal{E})$, la variété de $\mathcal{A}_p(\mathcal{E})$ constituée des tenseurs pouvant s'écrire comme somme de r tenseurs décomposés au plus.

VISUALISATION DES MATRICES

L'écriture matricielle $M = \sum_{i,j} \mu_{ij} e_i \otimes e_j$ correspond à :



En général, i désigne la ligne et j la colonne, par convention. En fait, il faut remarquer que $x \otimes y$ s'écrit, si $x = \sum_i \xi_i e_{1i}$ et $y = \sum_j \eta_j e_{2j}$:

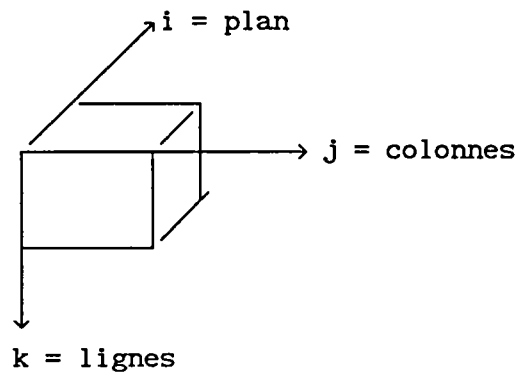


qui s'écrit également, si x' désigne le transposé de x :

$x \otimes y = yx'$

VISUALISTION DES TENSEURS D'ORDRE 3

Le parallélépipède des (α_{ijk}) est écrit par convention selon :



α_{ijk} :	$i^{\text{ième}}$	plan
	$j^{\text{ième}}$	colonne
	$k^{\text{ième}}$	ligne

CHAPITRE B

ANALYSES EN COMPOSANTES PRINCIPALES D'UNE MATRICE

1. QUELQUES REPÈRES D'HISTOIRE

L'Analyse en Composantes Principales est une technique usuelle en analyse de données. Les fondateurs souvent cités sont PEARSON en 1901 (On lines and planes of closest fit to systems of points in space, *Phil. Mag.*, 2(11), 559-572) et HOTELLING en 1933 (Analysis of a complex statistical variable into principal component. *J. Educ. Psy.* 24, 417-441 & 498-520).

Une technique proche est l'Analyse Factorielle, initiée par SPEARMAN en 1904 (General intelligence, objectively determined and measured. *Amer. J. Psychol.*, 15, 72-101) et développée par THURSTONE en 1947 (Multiple Factor Analysis, Chicago university Press). Une critique assez violente de l'Analyse Factorielle a été publiée dans par HILLS en 1977 (Book Review, *Appl. Statist.*, 26, 339-340), fondée notamment sur la difficulté d'interprétation.

Initialement, l'analyse en composantes principale est une technique claire, descriptive, de recherche d'une rotation d'un nuage de points pour une représentation optimale dans un espace de dimension moindre. L'objectif est de minimiser la perte d'inertie d'un nuage de point par le choix approprié d'un espace lors d'une projection. L'analyse factorielle, quant à elle, repose sur un modèle statistique supposé sous-jacent pour expliquer les covariances entres variables.

1. Rappel du cadre conceptuel de l'ACP

Le principal outil mathématique utilisé dès le début pour calculer effectivement les composantes est la décomposition d'une matrice en valeurs singulières (ECKART & YOUNG, 1936). Puis, les outils utilisés se sont enrichis de deux techniques plus géométriques : le schéma de dualité (CAZES, 1970) et le langage des opérateurs (ESCOUFIER & PAGES, 1976), mis au service des trois analyses les plus courantes :

- l'Analyse en Composantes Principales d'un tableau quantitatif centré et normé sur les colonnes
- l'Analyse Factorielle des Corespondances d'une table de contingence, étendue aux tableaux de variables qualitatives
- l'Analyse Canonique

De nombreux manuels présentent l'ACP et les techniques corrélées à un public toujours plus vaste d'utilisateurs, en privilégiant tel ou tel outil : le calcul matriciel (LEBART & FENELON, 1975, ; DIDAY & Al., 1982, LEBART & Al. 1982), le schéma de dualité (CAILLEZ & PAGES, 1976) ou le langage des opérateurs (FOUCART, 1984 ; LAVIT, 1988).

Si, mathématiquement, pour l'ACP, ces trois présentations sont équivalentes : une matrice est canoniquement associée à un opérateur linéaire agissant sur des espaces euclidiens qui conduisent naturellement à la notion de dualité, le maniement de ces outils est plus ou moins aisé selon le problème posé. Les isomorphismes canoniques entre $\mathcal{M}(\mathcal{E}, \mathcal{F})$ et $\mathcal{L}(\mathcal{E}, \mathcal{F})$ et entre \mathcal{E}^* et \mathcal{E} permettent de traduire moyennant une certaine agilité ou une longue habitude toute propriété dans les trois langages de présentation qui ont la préférence de tel ou tel auteur.

2. La Décomposition en valeurs singulières d'une matrice

l'ACP repose sur la représentation optimale d'une matrice donnée par une matrice de rang inférieur, dont l'article fondateur souvent cité est (ECKART & YOUNG, 1936), à tel point que la décomposition d'une matrice selon ses axes principaux est souvent appelée décomposition d'ECKART-YOUNG. Elle repose sur la décomposition en valeurs singulières d'une matrice rectangulaire.

La décomposition en valeurs singulières d'une matrice été proposée indépendamment par BELTRAMI en 1873 (sulle funzioni bilineari, *Giornale di Matematiche ud uso Degli Studenti Delle Universita*, 11, 98-106) et JORDAN en 1874 (Mémoire sur les formes bilinéaires, *Journal de Mathématiques Pures et Appliquées, Deuxième Série*, 19, 35-44), puis a fréquemment été "redécouverte" ensuite. Elle a été généralisée aux matrices complexes par AUTONNE en 1913 (Sur les matrices hypohermitiennes et unitaires, *C.R.A.S.*, 156, 858-860) et aux matrices rectangulaires par ECKART et YOUNG en 1936, (ECKART & YOUNG, 1936) mais aux opérateurs d'espaces hilbertiens dès 1907 par SCHMIDT (Zur theorie der linearen und nicht linearen Integralgleichungen. I Teil : Entwicklung willkürlichen Funktionen nach System vorgeschriebener, *Mathematische Annalen*, 63, 463-476) à l'occasion d'un travail sur la représentation des noyaux d'opérateurs intégraux. Le terme de valeur singulière d'un opérateur a été introduit par PICARD en 1910 (Sur un théorème général relatif aux équations intégrales de première espèce et sur quelques problèmes de physique mathématique. *Rendicondi del Circolo Matematico di Palermo*, 25, 79-97).

SCHMIDT, dans son travail de 1907, a démontré le lien entre les valeurs singulières du noyau d'un opérateur $K(x,y)$ et sa meilleure représentation comme somme d'un nombre fini fixé de produits du type $\phi(x)\psi(y)$. ECKART et YOUNG ont utilisé la norme de Frobenius, également appelée norme de Schur, cas particulier de la norme naturelle dans un espace hilbertien. Le résultat de SCHMIDT a été étendu pour les normes invariantes par des transformations unitaires par MIRSKY (MIRSKY, 1963). Aussi, la décomposition d'ECKART-YOUNG est-elle également connue sous le nom de théorème de SCHMIDT-MIRSKY.

3. Evolutions récentes

En restant dans le domaine du traitement des matrices, c'est à dire en travaillant dans des espaces vectoriels de dimension finie, les évolutions récentes concernent, si la matrice à étudier appartient à $\mathcal{E} \otimes \mathcal{F}$:

- le choix de métriques particulières sur \mathcal{E} et \mathcal{F}

- l'introduction de contraintes sur \mathcal{E} et \mathcal{F} ,
- les liens avec le modèle linéaire.

En particuliers, les deux grandes familles d'analyses maintenant établies sont :

- l'Analyse Factorielle des Correspondances (BENZECRI, 1973), généralisée à l'Analyse Factorielle des Correspondances Multiple (CAZES, 1980 & 1981),
- l'Analyse en Composantes Principales sous Double Contrainte Linéaire, ou ACP-DCL (CARROLL & Al., 1981 ; DENIS, 1990 ; RAO, 1964 ; ROBERT & ESCOUFIER, 1976 ; SABATIER, 1984 & 1987).

Un cas particulier d'ACP-DCL, lorsque la contrainte ne porte que sur un espace, en général l'espace image de la matrice, est connu parfois sous le nom d'ACP-VI (VI pour Variable Instrumentale), et est utilisé couramment en écologie (SABATIER & Al., 1989).

Les liens entre l'ACP et le modèle linéaire sont présentés dans DENIS (1983 & 1991), avec notamment un accent sur l'ACP-DCL dans ce dernier article.

4. L'Analyse en Composantes Principales sous Double Contrainte Linéaire

L'ACP-DCL a été proposée initialement comme technique particulière d'approximation d'une matrice, par RAO (1964). Elle fut ensuite redécouverte sous le nom de CANDELINC dans CARROLL & Al. (1981), et présentée à partir d'une note présentée au Congrès International de Psychiatrie, à Paris, en 1976 (J. D. CARROLL, P. E. Green & F. J. CARMONE : CANDELINC, a new method for multidimensional analysis with constrained solutions, cité par CARROLL & Al., 1981). Les auteurs partent d'une écriture matricielle de l'ACP : $M = AB'$, où la matrice M de dimensions (n,p) est approchée au mieux par le produit AB' , où A est de dimensions (n,r) et B de dimensions (p,r) . Des contraintes linéaires sont introduites pour les matrices A et B , et le modèle est calculé par des techniques simples d'optimisation en calcul matriciel. Une étude géométrique complète de ce modèle, à l'aide du schéma

de dualité, se trouve dans SABATIER (1987).

Il semble cependant que ce résultat était connu sous une autre présentation depuis longtemps tant chez les algébristes que dans la communauté du calcul numérique matriciel sous divers noms, dont procédure de Rayleigh-Ritz pour les matrices hermitiennes et procédure de Galerkin pour les matrices quelconque. Ces méthodes sont développée assez longuement dans PARLETT (1980) et plus succinctement dans CHATELIN (1988). Les publications initiales sont celles de RAYLEIGH en 1899 (RAYLEIGH, Lord [J. W. STRUTT] - 1899 - On the Calculation of the Frequency of Vibration of a system in its Gravest mode, with an Example from Hydrodynamics. *Philos. Mag.*, 47 556-572) et de RITZ en 1909 (RITZ W. - 1909 - Über eine neue Method zur Lösung Gewisser Variationsprobleme der Mathematischen Physik. *J. Reine Angew. Math.*, 135 1-161). La procédure de RAYLEIGH-RITZ (Procédure RR) consiste à approcher au mieux p valeurs propres d'une matrice A donnée et repose sur le fait qu'un sous espace invariant par A possède une base de vecteurs propres. Elle revient soit à calculer les valeurs propres de la projection de A sur un sous espace bien choisi, qui, en général, diminue la dimension du problème, ou à approcher au mieux les valeurs propres d'une matrice par des vecteurs contraints à appartenir à un sous espace vectoriel donné. Ces deux procédures sont reconnues équivalentes.

5. Analyse en Composantes Principales avec métriques

Une autre notion importante en analyse en composantes principales est celle de métrique. L'introduction de normes dans les espaces vectoriels de dimension quelconque a été réalisée par MINKOWSKI (publication posthume en 1911 : *Theorie der Konvexen Körper, insbesondere Begründung ihres Oberflächenbegriffs.* in Hilbert D. Ed., *Minkovski Abhandlung.* Teubner Verlag) , puis par une voie plus axiomatique par BANACH (Sur les opérations dans les ensembles abstraits et leurs applications aux équations intégrales. *Fundamenta Mathematicae*, 3, 133-181) et WIENER (Limit in terms of continuous transformations, *Bull. Soc. Math. de France*, 50, 119-134) indépendamment en 1922.

L'Analyse en composantes principales traite de l'approximation de matrices par des matrices choisies dans un ensemble précis. Elle fait donc largement appel à des notions de distance dans l'espace des matrices, donc à l'étude des normes et des structures euclidiennes sur l'espace des matrices.

Le premier travail introduisant fugitivement une notion de norme sur une matrice est celui de PEANO en 1888 (Intégration par série des équations différentielles linéaires, *Math. Ann.*, **32**, 450-456). La norme utilisée était la norme spectrale et la norme de Frobenius. SCHUR en 1909 (Über die charakteristischen Wurzeln einer linearen Substitution mit einer Anwendung auf die Theorie der Integralgleichungen. *Mathematische Annalen*, **63**, 433-476) a été le premier à utiliser la norme d'une matrice pour majorer les valeurs propres d'un opérateur. Mais il semble que ce soit FROBENIUS qui explora en détail les propriétés de ce que l'on appelle aujourd'hui la norme de FROBENIUS [$\|A\| = \text{Tr}(A'A)^{1/2}$] en 1911, parfois appelée également norme de Schur, et lui reconnut toutes les propriétés d'une norme. Mais les travaux les plus significatifs ont été obtenus lors d'études sur la norme d'opérateurs en analyse fonctionnelle, suite à l'émergence dans les années 1920 - 1930 de la notion d'espace vectoriel normé et surtout d'espace vectoriel topologique, notamment par von NEUMANN en 1937 (Some matrix inequalities and metrization of matrix-space. *Tomsk. Univ. Rev.* **1**, 286-300) et MIRSKY (1960).

Ces résultats ont ensuite diffusé par percolation lente dans plusieurs travaux sur les matrices, notamment pour l'étude du conditionnement et des perturbations dans les algorithmes de calcul matriciel : produit, inversion, recherche de valeurs propres et de vecteurs propres, ... (HOUSEHOLDER, 1964 ; WILKINSON, 1965), et plus précisément par ROBERT & ESCOUFIER (1976) pour les analyses en composantes principales.

L'un des aboutissements de ces travaux est la lecture de l'AFC comme cas particulier de l'ACP d'une matrice de $\mathcal{E} \otimes \mathcal{F}$, lorsque chacun des espaces \mathcal{E} et \mathcal{F} est muni d'une métrique définie par une MSDP. Les matrices définissant les métriques sont alors adaptées à la distance du χ^2 (ESCOUFIER, 1985).

2. LE THÉORÈME DE SCHMIDT-MIRSKY

L'analyse factorielle peut entièrement être construite à partir d'une conséquence importante du théorème de Mirsky (MIRSKY, 1963), qui est le théorème de Schmidt-Mirsky.

Notons \mathcal{E} et \mathcal{F} deux \mathbb{C} -espaces vectoriels, A et B deux matrices de $\mathcal{M}(\mathcal{E}, \mathcal{F})$, U et V deux matrices unitaires de $\mathcal{M}(\mathcal{F})$ et $\mathcal{M}(\mathcal{E})$ respectivement. (Remarquons que Si \mathcal{E} et \mathcal{F} sont deux \mathbb{R} -espaces vectoriels, les matrices U et V seront orthogonales).

1. LA DECOMPOSITION EN VALEURS SINGULIERES

Si A est de rang r , il existe deux matrices unitaires U et V telles que :

$$U^* A V = \begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix}$$

où D est une matrice diagonale d'éléments $\alpha_1 \geq \dots \geq \alpha_p > 0$.

Les réels $\alpha_1, \dots, \alpha_p$ sont appelés valeurs singulières de A , et sont les racines carrées positives des valeurs propres non nulles de $A^* A$.

Les colonnes de U sont les vecteurs propres de $A A^*$, et les colonnes de V les vecteurs propres de $A^* A$.

2. LE THEOREME DE SCHMIDT-MIRSKY

Le théorème de Schmidt-Mirsky est établi de façon générale pour des espaces vectoriels sur le corps \mathbb{C} des complexes. C'est dans ce cadre qu'il est présenté ci-dessous.

1. Norme invariante par transformation unitaire

1 - Une norme $\|\cdot\|$ sur $\mathcal{M}(\mathcal{E}, \mathcal{F})$ est dite invariante par transformation unitaire (NITU), ou unitairement invariante, si :

$$\forall V \in \mathcal{U}(\mathcal{E}), \forall U \in \mathcal{U}(\mathcal{F}), \|U^*AV\| = \|A\|$$

2 - Ainsi, la norme spectrale et la norme de Frobenius sont des NITU.

3 - Si les espaces vectoriels sont réels, la norme est dite invariante par transformation orthogonale (NITO) si :

$$\forall V \in \mathcal{O}(\mathcal{E}), \forall U \in \mathcal{O}(\mathcal{F}), \|U^tAV\| = \|A\|$$

2. Fonction symétrique de jauge

1 - On appelle Fonction Symétrique de Jauge Φ (FSJ) une norme sur \mathbb{R}^r invariante par toute permutation des coordonnées dans \mathbb{R}^n : si P est une matrice de permutation de $\mathcal{M}(\mathbb{R}^n)$, $\Phi(x) = \Phi(Px)$.

2 - Von NEUMANN, en 1937 (Some matrix inequalities and metrization of matrix-space. *Tomsk. Univ. Rev.* 1, 286-300), a établi un lien remarquable entre les NITU et les FSJ : à toute FSJ Φ on associe la fonction $\|\cdot\|_\Phi$ sur A définie par $\|A\|_\Phi = \Phi(\alpha_1, \dots, \alpha_p)$, où les α_i sont les valeurs singulières de A . Alors, $\|\cdot\|_\Phi$ est une NITU.

3. Théorème de Mirsky

Etant donnés

- Deux espaces vectoriels \mathcal{E} et \mathcal{F}
- Deux matrices A et B de $\mathcal{M}(\mathcal{E}, \mathcal{F})$
- Les valeurs singulières $\alpha_1, \dots, \alpha_p$ de A et β_1, \dots, β_p de B

Alors, pour toute norme $\|\cdot\|$ invariante par transformation unitaire :

$$\|\text{Diag}(\alpha_1 - \beta_1, \dots, \alpha_p - \beta_p)\| \leq \|A - B\|$$

4. Théorème de Schmidt-Mirsky

Etant donnés

- une matrice A de valeurs singulières $\alpha_1 \geq \dots \geq \alpha_p > 0$
- La DVS UDV^* de A
- une matrice B de rang $k \leq p$
- une FSJ Φ
- $\|\cdot\|_\Phi$ la NITU associée

Alors :

$$\|B - A\|_\Phi \geq \Phi(0, \dots, 0, \alpha_{k+1}, \dots, \alpha_p)$$

De plus, l'égalité est atteinte pour la matrice B_k définie par :

$$B_k = UD_k V^* \text{ avec } D_k = \text{Diag}(\alpha_1, \dots, \alpha_k, 0, \dots, 0)$$

qui est donc la meilleure approximation de rang k de A pour la norme $\|A\|_\Phi$.

3. PRÉSENTATION DE L'ACP

1. NOTATIONS ET CONVENTIONS

1. Notations et conventions sur les espaces vectoriels

□ Soient \mathcal{E} et \mathcal{F} deux espaces vectoriels euclidiens réels de dimensions respectives p et q . \mathcal{E} est muni du produit scalaire M et \mathcal{F} du produit scalaire N . Ces deux produits seront notés $\langle \cdot, \cdot \rangle_M$ et $\langle \cdot, \cdot \rangle_N$ respectivement en cas d'ambiguïté, et l'on a :

$$\begin{aligned} \forall x, y \in \mathcal{E}, \langle x, y \rangle_M &= \langle x, M.y \rangle \\ \forall u, v \in \mathcal{F}, \langle u, v \rangle_N &= \langle u, N.v \rangle \end{aligned}$$

□ Rappelons que nous notons $\Gamma_1(\mathcal{E}, \mathcal{F})$ le cône des matrices de rang 1 de $\mathcal{M}(\mathcal{E}, \mathcal{F})$, c'est à dire de la forme $x \otimes y$, où $x \in \mathcal{E}$ et $y \in \mathcal{F}$, et $\Gamma^r(\mathcal{E}, \mathcal{F})$ le cône des matrices de rang r de $\mathcal{M}(\mathcal{E}, \mathcal{F})$, c'est à dire qui s'écrivent sous la forme $\sum_{k=1}^r u_k \otimes v_k$ où $u_k \in \mathcal{E}$ et $v_k \in \mathcal{F}$.

2. Métrique sur $\mathcal{M}(\mathcal{E}, \mathcal{F})$

□ L'espace vectoriel $\mathcal{M}(\mathcal{E}, \mathcal{F})$ est un espace vectoriel réel de dimension pq . Il est muni d'un produit scalaire défini par une matrice symétrique définie positive T de $\mathcal{M}(\mathcal{M}(\mathcal{E}, \mathcal{F}))$, qui est donc une matrice carrée d'ordre pq :

$$\forall A, B \in \mathcal{M}(\mathcal{E}, \mathcal{F}), \langle A, B \rangle_T = \langle A, T.B \rangle$$

□ Les produits scalaires M sur \mathcal{E} et N sur \mathcal{F} induisent canoniquement un produit scalaire sur $\mathcal{M}(\mathcal{E}, \mathcal{F})$, qui est défini par le tenseur T suivant :

$$\forall x \in \mathcal{E}, \forall y \in \mathcal{F}, T.(x \otimes y) = (M.x) \otimes (N.y)$$

c'est à dire :

$$\forall x, y \in \mathcal{E}, \forall u, v \in \mathcal{F}, \langle x \otimes u, y \otimes v \rangle_T = \langle x, y \rangle_M \langle u, v \rangle_N = \langle x, M.y \rangle \cdot \langle u, N.v \rangle$$

□ T est un endomorphisme de $\mathcal{L}(\mathcal{M}(\mathcal{E}, \mathcal{F}))$, donc une matrice de $\mathcal{M}(\mathcal{M}(\mathcal{E}, \mathcal{F}))$. C'est donc une matrice carrée d'ordre pq . Par abus de notation, cette matrice T est souvent notée :

$$T = M \otimes N$$

Or, $M \in \mathcal{M}(\mathcal{E})$ et $N \in \mathcal{M}(\mathcal{F})$, donc $M \otimes N \in \mathcal{M}(\mathcal{M}(\mathcal{E}), \mathcal{M}(\mathcal{F}))$, et est une matrice à q^2 lignes et p^2 colonnes. L'isomorphisme σ entre $\mathcal{M}(\mathcal{M}(\mathcal{E}, \mathcal{F}))$ et $\mathcal{M}(\mathcal{M}(\mathcal{E}), \mathcal{M}(\mathcal{F}))$ que sous entend la notation précédente est une permutation, comme on le vérifie aisément sur une base quelconque de ces espaces. Pour éviter toute ambiguïté, nous noterons T_{MN} la matrice T telle que :

$$T_{MN} \cdot (x \otimes y) = (M \cdot x) \otimes (N \cdot y)$$

2. DUALITE DANS L'ESPACE $\mathcal{M}(\mathcal{E}, \mathcal{F})$

□ Soit X et Y deux matrices de $\mathcal{M}(\mathcal{E}, \mathcal{F})$ muni du produit scalaire défini par la matrice unité. Soit T une matrice de $\mathcal{M}(\mathcal{M}(\mathcal{E}, \mathcal{F}))$, c'est à dire matrice d'une application linéaire de $\mathcal{M}(\mathcal{E}, \mathcal{F})$ dans lui même. Alors, il existe une matrice T' de $\mathcal{M}(\mathcal{M}(\mathcal{E}, \mathcal{F}))$ telle que :

$$\forall X, \forall Y, \quad \langle X, T \cdot Y \rangle = \langle T' \cdot X, Y \rangle$$

□ Dans le cas particulier où la matrice T est de la forme T_{MN} , la dualité s'écrit, si $y \otimes z$ est une matrice de rang 1 de $\mathcal{M}(\mathcal{E}, \mathcal{F})$:

$$\begin{aligned} \langle X, T \cdot (y \otimes z) \rangle &= \langle X, M \cdot y \otimes N \cdot z \rangle \\ &= \langle X \cdot (M \cdot y), N \cdot z \rangle \\ &= \langle XM \cdot y, N \cdot z \rangle \\ &= \langle N' \cdot XM \cdot y, z \rangle \\ &= \langle N' \cdot XM, y \otimes z \rangle \end{aligned}$$

D'où :

$$T'_{MN} \cdot X = N' \cdot XM$$

□ Si M et N sont les matrices d'un produit scalaire dans \mathcal{E} et \mathcal{F}

respectivement, elles sont symétriques et $M' = M$ d'une part et $N' = N$ d'autre part. D'où, dans ce cas particulier :

$$T'_{MN}.X = NXM$$

3. ANALYSE EN COMPOSANTES PRINCIPALES

1. Définition

□ La recherche des r premières composantes principales d'une matrice A est la recherche de la projection de A sur le cône des matrices de rang r .

Donc, étant donnée une matrice A de $\mathcal{M}(\mathcal{E}, \mathcal{F})$, c'est rechercher la matrice X de $\Gamma^r(\mathcal{E}, \mathcal{F})$ telle que $\|A-X\|$ soit minimale. On notera alors :

$$X = f_r(A)$$

2. Différents cas particuliers

1 - Si la projection de A sur $\Gamma^r(\mathcal{E}, \mathcal{F})$ n'est soumise à aucune contraintes, l'ACP est dite sans contraintes.

2 - Elle est dite avec contraintes dans le cas contraire. Le cas le plus courant est celui où l'on impose aux composantes de $f_r(A)$ d'appartenir à des sous ensembles de \mathcal{E} et \mathcal{F} respectivement. Ces contraintes peuvent être linéaires ou non linéaires, selon que les sous ensembles de \mathcal{E} et \mathcal{F} imposés sont des SEV ou non.

3 - Si les contraintes sont linéaires, on étudiera le cas où les espaces imposés sont les images d'endomorphismes G et H : si x_k sont les composantes principales dans \mathcal{E} et y_k dans \mathcal{F} , $x_k \in \text{Im}(G)$ et $y_k \in \text{Im}(H)$.

4 - Si les contraintes sont non linéaires, on se restreindra au cas des contraintes algébriques. Un cas particulier important sera celui où les contraintes seront définies par des cônes, rencontrés lorsqu'il existe une

relation d'ordre entre les contraintes. Cette approche n'est pas développée dans ce document.

3. Recherche des r premières composantes principales

□ Soient donc :

- A une matrice de $M(\mathcal{E}, \mathcal{F})$,
- T le tenseur d'un produit scalaire sur $M(\mathcal{E}, \mathcal{F})$,
- Ω une variété algébrique de \mathcal{E}
- Ξ une variété algébrique de \mathcal{F} .

On recherche la matrice $X = f_r(A)$ de rang r, écrite sous la forme $\sum_{k=1}^r x_k \otimes y_k$, où $x_k \in \Omega$ et $y_k \in \Xi$, le plus proche de A au sens de la métrique définie par T, soit tel que :

$$d(A, X) = \|A - X\|_T^2$$

soit minimum.

□ Nous étudierons dans la suite les deux cas suivants :

- $\Omega = \mathcal{E}$ et $\Xi = \mathcal{F}$: ACP sans contraintes
- $\Omega = \text{Im}(G)$; $\Xi = \text{Im}(H)$: ACP avec doubles contraintes linéaires

notés respectivement : ACP et ACP-DCL.

Remarquons que l'ACP-DCL correspond exactement à l'ACP sous double contrainte définie par R.SABATIER (SABATIER, 1984) et retrouvée par DENIS (DENIS, 1991) dans le cadre du modèle linéaire avec contraintes.

4. ACP SANS CONTRAINTES

1. Cas de la métrique unité

Dans le cas d'une métrique euclidienne unité, le théorème de SCHMIDT-MIRSKY est une réponse au problème : si la DVS de A est telle que $A'A = \lambda_k u_k$, avec $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p \geq 0$ (si $p = \min(\dim(\mathcal{E}), \dim(\mathcal{F}))$) :

$$f_r(A) = \sum_{k=1}^r u_k \otimes v_k$$

avec $v_k = A.u_k$.

Soit, en normalisant les vecteurs u_k et v_k :

$$f_r(A) = \sum_{k=1}^r \sqrt{\lambda_k} u_k \otimes v_k$$

avec :

$$A' A . u_k = \lambda_k u_k$$

$$A A' . v_k = \lambda_k v_k$$

et :

$$A . u_k = \sqrt{\lambda_k} v_k$$

$$A' . v_k = \sqrt{\lambda_k} u_k$$

2. Cas d'une métrique T_{MN}

Soit donc M une MSDP définissant une métrique sur \mathcal{E} et N une MSDP définissant une métrique sur \mathcal{F} . Ces métriques ne sont pas invariantes par transformation orthogonale, aussi le théorème de SCHMIDT-MIRSKY ne s'applique pas pour la norme T_{MN} . Nous allons cependant nous ramener par des transformations de matrices aux conditions d'applications de ce théorème, et obtiendrons ainsi le résultat.

1 - Supposons que $f_r(A) = \sum_{k=1}^r x_k \otimes y_k$.

2 - Nous recherchons donc (x_k) et (y_k) tels que $\left\| A - \sum_{k=1}^r x_k \otimes y_k \right\|_{T_{MN}}$ soit minimum.

3 - Si A et B sont deux matrices de $\mathcal{M}(\mathcal{E}, \mathcal{F})$, nous savons que :

$$\langle A, B \rangle_{T_{MN}} = \langle A, T_{MN}.B \rangle = \langle A, NBM \rangle$$

4 - M et N étant des MSDP, il existe deux MSDP P et Q telles que $P^2 = M$ et $Q^2 = N$. D'où :

$$\begin{aligned} \langle A, B \rangle_{T_{MN}} &= \langle A, NBM \rangle = \langle A, Q^2 B P^2 \rangle = \langle QA, QBP^2 \rangle = \langle A'Q, P^2 B'Q \rangle \\ &= \langle PA'Q, PB'Q \rangle = \langle QAP, QBP \rangle \end{aligned}$$

5 - En particulier : $\|B\|_{T_{MN}}^2 = \|QBP\|^2$.

6 - Si maintenant nous choisissons $B = A - \sum_{k=1}^r x_k \otimes y_k$, l'égalité devient :

$$\begin{aligned} \left\| A - \sum_{k=1}^r x_k \otimes y_k \right\|_{T_{MN}}^2 &= \left\| Q \left(A - \sum_{k=1}^r x_k \otimes y_k \right) P \right\|^2 = \left\| QAP - Q \left(\sum_{k=1}^r x_k \otimes y_k \right) P \right\|^2 \\ &= \left\| QAP - \sum_{k=1}^r Q(x_k \otimes y_k) P \right\|^2 \end{aligned}$$

7 - Or, $\forall x \in \mathcal{E}, \forall y \in \mathcal{F}, \forall P \in \mathcal{M}(\mathcal{E}), \forall Q \in \mathcal{M}(\mathcal{F})$, nous avons l'égalité suivante : $Q(x \otimes y)P = (P.x) \otimes (Q.y)$

8 - Soit, en l'appliquant au paragraphe précédent :

$$\left\| A - \sum_{k=1}^r x_k \otimes y_k \right\|_{T_{MN}}^2 = \left\| QAP - \sum_{k=1}^r Q(x_k \otimes y_k) P \right\|^2 = \left\| QAP - \sum_{k=1}^r P.x_k \otimes Q.y_k \right\|^2$$

où l'on lit que $\sum_{k=1}^r P x_k \otimes Q y_k = f_r(QAP)$: les vecteurs $P.x_k$ et $Q.y_k$ sont les CP de la matrice QAP.

9 - Ils sont donc solution des équations :

$$\begin{cases} (QAP)'(QAP).(P.x_k) = \mu_k P.x_k \\ (QAP)(QAP)'.(Q.y_k) = \mu_k Q.y_k \end{cases}$$

soit :

$$\begin{cases} PA'Q^2AP.(P.x_k) = \mu_k P.x_k \\ QAP^2A'Q.(Q.y_k) = \mu_k Q.y_k \end{cases}$$

soit, puisque P et Q sont inversibles :

$$\begin{cases} A'Q^2AP.(P.x_k) = \mu_k.x_k \\ AP^2A'Q.(Q.y_k) = \mu_k.y_k \end{cases}$$

que l'on peut encore écrire :

$$\begin{cases} A'Q^2AP^2.x_k = \mu_k.x_k \\ AP^2A'Q^2.y_k = \mu_k.y_k \end{cases}$$

soit, en tenant compte de $P^2 = M$ et $Q^2 = N$:

$$\begin{cases} A'N AM .x_k = \mu_k.x_k \\ AM A'N .y_k = \mu_k.y_k \end{cases}$$

que l'on peut encore écrire :

$$\begin{cases} (NA)'(AM).x_k = \mu_k.x_k \\ (AM)(NA)'.y_k = \mu_k.y_k \end{cases}$$

D'où le résultat ;

□ Etant donnés

- deux espaces vectoriels euclidiens (\mathcal{E}, M) et (\mathcal{F}, N) ,
- une matrice A de $\mathcal{M}(\mathcal{E}, \mathcal{F})$,

les r premières composantes principales de A pour la métrique induite par les métriques M et N est la matrice

$$f_r(A) = \sum_{k=1}^r \mu_k x_k \otimes y_k$$

où x_k et y_k sont vecteurs propres normalisés associés aux r plus grandes valeurs propres des matrices $(NA)'(AM)$ et $(AM)(NA)'$ respectivement :

$$f_r(A) = \sum_{k=1}^r \sqrt{\mu_k} x_k \otimes y_k$$

avec :

$$(NA)'(AM).x_k = \mu_k x_k$$

$$(AM)(NA)'.y_k = \mu_k y_k$$

et :

$$(AM).x_k = \sqrt{\mu_k} y_k$$

$$(NA)'.y_k = \sqrt{\mu_k} x_k$$

□ Remarque

Les composantes principales ainsi calculées forment des systèmes orthogonaux dans \mathcal{E} et \mathcal{F} respectivement munis des métriques M et N .

3. Isométrie définie par une métrique T_{MN}

Il existe une façon plus rapide et très naturelle de retrouver ce résultat à partir de la notion d'isométrie.

■ Définition des isométries

1 - Soient (\mathcal{E}, I) et (\mathcal{F}, I) deux espaces vectoriels munis d'une structure euclidienne définie par la métrique unité I . Soit une autre structure euclidienne sur \mathcal{E} et \mathcal{F} respectivement définie par les MSDP M et N . Notons P et Q les MSDP telles que $P^2 = M$ et $Q^2 = N$. Alors, les applications suivantes sont des isométries :

$$\begin{array}{ccc} (\mathcal{E}, I) & \longrightarrow & (\mathcal{E}, M) \\ x & \longrightarrow & P^{-1} \cdot x \end{array}$$

$$\begin{array}{ccc} (\mathcal{F}, I) & \longrightarrow & (\mathcal{F}, M) \\ y & \longrightarrow & Q^{-1} \cdot y \end{array}$$

$$\begin{array}{ccc} (\mathcal{E} \otimes \mathcal{F}, I \otimes I) & \longrightarrow & (\mathcal{E} \otimes \mathcal{F}, T_{MN}) \\ A & \longrightarrow & Q^{-1} A P^{-1} \end{array}$$

2 - On en déduit une isométrie ι de $\mathbb{R} \oplus \mathcal{E} \oplus \mathcal{F} \oplus \mathcal{E} \otimes \mathcal{F}$ sur $\mathbb{R} \oplus \mathcal{E} \oplus \mathcal{F} \oplus \mathcal{E} \otimes \mathcal{F}$, avec :

$$\begin{aligned} \iota(\alpha) &= \alpha \\ \iota(x) &= P^{-1} \cdot x \\ \iota(y) &= Q^{-1} \cdot y \\ \iota(A) &= QAP \end{aligned}$$

si $\alpha \in \mathbb{R}$, $x \in \mathcal{E}$, $y \in \mathcal{F}$ et $A \in \mathcal{E} \otimes \mathcal{F}$.

3 - En effet, par exemple dans \mathcal{E} en remarquant que $P^{-1} M P^{-1} = I$:

$$\langle \iota(a), \iota(b) \rangle_M = \langle P^{-1} \cdot a, P^{-1} b \rangle_M = \langle P^{-1} \cdot a, M P^{-1} b \rangle = \langle a, P^{-1} M P^{-1} b \rangle = \langle a, b \rangle$$

4 - Le schéma suivant est alors commutatif :

$$\begin{array}{ccc} (\mathcal{E}, I) & \xrightarrow{A} & (\mathcal{F}, I) \\ \downarrow P^{-1} & & \downarrow Q^{-1} \\ (\mathcal{E}, M) & \xrightarrow{Q^{-1} A P^{-1}} & (\mathcal{F}, N) \end{array}$$

Il ne s'agit pas du schéma de dualité, mais ce diagramme en est très proche.

■ ACP avec métriques

1 - Si donc $\mu_k x_k \otimes y_k$ sont les composantes principales de A , $\|A - \sum_k \mu_k x_k \otimes y_k\|^2$ est minimum. Donc, $\|\iota(A - \sum_k \mu_k x_k \otimes y_k)\|_T^2$ est également minimum.

2 - D'où : $\iota(x_k)$ et $\iota(y_k)$ sont les composantes principales de $\iota(A)$ pour la métrique T.

3 - En choisissant $B = \iota^{-1}(A)$, et en réalisant l'ACP de B, on remarque que, si u_k et v_k sont les CP de B, $\iota(u_k)$ et $\iota(v_k)$ sont les CP de $\iota(B) = A$ pour la métrique T. En retournant l'assertion, nous avons le résultats suivant :

4 - Les CP x_k et y_k de A pour la métrique T sont $\iota(u_k)$, $\iota(v_k)$, où u_k et v_k sont les CP de $\iota^{-1}(A)$.

5 - Comme $\iota(A) = Q^{-1}AP^{-1}$, $B = \iota^{-1}(A) = QAP$. De même : $x_k = \iota(u_k) = P^{-1}.u_k$ et $y_k = \iota(v_k) = Q^{-1}.v_k$.

6 - Comme $B'B.u_k = \mu_k u_k$ et $BB'.v_k = \mu_k v_k$, on en déduit :

$$(QAP)'QAP.(P.x_k) = \mu_k P.x_k \Rightarrow A'Q^2AP^2.x_k = \mu_k x_k \Rightarrow A'NAM.x_k = \mu_k x_k$$

et

$$QAP(QAP)'.(Q.y_k) = \mu_k Q.y_k \Rightarrow AP^2A'Q^2.y_k = \mu_k y_k \Rightarrow AMA'N.y_k = \mu_k y_k$$

qui sont bien les résultats cherchés.

7 - Cette approche plus géométrique par isométrie se généralisera facilement à un ordre supérieur.

5. ACP AVEC CONTRAINTES LINEAIRES

1. Notations

Nous supposons les espaces rencontrés euclidiens avec métriques associées à la matrice unité.

On note :

- \mathcal{E} et \mathcal{F} deux EV sur \mathbb{R} de dimensions respectives m et n
- \mathcal{S} un SEV de \mathcal{E} de dimension p,

- \mathcal{H} " " \mathcal{F} " " q ,
- A une application linéaire de $\mathcal{L}(\mathcal{E}, \mathcal{F})$
- P la projection orthogonale dans \mathcal{E} sur \mathcal{F}
- Q " " " " \mathcal{F} " \mathcal{H}
- P^\perp et Q^\perp les projecteurs orthogonaux à P et Q respectivement.
- \mathfrak{p} la projection orthogonale de $\mathcal{E} \otimes \mathcal{F}$ sur $\mathcal{F} \otimes \mathcal{H}$, et \mathfrak{p}^\perp le projecteur orthogonal associé.

2. Rappels sur les projecteurs

1 - \mathcal{F} peut être défini par une base (g_1, \dots, g_p) . Soit G la matrice de $\mathbb{R}^p \otimes \mathcal{E}$ dont les colonnes sont les vecteurs de cette base. Tout vecteur de \mathcal{F} s'écrit sous la forme $G.x$, où $x \in \mathbb{R}^p$, et $\mathcal{F} \equiv \text{Im}(G)$.

2 - On rappelle alors que :

$$P = G(G'G)^{-1}G'$$

3 - Si \mathcal{E} est muni d'une structure euclidienne définie par la MSDP M , on rappelle que :

$$P = G(G'MG)^{-1}(MG)'$$

4 - Si P est la projection dans \mathcal{E} , et Q la projection dans \mathcal{F} , \mathfrak{p} est défini par :

$$\mathfrak{p}(A) = QAP$$

3. Définition

On appelle ACP de A sous double contrainte linéaire (ACP-DCL) définies par \mathcal{F} et \mathcal{H} une décomposition de la forme :

$$A = \sum_{i=1}^r x_i \otimes y_i + R$$

avec :

$$x_i \in \mathcal{G}$$

$$y_i \in \mathcal{H}$$

$\|R\|$ minimale

4. Lemme

Réaliser l'ACP-DCL de A sous les contraintes définies par \mathcal{G} et \mathcal{H} revient à réaliser l'ACP de la matrice $\mathcal{P}(A)$, projection orthogonale de A sur $\mathcal{G} \otimes \mathcal{H}$.

■ Démonstration

1 - Notons \mathcal{P} la projection orthogonale de $\mathcal{E} \otimes \mathcal{F}$ sur $\mathcal{G} \otimes \mathcal{H}$, et \mathcal{P}^\perp le projecteur orthogonal à \mathcal{P} (i.e. sur $(\mathcal{G} \otimes \mathcal{H})^\perp$).

2 - Notons, pour simplifier : $\Sigma_i = \sum_1 x_i \otimes y_i$. Ainsi :

$$\begin{aligned} \|A - \Sigma_i\|^2 &= \|(\mathcal{P} + \mathcal{P}^\perp)(A - \Sigma_i)\|^2 \\ &= \|\mathcal{P}(A - \Sigma_i)\|^2 + \|\mathcal{P}^\perp(A - \Sigma_i)\|^2 \text{ car } \mathcal{P} \text{ et } \mathcal{P}^\perp \text{ sont orthogonaux} \\ &= \|\mathcal{P}(A) - \Sigma_i\|^2 + \|\mathcal{P}^\perp(A)\|^2 \text{ car } \mathcal{P}(\Sigma_i) = \Sigma_i \text{ et } \mathcal{P}^\perp(\Sigma_i) = 0 \text{ car } \Sigma_i \in \mathcal{G} \otimes \mathcal{H} \end{aligned}$$

3 - Ainsi, $\|A - \Sigma_i\|^2$ est minimale lorsque $\|\mathcal{P}(A) - \Sigma_i\|$ l'est.

■ Remarque

Ce résultat, avec la même démonstration, se généralisera aisément aux tableaux à plus de deux entrées, que ce soit pour les modèles TUCKER ou PARAFAC/CANDECOMP.

■ Résultat

□ Etant donnés

- deux espaces vectoriels euclidiens \mathcal{E} et \mathcal{F}

- une matrice A de $\mathcal{M}(\mathcal{E}, \mathcal{F})$,
 - deux SEV \mathcal{E} et \mathcal{H} de \mathcal{E} et \mathcal{F} respectivement
- P la projection orthogonale de \mathcal{E} sur \mathcal{E}
- Q la projection orthogonale de \mathcal{F} sur \mathcal{H}

les r premières composantes principales de A sous la double contrainte linéaire définie par \mathcal{E} et \mathcal{H} est la matrice

$$\tilde{f}_r(A) = \sum_{k=1}^r \mu_k x_k \otimes y_k$$

où x_k et y_k sont vecteurs propres normalisés associés aux r plus grandes valeurs propres des matrices $PA'QAP$ et $QAPA'Q$ respectivement :

$$\tilde{f}_r(A) = \sum_{k=1}^r \sqrt{\mu_k} x_k \otimes y_k$$

avec :

$$PA'QAP \cdot x_k = \mu_k x_k$$

$$QAPA'Q \cdot y_k = \mu_k y_k$$

et :

$$QAP \cdot x_k = \sqrt{\mu_k} y_k$$

$$PA'Q \cdot y_k = \sqrt{\mu_k} x_k$$

5. Application

1 - Si \mathcal{E} est défini par une base (g_1, \dots, g_p) , et \mathcal{H} par une base (h_1, \dots, h_q) , il est facile d'écrire :

- $P = G(G'G^{-1})G'$
- $Q = H(H'H)^{-1}H'$
- $\mathfrak{p}(A) = QAP = H(H'H)^{-1}H'AG(G'G^{-1})G'$

2 - Réaliser l'ACP-DCL de A sous ces contraintes revient donc à réaliser l'ACP de $B = H(H'H)^{-1}H'AG(G'G^{-1})G'$, soit à calculer les valeurs et vecteurs propres de $B'B = (QAP)'(QAP) = (PA'Q)(QAP) = PA'QAP$, soit :

$$B'B = G(G'G^{-1})G'A'H(H'H)^{-1}H'AG(G'G^{-1})G'$$

6. ACP AVEC CONTRAINTES LINEAIRES : ACP-DCL

Il est maintenant facile de rassembler ces différents résultats pour proposer l'ACP-DCL avec des métriques définies par des MSDP M et N sur \mathcal{E} et \mathcal{F} respectivement.

1. Notations

Les notations sont les mêmes qu'en 5.1, les espaces \mathcal{E} et \mathcal{F} étant munis d'une métrique définie par les MSDP M et N respectivement. On rappelle qu'on associe canoniquement à (M,N) une métrique T sur $\mathcal{E} \otimes \mathcal{F}$ définie par :

$$\langle A, x \otimes y \rangle_T = \langle A, (M.x) \otimes (N.y) \rangle$$

2. Définition

Réaliser l'ACP-DCL de A avec les métriques M et N est rechercher une décomposition :

$$A = \sum_{i=1}^r \lambda_i x_i \otimes y_i + R$$

avec

$$x_i \in \mathcal{E} ; y_i \in \mathcal{F}$$

$$\|x_i\|_M = \|y_i\|_N = 1$$

$$\|R\|_T \text{ minimale}$$

3. Lemme

Si :

- P désigne la projection orthogonale sur \mathcal{E} dans \mathcal{E} relativement à la métrique définie par M,
- Q la projection orthogonale sur \mathcal{F} dans \mathcal{F} relativement à la métrique

définie par N

◦ \mathfrak{P} la projection orthogonale sur $\mathcal{G} \otimes \mathcal{H}$ dans $\mathcal{E} \otimes \mathcal{F}$ relativement à la métrique définie par T,

alors,

réaliser l'ACP-DCL de A sous les contraintes définies par \mathcal{G} et \mathcal{H} avec les métriques définies par M et N revient à réaliser l'ACP de $\mathfrak{P}(A)$ avec les métriques définies par M et N.

la démonstration est calquée sur celle du 5.4

4. Résultat

1 - Ainsi, $\mathfrak{P}(A) = B = QAP$. Réaliser l'ACP de B avec les métriques M et N revient à calculer les vecteurs et valeurs propres de $C = B'NBM$.

2 - En reprenant les notations du 5 :

$$\circ P = G(G'MG)^{-1}(MG)'$$

$$\circ Q = H(H'NH)^{-1}(NH)'$$

3 - D'où :

$$C = (QAP)'N(QAP)M$$

$$= PA'QNQAPM$$

$$= G(G'MG)^{-1}(MG)'A'H(H'NH)^{-1}(NH)'NH(H'NH)^{-1}(NH)'AG(G'MG)^{-1}(MG)'M$$

7. VERS LE MODELE DE TUCKER

1 - Dans l'ACP, l'inconnue est une matrice de $\mathcal{M}(\mathcal{E}, \mathcal{F})$, ou un opérateur de élément de $\mathcal{E} \otimes \mathcal{F}$. Comme les opérateurs cherchés sont de rang inférieur à l'opérateur approché, on a $f_r(A) \in \mathcal{M}(\mathcal{G}, \mathcal{H})$, où \mathcal{G} est un SEV de \mathcal{E} et \mathcal{H} un SEV

de \mathcal{F} . Nous avons naturellement $\text{Dim}(\mathcal{G}) = \text{Dim}(\mathcal{H}) = r$.

2 - Posons $\mathcal{G}_r = \text{Im}(f_r(A)')$ et $\mathcal{H}_r = \text{Im}(f_r(A))$. Il est facile de remarquer que $f_r(A)$ est la projection orthogonale de A sur $\mathcal{M}(\mathcal{G}_r, \mathcal{H}_r)$. On remarque ensuite que, parmi tous les espaces SEV \mathcal{X} de dimension r de \mathcal{E} , et \mathcal{Y} de même dimension r de \mathcal{F} , $(\mathcal{G}_r, \mathcal{H}_r)$ est le couple pour lequel la projection de A sur $\mathcal{M}(\mathcal{X}, \mathcal{Y})$ est la plus proche de A .

3 - Plus précisément, si :

- \mathfrak{X} désigne l'ensemble des SEV de \mathcal{E} de dimension r
- \mathfrak{Y} désigne l'ensemble des SEV de \mathcal{F} de dimension r
- \mathcal{X} un élément de \mathfrak{X} et \mathcal{Y} un élément de \mathfrak{Y}
- $\mathcal{P}_{\mathcal{X}, \mathcal{Y}}$ la projection orthogonale de $\mathcal{M}(\mathcal{E}, \mathcal{F})$ sur $\mathcal{M}(\mathcal{X}, \mathcal{Y})$
- $\mathcal{G}_r \in \mathfrak{X} = \text{Im}(f_r(A)')$ et $\mathcal{H}_r \in \mathfrak{Y} = \text{Im}(f_r(A))$

Alors :

$$\forall (\mathcal{X}, \mathcal{Y}) \in \mathfrak{X} \times \mathfrak{Y}, \quad \|A - \mathcal{P}_{\mathcal{G}_r, \mathcal{H}_r}(A)\| \leq \|A - \mathcal{P}_{\mathcal{X}, \mathcal{Y}}(A)\|$$

4 - On peut ainsi renverser le problème de l'ACP et le poser sous la forme suivante :

Etant donné :

- une application A de $\mathcal{E} \otimes \mathcal{F}$ [ou une matrice A de $\mathcal{M}(\mathcal{E}, \mathcal{F})$]
- un nombre r fixé tel que $r \leq \text{rang}(A)$

Trouver un SEV \mathcal{G} de \mathcal{E} de dimension r et un SEV \mathcal{H} de \mathcal{F} de dimension r tels que $\|A - \mathcal{P}_{\mathcal{G}, \mathcal{H}}(A)\|$ soit minimale.

6 - La solution est alors liée à l'ACP de A :

$$\mathcal{G} = \text{Im}(f_r(A)') \text{ et } \mathcal{H} = \text{Im}(f_r(A))$$

Ces remarques mènent naturellement à la notion de "modèle" de TUCKER :

1. Définition

□ Etant donnés :

- un EV \mathcal{E} de dimension p
- un EV \mathcal{F} de dimension q
- deux entiers fixés $r \leq p$ et $s \leq q$,
- \mathcal{X} l'ensemble des SEV \mathcal{G} de \mathcal{E} de dimension r
- \mathcal{Y} l'ensemble des SEV \mathcal{H} de \mathcal{F} de dimension s

□ on appelle modèle de TUCKER d'ordre (r,s) , et nous noterons $t_{rs}(A)$ la matrice T définie par :

- $T \in M(\mathcal{X}, \mathcal{Y})$
- $X \in \mathcal{X}$
- $Y \in \mathcal{Y}$

telle que $\|A-T\|$ minimale.

2. Remarque

On remarque immédiatement :

$$\tilde{f}_r(A) = t_{rr}(A)$$

CHAPITRE C

APPLICATIONS LINEAIRES ASSOCIEES

Ce chapitre est en fait une présentation et une extension de la méthode de l'analyse 3-modes.

L'analyse 3-modes est plus connue en France sous le nom de méthode STATIS (pour Structuration des Tableaux A Trois Indices de la Statistique) développée depuis 1976, qui, avec les modèles de TUCKER et le modèle CANDECOMP / PARAFAC, tous deux d'origine anglo-saxonne, fait partie des grandes familles de méthode permettant d'aborder l'étude des tableaux à plus de deux entrées.

L'élaboration de la méthode STATIS a débuté par l'introduction de variables vectorielles, et non plus scalaires, dans les ACP (ESCOUFIER, 1973), puis l'exploitation de cette technique pour l'analyse des suites de tableaux (X_i) de variables sur les mêmes individus (L'HERMIER DES PLANTES, 1976 ; PAGES, ESCOUFIER & CAZES, 1976). La méthode a été rendue accessible à un large public par l'ouvrage de LAVIT (1988).

Le compromis, qui est le coeur de la méthode STATIS, est la première CP d'une application linéaire construite à partir d'un tableau à trois entrées. Après avoir rappelé les différentes applications linéaires que l'on peut construire à partir d'un tableau à trois entrées, l'extension proposée suivra alors quatre directions naturelles :

- utiliser l'ensemble de l'ACP (ne pas se cantonner au seul compromis)
- introduire dans STATIS des métriques arbitraires
- introduire dans STATIS des contraintes linéaires
- étendre ces techniques aux tableaux à plus de trois entrées.

1. ESPACES CONSIDÉRÉS ET NOTATIONS

□ Rappelons que l'on note :

- $\mathcal{L}(\mathcal{E}, \mathcal{F})$ l'espace vectoriel des applications linéaires de \mathcal{E} dans \mathcal{F} ,
- $\mathcal{M}(\mathcal{E}, \mathcal{F})$ l'espace des matrices des opérateurs de $\mathcal{L}(\mathcal{E}, \mathcal{F})$
- $\mathcal{E} \otimes \mathcal{F}$ l'espace des formes bilinéaires sur $\mathcal{E} \times \mathcal{F}$,

□ Il existe un isomorphisme canonique d'espaces vectoriels entre ces trois espaces :

$$\mathcal{L}(\mathcal{E}, \mathcal{F}) \simeq \mathcal{M}(\mathcal{E}, \mathcal{F}) \simeq \mathcal{E} \otimes \mathcal{F}$$

□ Rappelons également que, comme nous avons identifié tout espace vectoriel avec son dual une fois pour toutes, nous identifions également, pour tout opérateur L de $\mathcal{L}(\mathcal{E}, \mathcal{F})$ l'opérateur transposé et l'opérateur adjoint de L , tous deux représentés par la matrice transposée de celle de L .

□ La transposée d'une application linéaire U se note U' :

$$\langle U(x), y \rangle = \langle x, U'(y) \rangle$$

□ Soient \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 trois espaces vectoriels réels euclidiens, de dimensions respectives n_1, n_2, n_3 et munis de bases orthonormées respectives (e_{1i}) , (e_{2j}) et (e_{3k}) .

□ On considère un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$:

$$A = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \sum_{k=1}^{n_3} \alpha_{ijk} e_{1i} \otimes e_{2j} \otimes e_{3k}.$$

que l'on écrit pour simplifier lorsque aucune ambiguïté n'existe sur la variation des indices :

$$A = \sum_{ijk} \alpha_{ijk} e_{1i} \otimes e_{2j} \otimes e_{3k} = \sum_{ijk} \alpha_{ijk} E_{ijk} = (\alpha_{ijk})$$

2. APPLICATIONS LINÉAIRES ASSOCIÉES

1. RAPPEL

On rappelle le lemme suivant de l'algèbre trilinéaire : étant donnés trois espaces vectoriels \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 , il y a isomorphisme entre :

- $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$
- $\mathcal{L}(\mathcal{E}_1, \mathcal{L}(\mathcal{E}_2, \mathcal{E}_3))$
- $\mathcal{L}(\mathcal{L}(\mathcal{E}_1, \mathcal{E}_2), \mathcal{E}_3)$

que l'on peut traduire par :

$$\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3 \simeq \mathcal{E}_1 \otimes (\mathcal{E}_2 \otimes \mathcal{E}_3) \simeq (\mathcal{E}_1 \otimes \mathcal{E}_2) \otimes \mathcal{E}_3$$

2. APPLICATIONS LINEAIRES ASSOCIEES

1. Définition

□ Soit A un tenseur de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$. On y associe les deux applications linéaires U_A et V_A définies par $U_A(x) = A.x$ et $V_A(M) = A.M$:

$$\begin{array}{ccc} U_A : \mathcal{E}_1 & \longrightarrow & \mathcal{E}_2 \otimes \mathcal{E}_3 \\ x & \longmapsto & A.x \end{array}$$

et :

$$\begin{array}{ccc} V_A : \mathcal{E}_1 \otimes \mathcal{E}_2 & \longrightarrow & \mathcal{E}_3 \\ M & \longmapsto & A.M \end{array}$$

□ Ces deux applications sont dites applications linéaires associées au tenseur A (ALA).

2. Ecriture matricielle de U_A

1 - Pour mettre en évidence l'écriture matricielle de U_A , écrivons :

$$A = \sum_{i,j,k} \alpha_{ijk} E_{ijk} = \sum_{i,j,k} \alpha_{ijk} e_{1i} \otimes e_{2j} \otimes e_{3k}$$

2 - En décomposant A sur la BON (e_{1i}) de \mathcal{E}_1 , il découle :

$$A = \sum_i e_{1i} \otimes A \cdot e_{1i} = \sum_i e_{1i} \otimes A_i \text{ avec } A_i = \sum_{jk} \alpha_{ijk} e_{2j} \otimes e_{3k}$$

qui est l'écriture sous forme d'une matrice M_A de $M(\mathcal{E}_1, \mathcal{L}(\mathcal{E}_2 \times \mathcal{E}_3))$ de l'application $\mathcal{L}(\mathcal{E}_1, \mathcal{L}(\mathcal{E}_2 \times \mathcal{E}_3))$, où la $i^{\text{ème}}$ colonne de M est le vecteur A_i .

3 - Cette matrice a donc $n_2 \cdot n_3$ ligne et n_1 colonnes, et est obtenue à partir du parallélépipède (α_{ijk}) en :

- découpant le parallélépipède A en n_1 tranches qui sont les matrices A_i à n_3 lignes et n_2 colonnes ;
- transformant les matrices-tranche en vecteurs à $n_2 n_3$ coordonnées. Il n'y a aucun algorithme canonique pour réaliser cette opération, qui consiste simplement à écrire les "vecteurs" [vecteurs car éléments de l'espace $\mathcal{L}(\mathcal{E}_2, \mathcal{E}_3)] e_{2j} \otimes e_{3k}$ dans un ordre convenu ;
- créant la matrice de $\mathcal{L}(\mathcal{E}_1, \mathcal{L}(\mathcal{E}_2 \times \mathcal{E}_3))$ qui a pour $i^{\text{ème}}$ vecteur colonne le $i^{\text{ème}}$ vecteur précédent.

4 - On peut remarquer que, dans cette opération, seule l'arête 1, indicée par i , joue un rôle particulier. Les arêtes 2 et 3, caractérisées par j et k , jouent un rôle symétrique, et le choix des arêtes ne joue que sur l'ordre des lignes de la matrice de $\mathcal{L}(\mathcal{E}_1, \mathcal{L}(\mathcal{E}_2 \times \mathcal{E}_3))$.

3. Ecriture matricielle de V_A

1 - L'écriture matricielle de V_A s'obtient par le même procédé.

2 - $V_A \in \mathcal{L}(\mathcal{L}(\mathcal{E}_1 \times \mathcal{E}_2), \mathcal{E}_3)$ et est représenté par une matrice N_A de $M(\mathcal{L}(\mathcal{E}_1 \times \mathcal{E}_2), \mathcal{E}_3) \equiv M(M(\mathcal{E}_1, \mathcal{E}_2), \mathcal{E}_3)$.

3 - Décomposons A sur la base $(e_{1i} \otimes e_{2j})$ de $\mathcal{L}(\mathcal{E}_1 \times \mathcal{E}_2)$:

$$A = \sum_{ij} e_{1i} \otimes e_{2j} \otimes A \cdot (e_{1i} \otimes e_{2j}) = \sum_{ij} e_{1i} \otimes e_{2j} \otimes a_{ij}$$

$$\text{avec } a_{ij} = A \cdot (e_{1i} \otimes e_{2j}) = \sum_k \alpha_{ijk} e_{3k}$$

4 - N_A a donc $n_1 n_2$ colonnes et n_3 lignes, et la colonne indicée par (i, j)

est a_{ij} . Comme dans le cas précédent pour les lignes, l'ordre des colonnes de N_A n'est pas canonique, mais dépend d'une convention choisie pour réordonner les "vecteurs" $e_{1i} \otimes e_{2j}$. Seule l'arête 3 indicée par k joue un rôle particulier, les deux arêtes 1 et 2 indicées par i et j jouant un rôle symétrique.

4. Matrices associées

A tout tenseur A , on associe donc deux matrices M et N telles que :

$\forall (x, y, z) \in \mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3, \langle A, x \otimes y \otimes z \rangle = \langle M.x, y \otimes z \rangle = \langle V.(x \otimes y), z \rangle$
soit :

$\begin{aligned} M.x &= A.x \\ N.(y \otimes z) &= A.(y \otimes z) \end{aligned}$
--

3. APPLICATIONS LINEAIRES ASSOCIEES ET PERMUTATIONS DES INDICES

1. Construction

En tenant compte des permutations σ de \mathcal{E}_3 et de leur action sur A , on peut associer à A douze applications linéaires, dites applications linéaires associées avec permutation (ALAAP) comme suit :

□ On considère :

- les six permutations σ_i de $\{1, 2, 3\}$
- les six triplets $\mathcal{E}_{\sigma_i(1)}, \mathcal{E}_{\sigma_i(2)}, \mathcal{E}_{\sigma_i(3)}$
- les deux fois six applications linéaires :

$$\begin{array}{ccc} \sigma_i U_A : \mathcal{E}_{\sigma_i(1)} & \longrightarrow & \mathcal{L}(\mathcal{E}_{\sigma_i(2)}, \mathcal{E}_{\sigma_i(3)}) \\ x & \longmapsto & \sigma_i A.x \end{array}$$

$$\begin{array}{ccc} \sigma_1 V_A : \mathcal{L}(\mathcal{E}_{\sigma_1(1)}, \mathcal{E}_{\sigma_1(2)}) & \longrightarrow & \mathcal{E}_{\sigma_1(3)} \\ M & \longmapsto & \sigma_1 A. M \end{array}$$

Ainsi, par abus de notation, on note $\sigma_1 U_A$ l'ALA à $\sigma_1 A$ (au lieu de $U_{\sigma_1 A}$, qui serait une notation plus lourde).

2. Exemple

Si l'on note :

σ	$\sigma(1,2,3)$
0	1,2,3
1	1,3,2
2	2,3,1
3	2,1,3
4	3,1,2
5	3,2,1

il en résulte, par exemple pour $i = 2$:

$$\bullet \mathcal{E}_{\sigma_2(1)} = \mathcal{E}_2 \quad ; \quad \mathcal{E}_{\sigma_2(2)} = \mathcal{E}_3 \quad ; \quad \mathcal{E}_{\sigma_2(3)} = \mathcal{E}_1$$

$$\bullet \sigma_2 V_A(M) \in \mathcal{L}(\mathcal{L}(\mathcal{E}_2, \mathcal{E}_3), \mathcal{E}_1)$$

$$\bullet \text{ Si } M \in \mathcal{M}(\mathcal{E}_2, \mathcal{E}_3), \sigma_2 V_A(M) = \sigma_2 A. M = y \in \mathcal{E}_1 \text{ avec } \sigma_2 A = \sum_{ijk} \alpha_{ijk} e_{2j} \otimes e_{3k} \otimes e_{1i}$$

• La matrice associée à $\sigma_2 V_A$ est donc une matrice de $\mathcal{M}(\mathcal{L}(\mathcal{E}_2, \mathcal{E}_3), \mathcal{E}_1)$, à n_1 colonnes et $n_2 n_3$ lignes.

3. Relations entre les ALAAP

□ Ces applications ALAAP sont deux à deux transposées l'une de l'autre.

□ Rappelons que l'on note U'_A la transposée de U_A :

$$\forall x \in \mathcal{E}_1, \forall M \in \mathcal{M}(\mathcal{E}_2, \mathcal{E}_3), \langle U_A(x), M \rangle = \langle x, U'_A(M) \rangle$$

□ Choisisant $x = e_{11}$, $M = e_{2j} \otimes e_{3k}$, il en découle :

$$\begin{aligned} \langle U_A(e_{11}), e_{2j} \otimes e_{3k} \rangle &= \langle A \cdot e_{11}, e_{2j} \otimes e_{3k} \rangle \\ &= \langle A, e_{11} \otimes e_{2j} \otimes e_{3k} \rangle \\ &= \langle \sigma_2 A, e_{2j} \otimes e_{3k} \otimes e_{11} \rangle \\ &= \langle \sigma_2 A \cdot (e_{2j} \otimes e_{3k}), e_{11} \rangle \\ &= \langle V_{\sigma_2 A} \cdot (e_{2j} \otimes e_{3k}), e_{11} \rangle \end{aligned}$$

D'où : $U'_A = \sigma_2 V_A$

Le tableau complet des transpositions est donné ci-après (en omettant le σ :

$\sigma_1 U_A \equiv {}_1U_A$) :

i	$\sigma_i(1,2,3)$	U'_A	V'_A
0	1,2,3	$2V_A$	$4U_A$
1	1,3,2	$5V_A$	$3U_A$
2	2,3,1	$4V_A$	U_A
3	2,1,3	$1V_A$	$1U_A$
4	3,1,2	V_A	$2U_A$
5	3,2,1	$3V_A$	$3U_A$

4. ACP SUR LES APPLICATIONS LINEAIRES ASSOCIEES AVEC PERMUTATIONS

Les applications linéaires associées peuvent être décomposées par analyse en composantes principales, comme toute application linéaire. La question étudiée ici est double :

- que signifie réaliser une ACP sur les ALA ?
- quels liens existe-t-il entre les douze ACP possibles ?

1. ACP sur les ALA

Rappelons que, si U est une application linéaire, ou une matrice, on note par $f_r(U)$ la meilleure approximation de rang r de U . Nous allons préciser $f_r(U_A)$.

1 - $f_r(U_A)$ est la projection de U_A sur les applications linéaires de rang r de $\mathcal{L}(\mathcal{E}_1, \mathcal{L}(\mathcal{E}_2, \mathcal{E}_3))$.

2 - Les r PCP sont les triplets (λ_k, u_k, U_k) tels que :

$$f_r(U_A) = \sum_{k=1}^r \sqrt{\lambda_k} u_k \otimes U_k$$

3 - Après normalisation, λ_k, u_k et U_k sont fournis par la DVS de U_A , soit :

- les λ_k sont les premières valeurs propres de $U'_A U_A$
- les u_k sont les premiers vecteurs propres de $U'_A U_A$
- les U_k sont les premiers vecteurs propres de $U_A U'_A$.

4 - Or $A = \sum_i e_{1i} \otimes A_i \Rightarrow U_A(x) = A \cdot x = \sum_i \xi_i A_i$ si $x = \sum_i \xi_i e_{1i}$.

D'où : $U'_A U_A = (\sum_i A_i \otimes e_{1i}) \circ (\sum_j e_{1j} \otimes A_j) = \sum_{ij} \langle A_i, A_j \rangle e_{1i} \otimes e_{1j}$

5 - Or : $A_i = \sum_{kl} \alpha_{ikl} e_{2k} \otimes e_{3l} \Rightarrow \mu_{ij} = \langle A_i, A_j \rangle = \sum_{kl} \alpha_{ikl} \alpha_{jkl}$.

6 - La matrice à diagonaliser est donc une matrice de $\mathcal{M}(\mathcal{E}_1)$, carrée d'ordre n_1 , de coefficient $\mu_{ij} = \langle A_i, A_j \rangle = \sum_{kl} \alpha_{ikl} \alpha_{jkl}$.

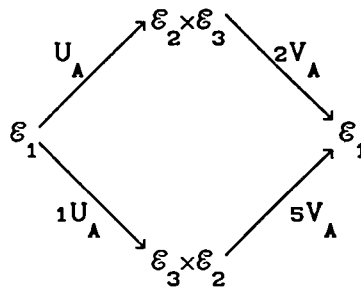
2. ACP indépendantes

□ Les douze ALA à un tenseur d'ordre trois étant deux à deux transposées l'une de l'autre, on se ramène aisément à six ACP différentes sur ces ALA,

en considérant que réaliser l'ACP de la matrice M est réaliser la DVS de M . Or, la DVS de M et celle de M' sont reliées par : $M = U'DV \Rightarrow M' = V'DU$.

□ On peut se ramener à trois ACP différentes en montrant que ces six ACP sont deux à deux confondues. En effet :

1 - Les ALAAP U_A , $1U_A$, $2V_A$ et $5V_A$ sont caractérisées par le schéma :



2 - Les deux endomorphismes $2V_A \circ U_A$ et $5V_A \circ 1U_A$ sont confondus, car les matrices associées à U_A et $1U_A$ ne diffèrent que par l'ordre des lignes, et $2V_A = U'_A$ d'une part et $5V_A = 1U'_A$ d'autre part.

8 - On montrerait de même que les ACP de $2U_A$ et $3U_A$ d'une part et de $4U_A$ et $5U_A$ d'autre part sont confondues. On est donc rammené à trois ACP différentes pour les douze ACP sur les douze ALAAP à un tenseur d'ordre trois.

3. Expression analytique

□ On peut expliciter l'ACP de l'exemple U_A précédent. La matrice M dont on réalise l'ACP est constituée de n_1 colonnes, chaque colonne ayant pour coordonnées les valeurs α_{ijk} , i fixe, j et k variant entre 1 et n_2 et n_3 respectivement. La matrice $M'M$ aura alors pour coordonnées :

$$\mu_{ij} = \sum_{mn} \alpha_{imn} \alpha_{jmn}$$

□ On calcule de même les coefficients des matrices à diagonaliser des autres ACP indépendantes.

□ D'où le tableau des résultats suivants , où :

- "ALAAP" désigne les applications linéaires associées dont l'ACP est confondue,
- "Matrice" désigne la matrice à diagonaliser pour réaliser l'ACP,
- " μ_{ij} " désigne le coefficient de ligne i et de colonne j de la matrice,
- "Ordre" désigne la dimension de la matrice à diagonaliser.

ALAAP	Matrice	μ_{ij}	Ordre
$U_A, {}^1U_A, {}^2V_A, {}^5V_A$	${}^2V_A \circ U_A, {}^5V_A \circ {}^1U_A$	$\sum_{kl} \alpha_{ikl} \alpha_{jkl}$	(n_1, n_1)
${}^2U_A, {}^3U_A, {}^4V_A, {}^1V_A$	${}^4V_A \circ {}^2U_A, {}^1V_A \circ {}^3U_A$	$\sum_{kl} \alpha_{kil} \alpha_{kjl}$	(n_2, n_2)
${}^4U_A, {}^5U_A, V_A, {}^3V_A$	$V_A \circ {}^4U_A, {}^3V_A \circ {}^5U_A$	$\sum_{kl} \alpha_{kli} \alpha_{klj}$	(n_3, n_3)

3. ACP SUR LES ALAAP AVEC MÉTRIQUE NON TRIVIALE

1. RAPPELS ET NOTATIONS

1 - Supposons que \mathcal{E}_1 soit muni d'une structure euclidienne définie par la MSDP M_1 , \mathcal{E}_2 par M_2 et \mathcal{E}_3 par M_3 .

2 - M_i est une MSDP de $\mathcal{M}(\mathcal{E}_i)$ de coefficients μ_{ipq} , pour $i = 1, 2, 3$.

3 - Nous reprenons alors les conventions et notations sur les espaces vectoriels définies au chapitre précédent.

2. ACP DE U_A

Pour introduire les ACP sur ALAAP avec métrique, nous allons traiter le cas de l'ALA U_A .

1. Lemme

Le calcul de l'ACP sur l'ALA U_A avec les métriques définies par M_1 , M_2 et M_3 est alors simple.

■ Enoncé

Soit A un tenseur de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$. Réaliser l'ACP de U_A avec les métriques définies par M_1 , M_2 et M_3 respectivement se ramène à diagonaliser la matrice B de $\mathcal{M}(\mathcal{E}_1, \mathcal{E}_1)$ de coefficients $\beta_{ip} = \sum_{jks tu} \alpha_{ijk} \alpha_{stu} \mu_{1ps} \mu_{2jt} \mu_{3ku}$

■ Démonstration

1 - Soit $A \in \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$. Alors : $U_A \in \mathcal{L}(\mathcal{E}_1, \mathcal{E}_2 \otimes \mathcal{E}_3)$. Rappelons qu' U_A est caractérisée par : $U_A \cdot x = A \cdot x$.

2 - \mathcal{E}_1 est muni d'une métrique définie par la MSDP M_1

3 - $\mathcal{E}_2 \otimes \mathcal{E}_3$ est muni d'une métrique définie par la MSDP $T_{M_2 M_3}$ de $\mathcal{M}(\mathcal{E}_2 \otimes \mathcal{E}_3)$

4 - L'ACP de U_A est alors simplement :

$$f_r(U_A) = \sum_{k=1}^r \sqrt{\mu_k} x_k \otimes Y_k$$

avec :

$$x_k \in \mathcal{E}_1 ; \|x_k\| = 1$$

$$Y_k \in \mathcal{E}_2 \otimes \mathcal{E}_3 ; \|Y_k\| = 1$$

$$(T_{M_2 M_3} \circ U_A)'(U_A \circ M_1).x_k = \mu_k Y_k$$

5 - On peut expliciter la matrice $B = (T_{M_2 M_3} \circ U_A)'(U_A \circ M_1)$, comme suit :

$$U_A = \sum_{ijk} \alpha_{ijk} e_{1i} \otimes (e_{2j} \otimes e_{3k})$$

$$M_1 = \sum_{pq} \mu_{1pq} e_{1p} \otimes e_{1q}$$

D'où :

$$\begin{aligned} U_A \circ M_1 &= \sum_{ijk} \alpha_{ijk} e_{1i} \otimes (e_{2j} \otimes e_{3k}) \circ \sum_{pq} \mu_{1pq} e_{1p} \otimes e_{1q} \\ &= \sum_{ijkpq} \alpha_{ijk} \mu_{1pq} [e_{1i} \otimes (e_{2j} \otimes e_{3k})] \circ (e_{1p} \otimes e_{1q}) \\ &= \sum_{ijkpq} \alpha_{ijk} \mu_{1pq} \langle e_{1i}, e_{1q} \rangle e_{1p} \otimes (e_{2j} \otimes e_{3k}) \\ &= \sum_{ijkp} \alpha_{ijk} \mu_{1pi} e_{1p} \otimes (e_{2j} \otimes e_{3k}) \end{aligned}$$

7 - De même :

$$T_{M_2 M_3} = \sum_{rstu} \tau_{rstu} e_{2r} \otimes e_{3s} \otimes e_{2t} \otimes e_{3u}$$

$$\begin{aligned} \text{avec } \tau_{rstu} &= \langle e_{2r} \otimes e_{3s}, e_{2t} \otimes e_{3u} \rangle_{T_{M_2 M_3}} \\ &= \langle e_{2r} \otimes e_{3s}, (M_2 \cdot e_{2t}) \otimes (M_3 \cdot e_{3u}) \rangle \\ &= \langle e_{2r} \otimes e_{3s}, (\sum_v \mu_{2tv} e_{2v}) \otimes (\sum_w \mu_{3uw} e_{3w}) \rangle \\ &= \mu_{2rt} \mu_{3su} \end{aligned}$$

D'où :

$$\begin{aligned}
T_{M2M3} \circ U_A &= \left(\sum_{rstu} \tau_{rstu} (e_{2r} \otimes e_{3s}) \otimes (e_{2t} \otimes e_{3u}) \right) \circ \left(\sum_{ijk} \alpha_{ijk} e_{1i} \otimes (e_{2j} \otimes e_{3k}) \right) \\
&= \sum_{ijk rstu} \alpha_{ijk} \tau_{rstu} \left((e_{2r} \otimes e_{3s}) \otimes (e_{2t} \otimes e_{3u}) \right) \circ (e_{1i} \otimes (e_{2j} \otimes e_{3k})) \\
&= \sum_{ijk rstu} \alpha_{ijk} \tau_{rstu} \langle e_{2r} \otimes e_{3s}, e_{2j} \otimes e_{3k} \rangle e_{1i} \otimes (e_{2t} \otimes e_{3u}) \\
&= \sum_{ijk tu} \alpha_{ijk} \tau_{jktu} e_{1i} \otimes (e_{2t} \otimes e_{3u})
\end{aligned}$$

D'où :

$$(T_{M2M3} \circ U_A)' = \sum_{ijk tu} \alpha_{ijk} \tau_{jktu} (e_{2t} \otimes e_{3u}) \otimes e_{1i}$$

8 - D'où, finalement :

$$B = (T_{M2M3} \circ U_A)' (U_A \circ M_1)$$

$$\begin{aligned}
&= \left(\sum_{ijk tu} \alpha_{ijk} \tau_{jktu} (e_{2t} \otimes e_{3u}) \otimes e_{1i} \right) \circ \left(\sum_{ijkp} \alpha_{ijk} \mu_{1pi} e_{1p} \otimes (e_{2j} \otimes e_{3k}) \right) \\
&= \sum_{ijk tufghp} \alpha_{ijk} \tau_{jktu} \alpha_{fgh} \mu_{1pf} \left((e_{2t} \otimes e_{3u}) \otimes e_{1i} \right) \circ (e_{1p} \otimes (e_{2g} \otimes e_{3h})) \\
&= \sum_{ijk tufghp} \alpha_{ijk} \alpha_{fgh} \tau_{jktu} \mu_{1pf} \langle e_{2t} \otimes e_{3u}, e_{2g} \otimes e_{3h} \rangle e_{1p} \otimes e_{1i} \\
&= \sum_{ijk tufp} \alpha_{ijk} \alpha_{ftu} \tau_{jktu} \mu_{1pf} e_{1p} \otimes e_{1i} \\
&= \sum_{ip} \beta_{ip} e_{1i} \otimes e_{1p}
\end{aligned}$$

avec :

$$\begin{aligned}
\beta_{ip} &= \sum_{jktuf} \alpha_{ijk} \alpha_{ftu} \tau_{jktu} \mu_{1pf} \\
&= \sum_{jkstu} \alpha_{ijk} \alpha_{stu} \mu_{1ps} \mu_{2jt} \mu_{3ku}
\end{aligned}$$

3. GENERALISATION

Le lemme précédent, au vu du tableau 3.4.3 de ce chapitre, se généralise aisément. Tous calculs faits, le tableau des matrices à diagonaliser pour une ACP sur une ALAAP avec métrique est le suivant :

ALAAP	β_{ij}	Ordre
$U_A, {}^1U_A, {}^2V_A, {}^5V_A$	$\sum_{klstu} \alpha_{ikl} \alpha_{stu} \mu_{1js} \mu_{2kt} \mu_{3lu}$	(n_1, n_1)
${}^2U_A, {}^3U_A, {}^4V_A, {}^1V_A$	$\sum_{klstu} \alpha_{kll} \alpha_{stu} \mu_{1ks} \mu_{2jt} \mu_{3lu}$	(n_2, n_2)
${}^4U_A, {}^5U_A, {}^1V_A, {}^3V_A$	$\sum_{klstu} \alpha_{kll} \alpha_{stu} \mu_{1ks} \mu_{2lt} \mu_{3ju}$	(n_3, n_3)

4 - ACP SUR LES ALAAP AVEC CONTRAINTES LINEAIRES

Il est difficile de fournir une expression explicite générale en terme de coordonnées de la matrice à diagonaliser dans l'ACP d'une ALAAP avec contraintes linéaires, car cette écriture est complexe. Une notation synthétique est plus facile à atteindre.

1. Notation et définitions

Soient trois espaces \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 , munis chacun de trois SEV \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 . Soient P_1 , P_2 , P_3 les projections orthogonales de \mathcal{E}_1 sur \mathcal{F}_1 , \mathcal{E}_2 sur \mathcal{F}_2 et \mathcal{E}_3 sur \mathcal{F}_3 .

Alors, réaliser l'ACP de U_A sous les contraintes définies par \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 est trouver la décomposition optimale :

$$\begin{aligned}
 U_A &= \sum_{i=1}^r \lambda_i u_i \otimes U_i \\
 \text{avec :} \\
 u_i &\in \mathcal{F}_1 \\
 U_i &\in \mathcal{F}_2 \otimes \mathcal{F}_3
 \end{aligned}$$

2. Lemme

■ Enoncé

Réaliser l'ACP de l'ALAAP U_A avec la triple contrainte définie par \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 revient à diagonaliser la matrice $P_1 U' (P_2 \hat{\otimes} P_3) U P_1$, où $P_2 \hat{\otimes} P_3$ est la projection orthogonale de $\mathcal{E}_2 \otimes \mathcal{E}_3$ sur $\mathcal{F}_2 \otimes \mathcal{F}_3$.

■ Démonstration

1 - Comme $\mathcal{F}_2 \otimes \mathcal{F}_3$ est un SEV de $\mathcal{E}_2 \otimes \mathcal{E}_3$, l'expression précédente est celle de l'ACP-DCL de U_A définie par les contraintes \mathcal{F}_1 et $\mathcal{F}_2 \otimes \mathcal{F}_3$, dont la solution est connue.

2 - La solution consiste à projeter U_A sur $\mathcal{F}_1 \otimes (\mathcal{F}_2 \otimes \mathcal{F}_3)$, et à en réaliser l'ACP.

3 - On sait que, si A est une matrice de $\mathcal{F}_1 \otimes (\mathcal{F}_2 \otimes \mathcal{F}_3)$, P une projection sur \mathcal{F}_1 et Q une projection sur $\mathcal{F}_2 \otimes \mathcal{F}_3$, la matrice projetée de A sur $\mathcal{F}_1 \otimes (\mathcal{F}_2 \otimes \mathcal{F}_3)$ est QAP .

4 - Or, on connaît $P \equiv P_1$. Si l'on note $P_2 \hat{\otimes} P_3$ la projection sur $\mathcal{F}_2 \otimes \mathcal{F}_3$ (où $\hat{\otimes}$ est apparenté au produit de Kronecker), nous avons alors le résultat suivant : réaliser l'ACP de l'ALAAP U_A sous la triple contrainte définie par \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 se ramène à réaliser l'ACP de la matrice $(P_2 \hat{\otimes} P_3) U P_1$, soit à diagonaliser la matrice $[(P_2 \hat{\otimes} P_3) U P_1]' (P_2 \hat{\otimes} P_3) U P_1 = P_1' U' (P_2 \hat{\otimes} P_3)' (P_2 \hat{\otimes} P_3) U P_1$.

5 - Or, un projecteur P est un opérateur linéaire symétrique tel que $P^2=P$.
La matrice à diagonaliser s'écrit donc plus simplement $P_1 U' (P_2 \hat{\otimes} P_3) U P_1$.

3. Généralisation

Sans détailler l'ensemble des calcul, il est facile d'aboutir au tableau récapitulatif suivant :

ALAAP	Matrice à diagonaliser	Ordre
$U_A, 1U_A, 2V_A, 5V_A$	$P_1 U' (P_2 \hat{\otimes} P_3) U P_1$	(n_1, n_1)
$2U_A, 3U_A, 4V_A, 1V_A$	$P_2 (2U') (P_3 \hat{\otimes} P_1) (2U) P_2$	(n_2, n_2)
$4U_A, 5U_A, V_A, 3V_A$	$P_3 (4U') (P_1 \hat{\otimes} P_2) (4U) P_3$	(n_3, n_3)

5. ALA AUX TENSEURS D'ORDRE P

L'étude des applications linéaires associées des tenseurs d'ordre p est l'analyse p -modes des tableaux à p entrées, et généralise l'analyse 3-modes des tableaux à trois entrées telle qu'elle est définie par exemple dans KROONENBERG, 1983.

1. RAPPELS ET NOTATIONS

1 - Rappelons que l'on note \mathcal{E}_i le i^{eme} espace vectoriel considéré, qu'il est de dimension n_i et muni d'une métrique définie par la MSDP M_i de coefficients μ_{ipq} .

2 - Une base de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ est $E_{1_1 \dots 1_p} = e_{1_1} \otimes \dots \otimes e_{1_p}$. Afin d'alléger les notations, nous noterons également $E_{1 \dots p}$ cette base.

3 - Ainsi, un tenseur A de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_n$ s'écrit :

$$\begin{aligned} \circ A &= \sum_{1_1 \dots 1_p} \alpha_{1_1 \dots 1_p} e_{1_1} \otimes \dots \otimes e_{1_p} \\ \circ A &= \sum_{1_1 \dots 1_p} \alpha_{1_1 \dots 1_p} E_{1_1 \dots 1_p} \\ \circ A &= \sum_{1 \dots p} \alpha_{1 \dots p} E_{1 \dots p} \\ \circ A &= \sum_i \alpha_i \vec{E}_i \end{aligned}$$

Nous choisirons la notation la plus concise en fonction du contexte.

2. APPLICATION LINEAIRE ASSOCIEE

1. Définition

Soit A un tenseur de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$. Rappelons que :

$$\forall i \leq p, \mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p \simeq (\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_i) \otimes (\mathcal{E}_{i+1} \otimes \dots \otimes \mathcal{E}_p)$$

Nous appelons $i^{\text{ème}}$ Application Linéaire Associée à A , et nous notons U_i , l'application linéaire de $\mathcal{L}(\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_i, \mathcal{E}_{i+1} \otimes \dots \otimes \mathcal{E}_p)$ définie par :

$$\forall X \in \mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_i, U_i(X) = A.X$$

Remarquons qu'il y a $p-1$ telles ALA.

2. Ecriture matricielle de U_i

1 - Notons : $A = \sum_{1 \dots p} \alpha_{1 \dots p} E_{1 \dots p}$

2 - Il en découle :

$$A = \sum_{1 \dots i, i+1 \dots p} \alpha_{1 \dots i, i+1 \dots p} E_{1 \dots i} \otimes E_{i+1 \dots p}$$

$$\begin{aligned}
&= \sum_{1 \dots i} E_{1 \dots i} \otimes \left[\sum_{1+1 \dots p} \alpha_{1 \dots p} E_{1+1 \dots p} \right] \\
&= \sum_{1 \dots i} E_{1 \dots i} \otimes A.E_{1 \dots i}
\end{aligned}$$

$$D'o\grave{u} : A.E_{1 \dots i} = \sum_{1+1 \dots p} \alpha_{1 \dots p} E_{1+1 \dots p}$$

3 - Or, $A.E_{1 \dots i} = U_i(E_{1 \dots i})$, et la matrice associée à U_i dans la base $E_{1 \dots i} \otimes E_{1+1 \dots p}$ a pour colonnes les vecteurs $U_i(E_{1 \dots i})$.

4 - La matrice de U_i est donc une matrice à $n_{1+1} \dots n_p$ lignes et $n_1 \dots n_i$ colonnes. Par exemple, si $p = 5$ et $\mathcal{E}_1 = \mathbb{R}^{10}$, la matrice U_2 a 1000 lignes et 10 colonnes. Un calcul sur de telles matrices ne peut être effectué que sur gros systèmes.

4 - Il y a $p-1$ telles matrices.

3. APPLICATIONS LINEAIRES ASSOCIEES AVEC PERMUTATION DES INDICES

1. Construction

La construction est simple, même si les notations ne le sont pas toujours.

1 - On peut effectuer $p!$ permutations dans $\{1, \dots, p\} \subset \mathbb{N}$. C'est le nombre de permutations sur les indices de A .

2 - Si σ_s désigne une de ces $p!$ permutations, on note ${}_s U_i$ la $i^{\text{ème}}$ ALA à $\sigma_s(A)$.

3 - Prenons un exemple dans $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3 \otimes \mathcal{E}_4 \otimes \mathcal{E}_5$.

Si $i = 2$, $U_2 \in \mathcal{L}(\mathcal{E}_1 \otimes \mathcal{E}_2, \mathcal{E}_3 \otimes \mathcal{E}_4 \otimes \mathcal{E}_5)$

Si $\sigma_{32}(1,2,3,4,5) = (2,4,1,5,3)$, alors ${}_{32}U_2 \in \mathcal{L}(\mathcal{E}_2 \otimes \mathcal{E}_4, \mathcal{E}_1 \otimes \mathcal{E}_5 \otimes \mathcal{E}_3)$

4 - Comme il y a $p-1$ ALA, et $p!$ permutations d'indices possibles, il y a $(p-1)p!$ ALAAP définies pour un tenseur A de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$.

*

2. Nombre d'ACP différentes sur les ALAAP

Nous nous intéressons aux ACP différentes que l'on peut réaliser sur ces $(p-1)p!$ ALAAP différentes. Avant de se lancer dans l'étude générale, suivons la démarche pour $p = 4$.

■ Démarche pour $p = 4$

1 - Il y a $3.4! = 72$ ALAAP différentes.

2 - Notons $1\ 2 \rightarrow 3\ 4$ par exemple, l'ALAAP de $\mathcal{L}(\varepsilon_1 \otimes \varepsilon_2, \varepsilon_3 \otimes \varepsilon_4)$.

3 - Les trois ALA U_1 , U_2 et U_3 sont définies par

- $U_1 : 1 \rightarrow 2\ 3\ 4$
- $U_2 : 1\ 2 \rightarrow 3\ 4$
- $U_3 : 1\ 2\ 3 \rightarrow 4$

4 - Les ALAAP se déduisent des U_i en faisant agir le groupe \mathcal{S}_4 , qui possède 24 éléments. Par exemple, en notant $\sigma_5(1,2,3,4) = (2,1,4,3)$, nous avons :

- $sU_1 : 2 \rightarrow 1\ 4\ 3$
- $sU_2 : 2\ 1 \rightarrow 4\ 3$
- $sU_3 : 2\ 1\ 4 \rightarrow 4$

et ainsi de suite pour les 24 permutations.

5 - Soit $\sigma_3(1,2,3,4) = (1,3,2,4)$, et comparons les ACP des ALAAP U_1 et $3U_1$

5.1 - Les matrices M et N de U_1 et $3U_1$ ont toutes deux $n_2 n_3 n_4$ lignes et n_1 colonnes.

5.2 Elles ne diffèrent en fait que par l'ordre des lignes. En effet :

la première colonne de U_1 , par exemple, est le vecteur $A.e_{11}$.

La première colonne de $3U_1$ est le vecteur $3A.e_{11}$.

$$\text{Or : } A.e_{11} = \sum_{jkl} \alpha_{1jkl} E_{jkl},$$

$$\text{et : } 3A.e_{11} = \sum_{ijk} \alpha_{1jkl} E_{kjl}$$

On retrouve bien les mêmes valeurs α_{1jkl} dans un ordre différent (en fait, l'ordre choisi pour ordonner les bases).

5.3 Donc, les matrices $(U_1)'U_1$ et $(3U_1)'(3U_1)$ sont égales, et les ACP de U_1 et $3U_1$ sont donc confondues.

6 - Plus généralement, les ACP de $i \ j \rightarrow k \ l$ sont invariantes par permutation de $\{i,j\}$ ou $\{k,l\}$

7 - Les 14 applications pouvant mener à des ACP différentes sont alors les suivantes (avec entre parenthèses le nombre d'ALAAP menant à une ACP confondue avec celle indiquée) :

1	$\rightarrow 2 \ 3 \ 4$	(6)
2	$\rightarrow 1 \ 3 \ 4$	(6)
3	$\rightarrow 1 \ 2 \ 4$	(6)
4	$\rightarrow 1 \ 2 \ 3$	(6)
1 2	$\rightarrow 3 \ 4$	(4)
1 3	$\rightarrow 2 \ 4$	(4)
1 4	$\rightarrow 2 \ 3$	(4)
2 3	$\rightarrow 1 \ 4$	(4)
2 4	$\rightarrow 1 \ 3$	(4)
3 4	$\rightarrow 1 \ 2$	(4)
1 2 3	$\rightarrow 4$	(6)
1 2 4	$\rightarrow 3$	(6)
1 3 4	$\rightarrow 2$	(6)
2 3 4	$\rightarrow 1$	(6)

Par exemple, les six ALAAP associées à $1 \rightarrow 2 \ 3 \ 4$ sont :

- $1 \rightarrow 2 \ 3 \ 4$
- $1 \rightarrow 2 \ 4 \ 3$
- $1 \rightarrow 3 \ 2 \ 4$

- $1 \rightarrow 3 \ 4 \ 2$
- $1 \rightarrow 4 \ 2 \ 3$
- $1 \rightarrow 4 \ 3 \ 2$

Et les quatre ALAAP associées à $1 \ 2 \rightarrow 3 \ 4$ sont

- $1 \ 2 \rightarrow 3 \ 4$
- $1 \ 2 \rightarrow 4 \ 3$
- $2 \ 1 \rightarrow 3 \ 4$
- $2 \ 1 \rightarrow 4 \ 3$

8 - De plus, les ALAAP

- $1 \rightarrow 2 \ 3 \ 4$
- $2 \ 3 \ 4 \rightarrow 1$

sont transposées l'une de l'autre. Les ACP de ces deux ALAAP sont donc dépendantes l'une de l'autre, puisque les valeurs et vecteurs propres de MM' se déduisent des mêmes grandeurs calculées pour $M'M$.

9 - Il reste donc 7 ALAAP qui mènent à des ACP indépendantes, qui sont (par exemple) :

$1 \rightarrow 2 \ 3 \ 4$	$\mathcal{E}_1 \rightarrow \mathcal{E}_2 \otimes \mathcal{E}_3 \otimes \mathcal{E}_4$
$2 \rightarrow 1 \ 3 \ 4$	$\mathcal{E}_2 \rightarrow \mathcal{E}_1 \otimes \mathcal{E}_3 \otimes \mathcal{E}_4$
$3 \rightarrow 1 \ 2 \ 4$	$\mathcal{E}_3 \rightarrow \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_4$
$4 \rightarrow 1 \ 2 \ 3$	$\mathcal{E}_4 \rightarrow \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$
$1 \ 2 \rightarrow 3 \ 4$	$\mathcal{E}_1 \otimes \mathcal{E}_2 \rightarrow \mathcal{E}_3 \otimes \mathcal{E}_4$
$1 \ 3 \rightarrow 2 \ 4$	$\mathcal{E}_1 \otimes \mathcal{E}_3 \rightarrow \mathcal{E}_2 \otimes \mathcal{E}_4$
$1 \ 4 \rightarrow 2 \ 3$	$\mathcal{E}_1 \otimes \mathcal{E}_4 \rightarrow \mathcal{E}_2 \otimes \mathcal{E}_3$

■ Cas général

La démarche dans le cas général est alors la suivante :

1 - Il y a $(p-1)p!$ ALAAP

2 - Notons

$$x_1, x_2, \dots \rightarrow y_1, \dots$$

1' ALAAP $\mathcal{L}(x_1 \otimes x_2 \otimes \dots, y_1 \otimes \dots)$

3 - les ACP des $x_1, x_2, \dots \rightarrow y_1, \dots$ sont confondues par permutation des x_i ou des y_j .

4 - Il y a $2^p - 2$ ensembles $\{x_1, x_2, \dots\}$ distincts : ce sont les parties de $\{1..p\} \subset \mathbb{N}$ qui n'ont ni 0 ni p éléments.

5 - Ces $2^p - 2$ ALAAP sont deux à deux transposées l'une de l'autre.

6 - Il y a donc $2^{p-1} - 1$ ALAAP qui mènent à des ACP indépendantes, soit, pour les premières valeurs de p :

◦ 3	3
◦ 4	7
◦ 5	15
◦ 6	31
◦ 7	63
◦ 8	127
etc...	

7 - Voici un exemple d'un ensemble de quinze ALAAP indépendantes d'un tableau à cinq entrées (avec les dimensions de la matrice associée à l'ALAAP, puis celles de la matrice à diagonaliser) :

ALAAP	Matrice	Mat. Diag.
1 → 2 3 4 5	$n_2 n_3 n_4 n_5 \times n_1$	$n_1 \times n_1$
2 → 1 3 4 5	$n_1 n_3 n_4 n_5 \times n_2$	$n_2 \times n_2$
3 → 1 2 4 5	$n_1 n_2 n_4 n_5 \times n_3$	$n_3 \times n_3$
4 → 1 2 3 5	$n_1 n_2 n_3 n_5 \times n_4$	$n_4 \times n_4$
5 → 1 2 3 4	$n_1 n_2 n_3 n_4 \times n_5$	$n_5 \times n_5$
1 2 → 3 4 5	$n_3 n_4 n_5 \times n_1 n_2$	$n_1 n_2 \times n_1 n_2$
1 3 → 2 4 5	$n_2 n_4 n_5 \times n_1 n_3$	$n_1 n_3 \times n_1 n_3$
1 4 → 2 3 5	$n_2 n_3 n_5 \times n_1 n_4$	$n_1 n_4 \times n_1 n_4$

1 5 → 2 3 4	$n_{234} \times n_{15}$	$n_{15} \times n_{15}$
2 3 → 1 4 5	$n_{145} \times n_{23}$	$n_{23} \times n_{23}$
2 4 → 1 3 5	$n_{135} \times n_{24}$	$n_{24} \times n_{24}$
2 5 → 1 4 5	$n_{145} \times n_{25}$	$n_{25} \times n_{25}$
3 4 → 1 2 5	$n_{123} \times n_{34}$	$n_{34} \times n_{34}$
3 5 → 1 2 4	$n_{124} \times n_{35}$	$n_{35} \times n_{35}$
4 5 → 2 3 4	$n_{234} \times n_{45}$	$n_{45} \times n_{45}$

8 - Par exemple, l'ALAAP 3 2 4 → 5 1 a une ACP identique à l'ALAAP 2 3 4 → 1 5, qui se ramène à celle de 1 5 → 2 3 4 par transposition.

9 - De même, si $\mathcal{E}_1 = \mathbb{R}^{10}$, une ALAAP quelconque se ramène toujours à la diagonalisation d'une matrice 100×100 au plus.

CHAPITRE D

MODELE DE TUCKER

1. ORIGINE ET DÉVELOPPEMENTS DU MODÈLE DE TUCKER

Le modèle de TUCKER d'ordre 3, souvent noté TUCKER3 dans la littérature anglo-saxonne, surtout depuis les publications de KROONENBERG & DE LEEUW (1980) et KROONENBERG (1983), est probablement le premier modèle publié d'analyse factorielle d'un tableau à plus de deux entrées. Son expression algébrique a été publiée dans TUCKER (1966), après plusieurs publications au début des années 60 : TUCKER (1963) et TUCKER (1964) notamment. L'idée de base reste une extension de l'analyse factorielle : représenter un tableau à trois entrées de dimensions n_1 , n_2 et n_3 respectivement pour chacune de ces entrées par un tableau à trois entrées de dimension moindre, mais fixée à l'avance, sur chacune des dimensions.

Dire que le tableau A de dimensions (n_1, n_2, n_3) est représenté par un tableau B de dimensions moindres (p_1, p_2, p_3) signifie que chaque élément α_{ijk} de A peut être calculé comme fonction des éléments β_{pqr} du tableau B . On dit également que A peut être reconstruit à partir de B . C'est bien une extension de l'analyse factorielle classique qui, sur matrices, revient à reconstruire de manière optimale une matrice de dimensions (n_1, n_2) fixée à partir des éléments d'une matrice de rang inférieur.

L'entreprise était difficile, car aucune des bonnes propriétés de l'AFC ne se généralise simplement aux tableaux à plus de deux entrées. D'ailleurs, l'émergence d'une vision claire de ce que signifie le modèle de TUCKER, et comment le calculer, a pris plus d'une vingtaine d'années.

□

Une présentation détaillée du modèle de TUCKER se trouve dans KROONENBERG & DE LEEUW (1980) et KROONENBERG (1983).

En 1966, dans son article original, Tucker écrivait que l'algorithme numérique qu'il proposait n'était pas optimal au sens des moindres carrés : il ne trouvait qu'une solution approchée du problème qu'il avait posé.

Le premier algorithme exact de résolution, par une méthode itérative de type moindres carrés alternés, est présentée dans KROONENBERG & DE LEEUW (1980). Elle a été reprise dans KROONENBERG (1983) en omettant les détails de calcul. Elle reste cependant complexe car elle développe le calcul sur indices : la seule notation synthétique utilisée est le produit de Kronecker sur les matrices, qui se rapproche du produit tensoriel, mais de façon non symétrique sur les dimensions du tableau.

KROONENBERG (1983), en reprenant ce résultat, insiste sur sa complexité, la difficulté de mise en oeuvre, et remarque que les premiers algorithmes de Tucker, qui ne réalisent pas d'itération, conduisent empiriquement à des résultats voisins de l'algorithme plus rigoureux proposé, probablement par un phénomène de relaxation.

Et, curieusement, dès 1983, KROONENBERG (1983), lorsqu'il étudie l'extension possible du modèle de TUCKER aux tableaux à plus de trois entrées, note à la fois que, conceptuellement, cette extension est facile, mais que, pratiquement, les notations utilisées traditionnellement, par un maniement exagérément complexe des indices, empêchent toute mise en oeuvre.

Une extension du modèle de TUCKER à 4 dimensions a été publiée par LASTOWICKA (1981), tout en rencontrant la même difficulté que KROONENBERG dans le maniement des indices.

Un généralisation du modèle de TUCKER à p modes (étude des tableaux à p entrées) a été proposée en 1986, par une équipe hollandaise de Tilburg, Amsterdam et Groningue (KAPTEYN, NEUDECKER & WANSBECK, 1986), à partir des

produits de Kronecker généralisés, en utilisant les matrices de permutation et la vectorialisation des tableaux. Cet article va droit au but mathématique, en proposant une notation claire, concise et synthétique, et les auteurs attribuent une bonne partie de leur réussite à cette notation. Les notations utilisées sont présentées de manière exhaustive dans HENDERSON & SEARLE (1981). La présentation de l'article s'éloigne de la démarche de KROONENBERG & DE LEEUW (1980), et gagne en simplicité.

Cependant, l'algorithme y est présenté sous le seul angle du cacul algébrique, sans référence précise au modèle de Tucker et sans liens explicites avec la programmation. Il reste complexe, car les auteurs n'ont pas exprimé le lien entre chaque phase de l'itération et l'ACP sous contrainte, et redémontrent un résultat d'ACP sous contrainte linéaire par les multiplicateurs de Lagrange.

Ce lien avec une ACP sous contrainte particulière est suggéré dans KIERS (1989), en relation avec le modèle STATIS, qui est une ACP 3 modes particulière sur un tableau à trois entrées. KIERS (1989) présente l'algorithme de résolution du modèle de Tucker ainsi, mais limité à trois entrées. Une lecture attentive du texte de KIERS (1989) [pp.11-12] permet de remarquer que la distinction entre itération et étape de l'itération y diparait mystérieusement. Or, c'est chaque étape de l'itération qui est une analyse STATIS sous contrainte, et non l'algorithme TUCKALS-3 de résolution de TUCKER-3.

Parallèlement, avec les travaux de JAFFRENOU (1978), POLIT (1986) et D'AUBIGNY & POLIT (1989), une école plus géométrique de pensée s'est constituée à Grenoble, permettant une généralisation du modèle de TUCKER d'un tableau à n entrées en tenant compte de métriques particulières sur chacun des espace associé à chacune des dimensions.

Cette approche géométrique, par les travaux de JAFFRENOU (1978), s'est d'abord attachée à établir les liens entre le modèle de TUCKER à 3 modes tel qu'il est présenté classiquement et une généralisation du schéma de dualité, qui reste une pierre angulaire de nombreux travaux français. Ensuite, POLIT (1986) a réussi à généraliser cette approche aux tableaux à n entrées, puis

D'AUBIGNY & POLIT (1989) ont introduit des métriques particulières sur les espaces.

□

Chacune de ces publications a apporté un regard toujours plus précis sur le modèle de Tucker le plus général :

- présentation dans TUCKER (1966)
- algorithme par itérations dans KROONENBERG & DE LEEUW (1980)
- généralisation du modèle pour 4 entrées dans LASTOWICKA (1981)
- généralisation algébrique à n entrées dans KAPTEYN & Al. (1986)
- généralisation géométrique à n entrées dans POLIT (1986)
- introduction de métriques particulières dans D'AUBIGNY & POLIT (1989)
- lien avec l'ACP-3 modes sous contraintes dans KIERS (1989).

Chacun de ces travaux s'est inscrit dans une trajectoire particulière, avec ses caractéristiques propres : il est aisé de distinguer un pôle hollandais, orienté vers les algorithmes de calcul, et un pôle grenoblois, plus marqué par la géométrie des schémas de dualité (sachant que l'initiateur est américain). Chacune des méthodes présente donc les lacunes de son approche : les travaux néerlandais se prêtent mal à la généralisation (hormis le travail de KAPTEYN & Al., curieusement resté sans suites), et les travaux grenoblois sont d'un abord algébrique qui parfois peut rebuter par son abstraction.

Nous proposons l'abandon du schéma de dualité, qui multiplie les espaces de travail, en ne conservant que des espaces $\mathcal{E}_1, \dots, \mathcal{E}_p$ munis respectivement de métriques euclidiennes définies par des MSDP M_1, \dots, M_p .

Bien qu'émise dans un contexte contestable, la remarque de KIERS (1989) sur les relations entre le modèle de TUCKER et l'ACP- n modes sous contraintes est capitale, et permet de bâtir une généralisation concise et simple d'algorithmes pour les tableaux à n entrées de l'algorithme de

KROONENBERG & DE LEEUW (1980). Cela permet de simplifier notablement la présentation, déjà concise, de KAPTEYN & Al. (1986).

Un léger glissement des notations (en utilisant le produit tensoriel, plus aisé à manipuler que les produits de Kronecker multiples qui génèrent des matrices de fortes dimensions) permet de retrouver rapidement les résultats de POLIT (1986), D'AUBIGNY & POLIT (1989), et une extension où des contraintes linéaires sont introduites sur les espaces des composantes.

2- MODÈLE DE TUCKER D'ORDRE 3

1. SOUS ESPACES VECTORIELS DE $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$

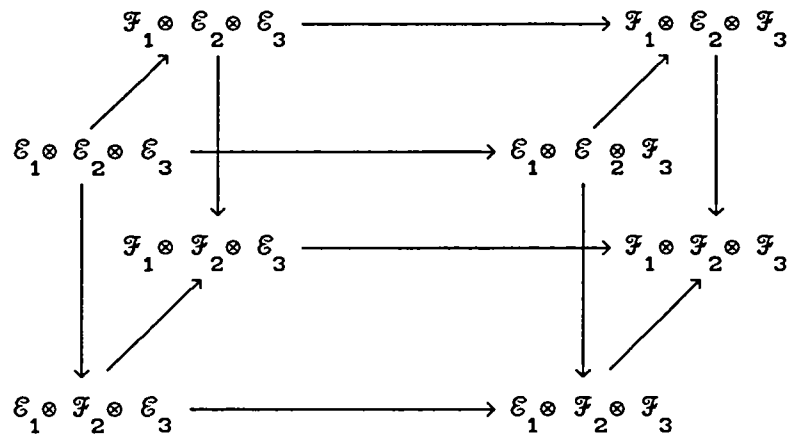
\mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 désignent des EVRE de dimensions respectives n_1 , n_2 et n_3 .

\mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 désignent des SEV de \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 de dimensions p_1 , p_2 et p_3 .

Rappelons que $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ est l'EV des formes trilinéaires sur \mathbb{R} .

1. Treillis des SEV de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$

1 - Rappelons que $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3 \subset \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$, et que l'on a plus généralement le diagramme d'inclusion suivant, où les flèches \longrightarrow désignent la relation \supset :



2. Projections

1 - Les espaces considérés étant des espaces euclidiens, "projection" signifiera dans la suite du chapitre "projection orthogonale".

2 - Il est trivial de rappeler que, dans le diagramme précédent, les projections selon les flèches forment un diagramme commutatif.

3 - On note par \mathcal{P} la projection sur $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$.

4 - Etant donné un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$, on peut donc le projeter par \mathcal{P} sur $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$.

2. ACP-MT (ANALYSE EN COMPOSANTES PRINCIPALES SELON UN MODELE DE TUCKER)

1. Présentation du problème

1 - Dans le paragraphe précédent, on se donnait trois sous espaces \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 , et l'on projetait un tenseur donné sur $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$. Rechercher $\mathcal{P}(A)$ est un problème de projection, et le problème est linéaire.

2 - Le problème de l'ACP-MT est différent :

- on connaît les nombres n_1 , n_2 et n_3 , dimensions des \mathcal{E}_i .
- on se donne les nombres p_1 , p_2 et p_3 , dimensions de SEV inconnus des \mathcal{E}_i
- on recherche les SEV \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 de dimensions p_1 , p_2 et p_3 tels que $\|A - \mathcal{P}(A)\|$ soit minimum.

3 - Les inconnues du problème sont donc les trois SEV \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 , et le problème n'est plus linéaire.

2. Expression du problème

On peut exprimer ce problème comme suit :

1 - Trouver une écriture de A sous la forme :

$$A = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \sum_{k=1}^{p_3} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k} + R$$

où : $\|R\|$ est minimum,

p_1, p_2, p_3 sont fixés,

les $(f_{pq})_q$ forment des systèmes orthonormés

2 - En termes de coordonnées, cette décomposition s'écrit :

$$\alpha_{pqr} = \sum_{ijk} \xi_{ijk} \varphi_{1ip} \varphi_{2jq} \varphi_{3jr} + \varepsilon_{pqr}$$

où :

- (α_{pqr}) sont les coordonnées de A ,
- (φ_{1ip}) les coordonnées d'une matrice F_1 dont la $i^{\text{ème}}$ colonne est f_{1i} ,
- (φ_{2jq}) les coordonnées d'une matrice F_2 dont la $j^{\text{ème}}$ colonne est f_{2j} ,
- (φ_{3kr}) les coordonnées d'une matrice F_3 dont la $k^{\text{ème}}$ colonne est f_{3k} .

3 - Le tenseur de composantes (ξ_{ijk}) est appelé "core matrix" dans la littérature anglo-saxonne (notamment dans KROONENBERG, 1983). Mais rappelons que, si le tenseur $\mathfrak{P}(A)$ est défini de façon unique, son expression en termes de coordonnées sur une base dépend de la base choisie : les inconnues sont trois SEV $\mathcal{F}_1, \mathcal{F}_2$ et \mathcal{F}_3 , et non les vecteurs f_{1i}, f_{2j} et f_{3k} ou leurs coordonnées.

4 - En particulier, si la solution $(\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3)$ est unique, la solution (f_{1i}, f_{2j}, f_{3k}) est indéterminée car il existe une infinité de BON d'un SEV \mathcal{F}_1 donné. Si (f_{1i}, f_{2j}, f_{3k}) est une solution en terme de base, tout système $(R_1 \cdot f_{1i}, R_2 \cdot f_{2j}, R_3 \cdot f_{3k})$ est également une solution, où R_1 est une matrice orthogonale de $\mathcal{M}(\mathcal{F}_1)$, correspondant à une rotation.

3. Solution du problème

1 - Les inconnues du problème sont trois SEV. La résolution numérique du problème de l'ACP-MT sera abordée par une méthode du type moindres carrés alternés sur les inconnues, donc sur les SEV, en utilisant un lemme présenté ci-après. Ce lemme permet de ramener l'algorithme de recherche de solution à une suite d'ACP sous contrainte linéaire.

2 - La méthode des moindres carrés fait intervenir une métrique sur l'espace des inconnues, qui sont ici des SEV. Rappelons que l'on peut en effet définir une distance dans l'ensemble des SEV d'un EVE de même dimension. L'outil généralement utilisé passe par la définition d'angles canoniques, tels que le plus grand d'entre eux approche 0 lorsque les deux espaces se rapprochent (STEWART & SUN, 1990). Cette définition remonte à JORDAN, en 1875, (Essai sur la géométrie à n dimensions, *Bulletin de la Société mathématique*, 3, 103-174) et a été fréquemment redécouverte ensuite par de nombreux auteurs. Nous contournerons la difficulté technique d'introduire une métrique sur un ensemble de SEV en n'utilisant que la projection $\mathcal{P}(A)$ d'un tenseur A sur les SEV.

3 - Les inconnues étant des SEV, nous noterons par \mathcal{F}_1 les SEV de \mathcal{E}_1 connus et fixés, et \mathcal{X}_1 les SEV de \mathcal{E}_1 recherchés.

4. Lemme

■ Enoncé

□ Soient :

- \mathcal{X}_1 l'ensemble des SEV de \mathcal{E}_1 de dimension p_1
- \mathcal{X}_1 un élément de \mathcal{X}_1
- \mathcal{F}_2 et \mathcal{F}_3 deux SEV fixés de \mathcal{E}_2 et \mathcal{E}_3 respectivement
- $\mathcal{P}_{\mathcal{X}_1}$ la projection orthogonale de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ sur $\mathcal{X}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$
- A un tenseur de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$

□ Alors, rechercher l'élément \mathcal{X}_1 de \mathcal{X}_1 tel que $\|A - \mathcal{P}_{\mathcal{X}_1}(A)\|$ soit minimum revient à réaliser l'ACP-DCL de $A \in \mathcal{L}(\mathcal{E}_1, \mathcal{E}_2 \otimes \mathcal{E}_3)$ sous les contraintes

définies par \mathcal{F}_2 et \mathcal{F}_3 .

□ \mathcal{X}_1 est donc le SEV de \mathcal{E}_1 engendré par les p_1 premières composantes principales de l'application $\mathfrak{p}(A)$, qui est la projection orthogonale de A sur $\mathcal{E}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$.

■ La démonstration est quasi superflue : c'est la définition de l'ACP-DCL qui se cache derrière cette écriture. Ce lemme est donc une simple remarque, mais sur laquelle est bâti notre algorithme de résolution.

□ Cette remarque découle immédiatement de l'écriture suivante :

$$\begin{aligned} 1 - A &= \sum_{ijk} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k} + R \\ &= \sum_i f_{1i} \otimes \left(\sum_{jk} \xi_{ijk} f_{2j} \otimes f_{3k} \right) + R \end{aligned}$$

2 - Comme $\sum_{jk} \xi_{ijk} f_{2j} \otimes f_{3k} \in \mathcal{F}_2 \otimes \mathcal{F}_3$, A est de la forme :

$$A = \sum_i f_{1i} \otimes M_i + R$$

avec $M_i \in \mathcal{F}_2 \otimes \mathcal{F}_3$ et $f_{1i} \in \mathcal{F}_1$, qui est bien celle de l'ACP-DCL de $A \in \mathcal{L}(\mathcal{E}_1, \mathcal{E}_2 \otimes \mathcal{E}_3)$ avec les contraintes définies par \mathcal{F}_1 sur \mathcal{E}_1 et $\mathcal{F}_2 \otimes \mathcal{F}_3$ sur $\mathcal{E}_2 \otimes \mathcal{E}_3$.

■ Remarques

1 - Le lemme précédent reste valide par permutation circulaire sur les SEV, en privilégiant \mathcal{E}_2 ou \mathcal{E}_3 .

5. Algorithme de résolution

1 - A partir du lemme précédent, Le problème de l'ACP-MT peut être résolu par un algorithme du type moindres carrés alternés, où les inconnues sont les SEV $\mathcal{F}_1, \mathcal{F}_2$ et \mathcal{F}_3 :

- On se donne un triplet de départ $(\mathcal{F}_1^0, \mathcal{F}_2^0, \mathcal{F}_3^0)$ de trois SEV
- On propose un algorithme itératif de calcul d'un triplet $(\mathcal{F}_1^{n+1}, \mathcal{F}_2^{n+1}, \mathcal{F}_3^{n+1})$ à partir d'un triplet $(\mathcal{F}_1^n, \mathcal{F}_2^n, \mathcal{F}_3^n)$ tel que, si \mathfrak{p}_n

désigne la projection de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ sur $\mathcal{F}_1^n \otimes \mathcal{F}_2^n \otimes \mathcal{F}_3^n$, alors :

$$\|A - \mathcal{P}_{n+1}(A)\| \leq \|A - \mathcal{P}_n(A)\|$$

2 - Considérons alors :

- \mathcal{X}_1 , ensemble des SEV de \mathcal{E}_1 de dimension p_1
- \mathcal{X}_2 , ensemble des SEV de \mathcal{E}_2 de dimension p_2
- \mathcal{X}_3 , ensemble des SEV de \mathcal{E}_3 de dimension p_3
- $(\mathcal{F}_1^n, \mathcal{F}_2^n, \mathcal{F}_3^n)$ le triplet de SEV à l'étape n.

3 - La solution numérique de l'ACP-MT sera alors obtenue en utilisant le lemme précédent alternativement sur les espaces \mathcal{F}_1^n , \mathcal{F}_2^n et \mathcal{F}_3^n .

4 - Considérons le triplet $(\mathcal{E}_1, \mathcal{F}_2^n, \mathcal{F}_3^n)$. Soit \hat{A}_n la projection orthogonale de A sur $\mathcal{E}_1 \otimes \mathcal{F}_2^n \otimes \mathcal{F}_3^n$. Alors, l'espace \mathcal{X}_1 , SEV de dimension p_1 fixée de \mathcal{E}_1 , tel que $\|A - \mathcal{P}_{\mathcal{X}_1}(A)\|$ soit minimum pour \mathcal{F}_2^n et \mathcal{F}_3^n fixés est le SEV engendré par les p_1 premières composantes principales de \hat{A}_n , et l'on a :

$$\|A - \mathcal{P}_{\mathcal{X}_1}(A)\| \leq \|A - \mathcal{P}_n(A)\|$$

5 - Posons alors $\mathcal{F}_1^{n+1} = \mathcal{X}_1$.

6 - Considérons le triplet $(\mathcal{F}_1^{n+1}, \mathcal{E}_2, \mathcal{F}_3^n)$. Soit \hat{B}_n la projection orthogonale de $2A$ sur $\mathcal{E}_2 \otimes \mathcal{F}_3^n \otimes \mathcal{F}_1^{n+1}$. Alors, l'espace \mathcal{Y}_2 , SEV de dimension p_2 fixée de \mathcal{E}_2 , tel que $\|A - \mathcal{P}_{\mathcal{Y}_2}(A)\|$ soit minimum pour \mathcal{F}_1^{n+1} et \mathcal{F}_3^n fixés est le SEV engendré par les p_2 premières composantes principales de \hat{B}_n , et l'on a :

$$\|A - \mathcal{P}_{\mathcal{Y}_2}(A)\| \leq \|A - \mathcal{P}_{\mathcal{X}_1}(A)\|$$

7 - Posons alors $\mathcal{F}_2^{n+1} = \mathcal{Y}_2$.

8 - Considérons le triplet $(\mathcal{F}_1^{n+1}, \mathcal{F}_2^{n+1}, \mathcal{E}_3)$. Soit \hat{C}_n la projection orthogonale de $4A$ sur $\mathcal{E}_3 \otimes \mathcal{F}_1^{n+1} \otimes \mathcal{F}_2^{n+1}$. Alors, l'espace \mathcal{Z}_3 , SEV de dimension p_3 fixée de \mathcal{E}_3 , tel que $\|A - \mathcal{P}_{\mathcal{Z}_3}(A)\|$ soit minimum pour \mathcal{F}_1^{n+1} et \mathcal{F}_2^{n+1} fixés est le SEV engendré par les p_3 premières composantes principales de \hat{C}_n , et l'on a :

$$\|A - \mathcal{P}_{Z_3}(A)\| \leq \|A - \mathcal{P}_{y_2}(A)\|$$

9 - Posons alors : $\mathcal{F}_3^{n+1} = Z_3$.

10 - On a alors construit un triplet $(\mathcal{F}_1^{n+1}, \mathcal{F}_2^{n+1}, \mathcal{F}_3^{n+1})$ tel que :

$$\|A - \mathcal{P}_{n+1}(A)\| = \|A - \mathcal{P}_{Z_3}(A)\| \leq \|A - \mathcal{P}_{y_2}(A)\| \leq \|A - \mathcal{P}_{x_1}(A)\| \leq \|A - \mathcal{P}_n(A)\|, \text{ d'où :}$$

$$\|A - \mathcal{P}_{n+1}(A)\| \leq \|A - \mathcal{P}_n(A)\|$$

11 - L'algorithme ainsi proposé est donc convergent, pour $\mathcal{P}_n(A)$.

3 - MODÈLE DE TUCKER D'ORDRE 3 AVEC DES MÉTRIQUES PARTICULIÈRES

1. Rappel des notations et présentation du problème

1 - Rappelons que les espaces \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 sont respectivement munis d'une métrique définie par les MSDP M_1, M_2 et M_3 . On associe alors canoniquement à $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ une métrique définie par le tenseur T telle que :

$$\langle A, B \rangle_T = \langle A, T.B \rangle$$

avec en particulier :

$$T.(x \otimes y \otimes z) = (M_1.x) \otimes (M_2.y) \otimes (M_3.z)$$

où $x \in \mathcal{E}_1$, $y \in \mathcal{E}_2$, $z \in \mathcal{E}_3$.

2 - Le problème de l'ACP-MT avec les métriques définies par M_1, M_2 et M_3 est alors le suivant :

- on connaît les nombres n_1 , n_2 et n_3 , dimensions des \mathcal{E}_i .
- on se donne les nombres p_1 , p_2 et p_3 , dimensions de SEV inconnus des \mathcal{E}_i
- on recherche les SEV \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 de dimensions p_1 , p_2 et p_3 tels que $\|A - \mathcal{P}(A)\|_T$ soit minimum.

3 - Par rapport au problème simple de l'ACP-MT, seule la métrique utilisée sur $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ est différente. Elle est définie par un tenseur T et intervient dans les calculs, qui suivent la même logique. Notamment, les inconnues du problème sont toujours les trois SEV \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 .

2. Expression du problème

Le problème s'exprime ainsi :

- 1 - Trouver une écriture de A sous la forme :

$$A = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \sum_{k=1}^{p_3} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k} + R$$

où : $\|R\|_T$ est minimum,

p_1, p_2, p_3 sont fixés,

les (f_{pq}) forment des systèmes orthonormés

pour les métriques définies par les MSDP M_p

2 - L'expression en termes de coordonnées n'a pas changée : seules les contraintes sur les ε_{pqr} changent :

$$\alpha_{pqr} = \sum_{ijk} \xi_{ijk} \varphi_{1ip} \varphi_{2jq} \varphi_{3kr} + \varepsilon_{pqr}$$

où :

- (α_{pqr}) sont les coordonnées de A ,
- (φ_{1ip}) les coordonnées d'une matrice F_1 dont la $i^{\text{ème}}$ colonne est f_{1i} ,
- (φ_{2jq}) les coordonnées d'une matrice F_2 dont la $j^{\text{ème}}$ colonne est f_{2j} ,
- (φ_{3kr}) les coordonnées d'une matrice F_3 dont la $k^{\text{ème}}$ colonne est f_{3k} .

3 - Les mêmes remarques que pour l'ACP-MT valent pour la matrice noyau.

3. Solution du problème

Nous présentons également deux solutions : la première résoud directement le problème, et la seconde utilise les isométries entre espaces définies par les métriques M_1, M_2 et M_3 . La deuxième solution faisant appel aux isométries ι est nettement préférable sur le plan du calcul : il n'y a pas à programmer spécifiquement l'ACP-MT avec métriques.

1 - La première technique directe pour trouver la solution est calquée sur celle de l'ACP-MT. Seule l'ACP à chaque itération change : il s'agit d'une ACP-DCL avec métriques.

2 - Les inconnues du problème sont trois SEV. La résolution numérique est abordée par une méthode du type moindres carrés alternés sur les inconnues, donc sur les SEV, en utilisant un lemme présenté ci-après. Ce lemme permet de ramener l'algorithme de recherche de solution à une suite d'ACP-DCL sous contrainte linéaire.

3 - Les inconnues étant des SEV, nous noterons à nouveau par \mathcal{F}_1 les SEV de \mathcal{E}_1 connus et fixés, et \mathcal{X}_1 les SEV de \mathcal{E}_1 recherchés.

4. Lemme

■ Enoncé

□ Soient :

- \mathcal{X}_1 l'ensemble des SEV de \mathcal{E}_1 de dimension p_1
- \mathcal{X}_1 un élément de \mathcal{X}_1
- \mathcal{F}_2 et \mathcal{F}_3 deux SEV fixés de \mathcal{E}_2 et \mathcal{E}_3 respectivement
- $\mathcal{P}_{\mathcal{X}_1}$ la projection orthogonale pour T de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ sur $\mathcal{X}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$
- A un tenseur de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$

□ Alors, rechercher l'élément \mathcal{X}_1 de \mathcal{X}_1 tel que $\|A - \mathcal{P}_{\mathcal{X}_1}(A)\|_T$ soit minimum revient à réaliser l'ACP-DCL de $A \in \mathcal{L}(\mathcal{E}_1, \mathcal{E}_2 \otimes \mathcal{E}_3)$ sous les contraintes définies par \mathcal{F}_2 et \mathcal{F}_3 avec les métriques définies par M_1, M_2 et M_3 .

□ \mathcal{X}_1 est donc le SEV de \mathcal{E}_1 engendré par les p_1 premières composantes principales de l'application $\mathcal{P}(A)$, projection orthogonale de A sur $\mathcal{E}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$ avec la métrique définie par T .

□ Cette remarque découle immédiatement de l'écriture suivante qui est rappelée :

$$\begin{aligned} 1 - A &= \sum_{ijk} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k} + R \\ &= \sum_i f_{1i} \otimes \left(\sum_{jk} \xi_{ijk} f_{2j} \otimes f_{3k} \right) + R \end{aligned}$$

2 - Comme $\sum_{j,k} \xi_{1jk} f_{2j} \otimes f_{3k} \in \mathcal{F}_2 \otimes \mathcal{F}_3$, A est de la forme :

$$A = \sum_i f_{1i} \otimes M_i + R$$

avec $M_i \in \mathcal{F}_2 \otimes \mathcal{F}_3$ et $f_{1i} \in \mathcal{F}_1$, qui est bien celle de l'ACP-DCL de $A \in \mathcal{L}(\mathcal{E}_1, \mathcal{E}_2 \otimes \mathcal{E}_3)$ avec les contraintes définies par \mathcal{F}_1 sur \mathcal{E}_1 et $\mathcal{F}_2 \otimes \mathcal{F}_3$ sur $\mathcal{E}_2 \otimes \mathcal{E}_3$ et les métriques définies par M_1, M_2 et M_3 .

3 - Le lemme précédent reste valide par permutation circulaire sur les SEV, en privilégiant \mathcal{E}_2 ou \mathcal{E}_3 .

4 - La résolution suit alors le même algorithme que celle de l'ACP-MT :

- On se donne un triplet de départ $(\mathcal{F}_1^0, \mathcal{F}_2^0, \mathcal{F}_3^0)$ de trois SEV
- On propose un algorithme itératif de calcul d'un triplet $(\mathcal{F}_1^{n+1}, \mathcal{F}_2^{n+1}, \mathcal{F}_3^{n+1})$ à partir d'un triplet $(\mathcal{F}_1^n, \mathcal{F}_2^n, \mathcal{F}_3^n)$ tel que, si \mathcal{P}_n désigne la projection de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ sur $\mathcal{F}_1^n \otimes \mathcal{F}_2^n \otimes \mathcal{F}_3^n$, alors :

$$\|A - \mathcal{P}_{n+1}(A)\|_T \leq \|A - \mathcal{P}_n(A)\|_T$$

5. Deuxième solution du problème

La deuxième solution du problème utilise directement l'isométrie entre différents espace induite par les matrices M_1, M_2 et M_3 .

Rappelons que si M_1 est la MSDP définissant la métrique sur \mathcal{E}_1 , nous notons P_1 la MSDP telle que $P_1^2 = M_1$.

■ Définition de l'isométrie

1 - Rappelons que l'on définit une isométrie ι pour chacun des espaces \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 comme suit :

$$\iota(x_1) = P_1^{-1} \cdot x_1$$

$$\iota(x_2) = P_2^{-1} \cdot x_2$$

$$\iota(x_3) = P_3^{-1} \cdot x_3$$

2 - Cette isométrie est alors étendue à $\mathcal{H}(\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3)$ par le produit tensoriel. En particulier :

$$\iota(x_1 \otimes x_2 \otimes x_3) = \iota(x_1) \otimes \iota(x_2) \otimes \iota(x_3) = (P_1^{-1} \cdot x_1) \otimes (P_2^{-1} \cdot x_2) \otimes (P_3^{-1} \cdot x_3)$$

■ Solution de l'ACP-MT avec métriques

La solution est alors très rapide.

1 - Soit $A = \sum_{ijk} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k} + R$ le modèle de l'ACP-MT de A avec métriques unités.

2 - $\|A - \sum_{ijk} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k}\|^2$ est donc minimum.

3 - $\|\iota(A - (\sum_{ijk} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k}))\|_T^2$ est donc minimum pour la norme définie par T .

4 - Ainsi, $\sum_{ijk} \xi_{ijk} \iota(f_{1i}) \otimes \iota(f_{2j}) \otimes \iota(f_{3k})$ sont les composantes du modèle ACP-MT $\iota(A)$ pour la norme définie par T .

5 - D'où : si $\xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k}$ sont les composantes du modèle ACP-MT de A pour la norme définie par T , alors, $\xi_{ijk} (P_1 \cdot f_{1i}) \otimes (P_2 \cdot f_{2j}) \otimes (P_3 \cdot f_{3k})$ sont les composantes du modèle de $\iota^{-1}(A)$ pour la norme unité, que l'on sait calculer.

■ Résultat

Pour calculer le modèle ACP-MT de A avec les métriques M_1 , M_2 et M_3 , on peut procéder comme suit :

1 - Calcul des matrices $P_1 = \sqrt{M_1}$, $P_2 = \sqrt{M_2}$ et $P_3 = \sqrt{M_3}$.

2 - Calcul de $B = \iota^{-1}(A)$ par $B = \sum_{ijk} P_1 \cdot e_{1i} \otimes P_2 \cdot e_{2j} \otimes P_3 \cdot e_{3k}$

3 - Calcul des BON (u_{1i}) , (u_{2j}) et (u_{3k}) ainsi que des coefficients ξ_{ijk} du modèle ACP-MT de B

4 - Calcul de $f_{1i} = P_1^{-1} \cdot u_{1i}$, $f_{2j} = P_2^{-1} \cdot u_{2j}$, $f_{3k} = P_3^{-1} \cdot u_{3k}$.

Cette solution est bien plus économique en temps de calcul, et sera retenue. De plus, elle se prête aisément à généralisation lorsque l'ordre augmente et devient > 3 .

4 - MODÈLE DE TUCKER D'ORDRE 3 AVEC DES CONTRAINTES LINÉAIRES

Le modèle de TUCKER avec contraintes linéaires revient à imposer des contraintes linéaires sur les composantes, ou f_{1i} , f_{2j} et f_{3k} , où encore à imposer que les espaces \mathcal{F}_i soient des SEV d'espaces \mathcal{H}_i fixés.

1. Définition

Réaliser l'ACP-MT du tenseur A avec la triple contrainte linéaire définie par les SEV \mathcal{H}_1 , \mathcal{H}_2 et \mathcal{H}_3 de \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 respectivement est trouver une écriture de la forme :

$$A = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \sum_{k=1}^{p_3} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k} + R$$

où : $\|R\|$ est minimum,

p_1, p_2, p_3 sont fixés,

les (f_{1q}) appartiennent à un SEV \mathcal{H}_1 de \mathcal{E}_1

les (f_{2q}) appartiennent à un SEV \mathcal{H}_2 de \mathcal{E}_2

les (f_{3q}) appartiennent à un SEV \mathcal{H}_3 de \mathcal{E}_3

les $(f_{pq})_{pq}$ forment des systèmes orthonormés

2. Lemme

■ Enoncé

Réaliser l'ACP-MT de A sous les contraintes définies par $\mathcal{H}_1, \mathcal{H}_2$ et \mathcal{H}_3 revient à réaliser dans $\mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$ l'ACP-MT de B , projection orthogonale de A dans $\mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$.

$$B = \mathcal{P}(A)$$

■ Démonstration

La démonstration est calquée sur la démonstration de l'ACP-DCL.

1 - Notons \mathcal{P} la projection orthogonale de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ sur $\mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$, et \mathcal{P}^\perp le projecteur orthogonal à \mathcal{P} (i.e. sur $(\mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3)^\perp$).

2 - Comme on cherche des décompositions de la forme :

$$A = \sum_i x_{1i} \otimes x_{2i} \otimes x_{3i} + R$$

où $x_{11} \in \mathcal{H}_1$, $x_{21} \in \mathcal{H}_2$ et $x_{31} \in \mathcal{H}_3$, notons, pour simplifier :

$$\Sigma_1 = \Sigma_1 x_{11} \otimes x_{21} \otimes x_{31}.$$

et remarquons que $\Sigma_1 \in \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3$.

3 - Ainsi :

$$\begin{aligned} \|A - \Sigma_1\|^2 &= \|(\mathcal{P} + \mathcal{P}^\perp)(A - \Sigma_1)\|^2 \\ &= \|\mathcal{P}(A - \Sigma_1)\|^2 + \|\mathcal{P}^\perp(A - \Sigma_1)\|^2 \text{ car } \mathcal{P} \text{ et } \mathcal{P}^\perp \text{ sont orthogonaux} \\ &= \|\mathcal{P}(A) - \Sigma_1\|^2 + \|\mathcal{P}^\perp(A)\|^2 \text{ car } \mathcal{P}(\Sigma_1) = \Sigma_1 \text{ et } \mathcal{P}^\perp(\Sigma_1) = 0 \text{ car } \Sigma_1 \in \mathcal{H}_1 \otimes \mathcal{H}_2 \otimes \mathcal{H}_3 \end{aligned}$$

4 - Ainsi, $\|A - \Sigma_1\|^2$ est minimale lorsque $\|\mathcal{P}(A) - \Sigma_1\|$ l'est.

5 - MODÈLE DE TUCKER D'ORDRE P

1 - Les notations et algorithmes utilisés pour le modèle de Tucker d'ordre 3 se généralisent sans difficulté au modèle de TUCKER d'ordre p.

2 - Dans la suite de ce paragraphe, nous supposons que les bases utilisées sont des BON, et nous noterons par $\langle \cdot, \cdot \rangle$ le produit scalaire quel que soit l'espace utilisé.

3 - $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ étant muni d'une structure euclidienne, "projection" signifiera dorénavant "projection orthogonale".

4 - On note par \mathcal{P} la projection sur $\mathcal{F}_1 \otimes \dots \otimes \mathcal{F}_p$.

1. ACP SELON UN MODELE DE TUCKER (ACPMT)

1. Expression du problème

1 - Le modèle de TUCKER d'ordre p s'exprime simplement par les notations proposées :

2 - Etant donnés :

- p dimensions p_i telles que $\forall i, p_i \leq n_i$,
- un tenseur A de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$,

Trouver p SEV \mathcal{F}_i de dimensions p_i , tels que :

$$\|A - \mathcal{P}(A)\| \text{ soit minimum.}$$

3 - L'expression en terme de décomposition sur une base est légèrement plus complexe :

$$A = \sum_{i_1=1}^{p_1} \dots \sum_{i_q=1}^{p_q} \xi_{i_1 \dots i_q} f_{i_1 1} \otimes \dots \otimes f_{i_q 1} + R$$

où : $\|R\|_T$ est minimum,
 p_1, \dots, p_q sont fixés,
les $(f_{k_i i_k})_k$ forment des systèmes orthonormés

modèle qui s'écrit également de façon plus compacte :

$$A = \sum_i \xi_i \left[\bigotimes_j f_{j i_j} \right]$$

4 - Quant à l'expression en termes de coordonnées, elle est assez complexe :

$$\alpha_{k_1 \dots k_q} = \sum_{i_1 \dots i_q} \xi_{i_1 \dots i_q} \varphi_{i_1 k_1} \varphi_{i_2 k_2} \dots \varphi_{i_q k_q} + \varepsilon_{k_1 \dots k_q}$$

qui se note (plus simplement ?) en :

$$\alpha_{\vec{k}} = \sum_{\vec{i}} \xi_{\vec{i}} \left[\prod_j \varphi_{j i_j k_j} \right]$$

où :

- $(\alpha_{k_1 \dots k_q})$ sont les coordonnées de A ,
- $(\varphi_{1 i_1 k_1})$ les coordonnées d'une matrice F_1 dont les colonnes sont les $f_{1 i_1}$
- $(\varphi_{2 i_2 k_2})$ " " " F_2 " " $f_{2 i_2}$
- ...
- $(\varphi_{q i_q k_q})$ " " " F_q " " $f_{q i_q}$.

5 - Les inconnues sont bien les SEV $\mathcal{F}_1, \dots, \mathcal{F}_p$. La solution sera obtenue numériquement au moyen d'un algorithme de type moindres carrés alternés : connaissant $p-1$ SEV \mathcal{F}_1 , on calculera le $p^{\text{ème}}$ en considérant les $p-1$ premiers comme fixés, et ce alternativement pour tous les SEV $\mathcal{F}_1, \dots, \mathcal{F}_p$.

2. Lemme

Soient :

- \mathcal{X}_1 l'ensemble des SEV de \mathcal{E}_1 de dimension p_1
- \mathcal{X}_1 un élément de \mathcal{X}_1
- $\mathcal{P}_{\mathcal{X}_1}$ la projection de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ sur $\mathcal{X}_1 \otimes \mathcal{F}_2 \otimes \dots \otimes \mathcal{F}_p$
- A un tenseur de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$

Alors, l'élément \mathcal{X}_1 de \mathcal{X}_1 tel que $\|A - \mathcal{P}_{\mathcal{X}_1}(A)\|$ soit minimum est le SEV de \mathcal{E}_1 engendré par les p_1 premières composantes principales de l'ACP-DCL de l'application linéaire A de $\mathcal{L}(\mathcal{E}_1, \mathcal{E}_2 \otimes \dots \otimes \mathcal{E}_p)$ sous la contrainte définie par : les composantes de $\mathcal{E}_2 \otimes \dots \otimes \mathcal{E}_p \in \mathcal{F}_2 \otimes \dots \otimes \mathcal{F}_p$.

C'est donc le SEV engendré par les p_1 PCP de l'application linéaire U_A définie par $U_A(x) = \hat{A}.x$, où \hat{A} est la projection de A sur $\mathcal{E}_1 \otimes \mathcal{F}_2 \otimes \dots \otimes \mathcal{F}_p$.

3. Remarque

Ce lemme reste valide par permutation circulaire sur les SEV, en privilégiant chacun des \mathcal{E}_i .

4. Solution du problème de l'ACPMT

1 - A partir du lemme précédent, Le problème de l'ACPMT peut être résolu assez simplement :

- On se donne un p-uplet de départ $(\mathcal{F}_1^0, \dots, \mathcal{F}_p^0)$ de p SEV
- On propose un algorithme itératif de calcul d'un p-uplet $(\mathcal{F}_1^{n+1}, \dots, \mathcal{F}_p^{n+1})$ à partir d'un p-uplet $(\mathcal{F}_1^n, \dots, \mathcal{F}_p^n)$ tel que, si \mathcal{P}_n désigne la projection de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ sur $\mathcal{F}_1^n \otimes \dots \otimes \mathcal{F}_p^n$, alors :

$$\|A - \mathcal{P}_{n+1}(A)\| \leq \|A - \mathcal{P}_n(A)\|$$

2 - Considérons :

- $\forall i \leq p$, \mathcal{X}_i , ensemble des SEV de \mathcal{E}_i de dimension p_i
- $(\mathcal{F}_1^n, \dots, \mathcal{F}_p^n)$ un p-uplet

3 - La technique proposée consiste à utiliser le lemme précédent par permutation circulaire sur les espaces $\mathcal{F}_1^n, \dots, \mathcal{F}_p^n$.

4 - Considérons le p-uplet $(\mathcal{E}_1, \mathcal{F}_2^n, \dots, \mathcal{F}_p^n)$. Soit \hat{A}_{1n} la projection orthogonale de A sur $\mathcal{E}_1 \otimes \mathcal{F}_2^n \otimes \dots \otimes \mathcal{F}_p^n$. Alors, l'espace \mathcal{X}_1 , SEV de dimension p_1 fixée de \mathcal{E}_1 , tel que $\|A - \mathcal{P}_{\mathcal{X}_1}(A)\|$ soit minimum pour $\mathcal{F}_2^n, \dots, \mathcal{F}_p^n$ fixés est le SEV engendré par les p_1 premières composantes principales de l'application linéaire U_{A1} de $\mathcal{L}(\mathcal{E}_1, \mathcal{F}_2^n \otimes \dots \otimes \mathcal{F}_p^n)$ associée à \hat{A}_{1n} , et l'on a :

$$\|A - \mathcal{P}_{\mathcal{X}_1}(A)\| \leq \|A - \mathcal{P}_n(A)\|$$

5 - Posons alors $\mathcal{F}_1^{n+1} = \mathcal{X}_1$.

6 - Considérons le triplet $(\mathcal{F}_1^{n+1}, \mathcal{E}_2, \mathcal{F}_3^n, \dots, \mathcal{F}_p^n)$. Soit \hat{A}_{2n} la projection orthogonale de A sur $\mathcal{F}_1^{n+1} \otimes \mathcal{E}_2 \otimes \mathcal{F}_3^n \otimes \dots \otimes \mathcal{F}_p^n$. Alors, l'espace \mathcal{X}_2 , SEV de dimension p_2 fixée de \mathcal{E}_2 , tel que $\|A - \mathcal{P}_{\mathcal{X}_2}(A)\|$ soit minimum pour $\mathcal{F}_1^{n+1}, \mathcal{F}_3^n, \dots, \mathcal{F}_p^n$ fixés est le SEV engendré par les p_2 premières composantes principales de \hat{A}_{2n} , et

l'on a :

$$\|A - \mathcal{P}_{\mathcal{X}_2}(A)\| \leq \|A - \mathcal{P}_{\mathcal{X}_1}(A)\|$$

7 - Posons alors $\mathcal{F}_2^{n+1} = \mathcal{X}_2$.

8 - Et ainsi de suite pour $i = 3, \dots, p$. Considérons le p-uplet $(\mathcal{F}_1^{n+1}, \dots, \mathcal{F}_{k-1}^{n+1}, \mathcal{E}_k, \mathcal{F}_{k+1}^n, \dots, \mathcal{F}_p^n)$. Soit \hat{A}_{kn} la projection orthogonale de A sur $\mathcal{F}_1^{n+1} \otimes \dots \otimes \mathcal{F}_{k-1}^{n+1} \otimes \mathcal{E}_k \otimes \mathcal{F}_{k+1}^n \otimes \dots \otimes \mathcal{F}_p^n$. Alors, l'espace \mathcal{X}_k , SEV de dimension p_k fixée de \mathcal{E}_k , tel que $\|A - \mathcal{P}_{\mathcal{X}_k}(A)\|$ soit minimum pour $\mathcal{F}_1^{n+1}, \dots, \mathcal{F}_{k-1}^{n+1}, \mathcal{F}_{k+1}^n, \dots, \mathcal{F}_p^n$ fixés est le SEV engendré par les p_k premières composantes principales de A_{kn} , et l'on a :

$$\|A - \mathcal{P}_{\mathcal{X}_k}(A)\| \leq \|A - \mathcal{P}_{\mathcal{X}_{k-1}}(A)\|$$

9 - Posons alors $\mathcal{F}_k^{n+1} = \mathcal{X}_k$.

10 - On a alors construit un p-uplet $(\mathcal{F}_1^{n+1}, \dots, \mathcal{F}_p^{n+1})$ tel que :

$$\|A - \mathcal{P}_{\mathcal{X}_{n+1}}(A)\| \leq \|A - \mathcal{P}_{\mathcal{X}_n}(A)\|$$

11 - L'algorithme ainsi proposé est donc convergent.

6 - EXEMPLE DU MODÈLE TUCKER-4

1. NOTATIONS

1 - Nous allons, à titre d'exemple, développer les calculs précédents dans le cas $p = 4$. Ce développement est assez fastidieux, et nous présenterons ensuite une façon plus compacte pour effectuer les calculs.

2 - Soit A un tenseur de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3 \otimes \mathcal{E}_4$, de coefficients (α_{ijkl}) , pour

$i = 1, \dots, n_1$; $j = 1, \dots, n_2$; $k = 1, \dots, n_3$; $l = 1, \dots, n_4$.

3 - Soient quatre dimensions p_1 , p_2 , p_3 et p_4 .

2. INITIALISATION

1 - Nous choisissons $(\mathcal{F}_1^0, \mathcal{F}_2^0, \mathcal{F}_3^0, \mathcal{F}_4^0)$ comme suit.

2 - Soit M_1 la matrice de l'application : $x \rightarrow A.x$. M_1 est une matrice à n_1 colonnes et $n_2 \times n_3 \times n_4$ lignes. \mathcal{F}_1^0 est alors l'espace vectoriel engendré par les p_1 premières composantes principales de M_1 .

3 - Pour cela, calculons $M_1' M_1$. $M_1' M_1$ est une matrice carrée à n_1 lignes et colonnes de coefficient général μ_{1rs} défini par :

$$\mu_{1rs} = \sum_j \sum_k \sum_l \alpha_{rjkl} \alpha_{sjkl}$$

4 - De même, \mathcal{F}_2^0 est défini comme l'espace vectoriel engendré par les p_2 premières composantes principales de la matrice M_2 de l'application : $x \rightarrow 2A.x$ ($x \in \mathcal{E}_2$). \mathcal{F}_2^0 est donc engendré par les p_2 premiers vecteurs propres de la matrice $M_2' M_2$ de coefficient μ_{2rs} défini par :

$$\mu_{2rs} = \sum_l \sum_k \sum_i \alpha_{lrkl} \alpha_{iskl}$$

5 - De même, \mathcal{F}_3^0 est engendré par les p_3 premiers vecteurs propres de la matrice de coefficient μ_{3rs} défini par :

$$\mu_{3rs} = \sum_l \sum_j \sum_i \alpha_{l j r i} \alpha_{l j s i}$$

et \mathcal{F}_4^0 par les p_4 premiers vecteurs propres de la matrice de coefficient μ_{4rs} défini par :

$$\mu_{4rs} = \sum_l \sum_j \sum_k \alpha_{l j k r} \alpha_{l j k s}$$

3. ITERATION

1 - Soit donc un quadruplet $(\mathcal{F}_1^n, \mathcal{F}_2^n, \mathcal{F}_3^n, \mathcal{F}_4^n)$. Nous allons en déduire un quadruplet $(\mathcal{F}_1^{n+1}, \mathcal{F}_2^{n+1}, \mathcal{F}_3^{n+1}, \mathcal{F}_4^{n+1})$ tel que $\|A - \mathcal{P}_n(A)\| \leq \|A - \mathcal{P}_{n+1}(A)\|$.

2 - Nous noterons par :

- v_{11}, \dots, v_{1p_1} une BON de \mathcal{F}_1^n ,
- v_{21}, \dots, v_{2p_2} " " " \mathcal{F}_2^n ,
- v_{31}, \dots, v_{3p_3} " " " \mathcal{F}_3^n ,
- v_{41}, \dots, v_{4p_4} " " " \mathcal{F}_4^n .

3 - Nous noterons également par :

- w_{11}, \dots, w_{1p_1} une BON de \mathcal{F}_1^{n+1} ,
- w_{21}, \dots, w_{2p_2} " " " \mathcal{F}_2^{n+1} ,
- w_{31}, \dots, w_{3p_3} " " " \mathcal{F}_3^{n+1} ,
- w_{41}, \dots, w_{4p_4} " " " \mathcal{F}_4^{n+1} .

4 - Les vecteurs v_{1i} , v_{2i} , v_{3i} et v_{4i} ont respectivement comme coordonnées sur les bases de travail $(e_{1j})_j$, $(e_{2j})_j$, $(e_{3j})_j$ et $(e_{4j})_j$: φ_{1ij} , φ_{2ij} , φ_{3ij} , et φ_{4ij} .

5 - De même, les vecteurs w_{1i} , w_{2i} , w_{3i} et w_{4i} ont respectivement comme coordonnées sur les bases de travail $(e_{1j})_j$, $(e_{2j})_j$, $(e_{3j})_j$ et $(e_{4j})_j$: ψ_{1ij} , ψ_{2ij} , ψ_{3ij} , et ψ_{4ij} .

1. Projection sur $\mathcal{E}_1 \otimes \mathcal{F}_2^n \otimes \mathcal{F}_3^n \otimes \mathcal{F}_4^n$

1 - La première étape est de projeter A sur $\mathcal{E}_1 \otimes \mathcal{F}_2^n \otimes \mathcal{F}_3^n \otimes \mathcal{F}_4^n$.

2 - $\mathcal{E}_1 \otimes \mathcal{F}_2^n \otimes \mathcal{F}_3^n \otimes \mathcal{F}_4^n$ est connu par une base orthonormée, car nous connaissons une base orthonormée de \mathcal{E}_1 , de \mathcal{F}_2^n , de \mathcal{F}_3^n et de \mathcal{F}_4^n .

3 - La projection orthogonale de A sur $\mathcal{E}_1 \otimes \mathcal{F}_2^n \otimes \mathcal{F}_3^n \otimes \mathcal{F}_4^n$ est donc définie par :

$$\hat{A}_{1n} = \sum_r \sum_s \sum_t \sum_u \langle A, e_{1r} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle e_{1r} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u}$$

4 - Sur la BON $e_{1i} \otimes e_{2j} \otimes e_{3k} \otimes e_{4l}$ de travail, les coefficients de \hat{A}_{1n}

s'écrivent donc :

$$\begin{aligned}\hat{\alpha}_{1nijkl} &= \langle \hat{A}_{1n}, e_{1i} \otimes e_{2j} \otimes e_{3k} \otimes e_{4l} \rangle \\ &= \sum_r \sum_s \sum_t \sum_u \langle A, e_{1r} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle \langle e_{1r} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u}, e_{1i} \otimes e_{2j} \otimes e_{3k} \otimes e_{4l} \rangle \\ &= \sum_s \sum_t \sum_u \langle A, e_{1i} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle \langle v_{2s}, e_{2j} \rangle \langle v_{3t}, e_{3k} \rangle \langle v_{4u}, e_{4l} \rangle\end{aligned}$$

5 - On peut écrire :

$$\langle v_{2s}, e_{2j} \rangle \langle v_{3t}, e_{3k} \rangle \langle v_{4u}, e_{4l} \rangle = \varphi_{2sj} \varphi_{3tk} \varphi_{4ul}.$$

Il reste donc à calculer $\langle A, e_{1r} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle$, calcul qui est l'objet du point suivant.

$$\begin{aligned}8 - \langle A, e_{1i} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle &= \\ &= \langle \sum_m \sum_w \sum_p \sum_q \alpha_{mwpq} e_{1m} \otimes e_{2w} \otimes e_{3p} \otimes e_{4q}, e_{1i} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle \\ &= \sum_m \sum_w \sum_p \sum_q \alpha_{mwpq} \langle e_{1m}, e_{1i} \rangle \langle e_{2w}, v_{2s} \rangle \langle e_{3p}, v_{3t} \rangle \langle e_{4q}, v_{4u} \rangle \\ &= \sum_w \sum_p \sum_q \alpha_{1wpq} \varphi_{2sw} \varphi_{3tp} \varphi_{4uq}\end{aligned}$$

9 - D'où le calcul final de $\hat{\alpha}_{1nijkl}$:

$$\begin{aligned}\hat{\alpha}_{1nijkl} &= \langle \hat{A}_{1n}, e_{1i} \otimes e_{2j} \otimes e_{3k} \otimes e_{4l} \rangle \\ &= \sum_s \sum_t \sum_u \langle A, e_{1i} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle \langle v_{2s}, e_{2j} \rangle \langle v_{3t}, e_{3k} \rangle \langle v_{4u}, e_{4l} \rangle \\ &= \sum_s \sum_t \sum_u \left(\sum_w \sum_p \sum_q \alpha_{1wpq} \varphi_{2sw} \varphi_{3tp} \varphi_{4uq} \right) \varphi_{2sj} \varphi_{3tk} \varphi_{4ul} \\ &= \sum_s \sum_t \sum_u \sum_w \sum_p \sum_q \alpha_{1wpq} \varphi_{2sw} \varphi_{2sj} \varphi_{3tp} \varphi_{3tk} \varphi_{4uq} \varphi_{4ul}\end{aligned}$$

qui est donc le coefficient général du tenseur \hat{A}_{1n} cherché.

2. Calcul de vecteurs orthonormés générant \mathcal{X}_1

1 - Les p_1 vecteurs générant \mathcal{X}_1 sont les p_1 premiers vecteurs propres de l'application linéaire : $U_{A1} : x \rightarrow \hat{A}_{1n} \cdot x$, application linéaire de \mathcal{E}_1 dans $\mathcal{F}_2^n \otimes \mathcal{F}_3^n \otimes \mathcal{F}_4^n$. En effet, comme on le vérifie aisément :

$$\hat{A}_{1n} \cdot x = \left(\sum_r \sum_s \sum_t \sum_u \langle A, e_{1r} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle e_{1r} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \right) \cdot x$$

$$= \sum_s \sum_t \sum_u \left(\sum_r \langle e_{1r}, x \rangle \langle A, e_{1r} \otimes v_{2s} \otimes v_{3t} \otimes v_{4u} \rangle \right) v_{2s} \otimes v_{3t} \otimes v_{4u}$$

qui est bien un tenseur de $\mathcal{F}_2^n \otimes \mathcal{F}_3^n \otimes \mathcal{F}_4^n$.

2 - \hat{A}_{1n} est un tenseur de coefficient $\hat{\alpha}_{1nijkl}$ défini précédemment. Les vecteurs générant l'espace \mathcal{X}_1 sont donc les p_1 premiers vecteurs propres de la matrice carrée d'ordre n_1 de coefficients μ_{rs} définis par :

$$\mu_{rs} = \sum_j \sum_k \sum_l \hat{\alpha}_{rjkl} \hat{\alpha}_{sjkl}$$

qu'il est inutile et fastidieux d'expliciter en fonction du paragraphe précédent, ce qui sera donc évité. Dans la mise en oeuvre du programme, ce développement n'est pas nécessaire : seule la connaissance des $\hat{\alpha}_{1nijkl}$ est nécessaire.

3. Poursuite de l'algorithme

L'algorithme se poursuit aux deux étapes suivantes en remplaçant les v_{1i} par les x_{1i} , BON (qui peut être choisie arbitrairement) de \mathcal{X}_1 , puis en réitérant les calculs précédents après permutation des indices.

4. REMARQUE

A la lumière de ce calcul précédents, qui n'est pas complexe mais fastidieux (la mise en oeuvre sous forme de programme exécutable à partir de cette écriture le sera certainement autant), il est possible de comprendre pourquoi ne pas disposer d'un outil de calcul synthétique (l'algèbre tensorielle) a freiné l'émergence d'algorithmes reposant sur le seul maniement des indices : le calcul final en termes de programmation n'est pas plus simple : il est structuré et la voie pour y parvenir est balisée de façon plus sûre.

7 - ACM-MT D'ORDRE P AVEC MÉTRIQUES

1. Rappel des notations et présentation du problème

Note : Nous notons habituellement $\mathcal{E}_1, \dots, \mathcal{E}_p$ une famille d'espaces vectoriels. Pour éviter toute confusion avec les dimensions p_1, \dots des SEV des \mathcal{E}_1 , nous notons dans ce chapitre $\mathcal{E}_1, \dots, \mathcal{E}_q$ une famille de q espaces vectoriels.

1 - Rappelons que les espaces $\mathcal{E}_1, \dots, \mathcal{E}_q$ sont respectivement munis d'une métrique définie par les MSDP M_1, \dots, M_q . On associe alors canoniquement à $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_q$ une métrique définie par le tenseur T telle que :

$$\langle A, B \rangle_T = \langle A, T.B \rangle$$

avec en particulier :

$$T.(x_1 \otimes \dots \otimes x_q) = (M_1.x_1) \otimes \dots \otimes (M_q.x_q)$$

où $x \in \mathcal{E}_1, \dots, x_q \in \mathcal{E}_q$.

2 - Le problème de l'ACP-MT avec les métriques définies par M_1, \dots, M_q est alors le suivant :

- on connaît les nombres n_1, \dots, n_q , dimensions de $\mathcal{E}_1, \dots, \mathcal{E}_q$.
- on se donne les nombres p_1, \dots, p_q , dimensions de SEV inconnus des \mathcal{E}_1
- on recherche les SEV $\mathcal{F}_1, \dots, \mathcal{F}_q$ de dimensions p_1, \dots, p_q tels que :

$\|A - \hat{p}(A)\|_T$ soit minimum.

3 - Par rapport au problème simple de l'ACP-MT, seule la métrique utilisée sur $\xi_1 \otimes \dots \otimes \xi_q$ est différente. Elle est définie par un tenseur T et intervient dans les calculs, qui suivent la même logique. Notamment, les inconnues du problème sont toujours les SEV $\mathcal{F}_1, \dots, \mathcal{F}_q$.

2. Expression du problème

Le problème s'exprime ainsi :

1 - Trouver une écriture de A sous la forme :

$$A = \sum_{i_1=1}^{p_1} \dots \sum_{i_q=1}^{p_q} \xi_{i_1 \dots i_q} f_{i_1 1} \otimes \dots \otimes f_{i_q 1} + R$$

où : $\|R\|_T$ est minimum,
 p_1, \dots, p_q sont fixés,
les $(f_{k i_k})_k$ forment des systèmes orthonormés
pour les métriques définies par les MSDP M_k

2 - L'expression en termes de coordonnées n'a pas changée : seules les contraintes sur les ε_{pqr} changent :

$$\alpha_{k_1 \dots k_q} = \sum_{i_1 \dots i_q} \xi_{i_1 \dots i_q} \varphi_{i_1 1 k_1} \varphi_{i_2 2 k_2} \dots \varphi_{i_q q k_q} + \varepsilon_{k_1 \dots k_q}$$

qui se note (plus simplement ?) en :

$$\alpha_{\vec{k}} = \sum_{\vec{i}} \xi_{\vec{i}} \left[\prod_j \varphi_{j i_j k_j} \right]$$

où nous rappelons que :

- $(\alpha_{k_1 \dots k_q})$ sont les coordonnées de A ,
- $(\varphi_{11_1 k_1})$ les coordonnées d'une matrice F_1 dont les colonnes sont les f_{11_1}
- $(\varphi_{21_2 k_2})$ " " F_2 " " f_{21_2}
- ...
- $(\varphi_{q1_q k_q})$ " " F_q " " f_{q1_q}

3. Solution du problème

Nous ne présentons que la solution utilisant les isométries entre différents espaces insuites par M_1, \dots, M_q : une solution directe est également possible, mais cela alourdirait inutilement les notations.

Rappelons que les matrices P_i sont les MSDP telles que $P_i^2 = M_i$.

■ Définition de l'isométrie

1 - Rappelons que l'on définit une isométrie ι pour chacun des espaces $\mathcal{E}_1, \dots, \mathcal{E}_q$ comme suit :

$$\iota(x_j) = P_j^{-1}(x_j)$$

2 - Cette isométrie est alors étendue à $\mathcal{M}(\mathcal{E}_1, \dots, \mathcal{E}_q)$ par le produit tensoriel. En particulier :

$$\iota(x_1 \otimes \dots \otimes x_q) = \iota(x_1) \otimes \dots \otimes \iota(x_q) = (P_1^{-1} \cdot x_1) \otimes \dots \otimes (P_q^{-1} \cdot x_q)$$

■ Solution de l'ACP-MT avec métriques

La solution est alors très rapide.

1 - Soit $A = \sum_{1 \dots q} \xi_{1 \dots q} f_{11_1} \otimes \dots \otimes f_{q1_q} + R$ le modèle de l'ACP-MT de A avec métriques unités.

2 - $\left\| A - \sum_{1..q} \xi_{1..q} f_{11_1} \otimes \dots \otimes f_{q1_q} \right\|^2$ est donc minimum.

3 - $\left\| \iota \left(A - \sum_{1..q} \xi_{1..q} f_{11_1} \otimes \dots \otimes f_{q1_q} \right) \right\|_T^2$ est donc minimum pour la norme définie par T.

4 - Ainsi, les $\xi_{1..q} \iota(f_{11_1}) \otimes \dots \otimes \iota(f_{q1_q})$ sont les composantes du modèle ACP-MT $\iota(A)$ pour la norme définie par T.

5 - D'où : si $\xi_{1..q} f_{11_1} \otimes \dots \otimes f_{q1_q}$ sont les composantes du modèle ACP-MT de A pour la norme définie par T, alors, $\xi_{1..q} (P_1.f_{11_1}) \otimes \dots \otimes (P_q.f_{q1_q})$ sont les composantes du modèle de $\iota^{-1}(A)$ pour la norme unité, que l'on sait calculer.

■ Résultat

Pour calculer le modèle ACP-MT de A avec les métriques M_1, \dots, M_q , on peut procéder comme suit :

1 - Calcul des matrices $P_1 = \sqrt{M_1}, \dots, P_q = \sqrt{M_q}$.

2 - Calcul de $B = \iota^{-1}(A)$ par $B = \sum_{1..q} P_1.e_{11_1} \otimes \dots \otimes P_q.e_{q1_q}$

3 - Calcul des BON $(u_{11_1}), \dots, (u_{q1_q})$ ainsi que des coefficients $\xi_{1_1 \dots 1_q}$ du modèle ACP-MT de B

4 - Calcul de $f_{11_1} = P_1^{-1}.u_{11_1}, \dots, f_{q1_q} = P_q^{-1}.u_{q1_q}$.

MODELE PARAFAC / CANDECOMP D'ORDRE 3

1. ORIGINES ET DÉVELOPPEMENTS DU MODÈLE PARAFAC / CANDECOMP

1. PROBLEME A RESOUDRE

Le modèle ParaFac / CanDecomp est une extension naturelle aux tenseurs d'ordre 3 (et plus) de la décomposition d'Eckart-Young des matrices (ECKART & YOUNG, 1936, MIRSKY, 1963 ; SCHMIDT, 1901) : étant donnée une matrice A de rang p , on recherche une matrice de même dimension mais de rang $q < p$ qui approche A au mieux au sens d'une métrique donnée. La métrique généralement retenue est celle de Frobenius : $\|A\|^2 = \text{Tr}(A'A)$, et $\langle A, B \rangle = \text{Tr}(A'B)$.

Le modèle PARAFAC/CANDECOMP peut s'intituler ainsi : étant donné un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ de rang r , trouver un tenseur \hat{A} de rang $q < r$ de mêmes dimension ($i, e \in \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$) tel que $\|A - \hat{A}\|$ minimum. Cette définition s'étend sans difficultés à un ordre p quelconque : étant donné un tenseur A de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ de rang r , trouver un tenseur \hat{A} de rang $q < r$ de mêmes dimension ($i, e \in \mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$) tel que $\|A - \hat{A}\|$ minimum.

L'existence et la recherche de \hat{A} , connaissant A , par voie algébrique directe ou numérique approchée, fait l'objet du présent chapitre.

2. ORIGINE DU MODELE PARAFAC/CANDECOMP

Ce modèle a été présenté quasi simultanément en 1970 sous deux appellations différentes, par deux équipes indépendantes : modèle CANDECOMP dans CARROLL & CHANG (1970) et modèle PARAFAC dans HARSHMAN (1970).

Une présentation du modèle ParaFac se trouve dans HARSHMAN & LUNDY, (1984) ; KROONENBERG (1983) ; KRUSKAL (1984). La méthode a été proposée en 1970 par HARSHMAN (HARSHMAN, 1970), qui l'a amélioré ensuite (HARSHMAN, 1972 et HARSHMAN, 1976). Malheureusement, la thèse de HARSHMAN (HARSHMAN, 1976) présentant sa méthode, n'a pas été publiée. Et si de nombreuses applications ont été réalisées outre atlantique, l'algorithme numérique répondant à la question (pourtant clairement posé) est quasi inconnu en France et, à notre connaissance, n'a pas été publié. Il repose cependant sur un algorithme d'approximation par moindres carrés alternés, extension de la méthode de WOLD (1966) pour la décomposition d'ECKART-YOUNG des matrices.

En revanche, CANDECOMP, acronyme de "Canonical Decomposition of General 3-Way tables", a été proposé par CARROLL & CHANG (1970), avec des exemples et la présentation de solutions numériques. Les auteurs l'étendent sans difficulté à des tables d'ordre n quelconque, dès l'article originel, où ils indiquent la réalisation par leurs soins d'un programme pour l'étude des tables à jusqu'à 7 entrées, 7 inclus. Aussi, c'est la méthode CANDECOMP qui nous servira de fil pour cette présentation.

La méthode numérique proposée par CARROLL & CHANG (1970) repose sur un rôle parfaitement symétrique des trois composantes, et une optimisation alternée sur ces composantes, qu'ils rattachent aux méthodes de relaxation pour l'optimisation des systèmes à plusieurs inconnues. Une fois le problème clairement posé, la méthode d'optimisation proposée par CARROLL & CHANG (1970) est naturelle, et mène simplement au résultat.

CARROLL & CHANG précisent que leur algorithme leur avait été suggéré par la lecture de deux travaux plus anciens : WOLD (1966) et LYTKENS (1966), qui développent cette méthode pour la recherche des q premières composantes principales d'une matrice. WOLD (1966) développe un algorithme numérique de recherche directe des premières composantes principales d'une matrice (Non Linear Iterative Least Square), que les auteurs de CanDecomp ont adapté aux tableaux à plus de deux entrées. WOLD (1966) a proposé dans son article l'extension aux tableaux à trois entrées, mais sans joindre de procédure de programmation. L'algorithme proposé par WOLD est certainement plus

ancien que 1966 : il est également présenté dans ANDERSON (1958).

Peut être grâce à sa simplicité limpide (après publication ...), le modèle CANDECOMP n'a pas fait l'objet d'aussi nombreux travaux que le modèle de TUCKER. Un progrès notable a été réalisé en 1980, par la prise en compte de contraintes linéaires sur les composantes, par le modèle CANDELINC (CARROLL, PRUZANSKY & KRUSKAL, 1980), qui est un modèle CANDECOMP sous contraintes.

Curieusement, alors que ce progrès ne posait pas de difficultés majeures, aucune publication n'a à notre connaissance étudié et résolu le problème d'un modèle CANDECOMP ou CANDELINC avec des métriques particulières sur les espaces.

3. DISTINCTION ENTRE LE MODELE CANDECOMP ET LE MODELE DE TUCKER

Pendant une longue période, une grande confusion a régné sur les relations entre le modèle CANDECOMP et le modèle de TUCKER, alors qu'à nos yeux ils sont de nature différente, et ne doivent pas être confondus.

Les relations entre le modèle de TUCKER et le modèle PARAFAC ont été abordées dans CARROLL & CHANG (1970), qui peut être ont bien involontairement contribué à diffuser cette confusion : ils remarquent simplement que leur modèle peut s'écrire comme un modèle de TUCKER particulier où le tenseur - noyau a tous ses termes diagonaux égaux à l'unité, et les autres nuls. Mais ils remarquent également que leur modèle répond à la recherche de la meilleure approximation, contrairement à celui de TUCKER (la publication KROONENBERG & DE LEEUW (1980) viendra 10 ans plus tard), et considèrent que leur modèle est plus simple.

Ce problème a également été abordé par JAFFRENOU (1978), qui présente le modèle CANDECOMP (sous le nom de modèle de CARROLL) comme suit : étant donné un cube (α_{ijk}) , trouver une décomposition de la forme $\alpha_{ijk} = \sum_{t=1}^r \lambda_{it} \mu_{jt} \nu_{kt}$. JAFFRENOU (1978), cependant, présente ainsi la

décomposition totale du tenseur A , et non sa meilleure approximation pour un r fixé : le problème étudié est celui du rang, et non celui du modèle CANDECOMP. D'ailleurs, aucune indication n'est fournie pour le choix ou l'estimation de r . Et si l'algorithme proposé par JAFFRENOU (1978) peut converger vers la décomposition totale cherchée, rien ne prouve qu'elle converge vers une solution de la meilleure approximation pour un rang donné : il s'agit de la même distinction, *mutatis mutandis*, qu'entre le modèle de décomposition de TUCKER et l'algorithme des moindres carrés de KROONENBERG & DE LEEUW (1980).

L'opinion de JAFFRENOU (1978) sur la similitude entre le modèle de TUCKER et le modèle CANDECOMP est également assez perturbante : dans la même page [p. 78], il affirme à la fois "le modèle de CARROLL est un cas particulier du modèle de TUCKER" et "notre opinion est qu'il n'y a pas de rapport entre les décompositions [...] proposées par TUCKER et CARROLL".

Plus en avant dans le temps, KROONENBERG (1983) présente le modèle PARAFAC comme un cas particulier du modèle de TUCKER, qui serait le modèle le plus général : il suffirait de choisir le tenseur identité pour le tenseur - noyau (core matrix) dans le modèle de TUCKER pour obtenir le modèle PARAFAC. A cette remarque, nous présentons deux objections : 1) rien n'impose que le tenseur - noyau soit cubique, et un tenseur identité parallélipédique n'a plus de sens ; 2) rien n'impose que les composantes du modèle PARAFAC soient orthogonales, alors qu'un modèle de TUCKER peut toujours être orthogonal. Un contre exemple de modèle CANDECOMP d'un tenseur de $\mathcal{A}_3(\mathbb{R}^3)$ à deux composantes non orthogonales peut facilement être construit.

Cette confusion est entretenue dans KROONENBERG (1989b), où, à l'occasion d'une généralisation de la décomposition d'une matrice en valeurs singulières aux tableaux à trois entrées, cet auteur écrit le modèle CANDECOMP avec des sommations sur trois indices :

$$z_{ijk} = \sum_{p=1}^P \sum_{q=1}^Q \sum_{r=1}^R a_{ip} b_{jq} c_{kr} g_{pqr}$$

en indiquant que $g_{pqr} = 0$ sauf si $p = q = r$. L'écriture devient alors :

$$z_{ijk} = \sum_{s=1}^S a_{is} b_{js} c_{ks} g_{sss}$$

Si $P = Q = R = S$.

Or, cette écriture est celle du modèle de TUCKER. La confusion est évidente car, outre l'écriture, KRONENBERG (1989b) traite de la généralisation de la décomposition en valeurs singulières d'une matrice. Une contrainte d'orthonormalité est imposée aux vecteurs a_i , b_i et c_i , ce qui est totalement absent du modèle CANDECOMP. Remarquons, à la décharge de l'auteur, que seul ce glissement de terminologie nous a intrigué dans l'article mentionné, et non le contenu et l'analyse des modèles.

Remarquons que la juste généralisation de la DVS d'une matrice est un problème qui n'a pas de solution générale pour les tableaux à 3 entrées ou plus, et est par là même un leurre. La DVS est à la base de l'ACP, qui ne peut se généraliser de façon canonique pour les tableaux à plusieurs entrées. l'analyse 3-modes, le modèle de TUCKER, les modèles Parafac / Candecomp sont autant d'extension de l'ACP aux tableaux à trois entrées et plus, qui ne peuvent d'agrèger en une méthode qui se résumerait à une généralisation de la DVS. Deux entrées est une situation privilégiée, qui éclate en plusieurs directions d'extensions possibles dès que ce seuil est franchi.

Cette la confusion est encore palpable dans CARROLL, PRUZANSKY & KRUSKAL (1981), à propos du modèle CANDELINC. Les auteurs établissent le modèle CANDELINC par un calcul algébrique rigoureux qui prend en compte de contraintes linéaires dans CANDECOMP, à l'aide d'un formalisme algébrique nouveau (les produits noté par \circ_n) qui sont en fait la combinaison d'un produit intérieur, ou une contraction, et d'une permutation. Ils étendent ensuite ce résultat au modèle de TUCKER en invoquant la parenté d'écriture entre les deux modèles.

La confusion continue dans les travaux de KIERS (1989), qui reprend l'idée d'aborder les modèles de la famille CANDECOMP comme des modèles de TUCKER en imposant des contraintes sur le tenseur - noyau (en l'occurrence,

d'être le tenseur identité). Le modèle retenu par KIERS (1989) est le modèle INDSCAL (Individual difference in orientation scaling), présenté comme un modèle CANDECOMP symétrique sur les deux premières dimensions : $\hat{S}_{11,j} = \sum_1 x_{11} x_{1,1} c_{j1}$. Sans explications, l'auteur enchaîne après cette présentation pour affirmer que la solution de ce modèle passe par une ACP sous contrainte, qui est en fait une étape de résolution du modèle de TUCKER, et propose de considérer le modèle INDSCAL comme un modèle de TUCKER sous contraintes.

□

Cette confusion peut être levée aisément en observant les éléments suivants :

- les inconnues du modèle de TUCKER sont des sous-espaces
- les inconnues du modèle CANDECOMP sont des composantes.

Etant donné un tenseur A donné, le modèle de TUCKER s'écrit :

$$A = \sum_{i=1}^{p_1} \sum_{j=1}^{p_2} \sum_{k=1}^{p_3} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k} + R$$

et le modèle CANDECOMP :

$$A = \sum_{i=1}^r x_i \otimes y_i \otimes z_i + R$$

Le modèle de TUCKER revient à chercher les trois sous espaces \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 de dimensions p_1 , p_2 et p_3 fixés tels que $\|A - \mathcal{P}(A)\|$ soit minimal, et le modèle CANDECOMP revient à chercher les r composantes décomposées $X_i = x_i \otimes y_i \otimes z_i$ telles que $\|A - \sum_i X_i\|$ soit minimale.

Remarquons enfin que le tenseur $\sum_{ijk} \xi_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k}$ est un tenseur de $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$, avec inclusion stricte de \mathcal{F}_1 dans \mathcal{E}_1 . Or, tous les SEV de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ ne sont pas de cette forme, notamment celui engendré par les X_i (qui est l'ensemble de tous les tenseurs de la forme $\sum_i \lambda_i X_i$). Ainsi, les meilleures approximations de A par le modèle de TUCKER pour un triplet (p_1, p_2, p_3) donné et par le modèle CANDECOMP pour un rang r donné sont recherchées dans des sous espaces de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ de nature très différentes.

Dans le modèle de TUCKER, on se donne une famille \mathcal{X} de SEV de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$: les SEV de la forme $\mathcal{T}_1 \otimes \mathcal{T}_2 \otimes \mathcal{T}_3$ à dimensions fixées, et on cherche l'espace \mathcal{X} de \mathcal{X} tel que $\|A - \mathcal{P}(A)\|$ soit minimale. Dans le modèle CANDECOMP, on se donne une variété gauche de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$: $\Gamma \equiv \Gamma^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$, et l'on cherche le point X de Γ tel que $\|A - X\|$ soit minimal.

Il n'y a, à nos yeux, pas de relations entre les modèles de TUCKER et le modèle CANDECOMP, qui répondent à deux questions différentes.

2. PRÉSENTATION DU MODÈLE PARAFAC / CANDECOMP D'ORDRE 3

Pour alléger la présentation, nous utilisons la notion de rang d'un tenseur, qui sera développée plus profondément dans le chapitre suivant.

1. RANG D'UN TENSEUR

Soient :

- \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 trois espaces vectoriels réels.
- Soit A un tenseur de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ de coefficient général α_{ijk} .
- r un nombre entier.

On dit que le tenseur A est de rang r s'il peut s'écrire sous la forme :

$$A = \sum_{m=1}^r x_m \otimes y_m \otimes z_m$$

et que r soit le plus petit nombre entier pour lequel une telle décomposition existe.

De façon analytique, les coefficients de A s'écrivent alors :

$$\alpha_{ijk} = \sum_{m=1}^r \xi_{im} \eta_{jm} \zeta_{km}$$

où $(\xi_{im})_i$, $(\eta_{jm})_j$ et $(\zeta_{km})_k$ sont respectivement les coordonnées de x_m , y_m et z_m .

Rappelons que l'on note $\Gamma^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ la variété de Sègre d'ordre r de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$, soit l'ensemble des tenseurs de rang r de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$.

2. MODELE PARAFAC / CANDECOMP D'ORDRE 3

1 - Soit A un tenseur de rang p . Etant donné un entier $r < p$, on appelle "modèle CanDecomp / ParaFac à l'ordre r de A " la meilleure approximation de A par un tenseur de rang r . On note \hat{A} cette approximation.

2 - \hat{A} est donc la projection de A sur $\Gamma^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$, soit le tenseur de $\Gamma^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ tel que :

$$\|A - \hat{A}\| \text{ minimum}$$

3 - Cette condition s'écrit également : étant donnés

- un tenseur $A \in \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$
- r un nombre entier fixé,

on cherche une décomposition :

$$A = \sum_{i=1}^r x_i \otimes y_i \otimes z_i + R$$

$$\text{où : } \forall i, x_i \in \mathcal{E}_1, y_i \in \mathcal{E}_2, z_i \in \mathcal{E}_3, R \in \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$$

$$\text{telle que } \|R\| \text{ minimale}$$

les tenseurs $x_i \otimes y_i \otimes z_i$ sont les composantes principales de A du modèle d'ordre r . On note

$$f_r(A) = \sum_{i=1}^r x_i \otimes y_i \otimes z_i$$

le tenseur somme de r composantes principales de A

4 - Remarquons que, contrairement à l'ACP, il n'y a pas emboîtement des composantes entre les modèles $f_r(A)$ et $f_{r+s}(A)$.

3. MÉTHODES GÉOMÉTRIQUES DE RÉOLUTION

1. PRESENTATION DES METHODES

1 - Le problème posé est en fait celui de trouver la projection de A sur la variété de Sègre de rang r : $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$. Si X est cette projection, X vérifie bien $\|A-X\|^2$ minimal, où $X \in \Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$. Deux méthodes géométriques sont alors envisagées, qui conduisent chacune à la résolution d'un système de plusieurs équations algébriques à plusieurs inconnues pour calculer la solution.

2 - L'une, de façon très générale, explicite le fait que, X étant la projection de A sur $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$, $A-X$ est orthogonal au plan tangent en X à $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$. Les conditions obtenues sont alors nécessaires, mais non suffisantes dans le cas général. Cette méthode est appelée dans la suite "projection sur une variété de Sègre".

3 - L'autre tire parti du fait que la variété de Sègre de rang r $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ est un cône. Les conditions obtenues sont également nécessaires, mais non suffisantes. Elle est appelée dans la suite : "projection sur un cône".

2. PROJECTION SUR UNE VARIETE DE SEGRE

1. Expression algébrique

1. Lemme

Rechercher les r premières composantes principales d'un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ revient à résoudre le système :

$$\forall i \leq r, \left\{ \begin{array}{l} (2A \cdot y_i) \cdot z_i = \sum_{j=1}^r \langle y_i, y_j \rangle \langle z_i, z_j \rangle x_j \\ (3A \cdot z_i) \cdot x_i = \sum_{j=1}^r \langle z_i, z_j \rangle \langle x_i, x_j \rangle y_j \\ (A \cdot x_i) \cdot y_i = \sum_{j=1}^r \langle x_i, x_j \rangle \langle y_i, y_j \rangle z_j \end{array} \right.$$

2 - Démonstration

1 - Désignons par :

• \mathcal{T} l'espace vectoriel tangent en X à $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$. D'où :

$$\forall T \in \mathcal{T}, \langle A-X, T \rangle = 0$$

• $\mathcal{X} \otimes M$ l'espace composé des tenseurs $X \otimes M$ où X décrit un ensemble donné \mathcal{X} .

2 - Sachant que :

$$\mathcal{T} = \sum_i \mathcal{E}_1 \otimes y_i \otimes z_i + x_i \otimes \mathcal{E}_2 \otimes z_i + x_i \otimes y_i \otimes \mathcal{E}_3.$$

on en déduit :

$$\forall (u_i, v_i, w_i)_i, \langle A-X, \sum_i u_i \otimes y_i \otimes z_i + x_i \otimes v_i \otimes z_i + x_i \otimes y_i \otimes w_i \rangle = 0.$$

3 - D'où, en particulier :

$$\forall i \leq r, \forall u_i \in \mathcal{E}_1, \forall v_i \in \mathcal{E}_2, \forall w_i \in \mathcal{E}_3, \left\{ \begin{array}{l} \langle A-X, u_i \otimes y_i \otimes z_i \rangle = 0 \\ \langle A-X, x_i \otimes v_i \otimes z_i \rangle = 0 \\ \langle A-X, x_i \otimes y_i \otimes w_i \rangle = 0 \end{array} \right.$$

4 - En se rappelant que $X = \sum_{i=1}^r x_i \otimes y_i \otimes z_i$, on en déduit, par exemple pour la dernière des trois égalités précédentes :

$$\forall i \leq r, \forall w_i, \langle A, x_i \otimes y_i \otimes z_i \rangle = \langle X, x_i \otimes y_i \otimes w_i \rangle$$

$$\Leftrightarrow \langle A, x_i \otimes y_i \otimes w_i \rangle = \langle \sum_{j=1}^r x_j \otimes y_j \otimes z_j, x_i \otimes y_i \otimes w_i \rangle$$

$$\Leftrightarrow (A \cdot x_i) \cdot y_i = \sum_j \langle x_i, x_j \rangle \langle y_i, y_j \rangle z_j$$

5 - Et, en reprenant le même calcul pour les deux autres équations, on obtient :

$$\forall i \leq r, (2A \cdot y_i) \cdot z_i = \sum_j \langle y_i, y_j \rangle \langle z_i, z_j \rangle x_j$$

$$\text{et : } \forall i \leq r, (3A \cdot z_i) \cdot x_i = \sum_j \langle z_i, z_j \rangle \langle x_i, x_j \rangle y_j$$

6 - D'où, si $\dim(\mathcal{E}) = n_1$, $\dim(\mathcal{E}_2) = n_2$, $\dim(\mathcal{E}_3) = n_3$, le système de $r(n_1+n_2+n_3)$ équations à $r(n_1+n_2+n_3)$ inconnues :

$$\forall i \leq r, \begin{cases} (2A \cdot y_i) \cdot z_i = \sum_{j=1}^r \langle y_i, y_j \rangle \langle z_i, z_j \rangle x_j \\ (3A \cdot z_i) \cdot x_i = \sum_{j=1}^r \langle z_i, z_j \rangle \langle x_i, x_j \rangle y_j \\ (A \cdot x_i) \cdot y_i = \sum_{j=1}^r \langle x_i, x_j \rangle \langle y_i, y_j \rangle z_j \end{cases}$$

2. Expression analytique

1. Lemme

L'expression analytique du système précédent est :

$$\forall i \leq r, \begin{cases} \forall m \leq n_1, \sum_{n=1}^{n_2} \sum_{p=1}^{n_3} \alpha_{mnp} \eta_{in} \zeta_{ip} = \sum_{j=1}^r \sum_{v=1}^{n_2} \sum_{w=1}^{n_3} \eta_{iv} \eta_{jv} \zeta_{iw} \zeta_{jw} \xi_{jm} \\ \forall n \leq n_2, \sum_{m=1}^{n_1} \sum_{p=1}^{n_3} \alpha_{mnp} \xi_{im} \zeta_{ip} = \sum_{j=1}^r \sum_{v=1}^{n_1} \sum_{w=1}^{n_3} \xi_{iv} \xi_{jv} \zeta_{iw} \zeta_{jw} \eta_{jn} \\ \forall p \leq n_3, \sum_{m=1}^{n_1} \sum_{n=1}^{n_2} \alpha_{mnp} \xi_{im} \eta_{in} = \sum_{j=1}^r \sum_{v=1}^{n_1} \sum_{w=1}^{n_2} \xi_{iv} \xi_{jv} \eta_{iw} \eta_{jw} \zeta_{jp} \end{cases}$$

avec :

$$\begin{aligned}
f_r(\mathbf{A}) &= \sum_{j=1}^r \mathbf{x}_j \otimes \mathbf{y}_j \otimes \mathbf{z}_j, \\
\mathbf{A} &= \sum_{i,j,k} \alpha_{ijk} \mathbf{E}_{ijk} \\
\mathbf{x}_i &= \sum_j \xi_{ij} \mathbf{e}_{1j} \\
\mathbf{y}_i &= \sum_j \eta_{ij} \mathbf{e}_{2j} \\
\mathbf{z}_i &= \sum_j \zeta_{ij} \mathbf{e}_{3j}
\end{aligned}$$

2. Démonstration

1 - Le système vectoriel précédent s'explicit de la façon suivante en fonction des coordonnées des différents vecteurs et tenseurs :

2 - Posons :

$$\begin{aligned}
\circ \mathbf{A} &= \sum_{i=1}^{n1} \sum_{j=1}^{n2} \sum_{k=1}^{n3} \alpha_{ijk} \mathbf{e}_{1i} \otimes \mathbf{e}_{2j} \otimes \mathbf{e}_{3k} \\
\circ \mathbf{x}_i &= \sum_{j=1}^{n1} \xi_{ij} \mathbf{e}_{1j} ; \quad \mathbf{y}_i = \sum_{j=1}^{n2} \eta_{ij} \mathbf{e}_{2j} ; \quad \mathbf{z}_i = \sum_{j=1}^{n3} \zeta_{ij} \mathbf{e}_{3j}.
\end{aligned}$$

3 - Sachant que $2\mathbf{A} = \sum_{i,j,k} \alpha_{ijk} \mathbf{e}_{2j} \otimes \mathbf{e}_{3k} \otimes \mathbf{e}_{1i}$, on en déduit, pour la première équation :

$$2\mathbf{A} \cdot (\mathbf{y}_i \otimes \mathbf{z}_i) = \sum_{j=1}^r \langle \mathbf{y}_i, \mathbf{y}_j \rangle \langle \mathbf{z}_i, \mathbf{z}_j \rangle \mathbf{x}_j$$

soit :

$$\begin{aligned}
&\left(\sum_{m=1}^{n1} \sum_{n=1}^{n2} \sum_{p=1}^{n3} \alpha_{mnp} \mathbf{e}_{2n} \otimes \mathbf{e}_{3p} \otimes \mathbf{e}_{1m} \right) \cdot \left(\sum_{t=1}^{n2} \sum_{u=1}^{n3} \eta_{it} \zeta_{iu} \mathbf{e}_{2t} \otimes \mathbf{e}_{3u} \right) \\
&= \sum_{j=1}^r \left(\sum_{v=1}^{n2} \eta_{iv} \eta_{jv} \right) \cdot \left(\sum_{w=1}^{n3} \zeta_{iw} \zeta_{jw} \right) \left(\sum_{q=1}^{n1} \xi_{iq} \mathbf{e}_{1q} \right)
\end{aligned}$$

résumé en :

$$\left(\sum_{mnp} \alpha_{mnp} \mathbf{e}_{2n} \otimes \mathbf{e}_{3p} \otimes \mathbf{e}_{1m} \right) \cdot \left(\sum_{tu} \eta_{it} \zeta_{iu} \mathbf{e}_{2t} \otimes \mathbf{e}_{3u} \right)$$

$$= \sum_j (\sum_v \eta_{iv} \eta_{jv}) (\sum_w \zeta_{iw} \zeta_{jw}) (\sum_q \xi_{qj} e_{1q})$$

4 - Or :

$$\begin{aligned} (\sum_{mnp} \alpha_{mnp} e_{2n} \otimes e_{3p} \otimes e_{1m}) \cdot (\sum_{tu} \eta_{it} \zeta_{iu} e_{2t} \otimes e_{3u}) &= \sum_{mnp} \alpha_{mnp} \eta_{in} \zeta_{ip} e_{1m} \\ &= \sum_m (\sum_{np} \alpha_{mnp} \eta_{in} \zeta_{ip}) e_{1m} \end{aligned}$$

et :

$$\begin{aligned} \sum_j (\sum_v \eta_{iv} \eta_{jv}) (\sum_w \zeta_{iw} \zeta_{jw}) (\sum_q \xi_{qj} e_{1q}) &= \sum_{jvwq} \eta_{iv} \eta_{jv} \zeta_{iw} \zeta_{jw} \xi_{jq} e_{1q} \\ &= \sum_q (\sum_{jvw} \eta_{iv} \eta_{jv} \zeta_{iw} \zeta_{jw} \xi_{jq}) e_{1q} \end{aligned}$$

5 - D'où, par égalité de ces deux membres :

$$\forall i \leq r, \forall m \leq n_1, \sum_{np} \alpha_{mnp} \eta_{in} \zeta_{ip} = \sum_{jvw} \eta_{iv} \eta_{jv} \zeta_{iw} \zeta_{jw} \xi_{jm}$$

soit, développé :

$$\forall i \leq r, \forall m \leq n_1, \sum_{n=1}^{n_2} \sum_{p=1}^{n_3} \alpha_{mnp} \eta_{in} \zeta_{ip} = \sum_{j=1}^r \sum_{v=1}^{n_2} \sum_{w=1}^{n_3} \eta_{iv} \eta_{jv} \zeta_{iw} \zeta_{jw} \xi_{jm}$$

6 - De même, sachant que $3A = \sum_{ijk} \alpha_{ijk} e_{3k} \otimes e_{1i} \otimes e_{2j}$, on en déduit, pour la deuxième équation :

$$3A. (z_i \otimes x_i) = \sum_{j=1}^r \langle z_i, z_j \rangle \langle x_i, x_j \rangle y_j$$

soit :

$$\begin{aligned} \left(\sum_{m=1}^{n_1} \sum_{n=1}^{n_2} \sum_{p=1}^{n_3} \alpha_{mnp} e_{3p} \otimes e_{1m} \otimes e_{2n} \right) \cdot \left(\sum_{t=1}^{n_3} \sum_{u=1}^{n_1} \zeta_{it} \xi_{iu} e_{3t} \otimes e_{1u} \right) \\ = \sum_{j=1}^r \left(\sum_{v=1}^{n_3} \zeta_{iv} \zeta_{jv} \right) \cdot \left(\sum_{w=1}^{n_1} \xi_{iw} \xi_{jw} \right) \left(\sum_{q=1}^{n_2} \eta_{jq} e_{2q} \right) \end{aligned}$$

résumé en :

$$\begin{aligned} (\sum_{mnp} \alpha_{mnp} e_{3p} \otimes e_{1m} \otimes e_{2n}) \cdot (\sum_{tu} \zeta_{it} \xi_{iu} e_{3t} \otimes e_{1u}) \\ = \sum_j (\sum_v \zeta_{iv} \zeta_{jv}) (\sum_w \xi_{iw} \xi_{jw}) (\sum_q \eta_{jq} e_{2q}) \end{aligned}$$

7 - Or :

$$\begin{aligned} (\sum_{mnp} \alpha_{mnp} e_{3p} \otimes e_{1m} \otimes e_{2n}) \cdot (\sum_{tu} \zeta_{1t} \xi_{1u} e_{3t} \otimes e_{1u}) &= \sum_{mnp} \alpha_{mnp} \zeta_{1p} \xi_{1m} e_{2n} \\ &= \sum_n (\sum_{mp} \alpha_{mp} \zeta_{1p} \xi_{1m}) e_{2n} \end{aligned}$$

et :

$$\begin{aligned} \sum_j (\sum_v \zeta_{1v} \zeta_{jv}) (\sum_w \xi_{1w} \xi_{jw}) (\sum_q \eta_{jq} e_{1q}) &= \sum_{jvwq} \zeta_{1v} \zeta_{jv} \xi_{1w} \xi_{jw} \eta_{jq} e_{1q} \\ &= \sum_q (\sum_{jvw} \zeta_{1v} \zeta_{jv} \xi_{1w} \xi_{jw} \eta_{jq}) e_{1q} \end{aligned}$$

8 - D'où, par égalité de ces deux membres :

$$\forall i \leq r, \forall n \leq n_2, \sum_{mp} \alpha_{mp} \zeta_{1p} \xi_{1m} = \sum_{jvw} \zeta_{1v} \zeta_{jv} \xi_{1w} \xi_{jw} \eta_{jn}$$

soit, développé :

$$\forall i \leq r, \forall n \leq n_2, \sum_{m=1}^{n1} \sum_{p=1}^{n3} \alpha_{mp} \xi_{1m} \zeta_{1p} = \sum_{j=1}^r \sum_{v=1}^{n1} \sum_{w=1}^{n3} \xi_{1v} \xi_{jv} \zeta_{1w} \zeta_{jw} \eta_{jn}$$

9 - Et enfin, pour la troisième équation :

$$A.(x_i \otimes y_i) = \sum_{j=1}^r \langle x_i, x_j \rangle \langle y_i, y_j \rangle z_j$$

nous avons le résultat suivant (fourni sans développement) :

$$\forall i \leq r, \forall p \leq n_3, \sum_{m=1}^{n1} \sum_{n=1}^{n2} \alpha_{mp} \xi_{1m} \eta_{1n} = \sum_{j=1}^r \sum_{v=1}^{n1} \sum_{w=1}^{n2} \xi_{1v} \xi_{jv} \eta_{1w} \eta_{jw} \zeta_{jp}$$

10 - D'où le résultat global : les r premières composantes principales

$$f_r(A) = \sum_{j=1}^r x_j \otimes y_j \otimes z_j,$$

avec :

$$A = \sum_{ijk} \alpha_{ijk} E_{ijk}$$

$$x_i = \sum_j \xi_{1j} e_{1j}$$

$$y_i = \sum_j \eta_{1j} e_{2j}$$

$$z_i = \sum_j \zeta_{1j} e_{3j}$$

sont solution du système :

$$\forall i \leq r, \begin{cases} \forall m \leq n_1, \sum_{n=1}^{n_2} \sum_{p=1}^{n_3} \alpha_{mnp} \eta_{in} \zeta_{ip} = \sum_{j=1}^r \sum_{v=1}^{n_2} \sum_{w=1}^{n_3} \eta_{iv} \eta_{jv} \zeta_{iw} \zeta_{jw} \xi_{jm} \\ \forall n \leq n_2, \sum_{m=1}^{n_1} \sum_{p=1}^{n_3} \alpha_{mnp} \xi_{im} \zeta_{ip} = \sum_{j=1}^r \sum_{v=1}^{n_1} \sum_{w=1}^{n_3} \xi_{iv} \xi_{jv} \zeta_{iw} \zeta_{jw} \eta_{jn} \\ \forall p \leq n_3, \sum_{m=1}^{n_1} \sum_{n=1}^{n_2} \alpha_{mnp} \xi_{im} \eta_{in} = \sum_{j=1}^r \sum_{v=1}^{n_1} \sum_{w=1}^{n_2} \xi_{iv} \xi_{jv} \eta_{iw} \eta_{jw} \zeta_{jm} \end{cases}$$

3. Remarque géométrique

Ces $r(n_1+n_2+n_3)$ équations à $r(n_1+n_2+n_3)$ inconnues représentent l'intersection de $r(n_1+n_2+n_3)$ surfaces de dimension 5 dans $\mathbb{R}^{r(n_1+n_2+n_3)}$.

4. Remarque arithmétique

Ce système peut avoir $5^{r(n_1+n_2+n_3)}$ solutions réelles. Par exemple, pour un cube de $\mathbb{R}^{10} \otimes \mathbb{R}^{10} \otimes \mathbb{R}^{10}$ avec 4 composantes, ce nombre vaut 5^{120} , soit environ $7.523 \ 16 \ 10^{83}$. En supposant qu'il faille une seconde pour calculer 10^6 solutions, et sachant qu'il y a $3.15 \ 576 \ 10^7$ secondes par an, le temps nécessaire pour les calculer toutes (afin de choisir la meilleure) est de $2.38 \ 10^{70}$ années, donc environ $1.7 \ 10^{61}$ fois l'âge de l'univers.

Le calcul des solutions du système précédent est donc hors de portée.

5. Exemple

Plus concrètement :

1 - Soit :

◦ e_0, \dots, e_9 une BON de \mathbb{R}^{10} .

- E_{ijk} la BON associée de $\mathcal{A}_3(\mathbb{R}^{10})$: $E_{ijk} = e_i \otimes e_j \otimes e_k$
- $A = \alpha_{000} E_{000} + \dots + \alpha_{999} E_{999}$ (1000 termes) un tenseur de $\mathcal{A}_3(\mathbb{R}^{10})$
- $r = 4$ l'ordre du modèle CanDecomp / ParaFac recherché

2 - Le modèle recherché est alors :

$A = x_1 \otimes y_1 \otimes z_1 + x_2 \otimes y_2 \otimes z_2 + x_3 \otimes y_3 \otimes z_3 + x_4 \otimes y_4 \otimes z_4 + R$, $\|R\|$ minimale
que l'on notera, pour éviter les confusions avec les indices de coordonnées sur une BON de \mathbb{R}^{10} :

$$A = x_i \otimes y_i \otimes z_i + x_j \otimes y_j \otimes z_j + x_k \otimes y_k \otimes z_k + x_l \otimes y_l \otimes z_l + R$$

3 - Si $x_a = x_{a0} e_0 + \dots + x_{a9}$, les inconnues sont :

- $x_{10}, x_{11}, \dots, x_{19}$
- $x_{j0}, x_{j1}, \dots, x_{j9}$
- $x_{k0}, x_{k1}, \dots, x_{k9}$
- $x_{l0}, x_{l1}, \dots, x_{l9}$

qui sont au nombre de 40 pour les vecteurs x_i, x_j, x_k et x_l , soit 120 pour les vecteurs x_a, y_a et z_a où $a \in \{i, j, k, l\}$.

4 - Le système précédent est bien un système de 120 équations algébriques à 120 inconnues, de degré 5, à coefficients réels. Il a donc en général 5^{120} solutions dans \mathbb{C} .

5 - Au delà du calcul proposé précédemment, le degré cinq se comprend heuristiquement par le raisonnement suivant :

- il faut minimiser $\|R\|^2$
- $\|R\|^2$ est un polynôme en x_{10}, \dots, z_{19} de degré 6, car comprenant des termes du type $x_{10}^2 y_{10}^2 z_{10}^2$.
- Par dérivation (par exemple $\partial \|R\|^2 / \partial x_{j3} = 0$), on obtient un polynôme de degré 5. Comme il y a 120 inconnues, il y a 120 dérivées à annuler, soit 120 équations.

6. Croissance du nombre de solutions à l'équation

Dans le cas particulier où $n_1 = n_2 = n_3 = n$, le nombre de solutions du

système d'équations algébriques est 5^{3rn} , soit 125^{rn} . Voici donc les premières puissances de 125, qui donnent une idée de la rapidité de la croissance du nombre de solutions :

rn	125^{rn}
1	125
2	15 625
3	1 953 125
4	244 140 625
5	3.05 175 10^{10}
etc ...	

3. PROJECTION SUR UN CONE

Une autre approche de la solution consiste à se ramener à rechercher un minimum sous contrainte.

1. Remarque géométrique

Nous cherchons $X \in \Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ tel que $\|A - X\|^2$ soit minimum. Comme $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ est un cône de pointe 0, cela revient à rechercher X tel que l'angle entre X et A soit minimum, soit $\langle A, X \rangle$ maximum en imposant à X d'être de norme 1.

2. Expression algébrique

1. Lemme

Rechercher les r premières composantes principales d'un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ revient à résoudre le système :

$$\begin{array}{l} \sum_{i=1}^r \langle A, x_i \otimes y_i \otimes z_i \rangle \text{ maximum} \\ \text{sous la contrainte } \left\| \sum_{i=1}^r x_i \otimes y_i \otimes z_i \right\|^2 = 1 \end{array}$$

2. Démonstration

1 - Notons : $X = \lambda.U$, avec $\|U\| = 1$. On en déduit :

$$\|A-X\|^2 = \langle A-\lambda U, A-\lambda U \rangle$$

2 - Or, $\lambda U \in \mathcal{T}$, et $\forall M \in \mathcal{T}$, $\langle A-\lambda U, M \rangle = 0$. Donc, $\langle A-\lambda U, \lambda U \rangle = 0$.

3 - D'où : $\langle A-\lambda U, A-\lambda U \rangle = \langle A, A-\lambda U \rangle = \|A\|^2 - \lambda \langle A, U \rangle$

4 - Or, $\langle A-\lambda U, \lambda U \rangle = 0 \implies \lambda \langle A, U \rangle = \lambda^2 \|U\|^2 = \lambda^2 \implies \langle A, U \rangle = \lambda$. D'où :

$$\|A-\lambda U\|^2 = \|A\|^2 - \langle A, U \rangle^2$$

5 - Comme on recherche λU tel que $\|A-\lambda U\|^2$ soit minimum, cela revient à rechercher U tel que $\langle A, U \rangle$ soit maximum sous la contrainte $\|U\|^2 = 1$.

6 - Soit, en posant $X = U = \sum_{i=1}^r x_i \otimes y_i \otimes z_i$ avec $\|X\| = 1$:

$$\left\{ \begin{array}{l} \langle A, X \rangle = \langle A, \sum_{i=1}^r x_i \otimes y_i \otimes z_i \rangle = \sum_{i=1}^r \langle A, x_i \otimes y_i \otimes z_i \rangle \text{ maximum} \\ \text{sous la contrainte } \left\| \sum_{i=1}^r x_i \otimes y_i \otimes z_i \right\|^2 = 1. \end{array} \right.$$

3. Remarque

$$\|X\|^2 = \left\| \sum_{i=1}^r x_i \otimes y_i \otimes z_i \right\|^2 = \sum_{i,j=1}^r \langle x_i, x_j \rangle \langle y_i, y_j \rangle \langle z_i, z_j \rangle$$

3. Expression analytique

1. Lemme

L'expression analytique du système algébrique précédent est :

$$\begin{aligned} & \sum_{i=1}^{n1} \sum_{j=1}^{n2} \sum_{k=1}^{n3} \sum_{l=1}^r \alpha_{ijk} \xi_{il} \eta_{lj} \zeta_{lk} \text{ maximum} \\ & \text{sous } \sum_{i,j=1}^r \sum_{k=1}^{n1} \sum_{l=1}^{n2} \sum_{m=1}^{n3} \xi_{ik} \xi_{jk} \eta_{il} \eta_{jl} \zeta_{im} \zeta_{jm} = 1 \end{aligned}$$

2. Démonstration

1 - Posons également :

$$\begin{aligned} \circ A &= \sum_{i=1}^{n1} \sum_{j=1}^{n2} \sum_{k=1}^{n3} \alpha_{ijk} e_{1i} \otimes e_{2j} \otimes e_{3k} \\ \circ x_1 &= \sum_{j=1}^{n1} \xi_{1j} e_{1j} ; \quad y_1 = \sum_{j=1}^{n2} \eta_{1j} e_{2j} ; \quad z_1 = \sum_{j=1}^{n3} \zeta_{1j} e_{3j}. \end{aligned}$$

2 - On en déduit :

$$\langle A, x_1 \otimes y_1 \otimes z_1 \rangle = \sum_{i=1}^{n1} \sum_{j=1}^{n2} \sum_{k=1}^{n3} \alpha_{ijk} \xi_{il} \eta_{lj} \zeta_{lk}$$

$$\begin{aligned} \text{et : } \left\| \sum_{i=1}^r x_i \otimes y_i \otimes z_i \right\|^2 &= \sum_{i,j=1}^r \langle x_i, x_j \rangle \langle y_i, y_j \rangle \langle z_i, z_j \rangle \\ &= \sum_{i,j=1}^r \left[\sum_{k=1}^{n1} \xi_{ik} \xi_{jk} \right] \cdot \left[\sum_{l=1}^{n2} \eta_{il} \eta_{jl} \right] \cdot \left[\sum_{m=1}^{n3} \zeta_{im} \zeta_{jm} \right] \\ &= \sum_{i,j=1}^r \sum_{k=1}^{n1} \sum_{l=1}^{n2} \sum_{m=1}^{n3} \xi_{ik} \xi_{jk} \eta_{il} \eta_{jl} \zeta_{im} \zeta_{jm} \end{aligned}$$

3 - D'où le système :

$$\left\{ \begin{array}{l} \sum_{i=1}^{n1} \sum_{j=1}^{n2} \sum_{k=1}^{n3} \sum_{l=1}^r \alpha_{ijk} \xi_{il} \eta_{lj} \zeta_{lk} \text{ maximum} \\ \text{sous } \sum_{i,j=1}^r \sum_{k=1}^{n1} \sum_{l=1}^{n2} \sum_{m=1}^{n3} \xi_{ik} \xi_{jk} \eta_{il} \eta_{jl} \zeta_{im} \zeta_{jm} = 1 \end{array} \right.$$

4. RECHERCHE D'UNE SOLUTION APPROCHÉE

1. JUSTIFICATION D'UNE SOLUTION APPROCHÉE

La recherche d'une solution algébrique exacte aux systèmes précédents se heurte à deux difficultés dès que la dimension des espaces considérés, ou le nombre de composante souhaitées, devient grand (en pratique dès que la dimension de l'un des espaces \mathcal{E}_i dépasse l'unité ...).

La résolution des systèmes par les procédures proposées en annexe de ce document (Résolution des systèmes de n équations algébriques à n inconnues) montre que, algébriquement, le problème est soluble. Mais, numériquement, il est hors de portée : faute de logiciel de calcul semi-formel performant, il mène à un nombre ingérable d'équations (éliminer une inconnue de degré p entre n équations revient à calculer n-1 déterminants de Sylvester, ce qui multiplie d'autant le degré des inconnues restant : une inconnue de degré q devient de degré 2pq dans les n-1 équations obtenue en identifiant à 0 les déterminants de Sylvester).

De plus, lorsque le degré d'un polynôme à une inconnue devient élevé (dès la dizaine, par exemple), les solutions numériques sont instables car très sensibles à une faible variation des coefficients (SCHWARTZ, 1989). C'est la raison pour laquelle, dans la recherche des valeurs propres d'une matrice, la méthode directe de calcul et résolution du polynôme caractéristique ($\text{Det}(A-\lambda I) = 0$) est techniquement faisable mais fortement non recommandée (SIBONY & MARDON, 1982), bien qu'il existe des méthodes

efficaces de calcul du polynôme caractéristique (méthode de Krylov, Leverrier, Souriau-Fadeev).

Aussi, cette combinaison de complexité et d'instabilité conduit naturellement à choisir une méthode numérique d'approximation pour le calcul des composantes du modèle ParaFac/CanDecomp d'un tenseur A donné. La méthode présentée s'étend sans difficultés à un tenseur d'ordre n arbitraire. La procédure proposée ici est une méthode itérative, proche de celle indiquée dans l'article original de CARROLL & CHANG (1970). Elle s'en éloigne légèrement par l'emploi des bases duales, qui permettent une extension naturelle par l'introduction de contraintes et de métriques définies par des MSDP, et par l'introduction d'un module d'accélération de la convergence.

2. REMARQUES SUR L'ALGORITHME PROPOSE

Plusieurs remarques importantes sont à faire concernant l'algorithme proposé, que nous avons programmé en langage PASCAL sur micro-ordinateur compatible PC.

1 - Il est certain que l'algorithme proposé conduit à un minimum local de $\|R\|^2$. Nous n'avons pas la démonstration que ce minimum local est toujours le minimum global, qui est recherché.

2 - Cependant, même si c'est une simple présomption, elle est forte car, lorsqu'on le connaît, le minimum trouvé est toujours le minimum optimal.

3 - Lorsque la dimension des espaces (n_1, n_2, n_3) où le rang du tenseur approchant (r) croissent (de l'ordre de quelques unités), l'algorithme devient lent (plusieurs secondes à chaque itération sur un 386 SX 20 avec coprocesseur arithmétique).

4 - La convergence est loin d'être quadratique : si ε_n est la distance entre la solution approchée à l'itération n et la solution recherchée, ε_n décroît comme $1/n^p$ avec $p < 1$: il faut parfois, plusieurs milliers d'itérations pour une précision de 10^{-9} , avec $r = 5$ et $n = 3$, donc plusieurs heures de

calculs.

5 - C'est pourquoi nous proposons un module d'accélération de la convergence, qui devient quadratique dès que l'on est proche de l'optimum cherché.

3. PRINCIPE DE L'ALGORITHME D'APPROXIMATION

R est une fonction de A , $(x_i)_i$, $(y_i)_i$, $(z_i)_i$, et l'on souhaite $\|R\|$ minimale :

$$\|A - \sum_i x_i \otimes y_i \otimes z_i\| \text{ minimale}$$

où les inconnues sont $(x_i)_i$, $(y_i)_i$ et $(z_i)_i$, et la somme étendue de 1 à r.

1 - Supposons $(x_i)_i$ et $(y_i)_i$ connus, et cherchons $(z_i)_i$ tels que le minimum soit réalisé. Posons pour simplifier :

$$D_i = x_i \otimes y_i$$

On écrit alors :

$$\|A - \sum_i D_i \otimes z_i\| \text{ minimale}$$

Le problème ainsi posé pour le calcul des z_i est un problème de régression linéaire, dont la solution est classique.

2 - Développons (1) :

$$\begin{aligned} \|A - \sum_i D_i \otimes z_i\|^2 &= \langle A - \sum_i D_i \otimes z_i, A - \sum_j D_j \otimes z_j \rangle \\ &= \|A\|^2 - 2\langle A, \sum_i D_i \otimes z_i \rangle + \langle \sum_i D_i \otimes z_i, \sum_j D_j \otimes z_j \rangle \\ &= \|A\|^2 - 2\sum_i \langle A D_i, z_i \rangle + \sum_i \langle \sum_j \langle D_i, D_j \rangle z_j, z_i \rangle \end{aligned}$$

3 - D'où, à l'optimum cherché de (1) en $(z_i)_i$:

$$\forall i \leq r, \sum_j \langle D_i, D_j \rangle z_j = A \cdot D_i$$

égalité obtenue par dérivation de (2) en z_i .

4 - Le système (3) est un système de r équations vectorielles linéaires en

z_1 . Si son déterminant est non nul, il a une solution et une seule, car, si n_3 est la dimension de \mathcal{E}_3 , où $\forall i, z_i \in \mathcal{E}_3$, ce système est équivalent à n_3 systèmes, un pour chacune des coordonnées. Le déterminant du système est celui de la matrice carrée $M \in \mathcal{M}_2(\mathbb{R}^r) = \mathcal{M}(\mathbb{R}^r)$ de coefficient $\mu_{ij} = \langle D_i, D_j \rangle$. Il est donc non nul si et seulement si le système formé par les D_i est libre, ce qui est le cas, par construction (sinon, on pourrait diminuer r , rang de X , d'au moins une unité).

5 - Nous allons calculer la solution du système (3) en utilisant le système dual $(D_i^*)_1$ du système D_j . Rappelons que les D_i^* sont combinaisons linéaires des D_j et que :

$$\forall i, j \leq r, \langle D_i^*, D_j \rangle = \delta_{ij}$$

6 - Soit alors :

$$D_i = \sum_j v_{ij} D_j^*$$

$$\Rightarrow \langle D_i, D_j \rangle = v_{ij}$$

$$\Rightarrow D_i = \sum_j \langle D_i, D_j \rangle D_j^*$$

$$\Rightarrow A \cdot D_i = \sum_j \langle D_i, D_j \rangle A \cdot D_j^*$$

$$\Rightarrow z_j = A \cdot D_j^* \text{ car la solution de (5) existe et est unique.}$$

7 - D'où le résultat suivant : étant donné un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$, et r couples (x_i, y_i) de vecteurs de $\mathcal{E}_1 \times \mathcal{E}_2$, les r vecteurs z_i solutions de

$$\|A - \sum_i x_i \otimes y_i \otimes z_i\| \text{ minimale}$$

sont calculés par :

$$z_i = A \cdot D_i^*$$

où $(D_i^*)_1$ est le système dual de $(x_i \otimes y_i)_1$, c'est à dire :

$$\langle D_i^*, x_j \otimes y_j \rangle = \delta_{ij}$$

8 - Le même raisonnement peut être mené avec $2A = \sum_{ijk} \alpha_{ijk} e_{2j} \otimes e_{3k} \otimes e_{1i}$: étant donné un tenseur $2A$ de $\mathcal{E}_2 \otimes \mathcal{E}_3 \otimes \mathcal{E}_1$, et r couples (y_i, z_i) de vecteurs de $\mathcal{E}_2 \times \mathcal{E}_3$, les r vecteurs x_i solutions de

$$\|2A - \sum_i y_i \otimes z_i \otimes x_i\| \text{ minimale}$$

sont calculés par :

$$x_i = 2A \cdot F_i^*$$

où $(F_i^*)_i$ est le système dual de $(y_i \otimes z_i)_i$, c'est à dire :

$$\langle F_i^*, y_j \otimes z_j \rangle = \delta_{ij}$$

9 - Or, $\|2A - \sum_i y_i \otimes z_i \otimes x_i\| = \|A - \sum_i x_i \otimes y_i \otimes z_i\|$. D'où le calcul des (x_i) connaissant les (y_i, z_i) .

10 - Le même calcul peut être mené pour $3A = \sum_{ijk} \alpha_{ijk} e_{3k} \otimes e_{1i} \otimes e_{2j}$, et permettre le calcul des (y_i) connaissant les (x_i, z_i) tels que $\|A - \sum_i x_i \otimes y_i \otimes z_i\|$ soit minimum.

4. ALGORITHME NUMERIQUE

1. Séquence proposée

L'algorithme proposé repose sur une optimisation alternée entre les (x_i) , (y_i) et (z_i) . Si ce triplet est noté $x_i^{(n)}$, $y_i^{(n)}$ et $z_i^{(n)}$ lors de la $n^{ième}$ itération, les inégalités suivantes sont en effet vérifiées :

$$\begin{aligned} \|A - \sum_i x_i^{(n)} \otimes y_i^{(n)} \otimes z_i^{(n)}\| &\leq \|A - \sum_i x_i^{(n)} \otimes y_i^{(n)} \otimes z_i^{(n+1)}\| \\ &\leq \|A - \sum_i x_i^{(n)} \otimes y_i^{(n+1)} \otimes z_i^{(n+1)}\| \\ &\leq \|A - \sum_i x_i^{(n+1)} \otimes y_i^{(n+1)} \otimes z_i^{(n+1)}\| \end{aligned}$$

Comme $\forall n, 0 \leq \|A - \sum_i x_i^{(n)} \otimes y_i^{(n)} \otimes z_i^{(n)}\|$, l'algorithme converge vers un optimum local. Cet optimum local a d'autant plus de chance d'être un optimum global que le point initial en est proche.

2. Choix d'un point initial

Le point initial sera calculé par des algorithmes classiques d'ACP sur des matrices associées à A.

1 - On décompose A selon ses composantes principales sur \mathcal{E}_1 :

$$A = \sum_i u_i \otimes M_i$$

2 - Chaque composante M_i est décomposée selon ses propres composantes principales :

$$M_i = \sum_j v_{ij} \otimes w_{ij}$$

3 - La décomposition de A peut alors s'écrire :

$$A = \sum_{ij} u_i \otimes v_{ij} \otimes w_{ij}$$

4 - Les triplets $(u_i \otimes v_{ij} \otimes w_{ij})$ sont deux à deux orthogonaux. On a donc l'égalité :

$$\|A\|^2 = \sum_i \|u_i \otimes v_{ij} \otimes w_{ij}\|^2$$

5 - On retient donc pour point initial de CanDecomp les r composantes de plus grande norme dans la décomposition précédente.

6 - On peut encore améliorer cette approximation en réalisant la même opération pour $2A$ et $3A$, et retenir la meilleure des trois.

5. ACCELERATION DE LA CONVERGENCE

La mise en oeuvre de cet algorithme sur un micro-ordinateur compatible PC et de nombreuses simulations numériques ont montré la pertinence de l'algorithme, mais sa lente convergence. En fait, l'algorithme est une méthode du type moindres carrés alternés, qui est robuste, mais parfois très lentes (PRESS & Al., 1989).

Un algorithme d'accélération de la convergence est alors souhaitable.

Pour le proposer, nous avons suivi les présentations claires des problèmes d'optimisation que l'on trouve par exemple dans MINOUX (1983) et PRESS & Al. (1989), où des algorithmes de convergence quadratique sont proposés. Leur inconvénient est de ne converger que vers des optimums locaux.

La fonction $\varphi(X,Y,Z)$ à minimiser étant polynômiale, sa représentation

graphique dans le repère défini par (X,Y,Z) et φ se présente comme une surface ondulée (en fait une courbe C^∞), chaque creux étant un attracteur vers un minimum local qui est le point le plus bas de la surface dans ce creux.

L'idée générale est alors la suivante :

(1) mettre en oeuvre l'algorithme des moindres carrés alternés pour se trouver dans le bon bassin d'attraction (celui du minimum global)

(2) accélérer la fin de la convergence par une méthode à convergence quadratique.

1. Présentation des méthodes et choix

Une accélération de la convergence peut être obtenue en tirant parti de deux informations qui n'ont pas été valorisées dans ce qui précède :

- le calcul du gradient de la fonction à minimiser
- le calcul du hessien de cette même fonction.

Soit f une fonction réelle deux fois différentiable d'une variable vectorielle x .

La connaissance du gradient ∇f permet de s'orienter vers des méthodes de descente : étant donné un point x de la fonction à minimiser, on choisit une direction d de descente, et l'on minimise sur λ la fonction $f(x-\lambda d)$. Les seules variables à conserver en mémoire temporairement pour le calcul sont des vecteurs d , et ce de façon itérative.

La méthode de descente la plus connue est la méthode de la plus forte pente, très ancienne, puisqu'elle remonte à CAUCHY, en 1847 (Méthode générale pour la résolution des systèmes d'équations simultanées, *C.R.A.S.*, XXV 536-538). La direction d choisie est $-\nabla f$. Cet algorithme peut également être lent, car c'est un algorithme de convergence linéaire de rapport α qui

peut être très proche de 1 si les valeurs propres du hessien de f sont rapprochées (LUENBERGER, 1973).

L'algorithme proposé est un algorithme de plus forte pente accéléré. Si x est un vecteur de \mathbb{R}^n :

- (1) on itère à partir de x_k n fois la procédure de descente par la plus forte pente pour obtenir y_1, y_2, \dots, y_n ,
- (2) puis on optimise depuis x_k dans la direction $d = y_n - x_k$.
- (3) L'optimum ainsi trouvé est x_{k+1} , et l'on réitère (MINOUX, 1983 ; PRESS & Al., 1989).

La convergence de cet algorithme est alors quadratique, comme la méthode des gradients conjugués, plus complexe à mettre en oeuvre (FORSYTHE, 1968).

Les méthodes faisant appel au hessien, ou au calcul de matrices dérivées du hessien, ne sont pas recommandées dans ce cas particulier, car elles nécessitent la manipulation de matrices $n \times n$, notamment des inversions (PRESS & Al., 1989). Or, n peut être grand dans le cas qui nous intéresse.

2. Mise en oeuvre de l'algorithme de descente d'ordre n

1 - La fonction à minimiser est :

$$\varphi(X, Y, Z) \equiv \varphi = \|A - \sum_1 x_1 \otimes y_1 \otimes z_1\|^2$$

avec

- $X = \sum_p e_{1k} \otimes x_k$
- $Y = \sum_q e_{2l} \otimes y_l$
- $Z = \sum_m e_{3m} \otimes z_m$

2 - Les inconnues sont les trois matrices X, Y, Z . On considèrera alors comme variable un triplet de trois matrices de $\mathbb{R}^r \otimes \mathcal{E}_1$, $\mathbb{R}^r \otimes \mathcal{E}_2$ et $\mathbb{R}^r \otimes \mathcal{E}_3$.

■ Calcul du gradient $\nabla \varphi$

1 - Posons : $D_1 = x_1 \otimes y_1$

2 - Ainsi : $\varphi = \|A - \sum_1 D_1 \otimes z_1\|^2$

3 - La fonction $g(Z) = \sum_1 D_1 \otimes z_1$ est linéaire. Notons la $g(Z) = K_Z \cdot Z$

4 - Ainsi : $\varphi = \|A - K_Z \cdot Z\|^2$

5 - D'où, si l'on note ∇_Z le gradient vis à vis de Z :

$$\nabla_Z = 2K'_Z(K_Z \cdot Z - A)$$

6 - De même, par permutation circulaire :

$$\nabla_X = 2K'_X(K_X \cdot X - A)$$

$$\nabla_Y = 2K'_Y(K_Y \cdot Y - A)$$

7 - D'où, finalement :

$$d\varphi = \langle \nabla_X, dX \rangle + \langle \nabla_Y, dY \rangle + \langle \nabla_Z, dZ \rangle$$

■ Descente selon la plus forte pente

1 - La descente selon la plus forte pente sera définie par une direction P, triplet de trois matrices (P_X, P_Y, P_Z) .

2 - Ce triplet est proportionnel à $-(\nabla_X, \nabla_Y, \nabla_Z)$, d'où :

$$P_X = -\lambda \nabla_X$$

$$P_Y = -\lambda \nabla_Y$$

$$P_Z = -\lambda \nabla_Z$$

3 - On recherche donc le scalaire $\lambda > 0$ tel que :

$$\varphi(\lambda) = \|A - \sum_1 (x_1 - \lambda \nabla_{X1}) \otimes (y_1 - \lambda \nabla_{Y1}) \otimes (z_1 - \lambda \nabla_{Z1})\|^2$$

soit minimal, si l'on note Δ_{U1} le $i^{ème}$ vecteur colonne de ∇_U .

4 - $\varphi(\lambda)$ est un polynôme du 6^{ème} degré en λ que l'on peut développer :

$$\begin{aligned} \varphi(\lambda) = & \| -\lambda^3 (\sum_1 \nabla_{X1} \otimes \nabla_{Y1} \otimes \nabla_{Z1}) \\ & + \lambda^2 (\sum_1 \nabla_{X1} \otimes \nabla_{Y1} \otimes z_1 + \nabla_{X1} \otimes y_1 \otimes \nabla_{Z1} + x_1 \otimes \nabla_{Y1} \otimes \nabla_{Z1}) \\ & - \lambda (\sum_1 \nabla_{X1} \otimes y_1 \otimes z_1 + x_1 \otimes y_1 \otimes \nabla_{Z1} + x_1 \otimes \nabla_{Y1} \otimes z_1) \end{aligned}$$

$$+ A - \sum_1 x_1 \otimes y_1 \otimes z_1 \|^2$$

qui s'écrit également sous la forme :

$$\begin{aligned} \varphi(\lambda) &= \|\lambda^3 T_3 + \lambda^2 T_2 + \lambda T_1 + T_0\|^2 \\ &= \lambda^6 \|T_3\|^2 \\ &\quad + \lambda^5 2\langle T_2, T_3 \rangle \\ &\quad + \lambda^4 \|T_2\|^2 + 2\langle T_1, T_3 \rangle \\ &\quad + \lambda^3 2\langle T_0, T_3 \rangle + 2\langle T_1, T_2 \rangle \\ &\quad + \lambda^2 \|T_1\|^2 + 2\langle T_0, T_2 \rangle \\ &\quad + \lambda 2\langle T_0, T_1 \rangle \\ &\quad + \|T_0\|^2 \\ &= \beta_0 + \beta_1 \lambda + \beta_2 \lambda^2 + \beta_3 \lambda^3 + \beta_4 \lambda^4 + \beta_5 \lambda^5 + \beta_6 \lambda^6 \end{aligned}$$

5 - λ est donc solution de $\varphi(\lambda)$ minimum, donc racine réelle d'un polynôme du conquième degré.

3. Résultat

On peut schématiser ainsi l'algorithme :

- 1 - Donnée de X_0, Y_0, Z_0
- 2 - Itérer le triplet $(X_{k+1}, Y_{k+1}, Z_{k+1})$ en fonction du triplet (X_k, Y_k, Z_k)
Tant qu'une condition de Test n'est pas remplie

avec l'algorithme suivant pour la procédure d'itération :

- 1 - Répéter $n_1 n_2 n_3$ fois la procédure de la descente selon le gradient.
Le résultat est un triplet (U_x, U_y, U_z)
- 2 - Réaliser la descente selon la direction $(X_k - U_x, Y_k - U_y, Z_k - U_z)$
- 2 - Le résultat est le triplet $(X_{k+1}, Y_{k+1}, Z_{k+1})$

L'algorithme de descente selon la direction définie par P est :

- 1 - Calculer T_0, T_1, T_2, T_3

2 - Calculer $\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6$

3 - Chercher le minimum de $\sum_{k=0}^6 \beta_k \lambda^k$

4 - Poser $X_{k+1} = X_k - \lambda P_X$, $Y_{k+1} = Y_k - \lambda P_Y$, $Z_{k+1} = Z_k - \lambda P_Z$

La descente selon le gradient demande le calcul de $\nabla_X, \nabla_Y, \nabla_Z$.

■ Développement des calculs

1 - Calcul des β_i

$$\beta_6 = \|T_3\|^2$$

$$\beta_5 = 2\langle T_2, T_3 \rangle$$

$$\beta_4 = \|T_2\|^2 + 2\langle T_1, T_3 \rangle$$

$$\beta_3 = 2\langle T_0, T_3 \rangle + 2\langle T_1, T_2 \rangle$$

$$\beta_2 = \|T_1\|^2 + 2\langle T_0, T_2 \rangle$$

$$\beta_1 = 2\langle T_0, T_1 \rangle$$

$$\beta_0 = \|T_0\|^2$$

2 - Calcul des T_i

$$T_3 = \sum_1 P_{x1} \otimes P_{y1} \otimes P_{z1}$$

$$T_2 = \sum_1 (P_{x1} \otimes P_{y1} \otimes z_1 + P_{x1} \otimes y_1 \otimes P_{z1} + x_1 \otimes P_{y1} \otimes P_{z1})$$

$$T_1 = \sum_1 (P_{x1} \otimes y_1 \otimes z_1 + x_1 \otimes y_1 \otimes P_{z1} + x_1 \otimes P_{y1} \otimes z_1)$$

$$T_0 = A - \sum_1 x_1 \otimes y_1 \otimes z_1$$

3 - Calcul des $\nabla_X, \nabla_Y, \nabla_Z$

C'est le calcul le plus long. Donnons en détail le calcul de ∇_Z

3.1 - On sait : $\nabla_Z = 2K'_Z(K_Z \cdot Z - A)$

3.2 - Le calcul de K_Z se fera sur la base $r_p \otimes e_{3q}$, où r_p est une BON de R^r et e_{3q} une BON de \mathcal{E}_3 . Notons $E_{pq} = r_p \otimes e_{3q}$. Nous allons calculer les coordonnées

sur cette base de K'_Z et de $K'_Z K_Z$

3.3 - Rappelons : $K_Z.Z = \sum_i x_i \otimes y_i \otimes z_i$.

$$\text{Or : } K_Z = \sum_{pq} E_{pq} \otimes K'_Z.E_{pq}$$

$$\text{Et : } K'_Z.E_{pq} = x_p \otimes y_p \otimes z_q$$

$$\text{D'où : } K_Z = \sum_{pq} E_{pq} \otimes x_p \otimes y_p \otimes z_q = \sum_{pq} r_{pq} \otimes e_{3q} \otimes x_p \otimes y_p \otimes z_q$$

$$\text{et } K'_Z = \sum_{pq} x_p \otimes y_p \otimes z_q \otimes E_{pq} = \sum_{pq} x_p \otimes y_p \otimes z_q \otimes r_{pq} \otimes e_{3q}$$

D'où :

$$\begin{aligned} K'_Z K_Z &= \left(\sum_{pqst} x_p \otimes y_p \otimes z_q \otimes E_{pq} \right) \left(\sum_{st} E_{st} \otimes x_s \otimes y_s \otimes z_t \right) \\ &= \sum_{pqst} \langle x_p, x_s \rangle \langle y_p, y_s \rangle \langle z_q, z_t \rangle E_{pq} \otimes E_{st} \end{aligned}$$

3.4 Le calcul peut donc se poursuivre ainsi :

$$\begin{aligned} V_Z &= 2K'_Z(K_Z.Z - A) \\ &= 2 K'_Z K_Z.Z - 2 K'_Z.A \end{aligned}$$

Avec :

$$\begin{aligned} K'_Z K_Z.Z &= \left(\sum_{pqst} \langle x_p, x_s \rangle \langle y_p, y_s \rangle \langle z_q, z_t \rangle E_{pq} \otimes E_{st} \right) \cdot \left(\sum_u r_{u} \otimes z_u \right) \\ &= \sum_{st} \left(\sum_{pq} \langle x_p, x_s \rangle \langle y_p, y_s \rangle \langle z_q, z_t \rangle \langle z_p, e_{3p} \rangle \right) E_{st} \end{aligned}$$

$$\begin{aligned} K'_Z.A &= \left(\sum_{pq} x_p \otimes y_p \otimes z_q \otimes E_{pq} \right) \cdot \left(\sum_{uvw} \alpha_{uvw} E_{uvw} \right) \\ &= \sum_{pq} \left(\sum_{uvw} \alpha_{uvw} \langle x_p, e_{1u} \rangle \langle y_p, e_{2v} \rangle \langle z_q, e_{3w} \rangle \right) E_{pq} \end{aligned}$$

3.5 Soit, si $x_p = \sum_u \xi_{pu} e_{1u}$; $y_p = \sum_v \eta_{pv} e_{2v}$; $z_p = \sum_w \zeta_{pw} e_{3w}$:

$$K'_Z K_Z.Z = \sum_{st} \chi_{st} E_{st}$$

avec :

$$\begin{aligned} \chi_{st} &= \sum_{pq} \langle x_p, x_s \rangle \langle y_p, y_s \rangle \langle z_q, z_t \rangle \langle z_p, e_{3p} \rangle \\ &= \sum_{pq} \left(\sum_u \chi_{pu} \chi_{su} \right) \left(\sum_v \eta_{pv} \eta_{sv} \right) \left(\sum_w \zeta_{qw} \zeta_{tw} \right) \zeta_{pp} \end{aligned}$$

et :

$$K'_Z \cdot A = \sum_{pq} \chi_{pq} E_{pq}$$

avec :

$$\begin{aligned} \chi_{pq} &= \sum_{uvw} \alpha_{uvw} \langle x_p, e_{1u} \rangle \langle y_p, e_{2v} \rangle \langle z_q, e_{3w} \rangle \\ &= \sum_{uvw} \alpha_{uvw} \xi_{pu} \eta_{pv} \zeta_{qw} \end{aligned}$$

L'ensemble de ces résultats permet de programmer l'algorithme d'optimisation par descente accélérée.

5. MODÈLE CANDECOMP AVEC MÉTRIQUE SUR LES ESPACES

1. NOTATIONS ET RAPPELS

1 - Notons par M_1 la métrique sur \mathcal{E}_1 , M_2 celle sur \mathcal{E}_2 et M_3 celle sur \mathcal{E}_3 . On y associe canoniquement une métrique T sur $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ telle que :

$$\langle a \otimes b \otimes c, x \otimes y \otimes z \rangle_T = \langle a, M_1 \cdot x \rangle \langle b, M_2 \cdot y \rangle \langle c, M_3 \cdot z \rangle$$

2 - Remarquons que, par exemple, l'espace $\mathcal{E}_1 \otimes \mathcal{E}_2$ est muni canoniquement d'une métrique définie par le tenseur noté T_{12} , tel que :

$$\forall P, Q \in \mathcal{E}_1 \otimes \mathcal{E}_2, \langle P, Q \rangle_{T_{12}} = \langle P, T_{12} \cdot Q \rangle$$

comme $\langle P, T_{12} \cdot (x \otimes y) \rangle = \langle P, (M_1 \cdot x) \otimes (M_2 \cdot y) \rangle = \langle P, M_2 (x \otimes y) M_1 \rangle = \langle M_2 P M_1, x \otimes y \rangle$,

$$\langle P, Q \rangle_{T_{12}} = \langle P, T_{12} \cdot Q \rangle = \langle M_2 P M_1, Q \rangle = \langle P, M_2 Q M_1 \rangle$$

3 - La même remarque vaut pour T_{23} définie sur $\mathcal{E}_2 \otimes \mathcal{E}_3$ et T_{31} définie sur $\mathcal{E}_3 \otimes \mathcal{E}_1$.

2. LE MODELE

Le modèle ParaFac/CanDecomp d'ordre r avec les métriques M_1 , M_2 et M_3 suit la définition naturelle suivante : étant donné un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ et la métrique T sur $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ canoniquement associée aux métriques M_1 , M_2 et M_3 sur \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 respectivement, on appelle modèle ParaFac / CanDecomp de A à l'ordre r relatif à T la décomposition :

$$A = \sum_1 x_1 \otimes y_1 \otimes z_1 + R$$

où $x_1 \in \mathcal{E}_1$; $y_1 \in \mathcal{E}_2$; $z_1 \in \mathcal{E}_3$
telle que
 $\|R\|_T$ minimale

soit, également :

$$\|A - \sum_1 x_1 \otimes y_1 \otimes z_1\|_T^2 \text{ minimale}$$

3. L'ALGORITHME D'APPROXIMATION

L'algorithme de recherche des composantes du modèle ParaFac / CanDecomp avec métriques est également un algorithme d'optimisation alternée sur chacune des familles $(x_1)_1$, $(y_1)_1$ et $(z_1)_1$.

1 - R est une fonction de A, $(x_1)_1$, $(y_1)_1$, $(z_1)_1$, et l'on souhaite $\|R\|$ minimale :

$$\|A - \sum_1 x_1 \otimes y_1 \otimes z_1\|_T \text{ minimale}$$

où les inconnues sont $(x_1)_1$, $(y_1)_1$ et $(z_1)_1$, et la somme étendue de 1 à r.

2 - Supposons les (x_1) et (y_1) connus, et cherchons sous cette condition les (z_1) optimaux. Rappelons que nous notons D_1 la matrice $x_1 \otimes y_1$ de $\mathcal{E}_1 \otimes \mathcal{E}_2$. Posons $Z = \sum_1 e_{r1} \otimes z_1$ où $(e_{r1})_1$ est une BON de \mathbb{R}^r . $Z \in \mathbb{R}^r \otimes \mathcal{E}_3$. Il existe alors une matrice K de $\mathcal{L}(\mathbb{R}^r \otimes \mathcal{E}_3, \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3)$, ou un tenseur K de $\mathbb{R}^r \otimes \mathcal{E}_3 \otimes \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$, tel que :

$$\sum_1 D_1 \otimes z_1 = K.Z$$

Seule l'existence de K est utile dans la suite : expliciter ses coefficients ne sera pas nécessaire (c'est une matrice à $n_1 n_2 n_3$ lignes et $r n_3$ colonnes).

Ainsi posé, le problème de la recherche de Z est un problème de régression linéaire avec la métrique définie par T, dont la solution est classique.

1. Premier lemme

Etant donné :

- un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$
- un tenseur K de $\mathbb{R}^r \otimes \mathcal{E}_3 \otimes \mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$
- une métrique T sur $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$

la matrice Z de $\mathbb{R}^r \otimes \mathcal{E}_3$ telle que $\|A - K \cdot Z\|_T$ soit minimale est solution de

$$K' \cdot T \cdot A = K' \cdot T \cdot K \cdot Z$$

□ Démonstration

Un calcul simple montre :

$$\begin{aligned} \|A - K \cdot Z\|_T^2 &= \|A\|_T^2 - 2 \langle A, K \cdot Z \rangle_T + \|K \cdot Z\|_T^2 \\ &= \|A\|_T^2 - 2 \langle T \cdot A, K \cdot Z \rangle + \langle T \cdot K \cdot Z, K \cdot Z \rangle \\ &= \|A\|_T^2 - 2 \langle K' \cdot T \cdot A, Z \rangle + \langle K' \cdot T \cdot K \cdot Z, K \cdot Z \rangle \end{aligned}$$

D'où, à l'optimum, $\frac{\partial}{\partial Z} (\|A - K \cdot Z\|_T^2) = 0$, soit : $K' \cdot T \cdot A = K' \cdot T \cdot K \cdot Z$

2. Deuxième lemme

Si nous notons :

- U la MSDP de $\mathcal{L}(\mathcal{E}_1 \otimes \mathcal{E}_2)$
- M la MSDP de $\mathcal{L}(\mathcal{E}_3)$
- X une matrice quelconque de $\mathcal{E}_1 \otimes \mathcal{E}_2$
- x un vecteur quelconque de \mathcal{E}_3

tels que

$$\langle A, X \otimes x \rangle_T = \langle A, (U \cdot X) \otimes (M \cdot x) \rangle,$$

et V une MSDP telle que

$$V^2 = U$$

la solution de l'équation matricielle précédente est :

$$\forall i, z_i = AV(V \cdot D_i)^*$$

□ Démonstration

1 - Nous cherchons Z telle que $K'.T.A = K'.T.K.Z$

2 - Puisque $K \in \mathbb{R}^r \otimes \mathcal{E}_3$, et que $(e_{r1} \otimes e_{3j})_{1j}$ est une BON de $\mathbb{R}^r \otimes \mathcal{E}_3$:

$$K = \sum_{1j} e_{r1} \otimes e_{3j} \otimes K.(e_{r1} \otimes e_{3j})$$

3 - Or, $K.(e_{r1} \otimes e_{3j}) = D_1 \otimes e_{3j}$, d'où :

$$K = \sum_{1j} e_{r1} \otimes e_{3j} \otimes D_1 \otimes e_{3j}$$

$$K' = \sum_{1j} D_1 \otimes e_{3j} \otimes e_{r1} \otimes e_{3j}$$

4 - D'où :

$$K.Z = \sum_i D_1 \otimes z_i$$

$$T.K.Z = \sum_i U.D_1 \otimes M.z_i$$

$$\begin{aligned} K'.T.K.Z &= \left(\sum_{1j} D_1 \otimes e_{3j} \otimes e_{r1} \otimes e_{3j} \right) \cdot \left(\sum_k U.D_k \otimes M.z_k \right) \\ &= \sum_{1jk} \langle D_1, U.D_k \rangle \langle e_{3j}, M.z_k \rangle e_{r1} \otimes e_{3j} \end{aligned}$$

5 - De même :

$$K'.T.A = \left(\sum_{1j} D_1 \otimes e_{3j} \otimes e_{r1} \otimes e_{3j} \right) \cdot T.A$$

$$= \sum_{1j} \langle D_1 \otimes e_{3j}, T.A \rangle e_{r1} \otimes e_{3j}$$

$$= \sum_{1j} \langle U.D_1 \otimes M.e_{3j}, A \rangle e_{r1} \otimes e_{3j}$$

6 - D'où, en rapprochant 4 & 5 :

$$\forall i, j : \sum_k \langle D_1, U.D_k \rangle \langle e_{3j}, M.z_k \rangle = \langle U.D_1 \otimes M.e_{3j}, A \rangle$$

$$\Rightarrow \sum_k \langle U.D_k, D_1 \rangle \langle z_k, M.e_j \rangle = \langle A, U.D_1 \otimes M.e_{3j} \rangle$$

$$\Rightarrow \langle \sum_k \langle U.D_k, D_1 \rangle z_k, M.e_j \rangle = \langle A.(U.D_1), M.e_{3j} \rangle$$

$$\Rightarrow \sum_k \langle U.D_k, D_1 \rangle z_k = A.(U.D_1) \text{ car } M \text{ est une MSDP donc est inversible}$$

7 - Or, en rappelant que $V^2 = U$:

$$V.D_1 = \sum_j \langle V.D_1, V.D_j \rangle (V.D_j)^* = \sum_j \langle U.D_1, D_j \rangle (V.D_j)^*$$

$$\begin{aligned}\Rightarrow U.D_1 &= V^2.D_1 = \sum_j \langle U.D_1, D_j \rangle V(V.D_j)^* \\ \Rightarrow A.(U.D_1) &= \sum_j \langle U.D_1, D_j \rangle A.(V(V.D_j)^*)\end{aligned}$$

8 - Or, comme le système (6) a une solution est une seule,

$$z_1 = A.(V(V.D_1)^*)$$

3. Calcul de V

1 - la métrique U n'intervient pas explicitement dans les calculs : seul est à calculer $V.D_1$ et $V(V.D_1)^*$, où V est telle que $V^2 = U$.

$$2 - D_1 = x_1 \otimes y_1 \Rightarrow U.D_1 = (M_1.x_1) \otimes (M_2.y_1).$$

3 - U étant une MSDP, si $U = \sum_i \lambda_i u_i \otimes u_i$, alors $V = \sum_i \sqrt{\lambda_i} u_i \otimes u_i$, où les u_i sont les vecteurs propres normés de U.

4 - Or, Si $(m_{1i})_i$ sont les vecteurs propres normés de M_1 , et $(m_{2j})_j$ les vecteurs propres normés de M_2 :

$$\begin{aligned}U.(m_{1i} \otimes m_{2j}) &= (M_1.m_{1i}) \otimes (M_2.m_{2j}) \\ &= \lambda_{1i} m_{1i} \otimes \lambda_{2j} m_{2j} \\ &= \lambda_{1i} \lambda_{2j} m_{1i} \otimes m_{2j}\end{aligned}$$

Le système $(m_{1i} \otimes m_{2j})_{ij}$ forme donc une BON de $\mathcal{E}_1 \otimes \mathcal{E}_2$ formée de vecteurs propres de U, et :

$$V = \sum_{i,j} \sqrt{\lambda_{1i} \lambda_{2j}} m_{1i} \otimes m_{2j} \otimes m_{1i} \otimes m_{2j}$$

5 - Notons $\sqrt{M_1}$ la MSDP telle que $(\sqrt{M_1})^2 = M_1$. V est alors la matrice telle que $V.(x \otimes y) = (\sqrt{M_1}.x) \otimes (\sqrt{M_2}.y)$. Un calcul simple montre en effet :

$$V^2.(x \otimes y) = V.((\sqrt{M_1}.x) \otimes (\sqrt{M_2}.y)) = (M_1.x) \otimes (M_2.y).$$

D'où :

$$V.(x \otimes y) = (\sqrt{M_1}.x) \otimes (\sqrt{M_2}.y)$$

Et, en particulier :

$$V.D_1 = (\sqrt{M_1}.x_1) \otimes (\sqrt{M_2}.y_1)$$

4. DEUXIEME CALCUL DE LA SOLUTION DU MODELE AVEC METRIQUES

Cette deuxième solution directe utilise l'isométrie introduite lors de l'étude de l'ACP avec métriques.

1. Définition de l'isométrie

1 - Rappelons que l'on définit une isométrie ι pour chacun des espaces \mathcal{E}_1 , \mathcal{E}_2 et \mathcal{E}_3 comme suit :

$$\iota(x_1) = P_1^{-1}.x_1$$

$$\iota(x_2) = P_2^{-1}.x_2$$

$$\iota(x_3) = P_3^{-1}.x_3$$

2 - Cette isométrie est alors étendue à $\mathcal{H}(\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3)$ par le produit tensoriel. En particulier :

$$\iota(x_1 \otimes x_2 \otimes x_3) = \iota(x_1) \otimes \iota(x_2) \otimes \iota(x_3) = (P_1^{-1}.x_1) \otimes (P_2^{-1}.x_2) \otimes (P_3^{-1}.x_3)$$

2. Solution du modèle Parafac/Candecomp

La solution est alors très rapide.

1 - Soit $A = \sum_1 x_1 \otimes y_1 \otimes z_1 + R$ le modèle Parafac / Candecomp de A à l'ordre r .

2 - $\|A - \sum_1 x_1 \otimes y_1 \otimes z_1\|^2$ est donc minimum.

3 - $\|\iota(A - \sum_1 x_1 \otimes y_1 \otimes z_1)\|_T^2$ est donc minimum pour la norme définie par T .

4 - Ainsi, $\sum_1 \iota(x_1) \otimes \iota(y_1) \otimes \iota(z_1)$ sont les composantes du modèle Candecomp /

Parafac de $\iota(A)$ pour la norme définie par T.

5 - D'où : si x_i, y_i, z_i sont les composantes du modèle Candecomp / Parafac de A pour la norme définie par T, alors, $P_1 \cdot x_i, P_2 \cdot y_i$ et $P_3 \cdot z_i$ sont les composantes du modèle de $\iota^{-1}(A)$ pour la norme unité, que l'on sait calculer.

3. Résultat

Pour calculer le modèle Parafac / Candecomp de A à l'ordre r avec les métriques M_1, M_2 et M_3 , on peut procéder comme suit :

1 - Calcul des matrices $P_1 = \sqrt{M_1}, P_2 = \sqrt{M_2}$ et $P_3 = \sqrt{M_3}$.

2 - Calcul de $B = \iota^{-1}(A)$ par $B = \sum_{ijk} P_1 \cdot e_{1i} \otimes P_2 \cdot e_{2j} \otimes P_3 \cdot e_{3k}$

3 - Calcul des composantes u_i, v_i et w_i du modèle Parafac / Candecomp de B

4 - Calcul de $x_i = P_1^{-1} \cdot u_i, y_i = P_2^{-1} \cdot v_i, z_i = P_3^{-1} \cdot w_i$.

Cette solution est bien plus économique en temps de calcul, et sera retenue. De plus, elle se prête aisément à généralisation lorsque l'ordre augmente et devient > 3 .

4. Identité entre les deux solutions

Montrons, par exemple pour le calcul de z_i à chaque itération, qu'il y a identité entre les deux solutions.

1 - On considère donc, pour une itération donnée, que les x_i et y_i sont fixes, et l'on cherche les z_i tels que

$$\|A - \sum_i x_i \otimes y_i \otimes z_i\|_T^2$$

minimum.

2 - Alors, $P_1.x_1$, $P_2.y_1$ et $P_3.z_1$ sont tels que :

$$\|\iota^{-1}(A) - \sum_1 P_1.x_1 \otimes P_2.y_1 \otimes P_3.z_1\|^2$$

est minimale.

3 - Donc, en particulier, si $C_1 = P_1.x_1 \otimes P_2.y_1 : P_3.z_1 = \iota^{-1}(A).C_1^*$

D'où : $z_1 = P_3^{-1}(\iota^{-1}(A).C_1^*)$

4 - Or : $\iota^{-1}(A) = \sum_{1mn} \alpha_{1mn} P_1.e_{11} \otimes P_2.e_{2m} \otimes P_3.e_{3n}$. D'où :

$$\begin{aligned} P_3^{-1}(\iota^{-1}(A).C_1^*) &= P_3^{-1}\left[\left(\sum_{1mn} \alpha_{1mn} P_1.e_{11} \otimes P_2.e_{2m} \otimes P_3.e_{3n}\right).C_1^*\right] \\ &= P_3^{-1}\left[\sum_{1mn} \alpha_{1mn} \langle P_1.e_{11} \otimes P_2.e_{2m}, C_1^* \rangle P_3.e_{3n}\right] \\ &= \sum_{1mn} \alpha_{1mn} \langle P_1.e_{11} \otimes P_2.e_{2m}, C_1^* \rangle e_{3n} \end{aligned}$$

Or, en remarquant que $P_1.e_{11} \otimes P_2.e_{2m} = V.(e_{11} \otimes e_{2m})$, nous avons :

$$\begin{aligned} P_3^{-1}(\iota^{-1}(A).C_1^*) &= \sum_{1mn} \alpha_{1mn} \langle V.(e_{11} \otimes e_{2m}), C_1^* \rangle e_{3n} \\ &= \sum_{1mn} \alpha_{1mn} \langle e_{11} \otimes e_{2m}, V.C_1^* \rangle e_{3n} \\ &= \left(\sum_{1mn} \alpha_{1mn} e_{11} \otimes e_{2m} \otimes e_{3n}\right). (V.C_1^*) \\ &= A.(V.C_1^*) \end{aligned}$$

Or, $C_1 = P_1.x_1 \otimes P_2.y_1 = V.(x_1 \otimes y_1) = V.D_1$. D'où :

$$z_1 = A.(V.(V.D_1)^*)$$

qui était bien la solution trouvée précédemment, directement.

6 - MODÈLE PARAFAC / CANDECOMP AVEC CONTRAINTES LINÉAIRES

1. DEFINITION

Un modèle PARAFAC / CANDECOMP avec contraintes linéaires est simplement un tel modèle où l'on impose aux composantes (x_i) , (y_i) et (z_i) d'appartenir à des sous espaces fixés \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 respectivement. Plus précisément :

On appelle modèle PARAFAC / CANDECOMP sous contraintes linéaires définies par les SEV \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 une décomposition de la forme :

$$A = \sum_{i=1}^r x_i \otimes y_i \otimes z_i + R$$

avec :

$$\forall i, x_i \in \mathcal{F}_1$$

$$\forall i, y_i \in \mathcal{F}_2$$

$$\forall i, z_i \in \mathcal{F}_3$$

r fixé

$\|R\|$ minimale

2. LEMME

■ Enoncé

Calculer le modèle ParaFac / Candecomp de A sous la triple contrainte définie par \mathcal{F}_1 , \mathcal{F}_2 , \mathcal{F}_3 revient à calculer ce modèle pour le tenseur $\mathcal{P}(A)$, projection orthogonale de A sur $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$.

■ Démonstration

La démonstration est calquée sur celle de l'ACP-DCL, comme cela avait été annoncé. Rappelons là :

1 - Notons \mathfrak{p} la projection orthogonale de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ sur $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$, et \mathfrak{p}^\perp le projecteur orthogonal (i.e. sur $(\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3)^\perp$).

2 - Notons également pour simplifier : $\Sigma_1 = \sum_1 x_1 \otimes y_1 \otimes z_1$. Ainsi :

$$\begin{aligned} \|\mathbf{A} - \Sigma_1\|^2 &= \|(\mathfrak{p} + \mathfrak{p}^\perp)(\mathbf{A} - \Sigma_1)\|^2 \\ &= \|\mathfrak{p}(\mathbf{A} - \Sigma_1)\|^2 + \|\mathfrak{p}^\perp(\mathbf{A} - \Sigma_1)\|^2 \quad \text{car } \mathfrak{p} \text{ et } \mathfrak{p}^\perp \text{ sont orthogonaux} \\ &= \|\mathfrak{p}(\mathbf{A}) - \Sigma_1\|^2 + \|\mathfrak{p}^\perp(\mathbf{A})\|^2 \quad \text{car } \mathfrak{p}(\Sigma_1) = \Sigma_1 \text{ et } \mathfrak{p}^\perp(\Sigma_1) = 0 \end{aligned}$$

3 - $\|\mathbf{A} - \Sigma_1\|^2$ est donc minimale lorsque $\|\mathfrak{p}(\mathbf{A}) - \Sigma_1\|^2$ l'est.

4 - Cette démonstration s'étend sans difficultés au modèle PARAFAC / CANDECOMP d'ordre p .

3. APPLICATION

■ Notations

1 - Soient donc trois SEV \mathcal{F}_1 , \mathcal{F}_2 et \mathcal{F}_3 .

2 - Soit P_1 le projecteur orthogonal de \mathcal{E}_1 sur \mathcal{F}_1 ,
 P_2 " " de \mathcal{E}_2 sur \mathcal{F}_2
 P_3 " " de \mathcal{E}_3 sur \mathcal{F}_3 .

3 - Soit \mathfrak{p} le projecteur orthogonal de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ sur $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$

■ Lemme

Rappelons qu'alors :

$$\mathfrak{p}(u \otimes v \otimes w) = P_1 u \otimes P_2 v \otimes P_3 w$$

En effet :

- l'opérateur \mathfrak{p} ainsi défini est bien une projection car $\mathfrak{p}^2 = \mathfrak{p}$
- on vérifie aisément que $\text{Im}(\mathfrak{p}) = \mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$

• on vérifie également que $X \in (\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3)^\perp \Rightarrow \mathfrak{p}(X) = 0$

■ Calcul de $\mathfrak{p}(A)$

Le calcul de $\mathfrak{p}(A)$ est alors aisé :

$$\mathfrak{p}(A) = \mathfrak{p}\left(\sum_{ijk} \alpha_{ijk} E_{ijk}\right) = \sum_{ijk} \alpha_{ijk} \mathfrak{p}(E_{ijk}) = \sum_{ijk} \alpha_{ijk} (P_1 e_{1i}) \otimes (P_2 e_{2j}) \otimes (P_3 e_{3k})$$

Si $P_1 = \sum_i e_{1i} \otimes p_{1i}$ (p_{1i} est alors le $i^{\text{ème}}$ vecteur colonne de la matrice P_1)

$$P_2 = \sum_j e_{2j} \otimes p_{2j}$$

$$P_3 = \sum_k e_{3k} \otimes p_{3k}$$

on peut également écrire cette égalité sous la forme :

$$\mathfrak{p}(A) = \sum_{ijk} \alpha_{ijk} p_{1i} \otimes p_{2j} \otimes p_{3k}$$

7 - MODÈLE PARAFAC / CANDECOMP D'ORDRE P

1. RAPPEL DES NOTATIONS

1 - Soit une famille de p espaces vectoriels \mathcal{E}_i réels, de dimensions respectives n_i .

2 - Désignons par $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \dots \otimes \mathcal{E}_p$, ou en abrégé $\otimes \mathcal{E}_i$, l'espace vectoriel des formes p -linéaires sur \mathbb{R} . C'est un espace de dimension $n_1 n_2 \dots n_p = \prod n_i$.

3 - Soit r un nombre entier.

4 - Etant donné un tenseur A de $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \dots \otimes \mathcal{E}_p$, désignons par kA le tenseur obtenu par permutation circulaire où le premier espace est \mathcal{E}_k :

$$k(x_1 \otimes \dots \otimes x_p) = x_k \otimes x_{k+1} \otimes \dots \otimes x_p \otimes x_1 \otimes \dots \otimes x_{k-1}$$

2. MODELE PARAFAC / CANDECOMP

L'extension du modèle PARAFAC / CANDECOMP pour des tableaux à p entrées (donc des tenseurs de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$) tel qu'il a été présenté ne pose guère de difficultés, si ce n'est parfois une écriture un peu lourde lorsque l'on cherche à aller au bout des calculs.

Une telle extension avait été esquissée dans CARROL & CHANG (1970), pour la métrique unité et l'absence de contraintes sur les composantes. La présentation qui suit permet de prendre en compte :

- des métriques quelconques définies par des MSDP M_i sur les espaces \mathcal{E}_i
- des contraintes définies par des SEV \mathcal{F}_i sur les composantes.

□

Pour la présentation du modèle CANDECOMP/PARAFAC, nous nous limiterons aux algorithmes itératifs permettant d'atteindre une solution approchée, puisque la complexité du calcul des solutions algébriques exactes dépasse les possibilités des ordinateurs actuels.

1. Définition

Soit A un tenseur de $\otimes \mathcal{E}_i$. On appelle modèle PARAFAC / CANDECOMP à l'ordre r de A une décomposition de la forme :

$$A = \sum_{j=1}^r x_{1j} \otimes x_{2j} \otimes \dots \otimes x_{pj} + R$$

avec $x_{ij} \in \mathcal{E}_i$

$\|R\|$ minimale

2. Algorithme

L'algorithme de recherche de la solution est la transposition naturelle de l'algorithme proposé pour le cas $p = 3$.

1 - Fixons x_1, \dots, x_{p-1} , et cherchons x_p tel que $\|R\|$ soit minimal. Pour cela, posons $D_1 = x_{11} \otimes \dots \otimes x_{p-1,1}$.

2 - Le problème revient à chercher x_{p1} tel que $\|A - \sum_1 D_1 \cdot x_{p1}\|$ minimal. Ainsi posé, le problème est un problème de régression, que nous avons déjà rencontré, et dont la solution est classique.

3 - Nous savons qu'alors : $x_{p1} = A \cdot D_1^*$, d'où :

$$x_{p1} = A \cdot (x_{11} \otimes \dots \otimes x_{p-1,1})^*$$

4 - Ce même algorithme peut être réalisé par permutation circulaire sur les x_{ki} : si le chapeau \wedge signifie qu'un tel élément est omis, appelons D_{k1} le tenseur $x_{11} \otimes \dots \otimes \hat{x}_{k1} \otimes \dots \otimes x_{p1}$. Alors, à $x_{11} \otimes \dots \otimes \hat{x}_{k1} \otimes \dots \otimes x_{p1}$ fixés, x_{k1} est solution de $\|(k+1)A - \sum_1 (k+1)D_{k1} \cdot x_{k1}\|$, soit : $x_{k1} = (k+1)A \cdot (k+1)D_{k1}^*$, ou encore :

$$x_{k1} = (k+1)A \cdot (x_{k+1,1} \otimes \dots \otimes x_{p,1} \otimes x_{11} \otimes \dots \otimes x_{k-1,1})^*$$

5 - Une séquence complète d'itérations consiste à calculer circulairement tous les x_{ki} par cette procédure.

3. MODEL CANDECOMP / PARAFAC AVEC METRIQUES

L'introduction de métriques particulières M_1 sur les espaces \mathcal{E}_1 peut être prise en compte naturellement et directement par cette procédure.

1. Rappels sur les métriques

1 - Rappelons qu'aux métriques M_1 nous associons canoniquement une métrique T sur $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ définie par :

$$\langle A, x_1 \otimes \dots \otimes x_p \rangle = \langle A, (M_1.x_1) \otimes \dots \otimes (M_p.x_p) \rangle$$

2 - Nous noterons $T_{1..p-1}$ la métrique associée à M_1, \dots, M_{p-1} sur $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_{p-1}$.

3 - De même, à toute MSDP M_1 , nous associons la MSDP P_1 telle que $(P_1)^2 = M_1$ également notée $\sqrt{M_1}$.

2. Le modèle

Le modèle ParaFac/CanDecomp d'ordre r avec les métriques M_1, \dots, M_p suit la définition naturelle suivante : étant donné un tenseur A de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ et la métrique T sur $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ canoniquement associée aux métriques M_1, \dots, M_p , on appelle modèle ParaFac / CanDecomp de A à l'ordre r relatif à T la décomposition :

$$A = \sum_{j=1}^r x_{1j} \otimes x_{2j} \otimes \dots \otimes x_{pj} + R$$

avec $x_{1j} \in \mathcal{E}_1$

$\|R\|_T$ minimale

soit, également :

$$\|A - \sum_j x_{1j} \otimes \dots \otimes x_{pj}\|_T^2 \text{ minimale}$$

3. L'algorithme d'approximation

L'algorithme de recherche des composantes du modèle ParaFac / CanDecomp avec métriques est également un algorithme d'optimisation alternée sur chacune des familles $(x_{ij})_j$.

2 - Supposons les $(x_{11}), \dots, (x_{p-1,1})$ connus, et cherchons sous cette condition les (x_{p1}) optimaux. Rappelons que nous notons D_1 le tenseur $x_{11} \otimes \dots \otimes x_{p-1,1}$ de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_{p-1}$. Posons $X_p = \sum_1 e_{r1} \otimes x_{p1}$ où $(e_{r1})_1$ est une BON de \mathbb{R}^r . $X_p \in \mathbb{R}^r \otimes \mathcal{E}_p$. Il existe alors une matrice K de $\mathcal{L}(\mathbb{R}^r \otimes \mathcal{E}_p, \mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p)$ telle que :

$$\sum_1 D_1 \otimes x_{p1} = K.X_p$$

Encore une fois, seule l'existence de K est utile dans la suite : expliciter ses coefficients ne sera pas nécessaire (c'est une matrice à $n_1 \dots n_p$ lignes et rn_p colonnes).

Ainsi posé, le problème est un problème de régression linéaire avec la métrique définie par T , dont la solution est classique.

■ Lemme

Etant donné :

- un tenseur A de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$
- une matrice K de $\mathcal{L}(\mathbb{R}^r \otimes \mathcal{E}_p, \mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p)$
- une métrique T sur $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_{p-1}$

la matrice X_p de $\mathbb{R}^r \otimes \mathcal{E}_p$ telle que $\|A - K.X_p\|_T$ soit minimale est solution de

$$K'.T.A = K'.T.K.X_p$$

La démonstration du cas $p = 3$ s'applique ici rigoureusement.

■ Lemme

Si nous notons U la MSDP de $\mathcal{L}(\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_{p-1})$ et M_p la MSDP de $\mathcal{L}(\mathcal{E}_p)$ telles que $\langle A, X \otimes x \rangle_T = \langle A, (U.X) \otimes (M_p.x) \rangle$, et V une MSDP telle que $V^2 = U$, la

solution de l'équation matricielle précédente est :

$$z_1 = AV(V.D_1)^*$$

□ Démonstration

Là encore, la démonstration du cas $p = 3$ s'applique rigoureusement (en changeant \mathcal{E}_3 en \mathcal{E}_p), ce qui démontre l'intérêt d'avoir décomposé la démonstration du cas $p = 3$ en deux lemmes.

■ Calcul explicite de V

1 - la métrique U n'intervient pas explicitement dans les calculs : seul est à calculer $V.D_1$ et $V(V.D_1)^*$, où V est telle que $V^2 = U$.

$$2 - D_1 = x_{11} \otimes \dots \otimes x_{p-1,1} \Rightarrow U.D_1 = (M_1.x_{11}) \otimes \dots \otimes (M_{p-1}.x_{p-1,1}).$$

4 - Or, Si $(m_{ki})_i$ sont les vecteurs propres normés de M_k :

$$\begin{aligned} U.(m_{11} \otimes \dots \otimes m_{p-1,1}) &= (M_1.m_{11}) \otimes \dots \otimes (M_{p-1}.m_{p-1,1}) \\ &= \lambda_{11} m_{11} \otimes \dots \otimes \lambda_{p-1,1} m_{p-1,1} \\ &= \lambda_{11} \dots \lambda_{p-1,1} m_{11} \otimes \dots \otimes m_{p-1,1} \end{aligned}$$

5 - Le système $(m_{11} \otimes \dots \otimes m_{p-1,1})$ forme donc une BON de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_{p-1}$ formée de vecteurs propres de U, et :

$$V = \sum_{i_1 \dots i_{p-1}} \sqrt{\lambda_{11} \dots \lambda_{p-1,1}} m_{11} \otimes \dots \otimes m_{p-1,1} \otimes m_{11} \otimes \dots \otimes m_{p-1,1}$$

6 - Notons P_k la MSDP telle que $P_k^2 = M_k$. Soit alors W la matrice telle que

$W.(x_{11} \otimes \dots \otimes x_{p-1,1}) = (P_1.x_{11}) \otimes \dots \otimes (P_{p-1}.x_{p-1,1})$. Un calcul simple montre :

$$\begin{aligned} W^2.(x_{11} \otimes \dots \otimes x_{p-1,1}) &= W.((P_1.x_{11}) \otimes \dots \otimes (P_{p-1}.x_{p-1,1})) \\ &= (M_1.x_{11}) \otimes \dots \otimes (M_{p-1}.x_{p-1,1}) \end{aligned}$$

D'où : $V = W$, et :

$$V.D_1 = (P_1.x_{11}) \otimes \dots \otimes (P_{p-1}.x_{p-1,1})$$

4. Deuxième calcul de la solution du modèle avec métriques

Cette deuxième solution directe utilise également l'isométrie introduite lors de l'étude de l'ACP avec métriques et déjà utilisée pour le cas de l'ordre 3.

■ Définition de l'isométrie

1 - Rappelons que l'on définit une isométrie ι pour chacun des espaces \mathcal{E}_1 comme suit :

$$\iota(x_1) = P_1^{-1}.x_1$$

2 - Cette isométrie est alors étendue à $\mathcal{H}(\mathcal{E}_1, \dots, \mathcal{E}_p)$ par le produit tensoriel. En particulier :

$$\iota(x_1 \otimes \dots \otimes x_p) = \iota(x_1) \otimes \dots \otimes \iota(x_p) = (P_1^{-1}.x_1) \otimes \dots \otimes (P_p^{-1}.x_p)$$

■ Solution du modèle Parafac/Candecomp

La solution est alors très rapide.

1 - Soit $A = \sum_1 x_{11} \otimes \dots \otimes x_{p1} + R$ le modèle Parafac / Candecomp de A à l'ordre r .

2 - $\|A - \sum_1 x_{11} \otimes \dots \otimes x_{p1}\|^2$ est donc minimum.

3 - $\|\iota(A - \sum_1 x_{11} \otimes \dots \otimes x_{p1})\|_T^2$ est donc minimum pour la norme définie par T .

4 - Ainsi, $\sum_1 \iota(x_{11}) \otimes \dots \otimes \iota(x_{p1})$ sont les composantes du modèle Candecomp / Parafac de $\iota(A)$ pour la norme définie par T .

5 - D'où : si x_{11}, \dots, x_{p1} sont les composantes du modèle Candecomp / Parafac de A pour la norme définie par T , alors, $P_1.x_{11}, \dots, P_p.x_{p1}$ sont les composantes du modèle de $\iota^{-1}(A)$ pour la norme unité, que l'on sait calculer.

■ Résultat

Pour calculer le modèle Parafac / Candecomp de A à l'ordre r avec les métriques M_1, \dots, M_p , on peut procéder par la séquence des calculs suivants :

1 - Calcul des matrices $P_1 = \sqrt{M_1}, \dots, P_p = \sqrt{M_p}$.

2 - Calcul de $B = \iota^{-1}(A)$ par $B = \sum_{1 \dots p} \alpha_{1 \dots p} P_1.e_{11_1} \otimes \dots \otimes P_p.e_{p1_p}$

3 - Calcul des composantes u_1, \dots, u_p du modèle Parafac / Candecomp de B

4 - Calcul de $x_1 = P_1^{-1}.u_1, \dots, x_p = P_p^{-1}.u_p$.

Remarquons que le calcul des composantes du modèle Parafac / Candecomp de B se fait avec la métrique définie par la matrice unité dans chacun des espaces considérés. L'algorithme se réduit alors à un seul prétraitement des données (1 & 2) et une transformation du résultat (4). Il n'est pas nécessaire d'écrire un programme spécifique pour le modèle Parafac / Candecomp avec métriques.

4. MODELE AVEC CONTRAINTES LINEAIRES

1. Définition

Un modèle PARAFAC / CANDECOMP avec contraintes linéaires est simplement un modèle où l'on impose aux composantes $(x_{11}), \dots, (x_{p1})$ d'appartenir à des sous espaces fixés $\mathcal{F}_1, \dots, \mathcal{F}_p$ respectivement. Plus précisément :

On appelle modèle PARAFAC / CANDECOMP sous contraintes linéaires

définies par les SEV $\mathcal{F}_1, \mathcal{F}_2$ et \mathcal{F}_3 une décomposition de la forme :

$$A = \sum_{i=1}^r x_{1i} \otimes \dots \otimes x_{pi} + R$$

avec :

$\forall i, x_{ji} \in \mathcal{F}_j$

r fixé

$\|R\|$ minimale

2. Lemme

■ Enoncé

Calculer le modèle ParaFac / Candecomp de A sous les contraintes définies par $\mathcal{F}_1, \dots, \mathcal{F}_p$ revient à calculer ce modèle pour le tenseur $\mathcal{P}(A)$, projection orthogonale de A sur $\mathcal{F}_1 \otimes \dots \otimes \mathcal{F}_p$.

■ Démonstration

La démonstration est calquée sur celle de l'ACP-DCL, et celle du modèle Candecomp / Parafac d'ordre 3, comme cela avait été annoncé. Rappelons là :

1 - Notons \mathcal{P} la projection orthogonale de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ sur $\mathcal{F}_1 \otimes \dots \otimes \mathcal{F}_p$, et \mathcal{P}^\perp le projecteur orthogonal (i.e. sur $(\mathcal{F}_1 \otimes \dots \otimes \mathcal{F}_p)^\perp$).

2 - Notons également pour simplifier : $\Sigma_1 = \sum_i x_{1i} \otimes \dots \otimes x_{pi}$. Ainsi :

$$\begin{aligned} \|A - \Sigma_1\|^2 &= \|(\mathcal{P} + \mathcal{P}^\perp)(A - \Sigma_1)\|^2 \\ &= \|\mathcal{P}(A - \Sigma_1)\|^2 + \|\mathcal{P}^\perp(A - \Sigma_1)\|^2 \quad \text{car } \mathcal{P} \text{ et } \mathcal{P}^\perp \text{ sont orthogonaux} \\ &= \|\mathcal{P}(A) - \Sigma_1\|^2 + \|\mathcal{P}^\perp(A)\|^2 \quad \text{car } \mathcal{P}(\Sigma_1) = \Sigma_1 \text{ et } \mathcal{P}^\perp(\Sigma_1) = 0 \end{aligned}$$

3 - $\|A - \Sigma_1\|^2$ est donc minimale lorsque $\|\mathcal{P}(A) - \Sigma_1\|^2$ l'est.

3. Application

■ Notations

1 - Soit P_1 le projecteur orthogonal de \mathcal{E}_1 sur \mathcal{F}_1 .

2 - Soit \mathcal{P} le projecteur orthogonal de $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_p$ sur $\mathcal{F}_1 \otimes \dots \otimes \mathcal{F}_p$

■ Lemme

Rappelons qu'alors :

$$\mathcal{P}(u_1 \otimes \dots \otimes u_p) = P_1.u_1 \otimes \dots \otimes P_p.u_p$$

En effet (la démonstration est calquée sur le cas $p = 3$) :

- l'opérateur \mathcal{P} ainsi défini est bien une projection car $\mathcal{P}^2 = \mathcal{P}$
- on vérifie aisément que $\text{Im}(\mathcal{P}) = \mathcal{F}_1 \otimes \dots \otimes \mathcal{F}_p$
- on vérifie également que $X \in (\mathcal{F}_1 \otimes \dots \otimes \mathcal{F}_p)^\perp \Rightarrow \mathcal{P}(X) = 0$

■ Calcul de $\mathcal{P}(A)$

Le calcul de $\mathcal{P}(A)$ est alors aisé :

$$\begin{aligned} \mathcal{P}(A) &= \mathcal{P}\left(\sum_i \alpha_i E_i\right) \\ &= \sum_i \alpha_i \mathcal{P}(E_i) \\ &= \sum_i \alpha_i (P_1 e_{11_1}) \otimes \dots \otimes (P_p e_{p1_p}) \end{aligned}$$

Si $P_1 = \sum_i e_{11_1} \otimes p_{11_1}$ (p_{11_1} est le $i_1^{\text{ème}}$ vecteur colonne de la matrice P_1)

$$P_2 = \sum_i e_{21_2} \otimes p_{21_2}$$

...

$$P_p = \sum_i e_{p1_p} \otimes p_{p1_p}$$

on peut également écrire cette égalité sous la forme :

$$p(\mathbf{A}) = \sum_{i_1 \dots i_p} \alpha_{i_1 \dots i_p} p_{1i_1} \otimes \dots \otimes p_{pi_{p_1}}$$

1. QUELQUES REPÈRES D'HISTOIRE

1. POSITION DU PROBLEME

1.1 Définition du rang

Le rang d'un tenseur d'ordre p de $\mathcal{A}(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$ est une généralisation naturelle du rang d'une matrice (considérée alors comme tenseur d'ordre 2), qui a été proposée, probablement indépendamment, par plusieurs auteurs, et notamment par (STRASSEN, 1972).

Plusieurs auteurs utilisent les définitions suivantes reconnues comme équivalentes pour un tenseur d'ordre 3 :

□ étant donnée une forme trilinéaire (α_{ijk}) opérant sur le triplet de vecteurs (x, y, z) , on appelle rang de cette forme le nombre r minimal tel qu'une représentation de la forme :

$$\langle A, x \otimes y \otimes z \rangle = \sum_{p=1}^r \langle u_p, x \rangle \cdot \langle v_p, y \rangle \cdot \langle w_p, z \rangle$$

existe (ATKINSON & LLOYD, 1980 ; HOWELL, 1978 ; LAFON, 1975).

Soit :

$$A = \sum_{p=1}^r u_p \otimes v_p \otimes w_p.$$

Cette écriture synthétique est souvent présentée sous forme développée en fonction des coordonnées (α_{ijk}) de A , (v_{pi}) , (φ_{pj}) et (ω_{pp}) de u_p , v_p et w_p respectivement :

$$\alpha_{ijk} = \sum_{p=1}^r v_{pi} \varphi_{pj} \omega_{pk}$$

□ Le rang d'un tenseur de coefficients (α_{ijk}) est le nombre minimal de matrices de rang 1 qui forment un système générateur de l'espace vectoriel engendré par les matrices A_k de coefficients général $(\alpha'_{ij} = \alpha_{ijk})$: (ATKINSON & LLOYD, 1980 ; BROCKETT & DOBKIN, 1978 ; HOWELL, 1978 ; JA' JA', 1979 ; LAFON, 1975).

□ Le rang d'un tenseur (α_{ijk}) est le rang de la matrice $B(x) = \sum_k \xi_k A_k$ où les A_k ont la même signification que précédemment, où $x = (\xi_1, \dots, \xi_p)$ est un vecteur de p paramètres (LAFON, 1975).

KRUSKAL (1977) propose une définition par un biais légèrement différent, qui est la possibilité d'effectuer des décompositions en valeurs singulières simultanées d'une famille de matrices, mais reconnaît l'équivalence avec les définitions précédentes dans (KRUSKAL, 1989).

2. Applications du rang d'un tenseur

Le rang d'une forme trilinéaire a été régulièrement étudié depuis une vingtaine d'années, car il est directement relié à l'évaluation du nombre minimal de multiplications nécessaires pour l'évaluation simultanée d'une famille de formes bilinéaire (ATKINSON & STEPHENS, 1979 ; GASTINEL, 1972 ; JA' JA', 1979 ; LAFON, 1975). Ce nombre est relié au temps de calcul pour la multiplication de deux matrices, de deux polynômes, et possède par là même des applications concrètes très importantes.

1. Application au produit de matrices

Par exemple, on peut montrer que le calcul du produit de deux matrices réelles d'ordre 3 peut se réaliser avec 23 multiplications seulement, au lieu des 27 habituellement réalisées (BROCKETT & DOBKIN, 1976 ; LADERMAN,

1976), de deux matrices d'ordre 4 avec 46 multiplications (BROCKETT & DOBKIN, 1976) et de deux matrices d'ordre 5 avec 103 multiplications (SCHACHTEL, 1978). Ces travaux trouvent des prolongements naturels dans l'optimisation du temps de calcul d'entités algébriques, et c'est un champ de recherches très actif, notamment depuis une vingtaine d'années.

□

STRASSEN (1972) a produit un algorithme permettant de calculer le produit de deux matrices 2×2 avec 7 multiplications seulement au lieu de 8. LAFON (1975) a montré que ce nombre est optimal. L'examen de cet algorithme et la preuve de son optimalité, dont la présentation est tirée de LAFON (1975), permet de saisir le lien entre produits de matrices et rang d'un tenseur.

Le problème étudié par LAFON (1975) est légèrement différent : étant donnée une famille (B_i) de p matrices, quel est le nombre minimal de multiplications nécessaires pour calculer les p valeurs $\langle B_i, x \otimes y \rangle$ pour des valeurs arbitraires de x et y ? LAFON (1975) montre alors que si r est le rang tensoriel de la famille (B_i) , il est possible de calculer les p valeurs $\langle B_i, x \otimes y \rangle$ en utilisant r multiplications seulement, d'une part, et que $r/2$ est un minorant du nombre minimal cherché, d'autre part.

1 - Si l'on considère par exemple deux matrices A et B de $M_2(K)$, définies par :

$$A = \begin{bmatrix} \alpha_1 & \alpha_3 \\ \alpha_2 & \alpha_4 \end{bmatrix} \text{ et } B = \begin{bmatrix} \beta_1 & \beta_3 \\ \beta_2 & \beta_4 \end{bmatrix}$$

$$\text{et } C = AB = \begin{bmatrix} \alpha_1\beta_1 + \alpha_3\beta_2 & \alpha_1\beta_3 + \alpha_3\beta_4 \\ \alpha_2\beta_1 + \alpha_4\beta_2 & \alpha_2\beta_3 + \alpha_4\beta_4 \end{bmatrix}$$

le problème revient à calculer simultanément les 4 formes bilinéaires $\alpha_1\beta_1 + \alpha_3\beta_2$, $\alpha_1\beta_3 + \alpha_3\beta_4$, $\alpha_2\beta_1 + \alpha_4\beta_2$ et $\alpha_2\beta_3 + \alpha_4\beta_4$ de K^4 , associées respectivement aux matrices :

$$B_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; B_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}; B_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; B_4 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

de $\mathcal{M}_4(K)$.

2 - Si a désigne le vecteur $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ de K^4 et b le vecteur $(\beta_1, \beta_2, \beta_3, \beta_4)$ de K^4 respectivement, nous avons à évaluer les quatre quantités $\langle B_i, a \otimes b \rangle$ pour $i = 1, \dots, 4$.

3 - Si r est le rang de la famille de matrices (B_i) , cette évaluation peut se réaliser en r multiplications seulement. Or, LAFON (1975) montre que le rang de (B_1, \dots, B_4) est 7. 7 multiplications sont donc suffisantes pour calculer le produit de deux matrices de $\mathcal{M}_2(K)$.

4 - Les calculs avec sept multiplications sont les suivants (d'après PRESS & Al., 1989) :

On calcule les sept quantités θ_i suivantes, qui chacune font appel à une multiplication et une seule :

- $\theta_1 = (\alpha_1 + \alpha_4) \cdot (\beta_1 + \beta_4)$
- $\theta_2 = (\alpha_3 + \alpha_4) \cdot \beta_1$
- $\theta_3 = \alpha_1 \cdot (\beta_2 - \beta_4)$
- $\theta_4 = \alpha_4 \cdot (-\beta_1 + \beta_2)$
- $\theta_5 = (\alpha_1 + \alpha_2) \cdot \beta_4$
- $\theta_6 = (-\alpha_1 + \alpha_2) \cdot (\beta_1 + \beta_2)$
- $\theta_7 = (\alpha_3 - \alpha_4) \cdot (\beta_2 + \beta_4)$

Chacun des coefficients γ_i s'obtient par somme ou différence des quantités θ_i , comme suit :

- $\gamma_1 = \theta_1 + \theta_4 - \theta_5 + \theta_7$
- $\gamma_2 = \theta_2 + \theta_4$
- $\gamma_3 = \theta_3 + \theta_5$

$$\circ \gamma_4 = \theta_1 - \theta_2 + \theta_3 + \theta_6$$

□

GASTINEL (1971), avec des notations légèrement différentes reposant sur le produit de Hadamard, a montré que le rang de la famille (B_i) associée au produit des matrices de $\mathcal{M}_n(\mathbb{R})$ est inférieur ou égal à $n^3 - n + 1$. Il est donc suffisant de calculer 7 produits pour le produit de deux matrices de $\mathcal{M}_2(\mathbb{R})$, 25 produits pour $\mathcal{M}_3(\mathbb{R})$, etc

□

Comme l'algorithme proposé par (GASTINEL, 1971 ; LAFON, 1975 ; STRASSEN, 1969) ne fait pas appel à la commutativité, il est applicable au produit des matrices par blocs, et peut être réitéré pour le produit des grandes matrices.

Si deux matrices $(2n \times 2n)$ sont décomposées chacune en n^2 blocs (2×2) , leur produit se ramène à n^3 multiplications de matrices (2×2) , donc à $7n^3$ multiplications, au lieu des $(2n)^3 = 8n^3$ par la méthode classique.

Si deux matrices (4×4) sont décomposées en 4 blocs (2×2) , leur produit nécessitera non pas l'évaluation de 8 produits de deux matrices (2×2) , mais de 7 seulement (algorithme de STRASSEN sur les blocs). Chacun de ces produits ne nécessitera que 7 multiplications. Le produit de deux matrices (4×4) peut donc se réaliser avec 49 multiplication (au lieu de $64 = 4^3$).

Plus généralement, si le produit de deux matrices $(2^n \times 2^n)$ nécessite r_n multiplications, $r_{n+1} = 7r_n$. Comme $r_2 = 7$, $r_n = 7^n$ [au lieu de $(2^n)^3 = (2^3)^n = 8^n$].

Plus généralement encore, si $m(n)$ est tel que le produit de deux matrices $(n \times n)$ peut se réaliser avec $m(n)$ multiplications au plus, il est immédiat que $m(pq) \leq m(p)m(q)$. Or, nous connaissons :

- $m(2) = 7$
- $m(3) \leq 23$
- $m(4) \leq 46$

$$\circ m(5) \leq 103$$

qui permettent de calculer des majorants intéressants de $m(2^p \cdot 3^q \cdot 5^r)$.

□

HOPCROFT & KERR (1971), par une adaptation de l'algorithme de STRASSEN (1969), ont étudié les produits de matrice AB , où $A \in \mathcal{M}(A^p, A^2)$ et $B \in \mathcal{M}(A^2, A^n)$ [on écrit pour simplifier de type $(p \times 2)$] et B de type $(2 \times n)$. Ils montrent alors que $m(p, n)$ [nombre maximal de multiplications nécessaires] est égal à $(3 \cdot p \cdot n + \max(n, p))/2$, en l'absence de commutativité dans les éléments de A , alors qu'un calcul élémentaire nécessite $2 \cdot p \cdot n$ multiplications.

Cet algorithme peut également être itéré pour étudier le produit de matrices $(p \times q)$ et $(q \times n)$. BROCKETT & DOBKIN (1976) ont étudié en détail le cas $(n \times q)$ et $(q \times n)$, et ont montré que, pourvu que $q \leq \log_2(n)$, $m(q, n) = n^2 + o(n^2)$, ce qui est agréablement surprenant, car q n'intervient pas.

Remarquons également que l'algorithme de HOPCROFT & KERR (1971) fournit $m(2, n) = (3n^2 + n)/2$. Si l'anneau A est commutatif (ce qui est le cas des corps usuels, notamment \mathbb{R} , mais ne permet pas d'itérer le résultat pour étudier les matrices par blocs), WASKMANN (cité par BROCKETT & DOBKIN, 1976) montre que $m(2, n) \leq n^2 - 2n + 1$, ce qui permet la multiplication de deux matrices (4×4) en 46 multiplications et de deux matrices (3×3) en 23 multiplications seulement.

3. Dépendance du rang selon le corps de base et extension aux modules

HOWELL (1978) semble être l'un des premiers à avoir exploré en profondeur la dépendance du rang d'un tenseur vis à vis du corps de base des espaces vectoriels sur lesquels étaient bâtis les tenseurs. Par exemple, le rang d'un même tenseur à coefficients rationnels n'est pas le même selon

qu'il est considéré comme élément de $\mathcal{A}_3(\mathbb{Q}^n)$, $\mathcal{A}_3(\mathbb{R}^n)$ ou $\mathcal{A}_3(\mathbb{C}^n)$.

Il étend même la notion de tenseur et de rang à des modules, en étudiant les formes multilinéaires sur un anneau A , (et non sur un corps comme pour les espaces vectoriels). Il étudie le rang en fonction des anneaux sur lesquels sont bâtis ces modules, notamment parce que les modules sur \mathbb{Z} interviennent fréquemment dans les opérations algébriques (le tenseur de la multiplication des matrices sur un corps K ne contient par exemple que des 0 et des 1).

Si $r(T, A)$ désigne le rang de T considéré comme tenseur à éléments dans A , il explicite deux lemmes simples mais féconds en applications :

- $A \subset B \Rightarrow r(T, A) \geq r(T, B)$
- si $B = \Phi(A)$ où Φ est un homomorphisme d'anneaux, $r(T, A) \geq r(T, B)$.

HOWELL (1978) établit également le théorème suivant : si $X = A[X]$ est l'anneau des polynômes d'un anneau A , et T un tenseur à éléments dans A , $r(T, X) = r(T, A)$. Il en déduit le théorème important suivant : si K est un corps algébriquement clos, et T un tenseur à éléments dans un anneau A contenant K , alors $r(T, A) = r(T, K)$.

4. Solution du problème pour $\mathcal{A}_3(\mathbb{R}^2 \times \mathbb{R}^n \times \mathbb{R}^n)$

Ce problème a été étudié notamment par ATKINSON & STEPHENS (1979) ; JA' JA' (1979) et LAFON (1975).

Un premier résultat important a été proposée par LAFON (1975) qui montre que le rang de $T \in \mathcal{A}_3(\mathbb{R}^2 \times \mathbb{R}^n \times \mathbb{R}^n)$, écrit sous la forme $T = i \otimes T_i + j \otimes T_j$, est n si la matrice $T_i^{-1}T_j$ est diagonalisable, et étend ce résultat de la façon suivante : étant donnée une famille (T_i) de matrices, le rang du tenseur $T = \sum_i e_i \otimes T_i$ est n si les matrices $T_i^{-1}T_j$ sont simultanément diagonalisables pour tout couple (i, j) .

Un résultat en un sens plus général a été publié par dans ATKINSON &

STEPHENS (1979), sous la forme suivante :

- si X est une matrice inversible de $M_n(\mathbb{C}) \equiv \mathcal{A}_2(\mathbb{C}^n, \mathbb{C}^n)$
 - si le polynôme caractéristique de $Y - \lambda X$ possède n racines distinctes,
- alors, pour tout couple (U, V) de matrices de $\mathcal{A}_2(\mathbb{C}^n \times \mathbb{C}^m)$, sachant que :
- les matrices $A = XU$ et $B = YV \in \mathcal{A}_2(\mathbb{C}^n, \mathbb{C}^m)$,
 - $T = i \otimes A + j \otimes B \in \mathcal{A}_3(\mathbb{C}^2 \times \mathbb{C}^n \times \mathbb{C}^m)$
- $$r(T) \leq n.$$

ATKINSON & STEPHENS (1979) en déduisent un majorant pour $r(n, n, n)$ de la forme : $r(n, n, n) \leq n^2/2 + O(n)$.

Le calcul explicite du rang en fonction de l'ordre du tenseur a été totalement résolu pour un couple de formes bilinéaires, donc un tenseur de $\mathcal{A}_3(\mathbb{R}^2 \times \mathbb{R}^n \times \mathbb{R}^n)$ par JA' JA' (1979), en utilisant les résultats sur les faisceaux de matrices obtenus par WEIERSTRASS en 1867 (Zur Theorie der bilinearen und quadratischen Formen, Monatsch. Akad. Wiss. Berlin, pp. 310-338) et KRONECKER en 1890 (Algebraische Reduction der Schaaren bilinearen Formen, S.-B. Akad. Berlin, pp. 763-776) et présentés par GRANDMATCHER (1969) .

2. DEFINITIONS

1. Rang d'un tenseur

Soit T un tenseur de $\mathcal{A}_3(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$. On appelle rang de T le plus petit nombre entier r , noté $r(T)$, tel qu'une décomposition :

$$T = \sum_{k=1}^r x_k \otimes y_k \otimes z_k$$

existe, où $x_k \in \mathcal{E}_1$, $y_k \in \mathcal{E}_2$, $z_k \in \mathcal{E}_3$.

Cette définition est équivalente à celle proposée par ATKINSON & LLOYD (1980), KRUSKAL (1977), KRUSKAL (1986), HOWELL (1978), LAFON (1975).

2. Equivalence avec la définition de KRUSKAL (1977)

La définition de KRUSKAL fait appel à un découpage en "tranches" du tableau de données à trois entrées. Il faut donc montrer que cette définition est indépendante du plan de découpage choisi, ce qu'a montré KRUSKAL (1991).

3. Rang Maximum d'un tenseur

Supposons que $\dim(\mathcal{E}_1) = n_1$, $\dim(\mathcal{E}_2) = n_2$, $\dim(\mathcal{E}_3) = n_3$. $\mathcal{A}_3(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ est alors isomorphe à $\mathcal{A}_3(\mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \mathbb{R}^{n_3})$.

Soit T un tenseur de $\mathcal{A}_3(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$, et $r(T)$ son rang. On sait que, $\forall T \in \mathcal{A}_3(\mathcal{E})$, $r(T) \leq n_1 \times n_2 \times n_3$. Lorsque T décrit $\mathcal{A}_3(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$, $r(T)$ décrit donc un sous-ensemble de \mathbb{N} , borné par $n_1 \times n_2 \times n_3$. Cet ensemble majoré possède un plus grand élément, que l'on appelle rang maximum d'un tenseur (RMT), et que l'on note $r(n_1, n_2, n_3)$.

Si $n_1 = n_2 = n_3 = n$, on note $r(n) = r(n, n, n)$.

3. MINORANT DU RANG MAXIMUM D'UN TENSEUR

Nous donnons dans ce qui suit un minorant du RMT.

1. Théorème

$$r(n_1, n_2, n_3) \geq \frac{n_1 n_2 n_3}{n_1 + n_2 + n_3 - 2}$$

■ Démonstration

1 - L'espace $\mathcal{A}_3(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ est une variété différentielle de dimension $n_1 n_2 n_3$.

2 - Si $r = r(n_1, n_2, n_3)$, $\mathcal{A}_3(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ est inclus dans $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$, puisque tout tenseur T s'écrit comme somme de r tenseurs décomposés, donc est un point de $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$.

3 - Soit G un point de $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$. G s'écrit :

$$G = \sum_{i=1}^r x_i \otimes y_i \otimes z_i.$$

et, si l'on impose : $\|x_i\| = \|y_i\| = \|z_i\| = 1$, ce que l'on peut faire en toute généralité :

$$G = \sum_{i=1}^r \lambda_i x_i \otimes y_i \otimes z_i$$

4 - On peut montrer que cette application est localement un homéomorphisme, si G est de rang r . Il existe donc un homéomorphisme entre un voisinage de G et un voisinage de $(\lambda_i x_i \otimes y_i \otimes z_i)_i \in (\mathbb{R} \times S_3 \times S_3 \times S_3)^r$, qui est de dimension $r[1+(n_1-1)+(n_2-1)+(n_3-1)] = r(n_1+n_2+n_3-2)$. En faisant un choix continu au voisinage d'un tenseur G donné, on montre ainsi que $r(n_1+n_2+n_3-2) \geq \dim(\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3))$.

5 - Comme $\Gamma_3^r(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ contient $\mathcal{A}_3(\mathcal{E}_1 \times \mathcal{E}_2 \times \mathcal{E}_3)$ de dimension n^3 , on a l'inégalité : $r(n_1+n_2+n_3-2) \geq n^3$, soit : $r \geq \frac{n_1 n_2 n_3}{n_1+n_2+n_3-2}$

2. Généralisation

Il est facile de généraliser ce théorème à $r(n_1, \dots, n_p)$, rang maximal des tenseurs de $\mathcal{A}_p(\mathbb{R}^{n_1} \times \dots \times \mathbb{R}^{n_p})$:

$$r(n_1, \dots, n_p) \geq \frac{n_1 \dots n_p}{n_1 + \dots + n_p - p + 1}$$

3. Remarques

1 - Si $n_1 = n_2 = n_3 = n$, le théorème devient :

$$r(n) \geq \frac{n^3}{3n-2}$$

2 - Si n devient grand : $r(n) \geq \frac{n^3}{3n-2} \approx \frac{n^2}{3}$ lorsque $n \rightarrow \infty$.

3 - Ce résultat est cité par ATKINSON & STEPHENS (1979) comme connu mais non publié par BROCKETT (1976), et a été publié par HOWELL (1978) dans un contexte plus général avec une démonstration autre.

4. Résultats publiés par Howell (1978)

Les résultats publiés par HOWELL (1978) sont les suivants :

1 - Si K est un corps de cardinal infini et T un tenseur de $\mathcal{A}_3(K^m \times K^n \times K^p)$, alors $r(T, K) \geq mnp/(m+n+p-2)$.

2 - Si K est un corps possédant q éléments et T un tenseur de $\mathcal{A}_3(K^m \times K^n \times K^p)$, alors $r(T, K) \geq mnp/[m+n+p-2\log_q(q-1)]$.

3 - Si A est un anneau commutatif unitaire fini ou infini et T un tenseur de $\mathcal{A}_3(A^m \times A^n \times A^p)$, alors $r(T, A) \geq mnp/(m+n+p)$.

4. MAJORANTS DU RANG MAXIMAL D'UN TENSEUR PROPOSES DANS LA LITTÉRATURE

1. Position du problème

La majoration du RMT a été bien étudiée depuis une quinzaine d'années. Les principaux résultats sont soit d'ordre général pour $r(m, n, p)$, soit des calculs explicites pour des petites valeurs données de m, n, p . Un rappel des résultats non triviaux est proposé dans ATKINSON & STEPHENS (1979), pour $r(m, n, p)$, avec $m \leq n \leq p$.

On y constate que l'évaluation du rang d'un tenseur d'ordre ≥ 3 reste un problème difficile, et ouvert. Les résultats partiels obtenus, notamment

en terme de majorants, le sont par des techniques parfois très différentes, (ATKINSON & LLOYD, 1980 ; ATKINSON & STEPHENS, 1979 ; HOWELL, 1978), souvent grâce à plusieurs lemmes intermédiaires.

2. Résultats généraux

- si $p \geq mn$, $r(m,n,p) = mn$ (DOBKIN, 1973)
- $r(2,n,p) = \min(2n, n + \lfloor p/2 \rfloor)$ (JA'JA, 1979, ATKINSON & STEPHENS, 1979)
- $r(3,n,n) \leq 2n$ (DOBKIN, 1973)
- $r(m,n,p) \leq m + \lfloor p/2 \rfloor n$ (ATKINSON & STEPHENS, 1979)
- $r(m,n,mn-k) = mn - k^2 + r(k,k,k^2-k)$ si $k \leq \min(m,n)$
- $r(m,n,p) \leq p + r(m-1,n-1,p)$ (HOWELL, 1978)
- $r(m,n,p) \leq n + r(m-1,n,p-1)$ (HOWELL, 1978)
- $r(m,n,p) \leq m + r(m,n-1,p-1)$ (HOWELL, 1978)
- $r(n,n,p) \leq \lfloor p/2 \rfloor n$ si p pair (ATKINSON & LLOYD, 1980)
- $r(n,n,n) \leq 3n^2/4$ (HOWELL, 1978)
- $r(n,n,n) \leq n^2/2 + O(n)$ (ATKINSON & STEPHENS, 1979)

3. Résultats particuliers

- $r(2,2,2) = 3$ (KRUSKAL, 1986)
- $r(3,3,3) = 5$ (KRUSKAL, non publié)
- $r(4,4,4) = 8$ ou 9 (LLOYD, cité dans ATKINSON & LLOYD, 1980)
- $r(4,4,12) = 14$ (ATKINSON & STEPHENS, 1979)

- $r(n,n,3) \leq 2n-1$
- $r(n,n,3) \geq \lfloor 7n/4 \rfloor$
- $r(n,n+1,3) \leq 2n$

ATKINSON & STEPHENS (1979) proposent des résultats complets pour $r(3,3,p)$:

- $r(3,3,1) = 1$
- $r(3,3,2) = 4$
- $r(3,3,3) = 5$
- $r(3,3,4) = 6$

$$\begin{aligned}
r(3,3,5) &= 6 \text{ ou } 7 \\
r(3,3,6) &= 7 \\
r(3,3,7) &= 8 \\
r(3,3,8) &= 8 \\
r(3,3,p) &= 9 \quad \forall p \geq 9
\end{aligned}$$

Ces résultats ont également été proposés par KRUSKAL (KRUSKAL, non publié)

4. Méthodes utilisées

■ Généralisation de l'algorithme d'élimination de Gauss

L'algorithme d'élimination de Gauss, plus connu pour la résolution numérique des systèmes linéaires et l'inversion des matrices, peut servir à la détermination du rang des matrices et des tenseurs (HOWELL, 1978 ; STRASSEN, 1969).

Soit

- M une matrice de $\mathcal{A}_2(\mathbb{R}^m \times \mathbb{R}^n)$
- c le premier vecteur colonne de M
- l le premier vecteur ligne de M
- μ_{11} le scalaire de cl_1 , supposé non nul

L'écriture suivante est alors classique :

$$M = l \otimes c / \mu_{11} + M_1$$

où $M_1 \in \mathcal{A}_2(\mathbb{R}^{m-1} \times \mathbb{R}^{n-1})$. Par itération de ce procédé sur M_1 , et au delà, M est décomposée en somme de r tenseurs élémentaires décomposés, et r est le rang de M.

Cet algorithme se généralise ainsi :

$$1 - \text{Soit } A \in \mathcal{A}_3(\mathbb{R}^m, \mathbb{R}^n, \mathbb{R}^p). \quad A = \sum_1 e_{11} \otimes A_1.$$

2 - Chacun des A_i est décomposé selon : $A_i = l_i \otimes c_i / \mu_{111} + B_i$ (algorithme de Gauss sur les A_i).

3 - D'où : $A = \sum_i e_{11} \otimes (l_i \otimes c_i / \mu_{111} + B_i) = \sum_i \mu_{111} e_{11} \otimes l_i \otimes c_i + \sum_i e_{11} \otimes B_i$

4 - Comme $B_i \in \mathcal{A}_2(\mathbb{R}^{n-1}, \mathbb{R}^{p-1})$, et que i varie de 1 à n , on en déduit :
 $r(m, n, p) \leq n + r(n, m-1, p-1)$.

Ce résultat a été retrouvé récemment par R. BAILEY et C. ROWLEY (DENIS, communication personnelle).

L'algorithme proposé ici n'est pas celui présenté par HOWELL (1978), et est de portée moins générale : il fait appel à la division dans \mathbb{R} , et est généralisable à tout triplet de K -espaces vectoriels \mathcal{E} , \mathcal{F} et \mathcal{G} de dimension respective m, n et p .

L'algorithme proposé par HOWELL (1978), par le biais de la recherche de PGCD entre plusieurs éléments, de changements de bases et l'introduction d'opérateurs particuliers, étend cette démonstration aux A -modules, où A est un anneau intègre.

HOWELL (1978) montre ensuite simplement, à partir de cette inégalité, que $r(n, n, n) \leq [3n^2/4]^+$, où $[x]^+$ est le plus petit entier supérieur ou égal à x . Cette égalité est vérifiée pour tout tenseur A de $\mathcal{A}_3(\mathcal{E})$, où \mathcal{E} est un A -module de dimension n sur un anneau intègre A . Ce résultat avait été publié par DOBKIN (1973) pour les \mathbb{C} -espaces vectoriels lorsque n est pair.

HOWELL (1978), toujours dans le même article, étend ses résultats à $\mathcal{A}_p(\mathcal{E}_1 \times \dots \times \mathcal{E}_p)$, où les \mathcal{E}_i sont des A -modules de dimension n_i sur un anneau intègre A :

$$r(n_1, \dots, n_p) \leq \frac{\prod_{i=1}^p n_i}{n_j n_k} + r(n_1, \dots, n_j-1, \dots, n_k-1, \dots, n_p)$$

d'où il déduit :

$$r(n, \dots, n) \leq \frac{p}{2(p-1)} n^{p-1} + o(p-1)$$

2. MAJORANTS DU RANG MAXIMUM D'UN TENSEUR

1. TROIS LEMMES IMPORTANTS

Dans ce qui suit, nous montrerons trois lemmes dont les résultats sont à la base de plusieurs majorations du rang d'un tenseur général. Aucun de ses lemmes n'est original. Mais une démonstration consistante avec les notations retenues est cependant proposée.

1. Premier lemme

■ Enoncé

$r(n) = r$ est équivalent à : tout sous espace \mathcal{N} de dimension n de $\mathcal{A}_2(\mathcal{E}) \equiv \mathcal{M}(\mathcal{E})$, engendré par n matrices M_j quelconques, peut être engendré par r matrices $y_i \otimes z_i$ de rang 1.

■ Démonstration

Soit $r = r(n)$. D'où : $\forall T \in \mathcal{A}_3(\mathcal{E}), T = \sum_{i=1}^r x_i \otimes y_i \otimes z_i$.

Posons : $x_i = \sum_{j=1}^n \xi_{ij} e_j$. D'où :

$$\begin{aligned} T &= \sum_{i=1}^r x_i \otimes y_i \otimes z_i \\ &= \sum_{i=1}^r \left[\sum_{j=1}^n \xi_{ij} e_j \right] \otimes y_i \otimes z_i \\ &= \sum_{j=1}^n e_j \otimes \left[\sum_{i=1}^r \xi_{ij} y_i \otimes z_i \right] \\ &= \sum_{j=1}^n e_j \otimes M_j \quad ; \quad M_j = \sum_{i=1}^r \xi_{ij} y_i \otimes z_i. \end{aligned}$$

Et on remarque que la famille $(y_i \otimes z_i)$ est une famille génératrice de l'espace vectoriel engendré par les $(M_j)_j$. Pour la réciproque, on remarque que, si T parcourt $\mathcal{A}_3(\mathcal{E})$, les (M_j) sont alors quelconques $\left[\text{il suffit de choisir } T = \sum_{j=1}^n e_j \otimes M_j \right]$.

■ Commentaire

Ce lemme, simple dans son énoncé et sa démonstration, permet de ramener un problème de rang sur $\mathcal{A}_3(\mathcal{E})$ à un problème analogue sur une famille de matrices de $\mathcal{A}_2(\mathcal{E})$. C'est donc un outil puissant. Ce lemme a également été proposé par JA' JA' (1979) et KRUSKAL (1991).

2. Deuxième lemme

■ Position du problème

Soit M une matrice de rang n décomposée en : $M = \sum_{i=1}^n y_i \otimes z_i$. Dans quelle mesure peut on choisir arbitrairement les n couples $(y_i \otimes z_i)$? M étant de rang n , on en déduit que les systèmes (y_i) et (z_i) sont libres.

■ Enoncé

Soient :

- \mathcal{E} et \mathcal{F} deux espaces vectoriels de dimension n
- M un isomorphisme de \mathcal{E} dans \mathcal{F}
- $(y_i)_i$ et $(z_i)_i$ deux bases de \mathcal{E} et \mathcal{F} respectivement.

Alors, M s'écrit sous la forme $M = \sum_{i=1}^n y_i \otimes z_i$ si, et seulement si : $\forall i \leq n$, $\langle y_i, M^{-1} z_j \rangle = \delta_{ij}$.

$$M = \sum_{i=1}^n y_i \otimes z_i \iff \forall i \leq n, \langle y_i, M^{-1} z_j \rangle = \delta_{ij}$$

■ Démonstration

[\Rightarrow] : la base (y_i) possède une base duale (y_i^*) telle que $\langle y_i, y_j^* \rangle = \delta_{ij}$. On en déduit : $z_i = M.y_i^*$, soit : $y_i^* = M^{-1}z_i$. D'où : $\langle y_i, M^{-1}z_j \rangle = \delta_{ij}$.

[\Leftarrow] : Réciproquement : si $\langle y_i, M^{-1}z_j \rangle = \delta_{ij}$, alors $M^{-1}z_i = y_i^*$, d'où $z_i = M.y_i^*$; or $M = \sum_{i=1}^n y_i \otimes M.y_i^* = \sum_{i=1}^n y_i \otimes z_i$ ■

3. Troisième lemme

Le troisième lemme est une extension du lemme précédent au cas où la matrice donnée n'est pas de plein rang.

■ Enoncé

Soit une matrice M de rang k de $\mathcal{A}_2(\mathcal{E})$, et décomposée sous la forme

$$M = \sum_{i=1}^k y_i \otimes z_i, \text{ alors :}$$

1) M est un isomorphisme de $\text{Im}(M')$ sur $\text{Im}(M)$

2) $\forall i \leq k, z_i \in \text{Im}(M)$ et $y_i \in \text{Im}(M')$

3) $\forall i, j \leq k, \langle y_i, \hat{M}^{-1} z_j \rangle = \delta_{ij}$ où \hat{M} est la restriction de M à $\text{Im}(M')$.

$$M = \sum_{i=1}^k y_i \otimes z_i$$

$$\iff \text{Im}(M') \simeq \text{Im}(M)$$

$$\iff \forall i \leq k, z_i \in \text{Im}(M) \text{ et } y_i \in \text{Im}(M')$$

$$\iff \forall i, j \leq k, \langle y_i, \hat{M}^{-1} z_j \rangle = \delta_{ij}$$

■ Démonstration

Nous allons montrer : (2) \Rightarrow (1) \Rightarrow (3)

1 - $(y_i)_i$ est une famille libre (sinon le rang de M serait au plus $k-1$), ainsi que $(z_i)_i$. Soient respectivement \mathcal{F} et \mathcal{E} les deux sous-espaces vectoriels de dimension k de \mathcal{E} engendrés par ces deux familles. $(y_i)_i$ est une base de \mathcal{F} et possède une base duale $(y_i^*)_i$ telle que $\langle y_i, y_j^* \rangle = \delta_{ij}$. Il en découle : $M \cdot y_j^* = \sum_{i=1}^k \langle y_i, y_j^* \rangle z_i = z_j$. D'où : $z_j \in \text{Im}(M)$.

2 - De même, en remarquant : $M' = \sum_{i=1}^k z_i \otimes y_i$, il découle : $y_i \in \text{Im}(M')$.

3 - De 2), il vient que $(y_i^*)_i$ est une base de $\text{Im}(M')$ et $(z_i)_i$ une base de $\text{Im}(M)$. Les bases duales $(y_i^*)_i$ et $(z_i^*)_i$ sont également des bases respectivement de $\text{Im}(M')$ et $\text{Im}(M)$. Comme $M \cdot y_i^* = z_i$ et $M' \cdot z_i^* = y_i$, M transforme une base de $\text{Im}(M')$ en une base de $\text{Im}(M)$. C'est donc bien un isomorphisme de $\text{Im}(M')$ sur $\text{Im}(M)$.

\rightarrow (3) : C'est l'application du deuxième lemme fondamental avec $\mathcal{E} = \text{Im}(M')$ et $\mathcal{F} = \text{Im}(M)$.

■ Conséquences

Dans ce qui suit, en utilisant ces trois lemmes, nous allons proposer plusieurs majorants du rang d'un tenseur général. L'idée simple est de majorer $r(n)$ par la construction explicite d'une famille de r matrices de rang 1 génératrices d'un espace vectoriel donné engendré par n matrices carrées arbitraires de $\mathcal{M}_n(\mathcal{E})$ (application du premier lemme). Ensuite, en utilisant les lemmes 2 et 3, on diminue le nombre de matrices de rang 1 du système générateur en montrant qu'on peut en choisir plusieurs égales entre elles. La démarche procède donc par construction de systèmes générateurs d'un espace engendré par n matrices arbitraires.

2. PREMIER MAJORANT DU RMT

■ Lemme

$$r(n) \leq n^2$$

□ Démonstration

Soit $T = \sum_{i,j,k=1}^n \lambda_{ijk} e_i \otimes e_j \otimes e_k$. Posons $t_{ij} = \sum_{k=1}^n \lambda_{ijk} e_k$. Il vient :

$$T = \sum_{i,j=1}^n e_i \otimes e_j \otimes t_{ij}. \text{ D'où : } r(n) \leq n^2 \blacksquare$$

3. DEUXIEME MAJORANT DU RMT

■ Enoncé

$$r(n) \leq n^2 - n + 1$$

■ Première Démonstration

□ La démonstration se réalise en plusieurs étapes.

• D'après le premier lemme fondamental, dire que $r(n) \leq n^2 - n + 1$ revient à dire que, étant donnée une famille de n matrices $(M_j)_{j=1,n}$ de $\mathcal{M}_2(\mathcal{E})$, il existe une famille de $n^2 - n + 1$ matrices de rang 1 générant les matrices M_j par combinaison linéaire.

• On peut montrer que l'on peut se ramener au cas où les matrices M_j sont toutes inversibles.

• On montre enfin qu'une telle famille de $n^2 - n + 1$ matrices de rang 1 existe bien.

□ Soit donc $(M_j)_{j=1,n}$ une famille de n matrices de $\mathcal{M}(\mathcal{E}) \equiv \mathcal{M}_2(\mathcal{E})$, décomposées de la façon suivante : $M_j = \sum_{i=1}^n \xi_{ij} y_{ij} \otimes z_{ij}$, soit de façon explicite :

$$[1] \quad \left\{ \begin{array}{l} M_1 = \xi_{11} y_{11} \otimes z_{11} + \dots + \xi_{n1} y_{n1} \otimes z_{n1} \\ M_2 = \xi_{12} y_{12} \otimes z_{12} + \dots + \xi_{n2} y_{n2} \otimes z_{n2} \\ \dots \\ M_n = \xi_{1n} y_{1n} \otimes z_{1n} + \dots + \xi_{nn} y_{nn} \otimes z_{nn} \end{array} \right.$$

□ On a donc une famille de n^2 matrices $y_{ij} \otimes z_{ij}$ de rang 1 générant les matrices M_j par combinaison linéaire. L'idée est alors d'identifier certains couples (y_{ij}, z_{ij}) afin d'en diminuer le nombre. Par exemple, dans quelle mesure peut on trouver a et b , vecteurs de \mathcal{E} , tels que : $\forall j \leq n, y_{1j} = a, z_{1j} = b$; soit : $M_j = \xi_{1j} a \otimes b + \sum_{i=2}^n \xi_{ij} y_{ij} \otimes z_{ij}$?

□ Imposons a . On peut toujours compléter $\{a\}$, de n façons différentes, par $n-1$ vecteurs $(y_{ij})_{i=2,n}$ pour former n bases de \mathcal{E} . Pour la matrice M_j et la base $\{a, y_{2j}, \dots, y_{nj}\}$, il vient, d'après le deuxième lemme fondamental : $\xi_{1j} \langle a, M_j^{-1} b \rangle = 1, \langle y_{ij}, M_j^{-1} b \rangle = 0, \langle a, M_j^{-1} z_{ij} \rangle = 0$ et $\langle y_{ij}, M_j^{-1} z_{ij} \rangle = \delta_{ij}$. C'est une condition nécessaire et suffisante d'existence de la décomposition $M_j = \xi_{1j} a \otimes b + \sum_{i=2}^n \xi_{ij} y_{ij} \otimes z_{ij}$.

□ Il est toujours possible de choisir un couple (a,b) et de calculer $n-1$ couples (y_{ij}, z_{ij}) vérifiant ces relations : choisissons b ; il vient : $\forall j, M_j^{-1} b \neq 0$, puisque M_j est inversible. Il suffit alors de choisir a comme vecteur n'appartenant pas à la réunion des n hyperplans supplémentaires orthogonaux des $M_j^{-1} b$. Ce qui est toujours possible, car la réunion de n hyperplan est négligeable dans \mathcal{E} . On choisit pour la famille $(y_{ij})_i$ une base de l'hyperplan orthogonal à $M_j^{-1} b$, et les z_{ij} se calculent en décomposant M_j sur la base $\{a, y_{2j}, \dots, y_{nj}\}$.

□ Nous avons donc construit une famille de décompositions :

$$M_j = \xi_{1j} a \otimes b + \sum_{i=2}^n \xi_{ij} y_{ij} \otimes z_{ij}.$$

$$\text{D'où : } T = \sum_{j=1}^n e_j \otimes M_j$$

$$\begin{aligned}
&= \sum_{j=1}^n e_j \otimes (\xi_{1j} a \otimes b + \sum_{i=2}^n \xi_{1j} y_{1j} \otimes z_{1j}) \\
&= a \otimes b \otimes (\sum_{j=1}^n \xi_{1j} e_j) + \sum_{j=1}^n e_j \otimes (\sum_{i=2}^n \xi_{1j} y_{1j} \otimes z_{1j}) \\
&= a \otimes b \otimes u + \sum_{j=1}^n \sum_{i=2}^n \xi_{1j} e_j \otimes y_{1j} \otimes z_{1j}
\end{aligned}$$

qui est la somme de $1+n(n-1)$ tenseurs décomposés, soit n^2-n+1 .

■ Application

□ Si $n=2$, $n^2-n+1 = 3$, d'où : $r(2) \leq 3$. En fait, $r(2) = 3$.

□ Nous allons, sur un exemple, construire une décomposition d'un tenseur de $\mathcal{A}_3(\mathbb{R}^2)$ comme somme de trois tenseurs élémentaires.

□ Soit alors le tenseur :

$$\begin{aligned}
M = & \quad i \otimes i \otimes i + 2 \, i \otimes i \otimes j + 3 \, i \otimes j \otimes i + 4 \, i \otimes j \otimes j \\
& + 5 \, j \otimes i \otimes i + 6 \, j \otimes i \otimes j + 7 \, j \otimes j \otimes i + 8 \, j \otimes j \otimes j.
\end{aligned}$$

□ D'où :

$$\begin{cases} M_1 = M \cdot i = i \otimes i + 2 \, i \otimes j + 3 \, j \otimes i + 4 \, j \otimes j \\ M_2 = M \cdot j = 5 \, i \otimes i + 6 \, i \otimes j + 7 \, j \otimes i + 8 \, j \otimes j \end{cases}$$

soit :

$$M_1 = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \quad ; \quad M_2 = \begin{bmatrix} 5 & 7 \\ 6 & 8 \end{bmatrix}$$

et :

$$M_1^{-1} = -\frac{1}{2} \begin{bmatrix} 4 & -3 \\ -2 & 1 \end{bmatrix} \quad ; \quad M_2^{-1} = -\frac{1}{2} \begin{bmatrix} 8 & -7 \\ -6 & 5 \end{bmatrix}$$

□ Choisissons : $b = i + j = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$. D'où :

$$M_1^{-1} \cdot b = -\frac{1}{2} \begin{bmatrix} 4 & -3 \\ -2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} = -\frac{1}{2} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = -1/2 \, i + 1/2 \, j$$

et :

$$M_2^{-1} \cdot b = -\frac{1}{2} \begin{bmatrix} 8 & -7 \\ -6 & 5 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix} = -\frac{1}{2} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = -1/2 \, i + 1/2 \, j$$

□ Choisissons a tel que $\langle a, M_1^{-1}.b \rangle \neq 0$ et $\langle a, M_2^{-1}.b \rangle \neq 0$, soit non orthogonal à $i-j$. Prenons $a = i$. D'où : $\langle a, M_1^{-1}.b \rangle = -1/2$, $\langle a, M_2^{-1}.b \rangle = -1/2$ et

$$a \otimes b = i \otimes i + i \otimes j = \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix}.$$

□ D'où :

$$\begin{cases} M_1 = \lambda_1 a \otimes b + x_1 \otimes y_1 \\ M_2 = \lambda_2 a \otimes b + x_2 \otimes y_2 \end{cases}$$

D'où : $\lambda_1 = 1/\langle a, M_1^{-1}.b \rangle = -2$ et $\lambda_2 = 1/\langle a, M_2^{-1}.b \rangle = -2$

$$\text{et : } M_1 - \lambda_1 a \otimes b = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} + 2 \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & 3 \\ 4 & 4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 3 \\ 4 \end{bmatrix}$$

$$\text{et : } M_2 - \lambda_2 a \otimes b = \begin{bmatrix} 5 & 7 \\ 6 & 8 \end{bmatrix} + 2 \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 7 & 7 \\ 8 & 8 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 7 \\ 8 \end{bmatrix}$$

□ Soit, en résumé :

$$M_1 = \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} = -2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 3 \\ 4 \end{bmatrix}$$

$$M_2 = \begin{bmatrix} 5 & 7 \\ 6 & 8 \end{bmatrix} = -2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 7 \\ 8 \end{bmatrix}$$

D'où :

$$M = i \otimes M_1 + j \otimes M_2$$

$$= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 5 & 7 \\ 6 & 8 \end{bmatrix}$$

$$= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \left(-2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 3 \\ 4 \end{bmatrix} \right) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \left(-2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 7 \\ 8 \end{bmatrix} \right)$$

$$= -2 \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 3 \\ 4 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 7 \\ 8 \end{bmatrix}$$

qui est bien une décomposition de la forme cherchée.

■ Deuxième démonstration

□ Supposons M matrice de plein rang, et écrivons : $M = \sum_1 e_i \otimes m_i$, c.a.d. que M est la matrice constituée des vecteurs colonnes (m_i) .

□ Soit (a, b) un couple de vecteurs. On cherche s'il existe un réel $\lambda \in \mathbb{R}$ tel que $M = \lambda a \otimes b + N$, où $\text{rang}(N) = n-1$, c.a.d. tel que $\text{Det}(N) = \text{Det}(M - \lambda a \otimes b) = 0$.

□ On en déduit :

$M = \sum_1 e_i \otimes m_i$; $\lambda a \otimes b = \sum_1 e_i \otimes \lambda \alpha_i b$, où a est le vecteur de coordonnées (α_i) .
D'où $N = M - \lambda a \otimes b$ est la matrice $\sum_1 e_i \otimes (m_i - \lambda \alpha_i b)$, donc de colonnes $(m_i - \lambda \alpha_i b)$.

□ D'où :

$$\begin{aligned} \text{Det}(N) &= \text{Det}(m_1 - \lambda \alpha_1 b, \dots, m_n - \lambda \alpha_n b) \\ &= \text{Det}(m_1, \dots, m_n) - \lambda \sum_1 \alpha_i \text{Det}(m_1, \dots, m_{i-1}, b, m_{i+1}, \dots, m_n) \end{aligned}$$

car tous les autres termes du développement du déterminant sont nuls car ils contiennent au moins deux fois le vecteur b .

□ Notons $\Delta_i(b) = \text{Det}(m_1, \dots, m_{i-1}, b, m_{i+1}, \dots, m_n)$. Il existe un vecteur b tel que tous les $\Delta_i(b)$ soient non tous nuls (par exemple $b = m_1$, puisque $\text{Det}(M) \neq 0$). b étant ainsi choisi, il existe alors un vecteur a tel que $\sum_1 \alpha_i \Delta_i \neq 0$. Et, en choisissant $\lambda_0 = \frac{\text{Det}(M)}{\sum_1 \alpha_i \Delta_i}$, on obtient $\text{Det}(M - \lambda_0 a \otimes b) = 0$.

4. TROISIEME MAJORANT DU RMT

■ Remarque liminaire

Ce troisième majorant n'est pas explicité sous forme d'une formule fonction de n , mais présenté sous forme d'un algorithme de décomposition d'un tenseur comme somme de r termes, d'où $r(n) \leq r$. Une évaluation numérique de $r(n)$, de l'ordre de $2/3n^2$, est fournie pour les premières valeurs de n .

■ Enoncé

Notons : $\beta(k) = \left\lceil \frac{n}{k-1} \right\rceil$ et $\alpha(k) = \lfloor n/\beta(k) \rfloor^+$

Il vient alors :

$$r(n) \leq \sum_{k=0}^{n-1} \alpha(k)$$

■ Algorithme de décomposition

□ Remarques préliminaires

□ Soit r un majorant de $r(n)$. D'après le premier lemme fondamental, il est équivalent d'affirmer que toute famille de n matrices de $\mathcal{A}_2(\mathcal{E})$, où $\dim(\mathcal{E}) = n$, est générée par combinaison linéaire de r matrices de rang 1 $y_1 \otimes z_1$. C'est ce dernier problème que nous allons étudier.

□ Position du problème

□ D'après le deuxième majorant de $r(n)$, toute famille de n matrices inversibles M_j peut être décomposée de la façon suivante :

$$\begin{cases} M_1 = \lambda_1 a \otimes b + N_1 \\ \dots \\ M_j = \lambda_j a \otimes b + N_j \\ \dots \\ M_n = \lambda_n a \otimes b + N_n \end{cases}$$

Où $\text{rang}(N_j) = n-1$.

Nous allons chercher, dans les décompositions $N_j = \sum_{i=2}^n \xi_{ij} y_{ij} \otimes z_{ij}$, à identifier des couples (y_{ij}, z_{ij}) afin de diminuer le nombre de matrices de rang 1 qui génèrent la famille M_j .

□ Nous avons donc : $M_j = \xi_{1j} a \otimes b + \sum_{i=2}^n \xi_{ij} y_{ij} \otimes z_{ij}$, soit, de façon explicite :

$$\left\{ \begin{array}{l} M_1 = \xi_{11} a \otimes b + \xi_{21} y_{21} \otimes z_{21} + \dots + \xi_{n1} y_{n1} \otimes z_{n1} \\ M_2 = \xi_{12} a \otimes b + \xi_{22} y_{22} \otimes z_{22} + \dots + \xi_{n2} y_{n2} \otimes z_{n2} \\ \dots \\ M_n = \xi_{1n} a \otimes b + \xi_{2n} y_{2n} \otimes z_{2n} + \dots + \xi_{nn} y_{nn} \otimes z_{nn} \end{array} \right.$$

□ Dans quelle mesure peut-on choisir des couples (y_{ij}, z_{ij}) semblables pour des valeurs semblables de i et différentes de j , c'est à dire égaux sur une même colonne du tableau précédent ? Dans quelle mesure peut-on poursuivre le procédé simultanément sur plusieurs colonnes de la décomposition ?

□ *Solution pour la deuxième colonne*

□ Utilisons le troisième lemme fondamental. Les n espaces $\mathcal{F}_j = \text{Im}(N_j)$ sont n hyperplans de \mathcal{E} de dimension $n-1$. Les $n-1$ premiers, par exemple, ont ainsi un vecteur u non nul en commun. De même, les $n-1$ hyperplans $\text{Im}(N'_j)$ auront un vecteur non nul v en commun. On peut donc construire une base $\{u, y_{3j}, \dots, y_{nj}\}$ de $\text{Im}(N'_j)$ et une base $\{v, z_{3j}, \dots, z_{nj}\}$ de $\text{Im}(N_j)$ telles que N_j s'écrive : $N_j = \xi_{2j} u \otimes v + \sum_{i=3}^n \xi_{ij} y_{ij} \otimes z_{ij}$.

□ D'après le troisième lemme fondamental, pour qu'une telle décomposition existe, il est nécessaire que : $\forall j, \langle u, \hat{N}_j^{-1} v \rangle \neq 0$. Or, u et v sont déterminés par : $u \in \bigcap_j (\text{Im}(N'_j))$ et $v \in \bigcap_j (\text{Im}(N_j))$. Il faut donc vérifier, dans l'algorithme de construction, que les vecteurs u et v ainsi déterminés satisfont à $\langle u, \hat{N}_j^{-1} v \rangle \neq 0$. De façon plus générale, il est nécessaire de vérifier que la décomposition obtenue satisfait au point 3) du troisième lemme fondamental, afin d'assurer son existence. Nous n'avons pas de preuve générale de ce résultat, et cette vérification doit être faite numériquement décomposition par décomposition.

□ *Solution pour plusieurs colonnes simultanément*

□ Le procédé peut être poursuivi colonne par colonne : supposons que l'on ait ainsi identifié par paquets les k premières colonnes du tableau [1]. Ce tableau peut s'écrire :

$$\begin{cases} M_1 = M_{1k} + N_{1k} \\ \dots \\ M_j = M_{jk} + N_{jk} \\ \dots \\ M_n = M_{nk} + N_{nk} \end{cases}$$

avec $\text{rang}(M_{jk}) = k$ et $\text{rang}(N_{jk}) = n-k$.

□ On peut écrire :

$$\begin{cases} N_{1k} = \sum_{i=k+1}^n \xi_{i1} y_{i1} \otimes z_{i1} \\ N_{2k} = \sum_{i=k+1}^n \xi_{i2} y_{i2} \otimes z_{i2} \\ \dots \\ N_{nk} = \sum_{i=k+1}^n \xi_{in} y_{in} \otimes z_{in} \end{cases}$$

et l'on cherche à identifier par paquets les couples de vecteurs $(y_{ij} \otimes z_{ij})_{j=1,n}$ de la première colonne du tableau précédent.

□ Utilisons le troisième lemme fondamental. Les vecteurs $(y_{ij})_i$ et $(z_{ij})_i$ appartiennent à $\text{Im}(N'_{jk})$ et $\text{Im}(N_{jk})$ respectivement. Ces espaces sont de dimension $n-k$. Pour pouvoir identifier les couples $(y_{k+1,j}, z_{k+1,j})$ pour j variable, il est nécessaire que les espaces $\text{Im}(N'_{jk})$ et $\text{Im}(N_{jk})$ respectivement aient une intersection non réduite au vecteur nul. Soient $\mathcal{F}_j = \text{Im}(N_{jk})$ et $\mathcal{G}_j = \text{Im}(N'_{jk})$. Choisissons q espaces \mathcal{F}_j et \mathcal{G}_j . Il est nécessaire que $\text{Dim}(\bigcap_{k=1}^q \mathcal{F}_k) \geq 1$, soit, puisque $\text{Dim}(\mathcal{F}_k) = n-k$, que $q(n-k) - n(q-1) \geq 1$, soit $n-kq \geq 1$, soit $q \leq \frac{n-1}{k}$. Puisque q est entier,

d'où : $q = \left\lfloor \frac{n-1}{k} \right\rfloor$. Et de même pour $\text{Dim}(\bigcap_{k=1}^q \mathcal{G}_k) \geq 1$. Choisissons donc $u_k \in \bigcap_{k=1}^q \mathcal{F}_k$ et $v_k \in \bigcap_{k=1}^q \mathcal{G}_k$.

□ D'après le troisième lemme fondamental, il est nécessaire que $\forall j \leq q, \langle u_k, \hat{N}_{jk}^{-1} v_k \rangle \neq 0$. Nous n'avons pas de démonstration générale que cette inégalité est respectée, et la vérification doit être faite décomposition par décomposition.

□ *Dénombrement des couples $(y_{k+1,j}, z_{k+1,j})$ distincts*

□ Posons $\beta(k) = \left\lfloor \frac{n-1}{k} \right\rfloor$. $\beta(k)$ est le nombre maximal de couples $(y_{k+1,j}, z_{k+1,j})$ distincts que l'on peut identifier. La $(k+1)$ -ième colonne de [1] comprendra donc au maximum $\alpha(k) = \left\lceil \frac{n}{\beta(k)} \right\rceil^+$ où $[\xi]^+$ désigne le plus petit nombre entier supérieur ou égal à ξ . Posons, par convention, $\alpha(0) = 1$. Il vient alors naturellement : $r(n) \leq \sum_{k=0}^{n-1} \alpha(k)$.

□ *Exemple numérique pour $n = 9$*

Nous allons calculer les différentes valeurs de $\alpha(k)$ et $\beta(k)$. Il vient :

□ $\beta(1)=[8/1]=8$; $\beta(2)=[8/2]=4$; $\beta(3)=[8/3]=2$; $\beta(4)=[8/4]=2$;

$\beta(5)=\beta(6)=\beta(7)=\beta(8)=1$

□ $\alpha(1)=[9/8]^+=2$; $\alpha(2)=[9/4]^+=3$; $\alpha(3)=[9/2]^+=5$; $\alpha(4)=[9/2]^+=5$;

$\alpha(5)=\alpha(6)=\alpha(7)=\alpha(8)=9$

□ $r(9) \leq 1+2+3+5+5+9+9+9+9 = 52$: $r(9) \leq 52$.

k	$\beta(k)$	$\alpha(k)$
0		1
1	8	2
2	4	3
3	2	5
4	2	5
5	1	9
6	1	9
7	1	9
8	1	9

CHAPITRE G

CAS PARTICULIER DE $\mathcal{A}_3(\mathbb{R}^2)$

Dans le cas particulier de $\mathcal{A}_3(\mathbb{R}^2)$, les calculs peuvent être menés jusqu'au bout.

1. NOTATIONS

1 - \mathbb{R}^2 est muni d'une BON (i, j) .

2 - Le produit tensoriel de vecteurs : $x \otimes y$ est parfois noté xy lorsqu'il n'y a aucune confusion possible, pour alléger l'écriture.

3 - Tout tenseur de $\mathcal{A}_3(\mathbb{R}^2)$ s'écrit donc de manière unique :

$$A = \alpha_{111} i \otimes i \otimes i + \alpha_{112} i \otimes i \otimes j + \alpha_{121} i \otimes j \otimes i + \alpha_{122} i \otimes j \otimes j \\ + \alpha_{211} j \otimes i \otimes i + \alpha_{212} j \otimes i \otimes j + \alpha_{221} j \otimes j \otimes i + \alpha_{222} j \otimes j \otimes j$$

noté pour simplifier :

$$A = \alpha_{111} iii + \alpha_{112} iij + \alpha_{121} iji + \alpha_{122} ijj \\ + \alpha_{211} jii + \alpha_{212} jij + \alpha_{221} jji + \alpha_{222} jjj$$

4 - Rappelons que l'on note par $r(A)$ le rang de A .

2. RANG D'UN TENSEUR

Les résultats suivants concernant le rang d'un tenseur de $\mathcal{A}_3(\mathbb{R}^2)$ ont notamment été énoncés dans KRUSKALL, 1986, mais l'auteur n'a pas publié ses démonstrations. Nous en proposons quelques unes, directes.

1. PREMIER LEMME

1.1 Enoncé

Etant donné un tenseur A de $\mathcal{A}_3(\mathbb{R}^2)$ de rang $r(A)$, on a toujours la majoration suivante :

$$r(A) \leq 3$$

1.2 Première démonstration

1 - Il suffit de faire $n = 2$ dans le deuxième majorant du rang d'un tenseur général (III-1.5)

1.3 Deuxième démonstration

Cette deuxième démonstration est directe.

1 - Soit A de coefficient général (α_{ijk}) avec $i, j, k \in \{1, 2\}$.

2 - On peut donc écrire :

$$\begin{aligned} A &= \alpha_{111}iii + \alpha_{112}iij + \alpha_{121}iji + \alpha_{122}ijj \\ &\quad + \alpha_{211}jii + \alpha_{212}ji j + \alpha_{221}jji + \alpha_{222}jjj \\ &= ii(\alpha_{111}i + \alpha_{112}j) + ij(\alpha_{121}i + \alpha_{122}j) \\ &\quad + ji(\alpha_{211}i + \alpha_{212}j) + jj(\alpha_{221}i + \alpha_{222}j) \\ &= iiu_{11} + iju_{12} + jiu_{21} + jj u_{22} \end{aligned}$$

3 - Or, $u_{ij} \in \mathbb{R}^2$. Supposons que u_{11} et u_{22} forme un système libre. u_{12} et u_{21} sont alors combinaison linéaire de u_{11} et u_{22} , soit :

$$\begin{aligned} u_{12} &= \xi_1 u_{11} + \eta_1 u_{22} \\ u_{21} &= \xi_2 u_{11} + \eta_2 u_{22} \end{aligned}$$

4 - D'où :

$$\begin{aligned}
 A &= i i u_{11} + i j u_{12} + j i u_{21} + j j u_{22} \\
 &= i i u_{11} + i j (\xi_1 u_{11} + \eta_1 u_{22}) + j i (\xi_2 u_{11} + \eta_2 u_{22}) + j j u_{22} \\
 &= i i u_{11} + \xi_1 i j u_{11} + \eta_1 i j u_{22} + \xi_2 j i u_{11} + \eta_2 j i u_{22} + j j u_{22} \\
 &= (i + \xi_2 j) i u_{11} + i j (\xi_1 u_{11} + \eta_1 u_{22}) + j (\eta_2 i + j) u_{22} \\
 &= x i u_{11} + i j u_{12} + j y u_{22}
 \end{aligned}$$

qui est bien une décomposition en somme de trois tenseurs de rang 1.

5 - Une autre décomposition possible est :

$$\begin{aligned}
 A &= i i u_{11} + \xi_1 i j u_{11} + \eta_1 i j u_{22} + \xi_2 j i u_{11} + \eta_2 j i u_{22} + j j u_{22} \\
 &= i (i + \xi_1 j) u_{11} + (\eta_1 i + j) j u_{22} + j i (\xi_2 u_{11} + \eta_2 u_{22}) \\
 &= i x' u_{11} + y' j u_{22} + j i u_{21}
 \end{aligned}$$

6 - Remarquons que ces décompositions sont loin d'être uniques : on choisit arbitrairement i , donc j au signe près, et x , y , u_{11} , u_{22} et u_{12} en découlent.

2. DEUXIEME LEMME

1 - Notons S la sphère unité de $\mathcal{A}_3(\mathbb{R}^2)$. C'est l'ensemble des tenseurs A de $\mathcal{A}_3(\mathbb{R}^2)$ tels que $\|A\| = 1$.

2 - Notons SR_i , pour i variant de 1 à 3, le sous ensemble de S composé des tenseurs de rang i . Alors :

$$S = SR_1 \cup SR_2 \cup SR_3$$

3 - S est un clairement un sous-espace mesurable de \mathbb{R}^8 , muni de la mesure habituelle : si Ω est une partie mesurable de S :

$$\mu(\Omega) = \int \dots \int_{\Omega} d\xi_1 \dots d\xi_8$$

2.1 Enoncé

Pour la mesure μ dans S :

- SR_1 est une partie négligeable de S
- SR_2 et SR_3 sont de mesure non nulle.

2.2 Démonstrations

□ SR_1 est une partie négligeable de S

En effet, SR_1 est composée des tenseurs de la forme $\lambda x \otimes y \otimes z$, où :

$$\|x\| = \|y\| = \|z\| = 1.$$

C'est donc une sous variété de dimension 4 de $\mathcal{A}_3(\mathbb{R}^2)$, qui est de dimension 8. C'en est donc une partie négligeable,

□ SR_2 et SR_3 sont de mesure non nulle.

1 - Chacun de ces deux endembles est une sous variété de dimension 8.

2 - Soit T un tenseur de $\mathcal{A}_3(\mathbb{R}^2)$, écrit sous la forme : $T = i \otimes M + j \otimes N$, où (i, j) est une BON de \mathbb{R}^2 , et $M, N \in \mathcal{A}_2(\mathbb{R}^2)$. On peut supposer que l'une au moins des deux matrices M ou N est inversible.

3 - Alors, T est de rang 2 si et seulement si les valeurs propres de $M^{-1}N$ sont toutes deux réelles. Ce résultat, sous une forme plus générale, a été montré par LAFON (1975), et retrouvé par plusieurs auteurs, notamment KRUSKAL pour le cas $\mathcal{A}_3(\mathbb{R}^2)$ et ten BERGE pour le cas général $\mathbb{R}^2 \times \mathbb{R}^n \times \mathbb{R}^n$. Une démonstration directe est facile :

3.1 Soient donc (u, v) les vecteurs propres de $M^{-1}N$:

$$M^{-1}N.u = \lambda u \iff N.u = \lambda M.u$$

$$M^{-1}N.v = \mu v \Leftrightarrow N.v = \mu M.v$$

3.2 Or, si (x,y) est une BON de \mathbb{R}^2 ,

$$M = x \otimes M.x^* + y \otimes M.y^*$$

$$N = x \otimes N.x^* + y \otimes N.y^*$$

3.3 Donc, en choisissant pour (u,v) la base duale des vecteurs propres de $M^{-1}N$, nous avons :

$$M = u^* \otimes M.u + v^* \otimes M.v$$

$$N = u^* \otimes N.u + v^* \otimes N.v$$

et :

$$T = i \otimes M + j \otimes N = i \otimes (u^* \otimes M.u + v^* \otimes M.v) + j \otimes (u^* \otimes N.u + v^* \otimes N.v)$$

soit, si $a = M.u$ et $b = M.v$:

$$T = i \otimes (u^* \otimes a + v^* \otimes b) + j \otimes (u^* \otimes \lambda a + v^* \otimes \mu b) = (i + \lambda j) \otimes u^* \otimes a + (i + \mu j) \otimes v^* \otimes b$$

et est donc de rang 2.

La réciproque est immédiate.

4 - $M^{-1}N$ est une matrice carrée réelle d'ordre 2. Ses valeurs propres sont solutions d'un polynôme du second degré : $\text{Det}(M^{-1}N - \lambda I) = 0$. Elles sont donc réelles si, et seulement si le discriminant de ce polynôme est > 0 .

$$5 - \text{Soit } M = \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{bmatrix} \text{ et } N = \begin{bmatrix} \nu_{11} & \nu_{12} \\ \nu_{21} & \nu_{22} \end{bmatrix}$$

alors :

$$M^{-1} = \frac{1}{\mu_{11}\mu_{22} - \mu_{12}\mu_{21}} \begin{bmatrix} \mu_{22} & -\mu_{12} \\ -\mu_{21} & \mu_{11} \end{bmatrix}$$

$$\text{et } M^{-1}N = \frac{1}{\mu_{11}\mu_{22} - \mu_{12}\mu_{21}} \begin{bmatrix} \mu_{22}\nu_{11} - \mu_{12}\nu_{21} & \mu_{22}\nu_{12} - \mu_{12}\nu_{22} \\ -\mu_{21}\nu_{11} + \mu_{11}\nu_{21} & -\mu_{21}\nu_{12} + \mu_{11}\nu_{22} \end{bmatrix}$$

$$6 - \text{Si } A = \begin{bmatrix} \alpha & \beta \\ \gamma & \delta \end{bmatrix}, \text{ Det}(A - \lambda I) = (\alpha - \lambda)(\delta - \lambda) - \beta\gamma = \lambda^2 - \lambda(\alpha + \delta) + \alpha\delta - \beta\gamma$$

D'où le discriminant de $\text{Det}(A-\lambda I)$: $\Delta = (\alpha+\beta)^2 - 4(\alpha\delta - \beta\gamma)$

D'où, si :

$$\alpha = \mu_{22} \nu_{11} - \mu_{21} \nu_{12}$$

$$\beta = \mu_{22} \nu_{21} - \mu_{21} \nu_{22}$$

$$\gamma = \mu_{12} \nu_{11} - \mu_{11} \nu_{12}$$

$$\delta = \mu_{12} \nu_{21} - \mu_{11} \nu_{22}$$

$$\Delta = (\mu_{22} \nu_{11} - \mu_{21} \nu_{12} + \mu_{22} \nu_{21} - \mu_{21} \nu_{22})^2 - 4 [(\mu_{22} \nu_{11} - \mu_{21} \nu_{12})(\mu_{12} \nu_{21} - \mu_{11} \nu_{22}) - (\mu_{22} \nu_{21} - \mu_{21} \nu_{22})(\mu_{12} \nu_{11} - \mu_{11} \nu_{12})]$$

7 - Donc, si $\Delta > 0$, T est de rang 2, et $T \in \text{SR}_2$

et si $\Delta < 0$, T est de rang 3, et $T \in \text{SR}_3$.

Δ étant un polynôme du quatrième degré en les coefficients de T. Donc, les mesures de SR_2 et SR_3 sont toutes deux non nulles.

3. APPROXIMATION D'UN TENSEUR

1. PREMIER CALCUL DE LA PREMIERE COMPOSANTE PRINCIPALE

1 - Rappelons que la première composante principale $\lambda x \otimes y \otimes z$ d'un tenseur A est solution de :

$$\langle A, x \otimes y \otimes z \rangle \text{ maximum}$$

$$\text{sous } \|x\| = \|y\| = \|z\| = 1$$

2 - Dans le cas particulier de $\mathcal{A}_3(\mathbb{R}^2)$, la recherche des solutions de (1) peut être ramenée à la recherche des solutions d'une équation du 8^{ème} degré. Cependant, le calcul, présenté dans la suite du paragraphe, est fastidieux, bien que mécanique. Il est irréaliste d'envisager une extension raisonnable de cette méthode pour des dimensions de $\mathcal{E} \geq 3$, sans l'aide d'un logiciel de calcul formel puissant.

3 - Puisque x , y et z sont des vecteurs normés, posons :

$$x = i \cos\theta + j \sin\theta$$

$$y = i \cos\varphi + j \sin\varphi$$

$$z = i \cos\psi + j \sin\psi$$

4 - On en déduit :

$$\begin{aligned} \langle A, x \otimes y \otimes z \rangle &= \alpha_{111} \cos\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{112} \cos\theta \cdot \cos\varphi \cdot \sin\psi \\ &+ \alpha_{121} \cos\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{122} \cos\theta \cdot \sin\varphi \cdot \sin\psi \\ &+ \alpha_{211} \sin\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{212} \sin\theta \cdot \cos\varphi \cdot \sin\psi \\ &+ \alpha_{221} \sin\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{222} \sin\theta \cdot \sin\varphi \cdot \sin\psi \end{aligned}$$

5 - Aux extremums, on a donc la condition :

$$\frac{\partial}{\partial\theta} \langle A, x \otimes y \otimes z \rangle = \frac{\partial}{\partial\varphi} \langle A, x \otimes y \otimes z \rangle = \frac{\partial}{\partial\psi} \langle A, x \otimes y \otimes z \rangle = 0$$

6 - Ainsi :

$$\begin{aligned} \frac{\partial}{\partial\theta} \langle A, x \otimes y \otimes z \rangle &= \frac{\partial}{\partial\theta} \left[\alpha_{111} \cos\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{112} \cos\theta \cdot \cos\varphi \cdot \sin\psi \right. \\ &+ \alpha_{121} \cos\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{122} \cos\theta \cdot \sin\varphi \cdot \sin\psi \\ &+ \alpha_{211} \sin\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{212} \sin\theta \cdot \cos\varphi \cdot \sin\psi \\ &+ \left. \alpha_{221} \sin\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{222} \sin\theta \cdot \sin\varphi \cdot \sin\psi \right] \\ &= -\alpha_{111} \sin\theta \cdot \cos\varphi \cdot \cos\psi - \alpha_{112} \sin\theta \cdot \cos\varphi \cdot \sin\psi \\ &- \alpha_{121} \sin\theta \cdot \sin\varphi \cdot \cos\psi - \alpha_{122} \sin\theta \cdot \sin\varphi \cdot \sin\psi \\ &+ \alpha_{211} \cos\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{212} \cos\theta \cdot \cos\varphi \cdot \sin\psi \\ &+ \alpha_{221} \cos\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{222} \cos\theta \cdot \sin\varphi \cdot \sin\psi \\ \frac{\partial}{\partial\varphi} \langle A, x \otimes y \otimes z \rangle &= \frac{\partial}{\partial\varphi} \left[\alpha_{111} \cos\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{112} \cos\theta \cdot \cos\varphi \cdot \sin\psi \right. \\ &+ \alpha_{121} \cos\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{122} \cos\theta \cdot \sin\varphi \cdot \sin\psi \\ &+ \left. \alpha_{211} \sin\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{212} \sin\theta \cdot \cos\varphi \cdot \sin\psi \right] \end{aligned}$$

$$\begin{aligned}
& + \alpha_{221} \sin\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{222} \sin\theta \cdot \sin\varphi \cdot \sin\psi \Big] \\
= & - \alpha_{111} \cos\theta \cdot \sin\varphi \cdot \cos\psi - \alpha_{112} \cos\theta \cdot \sin\varphi \cdot \sin\psi \\
& + \alpha_{121} \cos\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{122} \cos\theta \cdot \cos\varphi \cdot \sin\psi \\
& - \alpha_{211} \sin\theta \cdot \sin\varphi \cdot \cos\psi - \alpha_{212} \sin\theta \cdot \sin\varphi \cdot \sin\psi \\
& + \alpha_{221} \sin\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{222} \sin\theta \cdot \cos\varphi \cdot \sin\psi
\end{aligned}$$

$$\begin{aligned}
\frac{\partial}{\partial \psi} \langle A, x \otimes y \otimes z \rangle = \frac{\partial}{\partial \psi} \Big[& \alpha_{111} \cos\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{112} \cos\theta \cdot \cos\varphi \cdot \sin\psi \\
& + \alpha_{121} \cos\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{122} \cos\theta \cdot \sin\varphi \cdot \sin\psi \\
& + \alpha_{211} \sin\theta \cdot \cos\varphi \cdot \cos\psi + \alpha_{212} \sin\theta \cdot \cos\varphi \cdot \sin\psi \\
& + \alpha_{221} \sin\theta \cdot \sin\varphi \cdot \cos\psi + \alpha_{222} \sin\theta \cdot \sin\varphi \cdot \sin\psi \Big] \\
& - \alpha_{111} \cos\theta \cdot \cos\varphi \cdot \sin\psi + \alpha_{112} \cos\theta \cdot \cos\varphi \cdot \cos\psi \\
& - \alpha_{121} \cos\theta \cdot \sin\varphi \cdot \sin\psi + \alpha_{122} \cos\theta \cdot \sin\varphi \cdot \cos\psi \\
& - \alpha_{211} \sin\theta \cdot \cos\varphi \cdot \sin\psi + \alpha_{212} \sin\theta \cdot \cos\varphi \cdot \cos\psi \\
& - \alpha_{221} \sin\theta \cdot \sin\varphi \cdot \sin\psi + \alpha_{222} \sin\theta \cdot \sin\varphi \cdot \cos\psi
\end{aligned}$$

5 - Supposons que $\cos\theta \cdot \cos\varphi \cdot \cos\psi$ soit non nul (si tel était le cas, on pourrait toujours s'y ramener par une rotation en θ , φ ou ψ selon le cas). Posons alors :

$$\tau = \operatorname{tg}\theta ; \vartheta = \operatorname{tg}\varphi ; v = \operatorname{tg}\psi.$$

6 - Les équations du 4 deviennent alors, par division par $\cos\theta \cdot \cos\varphi \cdot \cos\psi$:

$$\begin{aligned}
- \alpha_{111} \tau - \alpha_{112} \tau v - \alpha_{121} \tau \vartheta - \alpha_{122} \tau \vartheta v + \alpha_{211} + \alpha_{212} v + \alpha_{221} \vartheta + \alpha_{222} \vartheta v &= 0 \\
- \alpha_{111} \vartheta - \alpha_{112} \vartheta v + \alpha_{121} + \alpha_{122} v - \alpha_{211} \tau \vartheta - \alpha_{212} \tau \vartheta v + \alpha_{221} \tau + \alpha_{222} \tau v &= 0 \\
- \alpha_{111} v + \alpha_{112} - \alpha_{121} \vartheta v + \alpha_{122} \vartheta - \alpha_{211} \tau v + \alpha_{212} \tau - \alpha_{221} \tau \vartheta v + \alpha_{222} \tau \vartheta &= 0
\end{aligned}$$

Soit, par simple réarrangement :

$$\alpha_{211} - \alpha_{111}\tau + \alpha_{221}\vartheta + \alpha_{212}v - \alpha_{121}\tau\vartheta - \alpha_{112}\tau v + \alpha_{222}\vartheta v - \alpha_{122}\tau\vartheta v = 0$$

$$\alpha_{121} + \alpha_{221}\tau - \alpha_{111}\vartheta + \alpha_{122}v - \alpha_{211}\tau\vartheta + \alpha_{222}\tau v - \alpha_{112}\vartheta v - \alpha_{212}\tau\vartheta v = 0$$

$$\alpha_{112} + \alpha_{212}\tau + \alpha_{122}\vartheta - \alpha_{111}v + \alpha_{222}\tau\vartheta - \alpha_{211}\tau v - \alpha_{121}\vartheta v - \alpha_{221}\tau\vartheta v = 0$$

qui est un système de trois équations du troisième degré en τ , ϑ , v .

7 - Pour le résoudre, nous pouvons par exemple exprimer v comme fonction de τ et ϑ selon la troisième équation, et reporter cette égalité dans les deux premières.

8 - En extrayant v de la troisième équation, on obtient :

$$\alpha_{112} + \alpha_{212}\tau + \alpha_{122}\vartheta - \alpha_{111}v + \alpha_{222}\tau\vartheta - \alpha_{211}\tau v - \alpha_{121}\vartheta v - \alpha_{221}\tau\vartheta v = 0$$

\Rightarrow

$$\begin{aligned}\alpha_{112} + \alpha_{212}\tau + \alpha_{122}\vartheta + \alpha_{222}\tau\vartheta &= \alpha_{111}v + \alpha_{211}\tau v + \alpha_{121}\vartheta v + \alpha_{221}\tau\vartheta v \\ &= v(\alpha_{111} + \alpha_{211}\tau + \alpha_{121}\vartheta + \alpha_{221}\tau\vartheta)\end{aligned}$$

$$\Rightarrow v = \frac{\alpha_{112} + \alpha_{212}\tau + \alpha_{122}\vartheta + \alpha_{222}\tau\vartheta}{\alpha_{111} + \alpha_{211}\tau + \alpha_{121}\vartheta + \alpha_{221}\tau\vartheta}$$

9 - Valeur que l'on peut reporter dans la première équation en multipliant les deux membres par $(\alpha_{111} + \alpha_{211}\tau + \alpha_{121}\vartheta + \alpha_{221}\tau\vartheta)$:

$$\begin{aligned}&\alpha_{211}(\alpha_{111} + \alpha_{211}\tau + \alpha_{121}\vartheta + \alpha_{221}\tau\vartheta) - \alpha_{111}\tau(\alpha_{111} + \alpha_{211}\tau + \alpha_{121}\vartheta + \alpha_{221}\tau\vartheta) \\ &+ \alpha_{221}\vartheta(\alpha_{111} + \alpha_{211}\tau + \alpha_{121}\vartheta + \alpha_{221}\tau\vartheta) + \alpha_{212}(\alpha_{112} + \alpha_{212}\tau + \alpha_{122}\vartheta + \alpha_{222}\tau\vartheta) \\ &- \alpha_{121}\tau\vartheta(\alpha_{111} + \alpha_{211}\tau + \alpha_{121}\vartheta + \alpha_{221}\tau\vartheta) - \alpha_{112}\tau(\alpha_{112} + \alpha_{212}\tau + \alpha_{122}\vartheta + \alpha_{222}\tau\vartheta) \\ &+ \alpha_{222}\vartheta(\alpha_{112} + \alpha_{212}\tau + \alpha_{122}\vartheta + \alpha_{222}\tau\vartheta) - \alpha_{122}\tau\vartheta(\alpha_{112} + \alpha_{212}\tau + \alpha_{122}\vartheta + \alpha_{222}\tau\vartheta)\end{aligned}$$

$$= 0$$

qui s'écrit sous la forme :

$$\beta_1 + \beta_2\tau + \beta_3\vartheta + \beta_4\tau^2 + \beta_5\tau\vartheta + \beta_6\vartheta^2 + \beta_7\tau^2\vartheta + \beta_8\tau\vartheta^2 + \beta_9\tau^2\vartheta^2$$

10 - De même, le report de v dans la deuxième équation mène à une équation qui peut s'écrire :

$$\gamma_1 + \gamma_2\tau + \gamma_3\vartheta + \gamma_4\tau^2 + \gamma_5\tau\vartheta + \gamma_6\vartheta^2 + \gamma_7\tau^2\vartheta + \gamma_8\tau\vartheta^2 + \gamma_9\tau^2\vartheta^2$$

11 - D'où le système de deux équation du quatrième degré en τ et ϑ :

$$\begin{cases} \beta_1 + \beta_2\tau + \beta_3\vartheta + \beta_4\tau^2 + \beta_5\tau\vartheta + \beta_6\vartheta^2 + \beta_7\tau^2\vartheta + \beta_8\tau\vartheta^2 + \beta_9\tau^2\vartheta^2 \\ \gamma_1 + \gamma_2\tau + \gamma_3\vartheta + \gamma_4\tau^2 + \gamma_5\tau\vartheta + \gamma_6\vartheta^2 + \gamma_7\tau^2\vartheta + \gamma_8\tau\vartheta^2 + \gamma_9\tau^2\vartheta^2 \end{cases}$$

qui peut également s'écrire, par réarrangement :

$$\begin{cases} \beta_1 + \beta_2\tau + \beta_4\tau^2 + \beta_3\vartheta + \beta_5\tau\vartheta + \beta_7\tau^2\vartheta + \beta_6\vartheta^2 + \beta_8\tau\vartheta^2 + \beta_9\tau^2\vartheta^2 \\ \gamma_1 + \gamma_2\tau + \gamma_4\tau^2 + \gamma_3\vartheta + \gamma_5\tau\vartheta + \gamma_7\tau^2\vartheta + \gamma_6\vartheta^2 + \gamma_8\tau\vartheta^2 + \gamma_9\tau^2\vartheta^2 \end{cases}$$

qui est un système de la forme :

$$\begin{cases} P_1(\tau) + P_2(\tau).\vartheta + P_3(\tau).\vartheta^2 = 0 \\ Q_1(\tau) + Q_2(\tau).\vartheta + Q_3(\tau).\vartheta^2 = 0 \end{cases}$$

où les P_i et Q_i sont des polynômes du second degré en τ :

$$\begin{aligned} P_1 &= \beta_1 + \beta_2\tau + \beta_4\tau^2 \\ P_2 &= \beta_3 + \beta_5\tau + \beta_7\tau^2 \\ P_3 &= \beta_6 + \beta_8\tau + \beta_9\tau^2 \\ Q_1 &= \gamma_1 + \gamma_2\tau + \gamma_4\tau^2 \\ Q_2 &= \gamma_3 + \gamma_5\tau + \gamma_7\tau^2 \\ Q_3 &= \gamma_6 + \gamma_8\tau + \gamma_9\tau^2 \end{aligned}$$

Les β_i et γ_j étant eux-mêmes des fonctions des α_{ijk} selon les 9 & 10.

12 - Ce système est un système de deux équations du second degré en ϑ , dont les coefficients sont des polynômes du second degré en τ . Rappelons qu'une condition nécessaire et suffisante pour que deux équations en ξ :

$$a \xi^2 + b \xi + c = 0$$

$$a' \xi^2 + b' \xi + c' = 0$$

aient une racine commune est que leur résultante soit nulle, soit :

$$\Delta = \begin{vmatrix} a & b & c & 0 \\ 0 & a & b & c \\ a' & b' & c' & 0 \\ 0 & a' & b' & c' \end{vmatrix} = 0$$

Expliciter cette condition dans le cas du système de deux équations en ϑ fournit un polynôme du 8^{ième} degré dont τ est racine. Pour le déroulement des calculs, il est très avantageux de conserver l'écriture en déterminants, et ne la développer qu'en dernier ressort pour obtenir l'écriture explicite du polynôme.

13 - En exprimant le déterminant, on obtient :

$$\Delta = \begin{vmatrix} P_3 & P_2 & P_1 & 0 \\ 0 & P_3 & P_2 & P_1 \\ Q_3 & Q_2 & Q_1 & 0 \\ 0 & Q_3 & Q_2 & Q_1 \end{vmatrix} = 0$$

15 - Soit, en remplaçant les P_i et Q_j par leur expression :

$$\Delta = \begin{vmatrix} \beta_6 + \beta_8\tau + \beta_9\tau^2 & \beta_3 + \beta_5\tau + \beta_7\tau^2 & \beta_1 + \beta_2\tau + \beta_4\tau^2 & 0 \\ 0 & \beta_6 + \beta_8\tau + \beta_9\tau^2 & \beta_3 + \beta_5\tau + \beta_7\tau^2 & \beta_1 + \beta_2\tau + \beta_4\tau^2 \\ \gamma_6 + \gamma_8\tau + \gamma_9\tau^2 & \gamma_3 + \gamma_5\tau + \gamma_7\tau^2 & \gamma_1 + \gamma_2\tau + \gamma_4\tau^2 & 0 \\ 0 & \gamma_6 + \gamma_8\tau + \gamma_9\tau^2 & \gamma_3 + \gamma_5\tau + \gamma_7\tau^2 & \gamma_1 + \gamma_2\tau + \gamma_4\tau^2 \end{vmatrix}$$

= 0.

déterminant que l'on peut développer comme polynôme en τ de degré 8. Ce développement, bien que purement mécanique (voir annexe 1), n'est pas présenté ici : sa simple écriture remplit plusieurs pages

2 - DEUXIEME CALCUL DE LA PREMIERE COMPOSANTE PRINCIPALE

1 - Une deuxième façon de calculer λ_{xoyz} , première composante principale de A est , où $\|x\| = \|y\| = \|z\| = 1$, est la suivante :

$$z = A.(xoy),$$

y est vecteur propre de $(A.x)'(A.x)$ associé à la plus grande VP

x est tel que cette plus grande valeur propre soit maximale

$$\lambda = \langle A, xoyz \rangle$$

Nous allons déterminer un tel x. Le calcul de y, z et λ en découleront ensuite facilement.

2 - Posons :

$$x = i.\cos\theta + j.\sin\theta$$

$$\text{et } A = i\otimes P + j\otimes Q.$$

D'où :

$$\begin{aligned} (A.x)'(A.x) &= (P.\cos\theta + Q.\sin\theta)'(P.\cos\theta + Q.\sin\theta) \\ &= P'P.\cos^2\theta + (P'Q + Q'P).\cos\theta.\sin\theta + Q'Q.\sin^2\theta. \end{aligned}$$

3 - On recherche donc λ , plus grande valeur propre de $(A.x)'(A.x)$, qui est

une fonction implicite de θ défini par :

$$\text{Det}(P'P \cdot \cos^2\theta + (P'Q + Q'P) \cdot \cos\theta \cdot \sin\theta + Q'Q \cdot \sin^2\theta - \lambda I) = 0.$$

4 - Posons :

$$P'P = i\otimes p_1 + j\otimes p_2 ; Q'Q = i\otimes q_1 + j\otimes q_2 ; P'Q + Q'P = i\otimes r_1 + j\otimes r_2.$$

L'équation du §3 s'écrit alors :

$$0 = \text{Det}(p_1 \cos^2\theta + q_1 \sin^2\theta + r_1 \cos\theta \sin\theta - \lambda i, p_2 \cos^2\theta + q_2 \sin^2\theta + r_2 \cos\theta \sin\theta - \lambda j)$$

$$= \lambda^2 \text{Det}(i, j)$$

$$- \lambda (\text{Det}(i, p_2 \cos^2\theta + q_2 \sin^2\theta + r_2 \cos\theta \sin\theta) - \text{Det}(p_1 \cos^2\theta + q_1 \sin^2\theta + r_1 \cos\theta \sin\theta, j))$$

$$+ \text{Det}(p_1 \cos^2\theta + q_1 \sin^2\theta + r_1 \cos\theta \sin\theta, p_2 \cos^2\theta + q_2 \sin^2\theta + r_2 \cos\theta \sin\theta)$$

qui est de la forme :

$$\lambda^2 + \lambda(\beta_1 \cos^2\theta + \beta_2 \cos\theta \sin\theta + \beta_3 \sin^2\theta) + \gamma_1 \cos^4\theta + \gamma_2 \cos^3\theta \sin\theta + \gamma_3 \cos^2\theta \sin^2\theta + \gamma_4 \cos\theta \sin^3\theta + \gamma_5 \sin^4\theta$$

avec :

$$\beta_1 = \text{Det}(p_1, j) + \text{Det}(p_2, i) = \text{Tr}(P'P) ;$$

$$\beta_2 = \text{Det}(r_1, j) + \text{Det}(r_2, i) = \text{Tr}(P'Q + Q'P) ;$$

$$\beta_3 = \text{Det}(q_1, j) + \text{Det}(q_2, i) = \text{Tr}(Q'Q) ;$$

$$\gamma_1 = \text{Det}(p_1, p_2) = \text{Det}(P'P) ;$$

$$\gamma_2 = \text{Det}(p_1, r_2) + \text{Det}(p_2, r_1) ;$$

$$\gamma_3 = \text{Det}(p_1, q_2) + \text{Det}(q_2, r_1) ;$$

$$\gamma_4 = \text{Det}(q_1, r_2) + \text{Det}(q_2, r_1) ;$$

$$\gamma_5 = \text{Det}(r_1, r_2) = \text{Det}(Q'Q) ;$$

que l'on peut donc calculer connaissant P et Q.

5 - La fonction du §4 est de la forme : $\lambda^2 + f(\theta)\lambda + g(\theta) = 0$.

On en déduit : $[2\lambda + f(\theta)]d\lambda + (\lambda f'(\theta) + g'(\theta))d\theta = 0$.

D'où, au maximum : $\lambda f'(\theta) + g'(\theta) = 0$.

Ainsi, θ est solution de l'équation :

$$\left(\frac{g'(\theta)}{f'(\theta)}\right)^2 - f(\theta) \cdot \left(\frac{g'(\theta)}{f'(\theta)}\right) + g(\theta) = 0$$

soit :

$$g'^2 - ff'g' + f'^2g = 0$$

qui est un polynôme trigonométrique en θ .

6 - En remplaçant dans (5) les valeurs de f, f', g, g' , on obtient :

$$f(\theta) = \beta_1 \cos^2 \theta + \beta_2 \cos \theta \sin \theta + \beta_3 \sin^2 \theta$$

$$\Rightarrow f'(\theta) = \beta_2 \cos^2 \theta + 2(\beta_3 - \beta_1) \sin \theta \cos \theta - \beta_2 \sin^2 \theta$$

$$g(\theta) = \gamma_1 \cos^4 \theta + \gamma_2 \cos^3 \theta \sin \theta + \gamma_3 \cos^2 \theta \sin^2 \theta + \gamma_4 \cos \theta \sin^3 \theta + \gamma_5 \sin^4 \theta$$

$$\begin{aligned} \Rightarrow g'(\theta) = & \gamma_2 \cos^4 \theta - 2(\gamma_3 - 2\gamma_1) \cos^3 \theta \sin \theta + 3(\gamma_4 - \gamma_2) \cos^2 \theta \sin^2 \theta \\ & + 2(2\gamma_5 - \gamma_3) \cos \theta \sin^3 \theta - \gamma_4 \sin^4 \theta \end{aligned}$$

7 - Comme f et f' sont un polynôme trigonométrique homogène de degré 2 en θ , et g et g' un polynôme trigonométrique homogène de degré 4 en θ , $g'^2 - ff'g' + f'^2g$ est un polynôme trigonométrique homogène de degré 8 en θ . θ est donc solution d'un polynôme trigonométrique homogène de degré 8, dont les coefficients peuvent être explicités en fonction du §6.

3. APPLICATION NUMERIQUE

Nous proposons dans ce qui suit une application numérique qui suit la démarche du §3.2, dont nous utiliserons la notation. Les calculs élémentaires sont omis, seuls les résultats sont fournis.

1 - Choix du tenseur initial

Les coefficients de ce tenseur sont choisis comme aléatoire à partir des premières décimales de π comme suit :

$$\pi = 3,1 \ 41 \ 59 \ 26 \ 53 \ 58 \ 97 \ 93 \ 23 \ 84 \ \dots$$

$$P = \begin{bmatrix} 0.31 & 0.41 \\ 0.59 & 0.26 \end{bmatrix} \text{ et } Q = \begin{bmatrix} 0.53 & 0.58 \\ 0.97 & 0.93 \end{bmatrix}$$

2 - Calcul de $P'P$, $Q'Q$, $P'Q+Q'P$

D'où :

$$P'P = \begin{bmatrix} 0.4442 & 0.2805 \\ 0.2805 & 0.2357 \end{bmatrix}$$

$$Q'Q = \begin{bmatrix} 1.2218 & 1.2095 \\ 1.2095 & 1.2013 \end{bmatrix}$$

$$P'Q + Q'P = \begin{bmatrix} 1.4732 & 1.198 \\ 1.198 & 0.9592 \end{bmatrix}$$

3 - Calcul des coefficients β_i

$$\beta_1 = \text{Tr}(P'P) = 0.6799$$

$$\beta_2 = \text{Tr}(Q'Q) = 2.4324$$

$$\beta_3 = \text{Tr}(P'Q+Q'P) = 2.4231$$

4 - Calcul des coefficients γ_i

$$\gamma_1 = \text{Det}(P'P) = 0.026\ 017\ 69$$

$$\begin{aligned} \gamma_2 &= |p_1, r_2| + |p_2, r_1| = 0.090\ 037\ 064 - 0.011\ 194\ 24 \\ &= 0.078\ 843\ 4 \end{aligned}$$

$$\begin{aligned} \gamma_3 &= |p_1, q_2| + |p_2, q_1| = 0.194\ 352\ 71 + 0.051\ 286\ 49 \\ &= 0.245\ 639\ 2 \end{aligned}$$

$$\begin{aligned} \gamma_4 &= |q_1, r_2| + |q_2, r_1| = -0.277\ 030\ 44 - 0.320\ 774\ 16 \\ &= -0.597\ 804\ 6 \end{aligned}$$

$$\gamma_5 = \text{Det}(Q'Q) = 0.004\ 858\ 09$$

5 - Calcul de f, f', g, g'

D'où :

$$f(\theta) = 0.6799 \cos^2 \theta + 2.4324 \cos \theta \sin \theta + 2.4231 \sin^2 \theta$$

$$\Rightarrow f'(\theta) = 2.4324 \cos^2 \theta + 3.4864 \cos \theta \sin \theta - 2.4324 \sin^2 \theta$$

et :

$$\begin{aligned} g(\theta) = & 0.026\,017\,6 \cos^4 \theta \\ & + 0.078\,843\,4 \cos^3 \theta \sin \theta \\ & + 0.245\,639\,2 \cos^2 \theta \sin^2 \theta \\ & - 0.597\,804\,6 \cos \theta \sin^3 \theta \\ & + 0.004\,858\,09 \sin^4 \theta \end{aligned}$$

$$\begin{aligned} \Rightarrow g'(\theta) = & 0.078\,843\,4 \cos^4 \theta \\ & + 0.387\,207\,64 \cos^3 \theta \sin \theta \\ & - 2.029\,944 \cos^2 \theta \sin^2 \theta \\ & - 0.471\,846\,04 \cos \theta \sin^3 \theta \\ & + 0.597\,804\,6 \sin^4 \theta \end{aligned}$$

6 - Equation vérifiée par θ

$$\begin{aligned} g'^2 - f f' g' + f'^2 g = & 0.029\,761\,4 \cos^8 \theta \\ & - 0.324\,916 \cos^7 \theta \sin \theta \\ & + 1.774\,17 \cos^6 \theta \sin^2 \theta \\ & + 11.044\,3 \cos^5 \theta \sin^3 \theta \\ & + 20.282\,8 \cos^4 \theta \sin^4 \theta \\ & + 7.034\,62 \cos^3 \theta \sin^5 \theta \\ & - 8.984\,67 \cos^2 \theta \sin^6 \theta \\ & - 8.477\,76 \cos \theta \sin^7 \theta \\ & + 3.903\,54 \sin^8 \theta \end{aligned}$$

7 - Solutions en θ

L'équation (6) du 8^{ème} degré en θ a deux solutions réelles en θ sur $[0, \pi[$, θ_1 et θ_2 . $\theta_1 + \pi$ et $\theta_2 + \pi$ sont également solution sur $[0, 2\pi[$. La fonction (6) est en effet périodique de période π , et non 2π , comme polynôme trigonométrique homogène de degré pair.

$$\theta_1 = 1.100\,10$$

$$\theta_2 = 1.151\ 64$$

8 - Valeurs propres associées

$$(P\cos\theta_1 + Q\sin\theta_1)'(P\cos\theta_1 + Q\sin\theta_1) = \begin{bmatrix} 1.657\ 32 & 1.502\ 65 \\ 1.502\ 65 & 1.390\ 40 \end{bmatrix}$$

dont la plus grande valeur propre est $\lambda_1 = 3.032\ 49$

$$(P\cos\theta_2 + Q\sin\theta_2)'(P\cos\theta_2 + Q\sin\theta_2) = \begin{bmatrix} 1.640\ 67 & 1.500\ 98 \\ 1.500\ 98 & 1.640\ 67 \end{bmatrix}$$

dont la plus grande valeur propre est $\lambda_2 = 3.025\ 19$

9 - Résultat pour la valeur propre

La plus grande valeur propre $\lambda(\theta)$ associée à la matrice $P.\cos\theta + Q.\sin\theta$ est $\lambda = 3.025\ 49$ obtenue pour $\theta = 1.100\ 10$.

10 - Calcul des vecteurs x, y et z

x est le vecteur suivant fonction de θ :

$$x = (\cos\theta, \sin\theta) = (0.453\ 507, 0.891\ 252)$$

y est vecteur propre unitaire de $(A.x)'(A.x)$

$$y = (0.737\ 720, 0.675\ 105)$$

$$z \text{ est calculé par } z = \frac{(A.x).y}{\|(A.x).y\|}$$

$$A.x = \begin{bmatrix} 0.612\ 951 & 0.702\ 864 \\ 1.132\ 08 & 0.946\ 776 \end{bmatrix}$$

et

$$z = (0.305\ 596 , 0.486\ 190)$$

11 - Résultat

La première composante principale $\lambda x \otimes y \otimes z$ avec $\|x\|=\|y\|=\|z\|=1$ de $A=i \otimes P + j \otimes Q$, avec :

$$P = \begin{bmatrix} 0.31 & 0.41 \\ 0.59 & 0.26 \end{bmatrix} \text{ et } Q = \begin{bmatrix} 0.53 & 0.58 \\ 0.97 & 0.93 \end{bmatrix}$$

est :

$$\lambda = 3.025\ 19$$

$$x = (0.453\ 507 , 0.891\ 252)$$

$$y = (0.737\ 720 , 0.675\ 105)$$

$$z = (0.305\ 596 , 0.486\ 190)$$

Soit, de façon reconstituée :

$$\lambda x \otimes y \otimes z = i \otimes \hat{P} + j \otimes \hat{Q} \text{ avec :}$$

$$\hat{P} = \begin{bmatrix} 0.31 & 0.28 \\ 0.49 & 0.45 \end{bmatrix} \text{ et } \hat{Q} = \begin{bmatrix} 0.60 & 0.55 \\ 0.96 & 0.88 \end{bmatrix}$$

Ce paragraphe présente succinctement comment les résultats sur le modèle PARAFAC / CANDECOMP peuvent être utilisés pour obtenir des modèles multiplicatifs polynomiaux (MMP) ajustés sur des jeux de données. Il est présenté sous forme d'ouverture, et comprend le regard algébrique sur ce problème. Le lien avec les modèles statistiques peut être trouvé par exemple dans DENIS (1983).

Etant donnée une variable à expliquer η , et r variables explicatives ξ_1, \dots, ξ_r , on appelle Modèle Multiplicatif Polynomial (MMP) sur un jeu de données un ajustement de la forme :

$$\eta_1 = K_1(\xi_{11}) \dots K_r(\xi_{r1}) + \varepsilon_1$$

où les K_j sont des polynômes sur les variables ξ_j . Par exemple :

$$\eta_1 = (\alpha_1 + \beta_1 \xi_{11})(\alpha_2 + \beta_2 \xi_{21})(\alpha_3 + \beta_3 \xi_{31}) + \varepsilon_1$$

est un MMP par trois polynômes du premier degré.

1. Notations

Désignons par \mathcal{P}_p l'espace vectoriel des polynômes sur \mathbb{R} de degré $\leq p$, muni de la métrique définies par le poids w :

$$\langle P, Q \rangle = \int_{\mathbb{R}} w(\xi) P(\xi) Q(\xi) d\xi$$

\mathcal{P}_p est de dimension $p+1$.

2. Polynômes orthogonaux

Soit L_i une famille de polynômes orthogonaux de \mathcal{P}_p :

$$\langle L_i, L_j \rangle = \delta_{ij}$$

$$\Leftrightarrow$$

$$\int_{\mathbb{R}} w(\xi) L_i(\xi) L_j(\xi) d\xi = \delta_{ij}$$

Rappelons que l'on note $P \otimes Q$ le polynôme sur \mathbb{R}^2 défini par :
 $P \otimes Q(\xi, \eta) = P(\xi)Q(\eta)$.

Alors, $L_i \otimes L_j$ définit une famille de polynômes orthogonaux sur \mathbb{R}^2 pour la métrique définie par $w \otimes w$:

$$\langle L_i \otimes L_j, L_m \otimes L_n \rangle = \delta_{im} \delta_{jn}$$

$$\Leftrightarrow$$

$$\iint_{\mathbb{R}^2} w(\xi) w(\eta) L_i(\xi) L_j(\eta) L_m(\xi) L_n(\eta) d\xi d\eta = \delta_{im} \delta_{jn}$$

(La démonstration est immédiate par le théorème de Fubini).

Et ainsi de suite pour la famille $L_{i_1} \otimes \dots \otimes L_{i_r}$ qui forme une famille de polynômes orthogonaux de $\mathcal{P}_p^{\otimes r}$ métrique $w \otimes \dots \otimes w$.

3. Rang et ACP d'un polynôme à deux inconnues

Soit P un polynôme de $\mathcal{P}_p^{\otimes r}$. Par analogie avec le rang d'une matrice M de $\mathcal{E} \otimes \mathcal{F}$ où \mathcal{E} et \mathcal{F} sont deux EV, on peut définir le rang de P comme le plus petit entier r tel qu'il existe une famille (K_i) et (L_i) de \mathcal{E} telle que

$$P = \sum_{i=1}^r K_i \otimes L_i, \text{ soit :}$$

$$P(\xi, \eta) = \sum_{i=1}^r K_i(\xi) \otimes L_i(\eta)$$

De même, réaliser l'ACP de P revient à rechercher la meilleure approximation de P par un polynôme de rang inférieur. En particulier, rechercher la première composante principale de P est rechercher les deux polynômes K et L de \mathcal{E} tels que $\|P - K \otimes L\|$ minimale, soit :

$$\iint_{\mathbb{R}^2} w(\xi)w(\eta) \left(P(\xi, \eta) - K(\xi)L(\eta) \right)^2 d\xi d\eta \text{ minimale}$$

On peut noter que ces résultats sont très classiques, puisque le développement de la décomposition en valeurs singulières d'une part, et de l'algèbre linéaire sous son formalisme le plus abstrait (espaces hilbertiens puis espaces de Banach) d'autres parts datent de recherches sur les noyaux, la décomposition et l'approximation d'endomorphismes d'espaces fonctionnels depuis la fin du siècle dernier.

4. Modèle multiplicatif : cas de deux variables

Soit y un vecteur de n observations d'une variable à expliquer, et x_1 et x_2 les vecteurs de n observations correspondantes de deux variables explicatives. Nous allons chercher la meilleure approximation de η_1 par un produit $K(\xi_{11})L(\xi_{21})$.

Donnons l'exemple simple où $p = 1$ (polynômes de degré 1), et posons :

$$\bullet K(\xi) = \alpha + \beta\xi$$

$$\bullet L(\xi) = \lambda + \mu\xi$$

Nous cherchons la meilleure approximation des η_1 de la forme :

$$\eta_1 = (\alpha + \beta\xi_{11})(\lambda + \mu\xi_{21}) + \varepsilon_1 \quad (1)$$

avec $\sum_1 \varepsilon_1^2$ minimale.

Si une telle approximation existe, alors, a fortiori, il existera une approximation de la forme :

$$\eta_1 = \rho + \tau\xi_{11} + \vartheta\xi_{21} + \upsilon\xi_{11}\xi_{21} + \varepsilon_1 \quad (2)$$

Or, une approximation de la forme (2) est aisément calculable, car c'est rechercher la projection de y sur l'espace engendré par les vecteurs e , x_1 , x_2 et x_{12} , où ' e ' a 1 pour $i^{\text{ème}}$ coordonnée, et x_{12} $\xi_{11}\xi_{21}$. On connaît donc le polynôme $P(\xi_1, \xi_2)$ de $\mathcal{P}_1 \otimes \mathcal{P}_1$ tel que $\sum_1 (\eta_1 - P(\xi_{11}, \xi_{21}))^2$ soit minimale.

Alors, une bonne approximation de la forme (1) pourra être obtenue en calculant la première composante principale de P , c'est à dire les polynômes $K(\xi_1)$ et $L(\xi_2)$ tels que $\int_{\mathbb{R}^2} (P(\xi_1, \xi_2) - K(\xi_1)L(\xi_2))^2$ soit minimale.

Cet exemple est aisément généralisable pour un degré p quelconque :
 Pour trouver une bonne approximation de la forme :

$$\eta_i = K(\xi_{1i})L(\xi_{2i}) + \varepsilon_i \quad (3)$$

où K et L sont deux polynômes de \mathcal{P}_p , on calcule dans un premier temps la meilleure approximation de la forme :

$$\eta_i = P(\xi_{1i}, \xi_{2i}) + \varepsilon_i \quad (4)$$

et on calcule ensuite la première composante principale $K \otimes L$ de P .

Ce résultat avait été proposé dans DENIS (1983), où la recherche d'un modèle multiplicatif était une conséquence de la régression factorielle, utilisée par cet auteur pour décomposer l'interaction dans un modèle additif bilinéaire. DENIS (1983) insistait déjà sur l'importance de la notation tensorielle, qui permettait de comprendre algébriquement les calculs sous-jacents à la décomposition de l'interaction.

5. Modèle multiplicatif : cas de plusieurs variables

Ce résultat se généralise aisément au cas de q variables.

■ Position du problème

Soit y un vecteur de n observations d'une variable à expliquer, et x_1, \dots, x_q les vecteurs de n observations correspondantes de q variables explicatives. Nous allons chercher la meilleure approximation de η_i par un produit $K(\xi_{1i}) \times \dots \times K_q(\xi_{qi})$.

Soit à rechercher le meilleur MMP de la forme :

$$\eta_1 = K_1(\xi_{11}) \times \dots \times K_q(\xi_{q1}) + \varepsilon_1$$

Si l'on note $K_j(x_1)$ le vecteur de coordonnées $K_j(\xi_{1j})$, $K(x_1, \dots, x_q)$ le vecteur de coordonnées $K(\xi_{11}, \dots, \xi_{q1})$, on recherche dans un premier temps par un calcul simple de projection le polynôme K de $\mathcal{P}_p \otimes \dots \otimes \mathcal{P}_p$ tel que $\|y - K(x_1, \dots, x_q)\|$ soit minimal.

On choisit ensuite pour K_1, \dots, K_q la première composante d'une approximation de type PARAFAC de K , c'est à dire le q-uplet tel que :

$$\int \dots \int_{\mathbb{R}^q} w(\xi_1) \dots w(\xi_q) [K(\xi_1, \dots, \xi_q) - K_1(\xi_1) \times \dots \times K_q(\xi_q)]^2 d\xi_1 \dots d\xi_q \quad (2)$$

minimale.

Malgré l'apparente complexité de cette écriture, ce problème est simple à résoudre, de deux manières :

■ Résolution directe

Soit E_1 une base naturelle de $\mathcal{P}_p \otimes \dots \otimes \mathcal{P}_p$, par exemple :

$$e, x_1, \dots, x_q, x_{12}, \dots, x_{q-1,q}, x_{123}, \dots$$

Le problème (2) est alors un problème PARAFAC avec métriques définie par : $\mu_{1j} = \langle E_1, E_j \rangle =$

$$\int \dots \int_{\mathbb{R}^q} w(\xi_1) \dots w(\xi_q) E_1(\xi_1, \dots, \xi_q) E_j(\xi_1, \dots, \xi_q) d\xi_1 \dots d\xi_q$$

par exemple, si :

- $q = 3$
- $w(x) = \chi_{[0,1]}$
- $E_2 = x_1$
- $E_5 = x_{12}$

ce calcul devient :

$$\mu_{25} = \int_0^1 \int_0^1 \int_0^1 \xi_1 \cdot \xi_1 \xi_2 d\xi_1 d\xi_2 d\xi_3$$

$$\begin{aligned}
&= \left(\int_0^1 \xi_1^2 d\xi_1 \right) \left(\int_0^1 \xi_2 d\xi_2 \right) \left(\int_0^1 d\xi_3 \right) \\
&= (1/3) \cdot (1/2) \cdot 1 = 1/6
\end{aligned}$$

■ Résolution par polynômes orthogonaux

Cette méthode revient à chercher une BON L_i de $\mathcal{P}_p \otimes \dots \otimes \mathcal{P}_p$ telle que $\langle L_i, L_j \rangle = \delta_{ij}$. Une telle base dépend des poids $w(\xi)$. Si K est exprimé dans une telle base, (2) se ramène à la recherche d'un modèle PARAFAC avec métrique unité. Cette méthode est numériquement plus stable.

Dans ce chapitre, nous proposons une ouverture possible de l'ACP-DCL des matrice aux contraintes non linéaires algébriques, et une extension pour l'analyse des multitableaux, à plus de deux entrées. Ce chapitre est présenté sous forme d'ouverture, c'est à dire de piste balisée à suivre : les compléments indispensable de géométrie différentielle, assez techniques et qui rendraient le lecture ardue, ne sont pas présentés dans cette version du travail. De même, certains calculs sont ébauchés jusqu'au point ou leur achèvement est purement mécanique. Cette partie mécanique a parfois été omise dans les généralisations.

□

L'ACP-DCL est la technique d'Analyse en Composante Principale d'une matrice sous Double Contrainte Linéaire. Rappelons que, si A est une telle matrice, on recherche une décomposition de la forme, à r fixé :

$$A = \sum_{i=1}^r x_i \otimes y_i + R$$

avec $\|R\|^2$ minimale et $x_i \in \mathcal{F}_1$; $y_i \in \mathcal{F}_2$, SEV fixés à l'avance. La solution consiste alors à projeter A dans l'espace $\mathcal{F}_1 \otimes \mathcal{F}_2$ et à réaliser l'ACP de la matrice projetée.

L'extension aux modèles PARAFAC / CANDECOMP et modèles de TUCKER des tableaux à 3 entrées ou plus se réalise alors sans difficultés, et a été étudiée aux chapitres D4, D8, E6, E7.

Dans ce chapitre, nous allons étudier un problème plus large, où les contraintes sur les composantes x_i , y_i , ... seront non linéaires : on impose

à ces composantes d'appartenir à une sous-variété gauche de $\mathcal{E}_1, \mathcal{E}_2, \dots$, et non plus simplement linéaire.

Dans un premier temps, nous préciserons les variétés retenues pour cette étude, à savoir les variétés à paramétrage polynomial. Puis, nous étudierons l'ACP d'une matrice avec doubles contraintes algébriques (ACP-DCA), et élargirons les résultats aux modèles de TUCKER et PARAFAC / CANDECOMP.

L'utilisation du formalisme de l'algèbre tensorielle apporte ici également une clarté qui permet assez simplement de suivre les calculs, et de proposer des algorithmes numériques simples et efficaces.

1. VARIETES A PARAMETRAGE POLYNOMIAL

Les variétés polynomiales que nous utilisons ici sont des variétés différentielles dont les fonctions de paramétrage sont des polynômes.

L'introduction de variétés différentielles de façon rigoureuse est toujours assez technique et délicate, que l'on adopte le point de vue intrinsèque, où l'on privilégie la notion de variété abstraite (DOUBROVINE & Al, 1982 ; MALLIAVIN, 1972), ou extrinsèque, où l'on privilégie la notion de sous variété (BERGER & GOSTIAUX, 1987 ; STOKER, 1969).

La principale difficulté vient de la nécessité de montrer que les définitions et résultats proposés ne dépendent pas du paramétrage choisi. Par exemple, dans les espaces vectoriels, le produit scalaire $\langle x, y \rangle$ ne dépend pas de la base choisie.

C'est ainsi que le paramétrage polynomial d'une variété n'est pas une propriété intrinsèque d'une variété : il est essentiellement lié au paramétrage choisi, de façon tautologique. Aussi, nous choisissons, une fois pour toute dans ce chapitre, une BON (e_i) d'un EV \mathcal{E} , et un paramétrage fixé

φ de \mathbb{R}^q dans \mathcal{E} qui définit une sous variété Ξ dans \mathcal{E} . Ξ est définie comme l'image de \mathbb{R}^q par φ .

1. Exemple et généralisation

1 - Un SEV \mathcal{F} d'un EV \mathcal{E} est une variété à paramétrage polynomial particulière, ou les polynômes sont de degré 1.

2 - Si (u_1, \dots, u_q) est une base de \mathcal{F} , alors tout vecteur y de \mathcal{F} s'écrit :

$$y = \eta_1 u_1 + \dots + \eta_q u_q$$

3 - Si (e_1, \dots, e_n) est une BON de \mathcal{E} , et que l'on décompose y et les u_i sur une telle base, on obtient facilement :

$$\langle y, e_i \rangle = \eta_1 \langle u_1, e_i \rangle + \dots + \eta_q \langle u_q, e_i \rangle$$

soit :

$$\zeta_i = \eta_1 \omega_{1i} + \dots + \eta_q \omega_{qi}$$

où $\omega_{ki} = \langle u_k, e_i \rangle$.

4 - Le SEV \mathcal{F} , en tant que sous variété de \mathcal{E} , est alors paramétré par une famille de polynômes du premier degré en (η_1, \dots, η_q) .

5 - La généralisation à une variété polynomiale générale est alors simple : chacun des ζ_i est alors paramétré par un polynôme de degré quelconque en (η_1, \dots, η_q) .

2. Définition

Etant donné un EV \mathcal{E} sur \mathbb{R} muni d'une BON fixée (e_i) , on appelle variété à paramétrage polynomial (VPP) de \mathcal{E} de dimension q un sous ensemble Ξ de \mathcal{E} qui soit l'image de \mathbb{R}^q par une fonction polynomiale.

$$\xi_i = P_i(\eta_1, \dots, \eta_q)$$

où P_i est un polynôme à q indéterminées.

3. Exemple

Dans \mathbb{R}^3 , une VPP de dimension 2 paramétrée par des polynômes de degré 2 est l'ensemble des vecteurs $\xi_1 e_1 + \xi_2 e_2 + \xi_3 e_3$ tels que :

$$\begin{cases} \xi_1 = \alpha_0 + \alpha_{11}\eta_1 + \alpha_{12}\eta_2 + \alpha_{21}\eta_1^2 + \alpha_{22}\eta_1\eta_2 + \alpha_{23}\eta_2^2 \\ \xi_2 = \beta_0 + \beta_{11}\eta_1 + \beta_{12}\eta_2 + \beta_{21}\eta_1^2 + \beta_{22}\eta_1\eta_2 + \beta_{23}\eta_2^2 \\ \xi_3 = \gamma_0 + \gamma_{11}\eta_1 + \gamma_{12}\eta_2 + \gamma_{21}\eta_1^2 + \gamma_{22}\eta_1\eta_2 + \gamma_{23}\eta_2^2 \end{cases}$$

où η_1 et η_2 parcourent \mathbb{R}^2 .

4. Ecriture tensorielle d'une VPP

La géométrie différentielle est certainement la discipline qui fait le plus appel au calcul tensoriel (DODSON & POSTON, 1977). C'en est d'ailleurs le berceau au début de ce siècle. Mais ce n'est pas par ce biais que nous introduirons les tenseurs sur les variétés polynomiales : nous resterons dans le domaine algébrique d'application du calcul tensoriel.

1 - Ainsi, par exemple, si nous désignons par :

- y le vecteur de \mathcal{E} de coordonnées (ξ_1, ξ_2, ξ_3)
- u le vecteur de \mathbb{R}^2 de coordonnées (η_1, η_2)

le système précédent peut s'écrire :

$$y = a + A.u + B.(u \otimes u)$$

2 - Cette écriture est l'écriture tensorielle du paramétrage de \mathcal{E} . Elle peut se généraliser aisément, comme suit.

3 - Soient

- \mathcal{E} un EV de dimension n
- \mathbb{R}^q , avec $q < n$
- $u^{\otimes k}$ le tenseur $u \otimes \dots \otimes u$ (k fois) de $\mathcal{A}_k(\mathbb{R}^q)$
- A_k un tenseur de $\mathcal{A}_k(\mathbb{R}^q)$.

Alors, l'écriture générale d'une VPP de \mathcal{E} de dimension q est de la forme :

$$y = \sum_{k \leq d} A_{k+1} \cdot u^{\otimes k}$$

où d est le degré maximal des polynômes participant au paramétrage.

5. Schéma de Horner pour le calcul d'un polynôme à plusieurs indéterminées

Le schéma de Hörner est une technique classique permettant l'évaluation rapide d'un polynôme connaissant la valeur de l'indéterminée. Il est particulièrement précieux dans les algorithmes numériques requérant un grand nombre d'évaluations d'un polynôme.

Nous nous proposons de l'étendre très simplement aux polynômes à plusieurs indéterminées, ce qui sera utile pour les calculs dans la suite, lorsque seront abordés les algorithmes numériques.

■ Rappel du schéma pour les polynômes à une indéterminée

1 - Rappelons que, pour le calcul de $\eta = P(\xi)$, où $P \in \mathbb{R}[X]$, le schéma de Hörner permet un calcul bien plus économique que le calcul effectif des monômes degré par degré (PRESS & Al., 1989 ; SIBONY & MARDON, 1982).

2 - Soit par exemple à évaluer :

$$\eta = \alpha_0 + \alpha_1 \xi + \alpha_2 \xi^2 + \alpha_3 \xi^3 + \alpha_4 \xi^4 + \alpha_5 \xi^5$$

La méthode simpliste consiste à calculer les 5 puissances de ξ (4 multiplications), puis $P(\xi)$ (5 multiplications), soit 9 multiplications au total.

3 - Le schéma de Hörner propose l'écriture suivante :

$$\eta = \alpha_0 + \xi(\alpha_1 + \xi(\alpha_2 + \xi(\alpha_3 + \xi(\alpha_4 + \xi\alpha_5))))$$

ou encore :

$$\eta = \alpha_0 + (\alpha_1 + (\alpha_2 + (\alpha_3 + (\alpha_4 + \alpha_5 \xi) \xi) \xi) \xi) \xi$$

qui ne requièrent que 5 multiplications.

4 - Ce schéma se généralise aisément à un degré quelconque, et l'évaluation d'un polynôme de degré q requiert q multiplications.

5 - Remarquons que cette valeur n'est pas optimale : l'évaluation d'un polynôme de degré 4 peut se faire avec 3 multiplications, de degré 5 avec 4 multiplications et de degré 5 avec 4 multiplications également (PRESS & Al., 1989). Ces résultats sont directement issus des études de la complexité des algorithmes sur le rang, présentées au chapitre F (LAFON, 1975, STRASSEN, 1972).

■ Généralisation à plusieurs indéterminées

1 - Soit par exemple le polynôme : $y = a + A_2.u + A_3.(u \otimes u)$.

2 - On peut également écrire : $y = a + (A_2 + A_3.u).u$,
puisque $A_3.(u \otimes u) = (A_3.u).u$

3 - Ce qui se généralise aisément :

$$a + A_2.u + A_3.u^{\otimes 2} + A_4.u^{\otimes 3} + A_5.u^{\otimes 4} = a + (A_2 + (A_3 + (A_4 + A_5.u).u).u).u$$

et ainsi de suite.

4 - Par exemple, si $u \in \mathbb{R}^2$, faisons le décompte des multiplications nécessaires.

■ Pour $a + (A_2 + (A_3 + (A_4 + A_5 \cdot u) \cdot u) \cdot u) \cdot u$

1 - $A_5 \in \mathcal{A}_5(\mathbb{R}^2)$, et possède $2^5 = 32$ éléments. $A_5 \cdot u$ requiert 32 multiplications.

2 - $(A_4 + A_5 \cdot u) \in \mathcal{A}_4(\mathbb{R}^2)$ et possède $2^4 = 16$ éléments. $(A_4 + A_5 \cdot u) \cdot u$ requiert 16 multiplications.

3 - Et ainsi de suite : le nombre de multiplications nécessaires est de $32 + 16 + 8 + 4 = 60$

■ Pour $a + A_2 \cdot u + A_3 \cdot u^{\otimes 2} + A_4 \cdot u^{\otimes 3} + A_5 \cdot u^{\otimes 4}$

1 - $A_2 \in \mathcal{A}_2(\mathbb{R}^2)$. Donc $A_2 \cdot u$ requiert 4 multiplications.

2 - $u \otimes u$ requiert 3 multiplications. A_3 possède $2^3 = 8$ éléments. $A_3 \cdot u^{\otimes 2}$ requiert donc 8 multiplications.

3 - Connaissant $u \otimes u$, qu'il faut donc stocker, $u^{\otimes 3}$ requiert 4 multiplications. A_4 possède $2^4 = 16$ éléments, et $A_4 \cdot u^{\otimes 3}$ requiert donc 16 multiplications.

4 - De même, $u^{\otimes 4}$ requiert 5 multiplications et $A_5 \cdot u^{\otimes 4}$ en requiert 32.

5 - Le nombre total de multiplications nécessaires est :
 $4 + (3+8) + (4+16) + (5+32) = 72$.

2. ACP D'UNE MATRICE AVEC CONTRAINTES ALGEBRIQUES

Après avoir posé la définition, nous développerons la technique de résolution proposée sur un exemple de degré peu élevé, puis généraliserons à un degré quelconque.

1. Définition

Soient Ξ et Ω deux VPP de \mathcal{E} et \mathcal{F} respectivement, et M une matrice de $\mathcal{E} \otimes \mathcal{F}$.

On appelle ACP de A sous double contrainte algébrique (ACP-DCA) définies par Ξ et Ω une décomposition de la forme :

$$M = \sum_{i=1}^r x_i \otimes y_i + R$$

avec :

$$x_i \in \Xi$$
$$y_i \in \Omega$$
$$\|R\| \text{ minimale}$$

2. Solution sur un exemple

Deux techniques seront nécessaires pour résoudre ce problème :

1 - Un algorithme qui prolonge dans le cas de contraintes non linéaires l'algorithme de WOLD (WOLD, 1966) présenté pour le calcul des CP sans contraintes. Il fait partie de la famille des algorithmes par moindres carrés alternés.

2 - Un algorithme de descente.

■ Présentation de l'exemple

1 - Dans l'exemple simple suivant, nous choisissons :

- $r = 1$
- $\mathcal{E} = \mathcal{F} = \mathbb{R}^3$
- Ξ et Ω de dimension et de degré 2

2 - Soit le paramétrage suivant de Ξ et Ω :

- $\Xi : x = a + A.u + A_3.u^{\otimes 2}$
- $\Omega : y = b + B.v + B_3.v^{\otimes 2}$

avec :

- $u, v \in \mathbb{R}^2$
- $a, b \in \mathbb{R}^3$
- $A, B \in \mathbb{R}^2 \otimes \mathbb{R}^3$
- $A_3, B_3 \in \mathbb{R}^2 \otimes \mathbb{R}^2 \otimes \mathbb{R}^3$

3 - Nous cherchons donc à minimiser ε défini par :

$$\varepsilon = \|M - x \otimes y\|^2,$$

avec $x \in \Xi$ et $y \in \Omega$.

4 - Cela revient donc à chercher u et v tels que :

$$\varepsilon = \|M - (a + A.u + A_3.u^{\otimes 2}) \otimes (b + B.v + B_3.v^{\otimes 2})\|^2 \text{ minimal}$$

■ Moindres carrés alternés

La méthode consiste à optimiser alternativement sur u et v , considérant l'autre vecteur comme fixé. Présentons par exemple l'optimisation sur v . Le calcul, même s'il peut parfois paraître fastidieux ou long, est purement mécanique grâce, là encore, à la notation synthétique induite par l'algèbre tensorielle. Un tel calcul est quasi impossible sur les indices directement.

1 - u étant fixé, posons : $x = a + A.u + A_3.u^{\otimes 2}$.

2 - ε devient : $\varepsilon = \|M - x \otimes (b + B.v + B_3.v^{\otimes 2})\|^2$

3 - Le calcul des solutions de $\frac{\partial \varepsilon}{\partial v} = 0$ mène à un système algébrique non soluble par des méthodes simples. La méthode de minimisation retenue pour ε sera donc une méthode de descente par la plus forte pente accélérée (MINOUX, 1983).

4 - Cette méthode requiert le calcul de $\nabla \varepsilon$, effectué en annexe de ce chapitre, et qui est de la forme :

$$\nabla \varepsilon = c + C.v + C_3.v^{\otimes 2} + C_4.v^3$$

5 - La descente selon la plus forte pente consiste, un vecteur v_k étant trouvé à la $k^{\text{ème}}$ itération, à calculer le vecteur v_{k+1} de la forme $v_k - \lambda \nabla \varepsilon$ tel que ε_{k+1} soit minimum (si $\varepsilon_k = \varepsilon(v_k)$).

6 - Or, ε_{k+1} est un polynôme du quatrième degré en λ . En chercher le minimum revient à résoudre une équation du troisième degré, ce qui est réalisable. Ce calcul est présenté en annexe de ce chapitre.

3. Généralisation

■ Problème à résoudre

1 - Soit donc à résoudre :

$$M = \sum_i x_i \otimes y_i + R$$

avec $x_i \in \Xi$

$$y_i \in \Omega$$

$\|R\|$ minimale.

2 - Nous supposons Ξ et Ω définies respectivement par :

$$\begin{aligned}\Xi &= \{x \in \mathcal{E} \mid x = \sum_{i < p} A_{i+1} \cdot u^{\otimes i}, u \in \mathbb{R}^k\} \\ \Omega &= \{y \in \mathcal{F} \mid y = \sum_{j < q} B_{j+1} \cdot v^{\otimes j}, v \in \mathbb{R}^l\}\end{aligned}$$

Pour plus de simplicité dans l'écriture, et cela ne restreint pas la gamme des paramétrages possibles, nous supposons que les tenseurs A_m et B_r sont totalement symétriques.

■ 2. Présentation de l'optimisation

1 - Recherchons la première composante principale de M. Cela revient à minimiser :

$$\|R\|^2 = \|M - (\sum_i A_{i+1} \cdot u^{\otimes i}) \otimes (\sum_j B_{j+1} \cdot v^{\otimes j})\|^2$$

2 - Un algorithme des moindres carrés alternés peut être mise en oeuvre sur

u et v, avec optimisation sur u et v par un algorithme de descente selon la plus forte pente accélérée.

■ Mise en oeuvre de l'optimisation

La mise en oeuvre est présentée pour l'optimisation sur v. L'optimisation sur u est analogue, après transposition. Deux étapes sont réalisées :

- le calcul du gradient de la quantité à minimiser
- l'optimisation dans la direction du gradient

■ Calcul de gradient

1 - u étant fixé, donc x, posons :

$$\varepsilon = \|M - (\sum_i A_{i+1} \cdot u^{\otimes i}) \otimes (\sum_j B_{j+1} \cdot v^{\otimes j})\|^2 = \|M - x \otimes (\sum_j B_{j+1} \cdot v^{\otimes j})\|^2$$

2 - Le gradient de ε se calcule ainsi :

$$\begin{aligned} \varepsilon &= \|M - x \otimes (\sum_j B_{j+1} \cdot v^{\otimes j})\|^2 \\ &= \|M\|^2 - 2 \langle M, x \otimes (\sum_j B_{j+1} \cdot v^{\otimes j}) \rangle + \|x \otimes (b + B \cdot v + B_3 \cdot v^{\otimes 2})\|^2 \\ &= \|M\|^2 - 2 \langle M \cdot x, \sum_j B_{j+1} \cdot v^{\otimes j} \rangle + \|x\|^2 \cdot \|\sum_j B_{j+1} \cdot v^{\otimes j}\|^2 \end{aligned}$$

3 - D'où :

$$\nabla \varepsilon = -2 \nabla \langle M \cdot x, \sum_j B_{j+1} \cdot v^{\otimes j} \rangle + \|x\|^2 \cdot \nabla \|b + B \cdot v + B_3 \cdot v^{\otimes 2}\|^2$$

4 - Or :

$$\begin{aligned} \nabla \langle M \cdot x, \sum_j B_{j+1} \cdot v^{\otimes j} \rangle &= \sum_j \nabla \langle M \cdot x, B_{j+1} \cdot v^{\otimes j} \rangle \\ &= \sum_j \nabla \langle (B_{j+1})' M \cdot x, v^{\otimes j} \rangle \\ &= \sum_j j \sigma[(B_{j+1})' M \cdot x] \cdot v^{\otimes j-1} \\ &= \sum_j C_j \cdot v^{\otimes(j-1)} \end{aligned}$$

où C_j est un tenseur calculé à partir de M , x et B_{j+1} .

■ Mise en oeuvre de la plus forte pente

1 - La descente selon la plus forte pente consiste, un vecteur v_k étant trouvé à la $k^{\text{ème}}$ itération, à calculer le vecteur v_{k+1} de la forme $v_k - \lambda \nabla \varepsilon$ tel que ε_{k+1} soit minimum (si $\varepsilon_k = \varepsilon(v_k)$).

2 - Or :

$$\varepsilon_{k+1} = \|M - x \otimes (\sum_j B_{j+1} \cdot (v_k - \lambda \nabla \varepsilon_k)^{\otimes j})\|^2$$

et :

$(v_k - \lambda \nabla \varepsilon_k)^{\otimes j}$ est un polynôme en λ de degré j .

3 - Donc :

ε_{k+1} est un polynôme en λ de degré $2j$, et l'optimisation sur λ demande le calcul de toutes les solutions réelles d'un polynôme de degré $2j-1$.

4. Prise en compte de plusieurs composantes

■ Remarques liminaires

Dans l'ACP d'une matrice, écrite sous la forme $A = \sum_i \lambda_i x_i \otimes y_i + R$, les composantes (x_i) et (y_i) forment respectivement un système orthonormé.

Cette propriété est conservée lorsque l'on introduit des contraintes linéaires sur les espaces.

En revanche, elle ne l'est plus lorsque les contraintes sont non linéaires, notamment parce que, si $x_1 \in \Xi$, rien ne prouve que $(x_1^\perp) \cap \Xi$ soit non vide.

Ainsi, lors de l'ACP avec contraintes non linéaires, toutes les composantes doivent être calculées une à une, et ne peuvent être le résultat s'une diagonalisation globale.

Un algorithme d'optimisation alternée sur chacune des composantes est alors possible, comme présenté ci-après, avec une notation encore plus compacte pour en suivre plus aisément le déroulement.

■ Notation

Soient $x_i = \sum_{j < p} A_{i,j+1} \cdot u^{\otimes j}$ un paramétrage de Ξ .
 et $y_i = \sum_{j < p} B_{i,j+1} \cdot v^{\otimes j}$ un paramétrage de Ω .

1 - Notons :

$$\begin{aligned} \circ U_i &= \sum_{k < p} u_i^{\otimes k} \\ \circ A_i &= \sum_{j+1} A_{i,j+1} \end{aligned}$$

et :

$$\begin{aligned} \circ V_i &= \sum_{k < p} v_i^{\otimes k} \\ \circ B_i &= \sum_{j+1} B_{i,j+1} \end{aligned}$$

Alors :

$$\begin{aligned} x_i &= A_i \cdot U_i \\ \text{et } y_i &= B_i \cdot V_i \end{aligned}$$

■ Algorithme d'optimisation alternée

1 - Nous pouvons alors écrire :

$$\begin{aligned} \varepsilon &= \|M - \sum_i x_i \otimes y_i\|^2 \\ &= \|M - \sum_i (A_i \cdot U_i) \otimes (B_i \cdot V_i)\| \end{aligned}$$

2 - En supposant les x_i fixés, et en cherchant, par exemple la composante o de y_i ($i=0$), nous pouvons écrire :

$$\begin{aligned} \varepsilon &= \|M - \sum_i (A_i \cdot U_i) \otimes (B_i \cdot V_i)\| \\ &= \|M - \sum_i x_i \otimes (B_i \cdot V_i)\| \\ &= \|M - \sum_{i \neq o} x_i \otimes (B_i \cdot V_i) - x_o \otimes (B_o \cdot V_o)\| \\ &= \|N_o - x_o \otimes (B_o \cdot V_o)\| \end{aligned}$$

3 - Ainsi, recgercher le composante 0 revient à chercher la première composante de $N_0 = M - \sum_{i \neq 0} x_i \otimes (B_i \cdot V_i)$, qui a été l'objet du paragraphe précédent.

3. EXTENSION AUX TABLEAUX A PLUSIEURS ENTREES

L'extension des calculs précédents ne pose pas de problème, si ce n'est technique d'écriture, pour une extension aux multitableaux.

1. Analyse n-mode

C'est l'extension la plus facile, car une analyse n-mode se ramène à une ACP. Les calculs précédents sont donc appliqués directement sur l'ACP qui se présente dans l'analyse n-mode.

2. Modèle Parafac / Candecomp

L'extension aux modèles Parafac / Candecomp se réalise sans difficulté, en deux étapes.

1 - Soit A un tenseur, et $\sum_i x_i \otimes \dots \otimes x_{i_p} + R$ une décomposition par un modèle Parafac / Candecomp.

2 - On impose aux x_i d'appartenir à une variété non linéaire Ξ_1 .

3 - Cette variété connaît un paramétrage polynomial, et nous notons :

$$\circ x_{ij} = A_{ij} \cdot U_{ij}$$

avec :

$$\circ U_{ij} = \sum_{k < p} u_{ij}^{\otimes k}$$

$$\circ A_{ij} = \sum_{j,k+1} A_{ij,k+1}$$

4 - On note alors :

$$A = \sum_i (A_{i1} \cdot U_{i1}) \otimes \dots \otimes (A_{ip} \cdot U_{ip})$$

et :

$$\varepsilon = \|A - \sum_1 (A_{i1} \cdot U_{i1}) \otimes \dots \otimes (A_{ip} \cdot U_{ip})\|^2$$

5 - Sur cette écriture, il est facile d'appliquer comme précédemment, notamment à l'aide des permutations, un algorithme d'optimisation alternée, et de bâtir un algorithme convergeant vers la solution.

3. Modèle de TUCKER

Si, techniquement, l'extension de ces procédures aux modèles de TUCKER ne pose guère de problèmes, en réalisant à chaque étape une ACP n-modes sous contraintes non linéaires, puis en montrant par un simple raisonnement topologique de continuité que la limite de la suite ainsi obtenue satisfait aux mêmes contraintes non linéaires car toute variété définie par un paramétrage polynomial est fermée, son principe même soulève une question : comme, en introduisant des contraintes non linéaires, on perd l'orthogonalité des composantes, le résultat serait-il différent d'un modèle Parafac / Candecomp correctement choisi en fonction de contraintes et dimensions du modèle de Tucker ?

■ Calcul du gradient $\nabla \varepsilon$

$$\begin{aligned}\varepsilon &= \|M - x \otimes (b + B.v + B_3.v^{\otimes 2})\|^2 \\ &= \|M\|^2 - 2\langle M, x \otimes (b + B.v + B_3.v^{\otimes 2}) \rangle + \|x \otimes (b + B.v + B_3.v^{\otimes 2})\|^2 \\ &= \|M\|^2 - 2\langle M.x, b + B.v + B_3.v^{\otimes 2} \rangle + \|x\|^2 \cdot \|b + B.v + B_3.v^{\otimes 2}\|^2\end{aligned}$$

D'où :

$$1 - \nabla \varepsilon = - 2\nabla \langle M.x, b + B.v + B_3.v^{\otimes 2} \rangle + \|x\|^2 \cdot \nabla \|b + B.v + B_3.v^{\otimes 2}\|^2$$

$$2 - \nabla \langle M.x, b + B.v + B_3.v^{\otimes 2} \rangle = \nabla \langle M.x, b \rangle + \nabla \langle M.x, B.v \rangle + \nabla \langle M.x, B_3.v^{\otimes 2} \rangle$$

Soit, puisque :

$$\begin{aligned}\bullet \nabla \langle M.x, B.v \rangle &= \nabla \langle B'.M.x, v \rangle = B'.M.x \\ \bullet \nabla \langle M.x, B_3.v^{\otimes 2} \rangle &= \nabla \langle B'_3.M.x, v^{\otimes 2} \rangle = 2 (B'_3.M.x).v\end{aligned}$$

$$\nabla \langle M.x, b + B.v + B_3.v^{\otimes 2} \rangle = B'.M.x + 2 (B'_3.M.x).v$$

3 - De même :

$$\nabla \|b + B.v + B_3.v^{\otimes 2}\|^2 = 2 \left[\frac{\partial}{\partial v} (b + B.v + B_3.v^{\otimes 2}) \right] \cdot (b + B.v + B_3.v^{\otimes 2})$$

Or :

$$\frac{\partial}{\partial v} (b + B.v + B_3.v^{\otimes 2}) = B + 2B_3.v$$

D'où :

$$\begin{aligned}\nabla \|b + B.v + B_3.v^{\otimes 2}\|^2 &= (B + 2B_3.v) \cdot (b + B.v + B_3.v^{\otimes 2}) \\ &= B.b + B^2.v + B(B_3.v^{\otimes 2}) + 2(B_3.v).b + 2.(B_3.v).(B.v) + 2(B_3.v).(B_3.v^{\otimes 2})\end{aligned}$$

4 - Ainsi, $\nabla \varepsilon$ est de la forme :

$$\nabla \varepsilon = c + C.v + C_3.v^{\otimes 2} + C_4.v^3$$

Dont l'expression à partir de b , B et B_3 est immédiate à partir de 1, 2 et 3, et donne :

$$\begin{aligned}
 \circ c &= -2B'M.x + \|x\|^2 B.b \\
 \circ C.v &= -4(B'_3.M.x).v + \|x\|^2 (B^2.v + 2(B_3.v).b) \\
 \circ C_3.v^{\otimes 2} &= \|x\|^2 (B(B_3.v^{\otimes 2}) + 2.(B_3.v).(B.v)) \\
 \circ C_4.v^{\otimes 3} &= \|x\|^2 (2(B_3.v).(B_3.v^{\otimes 2}))
 \end{aligned}$$

■ Calcul de ε_{k+1}

$$\varepsilon_{k+1} = \|M - x \otimes (b + B.(v_k - \lambda \nabla \varepsilon_k) + B_3.(v_k - \lambda \nabla \varepsilon_k)^{\otimes 2})\|^2$$

1 - Or :

$$\begin{aligned}
 &M - x \otimes (b + B.(v_k - \lambda \nabla \varepsilon_k) + B_3.(v_k - \lambda \nabla \varepsilon_k)^{\otimes 2}) \\
 &= M - x \otimes (b - B.v_k - \lambda \nabla \varepsilon_k) + x \otimes B_3.((v_k - \lambda \nabla \varepsilon_k) \otimes (v_k - \lambda \nabla \varepsilon_k)) \\
 &= (M - x \otimes b - x \otimes B.v_k + x \otimes B_3.v_k^{\otimes 2}) \\
 &\quad + \lambda (x \otimes B.\nabla \varepsilon_k - x \otimes B_3(v_k \otimes \nabla \varepsilon_k + \nabla \varepsilon_k \otimes v_k)) \\
 &\quad + \lambda^2 x \otimes B_3.\nabla \varepsilon_k^{\otimes 2}
 \end{aligned}$$

qui est de la forme :

$$a_0 + \lambda a_1 + \lambda^2 a_2$$

2 - D'où :

$$\begin{aligned}
 \varepsilon_{k+1} &= \|M - x \otimes (b + B.(v_k - \lambda \nabla \varepsilon_k) + B_3.(v_k - \lambda \nabla \varepsilon_k)^{\otimes 2})\|^2 \\
 &= \|a_0 + \lambda a_1 + \lambda^2 a_2\|^2 \\
 &= \|a_0\|^2 + 2\lambda \langle a_0, a_1 \rangle + \lambda^2 (\|a_1\|^2 + 2\langle a_0, a_2 \rangle) + 2\lambda^3 \langle a_1, a_2 \rangle + \lambda^4 \|a_2\|^2
 \end{aligned}$$

Les calculs algébriques présentés dans ce travail mènent directement à des algorithmes numériques de résolution. Leur écriture complète, la vérification, les tests et la diffusion demandent plusieurs mois de travail d'une équipe - car il faut réunir des compétences multiples - que nous n'avons réalisé que partiellement dans le cadre de cette thèse.

Cependant, il apparaît opportun d'en assurer la faisabilité, en présentant un ensemble de logiciels qui permettent provisoirement la mise en oeuvre de ces calculs, sous forme de faisabilité. C'est l'objet des programmes présentés ci-après, qui ne sont qu'une ébauche de ce qu'un logiciel performant pourrait réaliser.

1. Choix d'un langage

La faisabilité a été étudiée sur micro-ordinateur compatible PC, de type 386 SX 20, équipé d'un co-processeur arithmétique, sous DOS, sans extension mémoire, avec disque dur de 40 kO.

Le langage retenu est le TURBO PASCAL de Borland, version 6.0, orienté objet.

Ce choix a été guidé par des considérations pratiques d'exécution dans un temps le plus court possible. Différentes extensions sont envisageables :

- Passer à TURBO-PASCAL sous WINDOWS pour pouvoir exploiter la mémoire vive au delà de 640 kO
- Passer à un langage PASCAL sur gros système

- Passer au langage C++, qui est un langage structuré orienté objet, qui permet notamment de surcharger les opérations. Cette dernière possibilité est importante pour qui souhaite conserver une programmation proche des calculs algébriques.

2. Architecture d'ensemble

Pour parvenir à une programmation correcte qui respecte au mieux la structure algébrique des calculs de ce document, nous avons choisi une programmation structurée orientée objet.

Rappelons que nous avons défini :

- des tenseurs,
- des opérations élémentaires sur les tenseurs,
- des opérations plus complexes sur les tenseurs,
- des algorithmes faisant appel à ces opérations,

chaque ensemble d'opérations ainsi définies appelant les opérations ou définitions de l'ensemble précédent.

Par exemple, nous pouvons définir :

- deux tenseurs A , B
- la différence de deux tenseurs : $A-B$
- la norme de la différence de deux tenseurs : $\|A-B\|$
- une approximation de type Parafac : $\|A - \sum_1 x_1 \otimes y_1 \otimes z_1\|$ minimale.

C'est pourquoi l'ensemble des programmes est organisé en couches, chacune étant une unité en Turbo-Pascal. Ces couches sont :

Couche 1 : Déclare

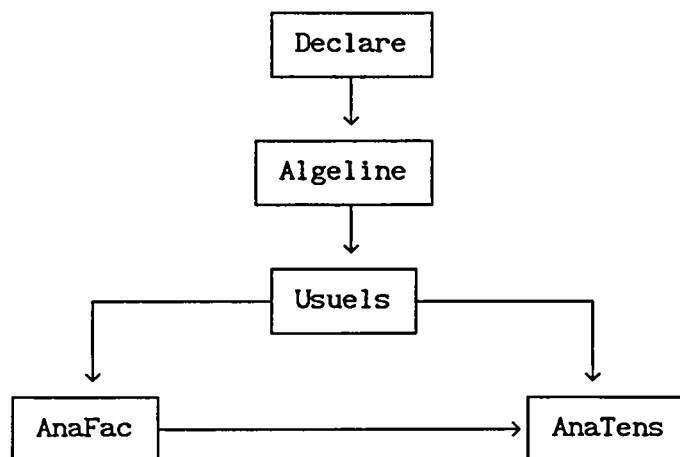
Couche 2 : Algéline

Couche 3 : Usuels

Couche 4 : AnaFac

Couche 5 : AnaTens

avec l'articulation suivante :



Le contenu des différents modules est indiqué en petits caractères.

UNIT DECLARE;

Regroupe les principales déclarations de types pour les vecteurs, matrices et tenseurs d'ordre 3.

INTERFACE

USES Crt;

CONST

nMax = 125; Dimension maximale des tableaux.

TYPE

Vector = Array[1..nMax] of Real;

Tableau = Object

 Coor : Vector;

End;

Vecteur = Object(Tableau)

 Dim : Integer;

 FUNCTION Norme : Real;

 PROCEDURE Affiche;

 PROCEDURE Aleatoire;

 PROCEDURE InitDim (Var n : Integer);

 PROCEDURE InitNul;

 PROCEDURE Normalisation;

 PROCEDURE Saisie;

End;

Matrice = Object(Vecteur)

 NombLign : Integer;

 NombCol : Integer;

 FUNCTION CoorVec(i,j : Integer) : Integer;

 FUNCTION Ind(i,j : Integer) : Real;

 PROCEDURE Affiche;

 PROCEDURE Aleatoire;

 PROCEDURE EcriCol (Var c : Integer ; Var X : Vecteur);

 PROCEDURE ExtractCol (Var c : Integer; Var X : Vecteur);

 PROCEDURE ExtractLign (Var l : Integer; Var X : Vecteur);

 PROCEDURE InitDimM (Var l,c : Integer);

 PROCEDURE InitNombLign (Var n : Integer);

 PROCEDURE InitNombCol (Var n : Integer);

 PROCEDURE Inverse;

 PROCEDURE Jacobi (Var ValPro : Vecteur ; Var VecPro : Matrice);

 PROCEDURE Lec (Var i,j : Integer ; Var x : Real);

 PROCEDURE NormaCol;

 PROCEDURE Saisie;

 PROCEDURE Transpose (Var Mat : Matrice);

End;

Tenseur = Object(Vecteur)

 NombLign : Integer;

```

NombCol : Integer;
NombPlan : Integer;
FUNCTION CoorVec(i,j,k : Integer) : Integer;
FUNCTION Ind(i,j,k : Integer) : Real;
PROCEDURE Aleatoire;
PROCEDURE Affiche;
PROCEDURE EcriPlan      (Var P1 : Integer ; Var Mat : Matrice);
PROCEDURE ExtractPlan  (Var P1 : Integer ; Var Mat : Matrice);
PROCEDURE InitDimT      (Var l,c,p : Integer);
PROCEDURE InitNombLign  (Var n : Integer);
PROCEDURE InitNombCol   (Var n : Integer);
PROCEDURE InitNombPlan  (Var n : Integer);
PROCEDURE Lec           (Var i,j,k : Integer ; Var x : Real);
PROCEDURE Permu2        (Var T2 : Tenseur);
PROCEDURE Permu3        (Var T3 : Tenseur);
PROCEDURE Saisie;
PROCEDURE Vectorialise  (Var Mat : Matrice);
End;

```

IMPLEMENTATION

```

FUNCTION Matrice.CoorVec(i,j : Integer) : Integer;
Calcule l'indice CoorVec dans un tableau à un indice correspondant à un
2-indice (i,j) d'une matrice.
BEGIN
    CoorVec:=j+NombCol*(i-1);
END;

```

```

FUNCTION Matrice.Ind(i,j : Integer) : Real;
Renvoie la valeur Mat(i,j)
BEGIN
    Ind:=Coor[CoorVec(i,j)];
END;

```

```

FUNCTION Tenseur.CoorVec(i,j,k : Integer) : Integer;
Calcule l'indice CoorVec dans un tableau à un indice correspondant à un
3-indice (i,j,k) d'un tenseur.
BEGIN
    CoorVec:=k+NombCol*(j-1)+NombLign*NombCol*(i-1);
END;

```

```

FUNCTION Tenseur.Ind(i,j,k : Integer) : Real;
Renvoie la valeur Tenseur(i,j,)
BEGIN
    Ind:=Coor[CoorVec(i,j,k)];
END;

```

```

PROCEDURE Matrice.Affiche;
Affiche à l'écran la matrice MATRICE
VAR i,j : Integer;
BEGIN
    For i:=1 to NombLign do
        Begin

```

```

        For j:=1 to NombCol do Write(Ind(i,j):6:2,' ');
        WriteLn;
    End;
    WriteLn;
END;

PROCEDURE Matrice.Aleatoire;
Crée une matrice MATRICE de dimensions données à coefficients aléatoires
compris entre 0 et 1.
    VAR i : Integer;
    BEGIN
        Randomize;
        For i:=1 to NombLign*NombCol do Coor[i]:=Random;
    END;

PROCEDURE Matrice.EcriCol(Var c : Integer ; Var X : Vecteur);
Ecrit le vecteur X à la colonne c de la matrice MATRICE
    VAR i : Integer;
        Valeur : Real;
    BEGIN
        For i:=1 to NombLign do Begin
            Valeur:=X.Coor[i];
            Lec(i,c,Valeur);
        End;
    END;

PROCEDURE Matrice.ExtractCol(Var c : Integer ; Var X : Vecteur);
Crée le vecteur X, colonne c de la matrice MATRICE
    VAR i : Integer;
    BEGIN
        For i:=1 to NombLign do X.Coor[i]:= Ind(i,c);
        X.InitDim(NombLign);
    END;

PROCEDURE Matrice.ExtractLign(Var l : Integer ; Var X : Vecteur);
Crée le vecteur X, ligne l de la matrice MATRICE
    VAR i : Integer;
    BEGIN
        For i:=1 to NombCol do X.Coor[i]:= Coor[CoorVec(l,i)];
        X.InitDim(NombCol);
    END;

PROCEDURE Matrice.InitDimM (Var l,c : Integer);
Initialise les dimensions (l,c) de la matrice MATRICE.
    l : Nombre de lignes
    c : Nombre de colonnes
    BEGIN
        InitNombLign(l);
        InitNombCol(c);
    END;

PROCEDURE Matrice.InitNombLign(Var n : Integer);
Initialise le nombre de lignes l de la matrice MATRICE
    BEGIN

```

```

    NombLign := n;
End;

PROCEDURE Matrice.InitNombCol(Var n : Integer);
Initialise le nombre de colonnes c d'une matrice MATRICE
BEGIN
    NombCol := n;
    Dim:=NombLign*NombCol;
    If Dim > nMax Then Begin
        Write('Arrêt : Dimensions trop importantes');
        Repeat Until KeyPressed;
        Exit;
    End;
End;

PROCEDURE Matrice.Inverse;
Inverse la matrice MATRICE. Le résultat est écrit en lieu et place de la
matrice initiale. Un test évalue en début de calcul si la matrice est
carrée. L'algorithme utilisé est la méthode du pivot de GAUSS, telle qu'elle
est présentée dans SCHWARTZ (1989), p 46.)
VAR i,j,k : Integer;
    h,Max,Pivot,q,s : Extended;
    P : Array[1..nMax] of Integer;
BEGIN
    If NombLign<>NombCol Then Begin
        WriteLn('Inversion impossible : matrice non carrée');
        Repeat Until KeyPressed;
        Exit;
    End;
    For k:=1 to NombLign do P[k]:=k;
    For k:=1 to NombLign do Begin
        Max:=0; P[k]:=0;
        For i:=k to NombLign do Begin
            s:=0;
            For j:=k to NombLign do s:=s+Abs(Coor[CoorVec(i,j)]);
            If s = 0 Then Begin
                Write('Matrice non inversible');
                Repeat Until KeyPressed;
                Exit;
            End;
            q:=Abs(Coor[CoorVec(i,k)]/s);
            If q>Max Then Begin
                Max:=q;
                P[k]:=i;
            End;
        End;
    End;
    If Max=0 Then Begin
        WriteLn('Processus interrompu'); WriteLn;
        Exit;
    End;
    If P[k]<>k Then Begin
        For j:=1 to NombLign do Begin
            h:=Coor[CoorVec(k,j)];
            Coor[CoorVec(k,j)]:=Coor[CoorVec(P[k],j)];
            Coor[CoorVec(P[k],j)]:=h;
        End;
    End;

```

```

End;
Pivot:=Coor[CoorVec(k,k)];
For j:=1 to NombLign do Begin
  If j<>k Then Begin
    Coor[CoorVec(k,j)]:= -Coor[CoorVec(k,j)]/Pivot;
    For i:=1 to NombLign do Begin
      If i<>k Then
        Coor[CoorVec(i,j)]:=Coor[CoorVec(i,j)]+
          Coor[CoorVec(i,k)*Coor[CoorVec(k,j)];
      End;
    End;
  End;
  For i:=1 to NombLign do Coor[CoorVec(i,k)]:=Coor[CoorVec(i,k)]/Pivot;
  Coor[CoorVec(k,k)]:=1/Pivot;
End;
For k:= NombLign-1 DownTo 1 do Begin
  If P[k]<>k Then Begin
    For i:=1 to NombLign do Begin
      h:=Coor[CoorVec(i,k)];
      Coor[CoorVec(i,k)]:=Coor[CoorVec(i,P[k])];
      Coor[CoorVec(i,P[k])]:=h;
    End;
  End;
End;
END;

```

PROCEDURE Matrice.Jacobi (Var ValPro : Vecteur ;
 Var VecPro : Matrice);
 Diagonalise la matrice MATRICE, et calcule les vecteurs propres dans la
 matrice VecPro. Les valeurs propres sont écrites dans la matrice ValPro. La
 matrice initiale est détruite dans ce processus. Le programme est extrait
 des Numerical Recipes.

CONST

np = 20;

TYPE

RealArrayNPbyNP = ARRAY [1..np,1..np] OF real;

RealArrayNP = ARRAY [1..np] OF real;

VAR

Mat,v : RealArrayNPbyNP;

i,j,n,nrot : Integer;

d : RealArrayNP;

PROCEDURE JacobiRecipes(VAR a: RealArrayNPbyNP;

n: integer;

VAR d: RealArrayNP;

VAR v: RealArrayNPbyNP;

VAR nrot: integer);

Procédure Jacobi du manuel Numerical Recipes, p 387

LABEL 99;

VAR

j,iq,ip,i: integer;

tresh,theta,tau,t,sm,s,h,g,c: real;

b,z: ^RealArrayNP;

BEGIN

new(b);

new(z);

```

FOR ip := 1 TO n DO BEGIN
  FOR iq := 1 TO n DO v[ip,iq] := 0.0;
  v[ip,ip] := 1.0
END;
FOR ip := 1 TO n DO BEGIN
  b^[ip] := a[ip,ip];
  d[ip] := b^[ip];
  z^[ip] := 0.0
END;
nrot := 0;
FOR i := 1 TO 50 DO BEGIN
  sm := 0.0;
  FOR ip := 1 TO n-1 DO
    FOR iq := ip+1 TO n DO
      sm := sm+abs(a[ip,iq]);
  IF sm = 0.0 THEN GOTO 99;
  IF i < 4 THEN tresh := 0.2*sm/sqr(n)
  ELSE tresh := 0.0;
  FOR ip := 1 TO n-1 DO BEGIN
    FOR iq := ip+1 TO n DO BEGIN
      g := 100.0*abs(a[ip,iq]);
      IF (i > 4) AND (abs(d[ip])+g = abs(d[iq]))
        AND (abs(d[iq])+g = abs(d[iq])) THEN a[ip,iq] := 0.0
      ELSE IF abs(a[ip,iq]) > tresh THEN BEGIN
        h := d[iq]-d[ip];
        IF abs(h)+g = abs(h) THEN
          t := a[ip,iq]/h
        ELSE BEGIN
          theta := 0.5*h/a[ip,iq];
          t := 1.0/(abs(theta)+sqrt(1.0+sqr(theta)));
          IF theta < 0.0 THEN t := -t
        END;
        c := 1.0/sqrt(1+sqr(t));
        s := t*c;
        tau := s/(1.0+c);
        h := t*a[ip,iq];
        z^[ip] := z^[ip]-h;
        z^[iq] := z^[iq]+h;
        d[ip] := d[ip]-h;
        d[iq] := d[iq]+h;
        a[ip,iq] := 0.0;
        FOR j := 1 TO ip-1 DO BEGIN
          g := a[j,ip];
          h := a[j,iq];
          a[j,ip] := g-s*(h+g*tau);
          a[j,iq] := h+s*(g-h*tau)
        END;
        FOR j := ip+1 TO iq-1 DO BEGIN
          g := a[ip,j];
          h := a[j,iq];
          a[ip,j] := g-s*(h+g*tau);
          a[j,iq] := h+s*(g-h*tau)
        END;
        FOR j := iq+1 TO n DO BEGIN
          g := a[ip,j];
          h := a[iq,j];

```

```

        a[ip,j] := g-s*(h+g*tau);
        a[iq,j] := h+s*(g-h*tau)
    END;
    FOR j := 1 TO n DO BEGIN
        g := v[j,ip];
        h := v[j,iq];
        v[j,ip] := g-s*(h+g*tau);
        v[j,iq] := h+s*(g-h*tau)
    END;
    nrot := nrot+1
END
END
END;
FOR ip := 1 TO n DO BEGIN
    b^[ip] := b^[ip]+z^[ip];
    d[ip] := b^[ip];
    z^[ip] := 0.0
END
END;
writeln('pause in routine JACOBI');
writeln('50 iterations should not happen');
readln;
99:
    dispose(z);
    dispose(b);
END;    de la procédure JACOBI de Numerical Recipes

PROCEDURE eigsort(VAR d: RealArrayNP;
                  VAR v: RealArrayNPbyNP;
                  n: integer);

Tri des valeurs propres dans Numerical Recipes, p. 388

VAR
    k,j,i: integer;
    p: real;
BEGIN
    FOR i := 1 TO n-1 DO BEGIN
        k := i;
        p := d[i];
        FOR j := i+1 TO n DO BEGIN
            IF d[j] >= p THEN BEGIN
                k := j;
                p := d[j]
            END
        END;
        IF k <> i THEN BEGIN
            d[k] := d[i];
            d[i] := p;
            FOR j := 1 TO n DO BEGIN
                p := v[j,i];
                v[j,i] := v[j,k];
                v[j,k] := p
            END
        END
    END
END;
END;
END;

```



```

BEGIN de la procédure JACOBI
  If NombLign<>NombCol Then Begin
    Write('Matrice non carrée ; Diagonalisation inadéquate');
    Exit;
  End;
  n:= NombLign;
  For i:=1 to n do Begin
    For j:=1 to n do Mat[i,j]:=Ind(i,j);
  End;
  JacobiRecipes(Mat,n,d,v,nrot);
  EigSrt(d,v,n);
  ValPro.InitDim(n);
  For i:= 1 to n do ValPro.Coor[i]:= d[i];
  VecPro.InitNombLign(n);
  VecPro.InitNombCol(n);
  For i:=1 to n do Begin
    For j:=1 to n do VecPro.Lec(i,j,v[i,j]);
  End;
END;

```

```

PROCEDURE Matrice.Lec(Var i,j : Integer ; Var x : Real);
Ecrit la valeur x en MATRICE(i,j)
BEGIN
  Coor[CoorVec(i,j)]:=x;
END;

```

```

PROCEDURE Matrice.NormaCol;
Remplace la matrice MATRICE par une matrice de mêmes dimensions dont chaque
colonne C est la colonne C de MATRICE, normalisée.
VAR i : Integer;
    Zcol : ^Vecteur;
BEGIN
  New(Zcol);
  Zcol^.InitDim(NombLign);
  For i:=1 to NombCol do Begin
    ExtractCol(i,Zcol^);
    Zcol^.Normalisation;
    EcriCol(i,Zcol^);
  End;
  Dispose(Zcol);
END;

```

```

PROCEDURE Matrice.Saisie;
Saisie la matrice MATRICE à l'écran
VAR i,j : Integer;
BEGIN
  For i:=1 to NombLign do
    Begin
      For j:=1 to NombCol do
        Begin
          Write(i,j : 4, ' ');

```

```

        Write('Coefficient = : ');
        ReadLn(Coor[CoorVec(i,j)]);
    End;
    WriteLn;
End;
END;

PROCEDURE Matrice.Transpose(Var Mat : Matrice);
Transpose la matrice MATRICE. Ecrit le résultat dans une matrice MAT.
    VAR i,j : Integer;
        Valeur : Real;
    BEGIN
        Mat.InitNombLign(NombCol);
        Mat.InitNombCol(NombLign);
        For i:=1 to Mat.NombLign do Begin
            For j:=1 to Mat.NombCol do Begin
                Valeur:=Ind(j,i);
                Mat.Lec(i,j,Valeur);
            End;
        End;
    End;
END;

PROCEDURE Tenseur.Affiche;
Affiche le tenseur TENSEUR à l'écran
    VAR i : Integer;
        M : ^Matrice;
    BEGIN
        New(M);
        For i:=1 to NombPlan do Begin
            ExtractPlan(i,M^);
            M^.Affiche;
            WriteLn;
        End;
        Dispose(M);
    END;

PROCEDURE Tenseur.Aleatoire;
Crée un tenseur TENSEUR de dimensions données, de coefficients aléatoires
compris entre 0 et 1.
    VAR i : Integer;
    BEGIN
        Randomize;
        For i:=1 to NombLign*NombCol*NombPlan do Coor[i]:=Random;
    END;

PROCEDURE Tenseur.EcriPlan (Var Pl : Integer ; Var Mat : Matrice);
Ecrit la matrice MAT au plan Pl du tenseur TENSEUR.
    VAR i,j : Integer;
        Valeur : Real;
    BEGIN
        If NombLign <> Mat.NombLign Then Begin
            Write('Ecriture impossible : lignes incompatibles');
            Repeat Until KeyPressed;
            Exit;
        End;
    END;

```

```

    If NombCol <> Mat.NombCol Then Begin
        Write('Ecriture impossible : colonnes incompatibles');
        Repeat Until KeyPressed;
        Exit;
    End;
    For i:=1 to NombLign do Begin
        For j:=1 to NombCol do Begin
            Valeur := Mat.Ind(i,j);
            Lec(Pl,i,j,Valeur);
        End;
    End;
End;
END;

PROCEDURE Tenseur.ExtractPlan(Var Pl : Integer ; Var Mat : Matrice);
Crée une matrice Mat qui est le plan Pl du tenseur TENSEUR
VAR i,j : Integer;
    Valeur : Real;
BEGIN
    Mat.InitNombLign(NombLign);
    Mat.InitNombCol(NombCol);
    For i:=1 to NombLign do begin
        For j:=1 to NombCol do Begin
            Valeur := Coor[CoorVec(Pl,i,j)];
            Mat.Lec(i,j,Valeur);
        End;
    End;
END;

PROCEDURE Tenseur.InitDimT (Var l,c,p : Integer);
Initialise les dimensions d'un tenseur TENSEUR
l : Nombre de lignes
c : Nombre de colonnes
p : Nombre de plans
BEGIN
    InitNombLign(l);
    InitNombCol(c);
    InitNombPlan(p);
END;

PROCEDURE Tenseur.InitNombLign (Var n : Integer);
Initialise le nombre n de lignes du tenseur TENSEUR
BEGIN
    NombLign:=n;
END;

PROCEDURE Tenseur.InitNombCol (Var n : Integer);
Initialise le nombre n de colonnes du tenseur TENSEUR
BEGIN
    NombCol:=n;
END;

PROCEDURE Tenseur.InitNombPlan (Var n : Integer);
Initialise le nombre n de plans de TENSEUR
BEGIN
    NombPlan:=n;

```

```

    Dim := NombLign*NombCol*NombPlan;
END;

```

```

PROCEDURE Tenseur.Lec(Var i,j,k : Integer ; Var x : Real);
Fournit le réel x, valeur de TENSEUR(i,j,k)
BEGIN
    Coor[CoorVec(i,j,k)] := x;
END;

```

```

PROCEDURE Tenseur.Permu2(Var T2 : Tenseur);

```

```

    VAR i,j,k : Integer;
        Valeur : Real;
BEGIN
    T2.InitNombLign(NombPlan);
    T2.InitNombCol(NombLign);
    T2.InitNombPlan(NombCol);
    For k:=1 to T2.NombPlan do Begin
        For i:=1 to T2.NombLign do Begin
            For j:=1 to T2.NombCol do Begin
                Valeur := Ind(i,j,k);
                T2.Lec(k,i,j,Valeur);
            End;
        End;
    End;
END;

```

```

PROCEDURE Tenseur.Permu3(Var T3 : Tenseur);

```

```

Si TENSEUR ∈ E1×E2×E3, crée le tenseur T3 permuté de TENSEUR, dans E3×E1×E2.
VAR i,j,k : Integer;
    Valeur : Real;
BEGIN
    T3.InitNombLign(NombCol);
    T3.InitNombCol(NombPlan);
    T3.InitNombPlan(NombLign);
    For j:=1 to T3.NombPlan do Begin
        For k:=1 to T3.NombLign do Begin
            For i:=1 to T3.NombCol do Begin
                Valeur := Ind(i,j,k);
                T3.Lec(j,k,i,Valeur);
            End;
        End;
    End;
END;

```

```

PROCEDURE Tenseur.Saisie;

```

```

Saisie un tenseur à l'écran
VAR i,j,k : Integer;
    Valeur : Real;
BEGIN
    ClrScr;
    For i:=1 to NombPlan do Begin
        Write('Plan ');WriteLn(i);
        For j:=1 to NombLign do Begin
            For k := 1 to NombCol do Begin

```

```

        Write(' ');Write(j);Write(' ');Write(k);Write(' ');
        ReadLn(Valeur);
        Lec(i,j,k,Valeur);
    End;
    WriteLn;
End;
End;
END;

```

PROCEDURE Tenseur.Vectorialise(Var Mat : Matrice);

Etant donné un tenseur de $E1 \times E2 \times E3$, crée une matrice MAT de $E1 \times (E2 \times E3)$ dont la colonne c est le plan c de TENSEUR, vectorialisé.

```

    VAR i,D : Integer;
        Vec : ^Vecteur;
        MatAux : ^Matrice;
    BEGIN
        New(Vec);New(MatAux);
        D:=NombLign*NombCol;
        Mat.InitDimM(D,NombPlan);
        Vec^.InitDim(D);
        For i:=1 to Mat.NombCol do Begin
            ExtractPlan(i,MatAux^);
            Vec^.Coor:=MatAux^.Coor;
            Mat.EcriCol(i,Vec^);
        End;
        Dispose(Vec);Dispose(MatAux);
    END;

```

PROCEDURE Vecteur.InitNul;

Crée un vecteur dont toutes les coordonnées sont nulles

```

    VAR i : Integer;
    BEGIN
        For i:=1 to Dim do Coor[i]:=0;
    END;

```

FUNCTION Vecteur.Norme : Real;

Calcule la norme d'un vecteur

```

    VAR
        i : Integer;
        Valeur : Extended;
    BEGIN
        Valeur := 0;
        For i := 1 to Dim do Valeur:=Valeur+Sqr(Coor[i]);
        Norme := Sqrt(Valeur);
    END;

```

PROCEDURE Vecteur.Affiche;

Affiche à l'écran un vecteur

```

    VAR i : Integer;
    BEGIN
        For i:=1 to Dim do Write (Coor[i]:6:2 , ' ');
        WriteLn;
    END;

```

PROCEDURE Vecteur.Aleatoire;

Crée un vecteur aléatoire de dimensions données, dont les coordonnées sont des réels compris entre 0 et 1.

```
VAR i : Integer;  
BEGIN  
    Randomize;  
    For i:=1 to Dim do Coor[i]:=Random;  
END;
```

PROCEDURE Vecteur.InitDim(Var n : Integer);

Initialise la dimension n d'un vecteur

```
BEGIN  
    Dim := n;  
END;
```

PROCEDURE Vecteur.Normalisation;

Normalise un vecteur

```
VAR Norm : Real;  
    i : Integer;  
BEGIN  
    Norm := Norme;  
    For i:=1 to Dim do Coor[i]:=Coor[i]/Norm;  
END;
```

PROCEDURE Vecteur.Saisie;

Saisie un vecteur à l'écran

```
VAR i : Integer;  
BEGIN  
    For i:=1 to Dim do Begin  
        Write('Coefficient = : '); ReadLn(Coor[i]);  
    End;  
END;  
END.
```

UNIT ALGELINE;

Regroupe les principales fonctions de l'algèbre linéaire

INTERFACE

USES Crt, Declare;

```
FUNCTION ProScal (V1,V2 : Vecteur) : Real;
FUNCTION ProScalMat (M1,M2 : Matrice) : Real;
FUNCTION ProscalTens (T1,T2 : Tenseur) : Real;
PROCEDURE CombiLine (Var a,b : Real ; Var X,Y,V : Vecteur);
PROCEDURE CombiLineMat (Var x,y : Real ; Var A,B,Mat : Matrice);
PROCEDURE ProduitMat (Var A,B,Mat : Matrice);
PROCEDURE ProdTensEtMat (Var Tens : Tenseur ; Var Mat : Matrice;
    Var Vec : Vecteur);
PROCEDURE ProdTensEtVec (Var Tens : Tenseur ; Var Vec : Vecteur ;
    Var Mat : Matrice);
PROCEDURE ProdMatEtVec (Var Mat : Matrice ; Var Vec1, Vec : Vecteur);
PROCEDURE ProReelEtVec (Var x : Real ; Var V1,V2 : Vecteur);
PROCEDURE ProTens2V (Var X,Y : Vecteur ; Var Mat : Matrice);
PROCEDURE ProTens3V (Var V1,V2,V3 : Vecteur ; Var T : Tenseur);
PROCEDURE Somme (Var X,Y,V : Vecteur);
```

IMPLEMENTATION

FUNCTION ProScalMat(M1,M2 : Matrice) : Real;

Effectue le produit scalaire des deux matrices M1 & M2

```
VAR i : Integer;
    Valeur : Real;
BEGIN
    If M1.NombLign<>M2.NombLign Then Begin
        Write('Produit scalaire impossible : lignes incompatibles');
        ReadLn;
        Exit;
    End;
    If M1.NombCol<>M2.NombCol Then Begin
        Write('Produit scalaire impossible : Colonnes incompatibles');
        ReadLn;
        Exit;
    End;
    Valeur:=0;
    For i:=1 to M1.NombLign*M1.NombCol do
        Valeur:=Valeur+M1.Coor[i]*M2.Coor[i];
    ProScalMat:=Valeur;
END;
```

FUNCTION ProscalTens(T1,T2 : Tenseur) : Real;

Effectue le produit scalaire des deux tenseurs T1 & T2

```
VAR i,j,k : Integer;
    Valeur : Real;
BEGIN
    If T1.NombPlan<>T2.NombPlan Then Begin
```

```

    WriteLn('Produit Scalaire Impossible : Plans Incompatibles');
    Exit;
End;
If T1.NombLign<>T2.NombLign Then Begin
    WriteLn('Produit Scalaire Impossible : Lignes Incompatibles');
    Exit;
End;
If T1.NombCol<>T2.NombCol Then Begin
    WriteLn('Produit Scalaire Impossible : Colonnes Incompatibles');
    Exit;
End;
Valeur := 0;
For i:=1 to T1.NombPlan do Begin
    For j:= 1 to T1.NombLign do begin
        For k:=1 to T1.NombCol do Valeur :=
Valeur+T1.Ind(i,j,k)*T2.Ind(i,j,k);      End;
    End;
    ProScalTens:= Valeur;
END;

```

```

FUNCTION ProScal(V1,V2 : Vecteur) : Real;
Effectue le produit scalaire des deux vecteurs V1 & V2
VAR i      : Integer;
    Valeur : Extended;
BEGIN
    If V1.Dim<>V2.Dim Then Begin
        Write('Produit scalaire impossible : dimensions incompatibles');
        ReadLn;
        Exit;
    End;
    Valeur:=0;
    For i:=1 to V1.Dim do Valeur:=Valeur+V1.Coor[i]*V2.Coor[i];
    ProScal:=Valeur;
END;

```

```

PROCEDURE CombiLine(Var a,b : Real ; Var X,Y,V : Vecteur);
Calcule le vecteur V = aX+bY
VAR i : Integer;
BEGIN
    If X.Dim<>Y.Dim Then Begin
        Write('Combinaison impossible : les vecteurs n''ont pas la même dimension');
        Repeat Until KeyPressed;
        Exit;
    End;
    V.InitDim(X.Dim);
    For i:=1 to X.Dim do V.Coor[i]:=a*X.Coor[i]+b*Y.Coor[i]
END;

```

```

PROCEDURE CombiLineMat(Var x,y : Real;
                        Var A,B,Mat : Matrice);
Calcule la matrice M = xA+yB
VAR i : Integer;
BEGIN
    If A.NombLign <> B.NombLign Then Begin

```



```

    WriteLn('Combinaison impossible : Dimensions en Lignes incompatibles');
    Repeat Until KeyPressed;
    Exit;
End;
If A.NombCol<>B.NombCol Then Begin
    WriteLn('Combinaison impossible : Dimensions en Colonnes incompatibles');
    Repeat Until KeyPressed;
    Exit;
End;
Mat.InitNombLign(A.NombLign);
Mat.InitNombCol(A.NombCol);
For i:=1 to A.NombCol*A.NombCol do Mat.Coor[i]:=x*A.Coor[i]+y*B.Coor[i];
END;

```

PROCEDURE ProduitMat(Var A,B,Mat : Matrice);

Calcule la matrice Mat = AB

VAR

i,j : Integer;

Valeur : Real;

VecA,VecB : ^Vecteur;

LABEL 99;

BEGIN

New(VecA);New(VecB);

If A.NombCol<>B.NombLign Then Begin

WriteLn('Produit impossible : dimensions incompatibles');

Repeat Until KeyPressed;

Goto 99;

End;

VecA^.InitDim(A.NombCol); VecB^.InitDim(B.NombLign);

Mat.InitNombLign(A.NombLign); Mat.InitNombCol(B.NombCol);

For i:=1 to Mat.NombLign **do** **Begin**

A.ExtractLign(i,VecA^);

For j:=1 to Mat.NombCol **do** **Begin**

B.ExtractCol(j,VecB^);

Valeur := ProScal(VecA^,VecB^);

Mat.Lec(i,j,Valeur);

End;

End;

99: Dispose(VecA);Dispose(VecB);

END;

PROCEDURE ProdTensEtMat(Var Tens : Tenseur ; Var Mat : Matrice;

Var Vec : Vecteur);

Calcule le vecteur Vec = Tens.Mat

VAR i,j,k : Integer;

Valeur : Real;

BEGIN

If Tens.NombPlan<>Mat.NombCol Then Begin

Write('ProdTens&Mat : Produit impossible : dimensions incompatibles');

Repeat Until KeyPressed;

Exit;

End;

If Tens.NombCol<>Mat.NombLign Then Begin

Write('ProdTens&Mat : Produit impossible : dimensions incompatibles');

```

    Repeat Until KeyPressed;
    Exit;
End;
Vec.InitDim(Tens.NombLign);
For k:=1 to Tens.NombLign do Begin
    Valeur:=0;
    For i:=1 to Tens.NombPlan do Begin
        For j:=1 to Tens.NombCol do
            Valeur:=Valeur+Tens.Ind(i,k,j)*Mat.Ind(j,i);      End;
        Vec.Coor[k]:=Valeur;
    End;
End;
END;

PROCEDURE ProdTensEtVec (Var Tens : Tenseur ; Var Vec : Vecteur ;
                        Var Mat : Matrice);
Calculer la matrice Mat = Tens.Vec
VAR i,j,k : Integer;
    Valeur : Real;
BEGIN
    If Vec.Dim <> Tens.NombPlan Then Begin
        Write('Produit impossible : dimensions incompatibles');
        Repeat Until KeyPressed;
        Exit;
    End;
    Mat.InitDimM(Tens.NombLign, Tens.NombCol);
    Mat.InitNul;
    For i:= 1 to Mat.NombLign do Begin
        For j:=1 to Mat.NombCol do Begin
            Valeur:=0;
            For k:=1 to Tens.NombPlan do
                Valeur:=Valeur+Vec.Coor[k]*Tens.Coor[Tens.C    Mat.Lec(i,j,Valeur);
            End;
        End;
    End;
END;

PROCEDURE ProdMatEtVec(Var Mat : Matrice ; Var Vec1, Vec : Vecteur);
Calculer le vecteur Vec = Mat.Vec1
VAR i,j : Integer;
    V : ^Vecteur;
BEGIN
    New(V);
    If Vec1.Dim <> Mat.NombCol then Begin
        Write ('Produit impossible car dimensions non compatibles');
        Repeat Until KeyPressed;
        Exit;
    End;
    Vec.InitDim(Mat.NombLign);
    For i:=1 to Mat.NombLign do Begin
        Mat.ExtractLign(i,V^);
        Vec.Coor[i]:= ProScal(V^,Vec1);
    End;
    Dispose(V);
END;

```

```

PROCEDURE ProReelEtVec(Var x : Real ; Var V1,V2 : Vecteur);
Calcule le vecteur V2 = xV1
  VAR i : Integer;
  BEGIN
    V2.InitDim(V1.Dim);
    For i:=1 to V2.Dim do V2.Coor[i]:=x*V1.Coor[i];
  END;

```

```

PROCEDURE ProTens2V(Var X,Y : Vecteur ; Var Mat : Matrice);
Calcule la matrice Mat = x@y
  Var i,j : Integer;
  BEGIN
    Mat.InitNombLign(Y.Dim);
    Mat.InitNombCol(X.Dim);
    For i:=1 to Mat.NombLign do Begin
      For j:=1 to Mat.NombCol do Begin
        Mat.Coor[Mat.CoorVec(i,j)]:=X.Coor[j]*Y.Coor[i]
      End;
    End;
  END;

```

```

PROCEDURE ProTens3V(Var V1,V2,V3 : Vecteur ; Var T : Tenseur);
Calcule le tenseur T = V1@V2@V3
  VAR i,j,k : Integer;
  R1,R2,Valeur : Real;
  BEGIN
    T.InitNombLign(V3.Dim);
    T.InitNombCol(V2.Dim);
    T.InitNombPlan(V1.Dim);
    For i:=1 to T.NombPlan do Begin
      R1:=V1.Coor[i];
      For j:= 1 to T.NombLign do Begin
        R2:=R1*(V3.Coor[j]);
        For k:=1 to T.NombCol do Begin
          Valeur:=R2*(V2.Coor[k]);
          T.Lec(i,j,k,Valeur);
        End;
      End;
    End;
  END;

```

```

PROCEDURE SommeMat(Var A,B,Mat : Matrice);
Calcule la matrice Mat = A+B
  Var i : Integer;
  BEGIN
    If A.NombLign <> B.NombLign Then Begin
      WriteLn('Somme impossible : Dimensions en Lignes incompatibles');
      Repeat Until KeyPressed;
      Exit;
    End;
    If A.NombCol <> B.NombCol Then Begin
      WriteLn('Somme impossible : Dimensions en Colonnes incompatibles');
      Repeat Until KeyPressed;
      Exit;
    End;
  END;

```

```

Mat.InitNombLign(A.NombLign);
Mat.InitNombCol(A.NombCol);
For i:=1 to A.NombCol*A.NombCol do Mat.Coor[i]:=A.Coor[i]+B.Coor[i];
END;

```

```

PROCEDURE Somme(Var X,Y,V : Vecteur);
Calcule le vecteur V = x+y
VAR i : Integer;
BEGIN
  If X.Dim<>Y.Dim Then Begin
    Write('Somme impossible : les vecteurs n''ont pas la même dimension');
    Repeat Until KeyPressed;
    Exit;
  End;
  V.InitDim(X.Dim);
  For i:=1 to X.Dim do V.Coor[i]:=X.Coor[i]+Y.Coor[i]
END;

END.

```

UNIT Usuels;

Regroupe les utilitaires les plus usuels

INTERFACE

USES Crt, Declare, Algeline;

FUNCTION Distance (Var V1,V2 : Vecteur) : Real;

PROCEDURE Dual (Var M,Res : Matrice);

PROCEDURE DualMat (Var T,Res : Tenseur);

PROCEDURE InitAleaMat (Var Mat : Matrice);

PROCEDURE InitAleaTens (Var Tens : Tenseur);

PROCEDURE InitAleaVec (Var Vec : Vecteur);

PROCEDURE InitSaisieMat (Var Mat : Matrice);

PROCEDURE InitSaisieTens (Var Tens : Tenseur);

PROCEDURE InitSaisieVec (Var Vec : Vecteur);

PROCEDURE MatCorr (Var M,Res : Matrice);

PROCEDURE Pause;

IMPLEMENTATION

FUNCTION Distance(Var V1,V2 : Vecteur) : Real;

Calcule la distance entre les vecteurs V1 et V2 : $\|V1-V2\|$

VAR r1,r2 : Real;

Vec : ^Vecteur;

BEGIN

New(Vec);

r1:=1;r2:=-1;

Combiline(r1,r2,V1,V2,Vec^);

Distance := Vec^.Norme;

Dispose(Vec);

END;

PROCEDURE Dual(Var M,Res : Matrice);

Calcule la matrice Res dont les colonnes forment un système dual du système des colonnes de M

VAR M2 : ^Matrice;

BEGIN

Res.InitNombLign(M.NombLign); Res.InitNombCol(M.NombCol);

New(M2);

MatCorr(M,M2^);

M2^.Inverse;

ProduitMat(M,M2^,Res);

Dispose(M2);

END;

PROCEDURE DualMat(Var T,Res : Tenseur);

Etant donné un tenseur T dont les plans sont les matrices T(i), calcule un tenseur Res de même dimension dont les tranches Res(i) sont les matrices duales des matrices T(i). C'est à dire que le système Res(i) est le système dual du système T(i) dans l'espace vectoriel des matrices.

VAR i,D : Integer;

MatAux,MatAuxDual,MatPlan : ^Matrice;

```

    VecAux : ^Vecteur;
BEGIN
    New(MatAux);New(MatPlan);
    New(VecAux);
    D:=T.NombLign*T.NombCol;
    Res.InitDimT(T.NombLign,T.NombCol,T.NombPlan);
    MatAux^.InitNombLign(D);MatAux^.InitNombCol(T.NombPlan);
    VecAux^.InitDim(D);
    For i:=1 to T.NombPlan do Begin
        T.ExtractPlan(i,MatPlan^);
        VecAux^.Coor:=MatPlan^.Coor;
        MatAux^.EcriCol(i,VecAux^);
    End;
    Dispose(VecAux);Dispose(MatPlan);
    New(MatAuxDual);New(VecAux);New(MatPlan);
    MatPlan^.InitDimM(Res.NombLign,Res.NombCol);
    VecAux^.InitDim(MatAux^.NombLign);
    Dual(MatAux^,MatAuxDual^);
    For i:=1 to Res.NombPlan do Begin
        MatAuxDual^.ExtractCol(i,VecAux^);
        MatPlan^.Coor := VecAux^.Coor;
        Res.EcriPlan(i,MatPlan^);
    End;
    Dispose(MatAux);Dispose(MatAuxDual);Dispose(VecAux);Dispose(MatPlan);
END;
```

PROCEDURE InitAleaMat(Var Mat : Matrice);

Crée une matrice de dimensions (m,n) à coefficients aléatoires compris entre 0 et 1.

```

    VAR m,n : Integer;
    BEGIN
        Write('Nombre de Lignes = ');ReadLn(m);
        Write('Nombre de Colonnes = ');ReadLn(n);
        Mat.InitNombLign(m);
        Mat.InitNombCol(n);
        Mat.Aleatoire;
    END;
```

PROCEDURE InitSaisieMat(Var Mat : Matrice);

Permet la saisie à l'écran d'une matrice de dimensions (m,n).

```

    VAR m,n : Integer;
    BEGIN
        Write('Nombre de Lignes = '); ReadLn(m);
        Write('Nombre de Colonnes = ');ReadLn(n);
        Mat.InitNombLign(m);
        Mat.InitNombCol(n);
        Mat.Saisie;
    END;
```

PROCEDURE InitSaisieTens(Var Tens : Tenseur);

Permet la saisie à l'écran d'un tenseur de dimensions (m,n,p).

```

    VAR m,n,p : Integer;
    BEGIN
        Write('Nombre de Lignes = '); ReadLn(m);
        Write('Nombre de Colonnes = ');ReadLn(n);
        Write('Nombre de Plans = ');ReadLn(p);
```

```

Tens.InitNombLign(m);
Tens.InitNombCol(n);
Tens.InitNombPlan(p);
Tens.Saisie;
END;

```

PROCEDURE InitAleaTens(Var Tens : Tenseur);

Crée un tenseur de dimensions (m,n,p) à coefficients aléatoires compris entre 0 et 1.

```

VAR m,n,p,Dim : Integer;
BEGIN
  Write('Nombre de Lignes = '); ReadLn(m);
  Write('Nombre de Colonnes = '); ReadLn(n);
  Write('Nombre de Plans = '); ReadLn(p);
  Tens.InitNombLign(m);
  Tens.InitNombCol(n);
  Tens.InitNombPlan(p);
  Tens.Aleatoire;
END;

```

PROCEDURE InitAleaVec(Var Vec : Vecteur);

Crée un vecteur de dimensions (n) à coefficients aléatoires compris entre 0 et 1.

```

VAR n : Integer;
BEGIN
  Write('Dimension = '); ReadLn(n);
  Vec.InitDim(n);
  Vec.Aleatoire;
END;

```

PROCEDURE InitSaisieVec(Var Vec : Vecteur);

Permet la saisie à l'écran d'un vecteur de dimensions (n).

```

VAR n : Integer;
BEGIN
  Write('Dimension = '); ReadLn(n);
  Vec.InitDim(n);
  Vec.Saisie;
END;

```

PROCEDURE MatCorr(Var M,Res : Matrice);

Etant donnée une matrice M, calcule la matrice Res = M'M.

```

VAR i,j : Integer;
    Valeur : Real;
    L,C : ^Vecteur;
BEGIN
  New(L); New(C);
  Res.InitNombLign(M.NombCol);
  Res.InitNombCol(M.NombCol);
  For i:=1 to Res.NombLign do Begin
    For j:=1 to Res.NombCol do Begin
      M.ExtractCol(i,C^);
      M.ExtractCol(j,L^);
      Valeur:=Proscal(L^,C^);
      Res.Lec(i,j,Valeur);
    End;
  End;

```

```
        Res.Lec(j,i,Valeur);  
    End;  
    End;  
    Dispose(L); Dispose(C);  
END;
```

PROCEDURE PAUSE;

Permet une pause jusqu'à frappe d'une touche quelconque.

```
    Var ch : Char;
```

BEGIN

```
    Repeat Until KeyPressed;
```

```
    Ch:=ReadKey;
```

END;

END.

UNIT ANAFAC;

Regroupe les principales procédures de l'analyse factorielle

INTERFACE

USES Crt , Declare, Algeline, Usuels ;

FUNCTION PluGranValPro(Var M : Matrice) : Real;

PROCEDURE ACP(Var MatDon : Matrice;
Var VecCoef : Vecteur;
Var MatCompX, MatCompY : Matrice) ;

PROCEDURE ReconStruACP(Var CompX, CompY : matrice;
Var Coef : Vecteur;
Var Mat : Matrice);

IMPLEMENTATION

FUNCTION PluGranValPro(Var M : Matrice) : Real;

Calcule la plus grande valeur propre d'une matrice symétrique M

VAR V : ^Vecteur;
M1 : ^Matrice;

BEGIN

New(V);New(M1);
M.Jacobi(V^,M1^);
PluGranValPro:=V^.Coor[1];
Dispose(V);Dispose(M1);

END;

PROCEDURE ACP(Var MatDon : Matrice;
Var VecCoef : Vecteur;
Var MatCompX, MatCompY : Matrice);

Réalise l'ACP de la matrice MatDon sous la forme $\sum L_i X_i \otimes Y_i$ -

Les coefficients L_i sont rangés dans le vecteur VecCoef -

Les vecteurs X_i la matrice MatCompX -

Les vecteurs Y_i la matrice MatCompY -}

VAR MatCo : ^Matrice;
V1,V2 : ^Vecteur;
ValPro : ^Vecteur;
VecPro : ^Matrice;
Col,j : Integer;
Valeur : Real;

PROCEDURE AcpInit;

BEGIN

New(MatCo);
New(V1);
New(V2);
New(ValPro);
New(VecPro);

END;

```

PROCEDURE AcpInitDim;
BEGIN
  VecCoef.InitDim(MatDon.NombCol);
  MatCompX.InitDimM(MatDon.NombCol , MatDon.NombCol);
  MatCompY.InitDimM(MatDon.NombLign , MatDon.NombCol);
END;

PROCEDURE AcpCalcul;

PROCEDURE AcpCalculCompX;
BEGIN
  MatCo^.Jacobi(ValPro^,MatCompX);
END;

PROCEDURE AcpCalculCompY;
VAR Col,j : Integer;
BEGIN
  V1^.InitDim(MatDon.NombCol);
  For Col:=1 to MatDon.NombCol do Begin
    MatCompX.ExtractCol(Col,V1^);
    ProdMatEtVec(MatDon,V1^,V2^);
    V2^.Normalisation;
    For j:=1 to MatDon.NombLign do Begin
      Valeur:=V2^.Coor[j];
      MatCompY.Lec(j,Col,Valeur);
    End;
  End;
END;

PROCEDURE AcpCalculCoef;
VAR j : Integer;
BEGIN
  For j:=1 to MatDon.NombCol do
    VecCoef.Coor[j] := Sqrt(ValPro^.Coor[j]);
  END;

BEGIN
  MatCorr(MatDon,MatCo^);          Calcul de MatCo = MatDon' * MatDon
  AcpCalculCompX;                  Calcul des composantes X(1)
  AcpCalculCompY;                  Calcul des composantes Y(1)
  AcpCalculCoef;                   Calcul des coefficients L(1)
END;

PROCEDURE AcpDone;
BEGIN
  Dispose(VecPro);
  Dispose(ValPro);
  Dispose(V2);
  Dispose(V1);
  Dispose(MatCo);
END;

BEGIN
  AcpInit;          Initialise les pointeurs
  AcpInitDim;       Initialise la dimension des tableaux
  AcpCalcul;         Calcul

```

```

    AcpDone;
    Ferme les pointeurs
END;

```

```

PROCEDURE ReconStruACP(Var CompX, CompY : matrice;
    Var Coef : Vecteur;
    Var Mat : Matrice);
Reconstitue la matrice de l'ACP :  $Mat = \sum Coeff(i) X(i) \otimes Y(i)$ 
VAR i : Integer;
    Valeur : Real;
    MatAux : ^Matrice;
    VecX, VecY : ^Vecteur;
BEGIN
    New(MatAux);
    New(VecX); New(VecY);
    MatAux^.InitNombLign(CompY.NombLign);
    MatAux^.InitNombCol(CompX.NombLign);
    MatAux^.InitNul;
    Mat.InitNombLign(CompY.NombLign);
    Mat.InitNombCol(CompX.NombLign);
    Mat.InitNul;
    For i:=1 to CompX.NombCol do Begin
        Valeur:=Coef.Coor[i];
        CompX.ExtractCol(i,VecX^);
        CompY.ExtractCol(i,VecY^);
        ProTens2V(VecX^,VecY^,MatAux^);
        ProReelEtVec(Valeur,MatAux^,MatAux^);
        Somme(MatAux^,Mat,Mat);
    End;

    Dispose(MatAux);
    Dispose(VecX); Dispose(VecY);
END;

END.

```

UNIT ANATENS;

Regroupe les principales procédures de l'analyse factorielle sur tenseurs d'ordre 3

INTERFACE

USES Crt , Declare, Algeline, Usuels, AnaFac;

CONST Epsi =0.00000001; précision seuil pour les tests d'arrêt

PROCEDURE Acp3Mode (Var T : Tenseur;
Var L : Vecteur;
Var MVec : Matrice;
Var TMat : Tenseur);

PROCEDURE ConstrParaFac(Var X,Y,Z : Matrice; Var Tens : Tenseur);

PROCEDURE PARAFAC(Var T : Tenseur;
Var r : Integer;
Var X,Y,Z : Matrice);

PROCEDURE PremParaFac(Var T : Tenseur;
Var Coef : Real;
Var X,Y,Z : Vecteur);

IMPLEMENTATION

PROCEDURE Acp3Mode(Var T : Tenseur;
Var L : Vecteur;
Var MVec : Matrice;
Var TMat : Tenseur);

Réalise l'ACP 3 Mode du Tenseur T sous forme $\sum_i (\lambda_i \text{ Vi} \otimes \text{Mi})$ -

Les coefficients λ_i sont rangés dans le vecteur L -
Les vecteurs Vi la matrice MVec -
Les matrices Mi le tenseur TMat -

VAR MatAux : ^Matrice;
VecX,VecY : ^Matrice;

PROCEDURE Acp3ModeInit;
BEGIN
New(MatAux);
New(VecX);
New(VecY);
END;

PROCEDURE Acp3ModeCalcul;

PROCEDURE Acp3ModeCalculResultat;
BEGIN
END;

```

BEGIN
  T.Vectorialise(MatAux^);
  ACP(MatAux^,L,MVec,VecY^);
  Acp3ModeCalculResultat;
END;

PROCEDURE Acp3ModeDone;
BEGIN
  Dispose(VecY);
  Dispose(VecX);
  Dispose(MatAux);
END;

BEGIN
  Acp3ModeInit;
  Acp3ModeCalcul;
  Acp3ModeDone;
END;

PROCEDURE ConstrParaFac(Var X,Y,Z : Matrice; Var Tens : Tenseur);
  Calcule le tenseur  $T = \sum_i X(i) \otimes Y(i) \otimes Z(i)$ , où  $X(i)$ ,  $Y(i)$ ,  $Z(i)$  sont les
  vecteurs colonnes des matrices X, Y, Z.
  VAR i : Integer;
  Xi,Yi,Zi : ^Vecteur;
  T2,T3 : ^Tenseur;
BEGIN
  New(Xi);New(Yi);New(Zi);
  New(T2);New(T3);
  If X.NombCol <> Y.NombCol Then Begin
    Write('Composition impossible : dimensions incompatibles');
    Exit;
  End;
  If X.NombCol <> Z.NombCol Then Begin
    Write('Composition impossible : dimensions incompatibles');
    Exit;
  End;
  Tens.InitDimT(Y.NombLign,Z.NombLign,X.NombLign);
  T2^.InitDimT(Y.NombLign,Z.NombLign,X.NombLign);
  T3^.InitDimT(Y.NombLign,Z.NombLign,X.NombLign);
  T2^.InitNul;
  For i:=1 to X.NombCol do Begin
    X.ExtractCol(i,Xi^);Y.ExtractCol(i,Yi^);Z.ExtractCol(i,Zi^);
    ProTens3V(Xi^,Yi^,Zi^,T3^);
    Somme(T2^,T3^,T2^);
  End;
  Tens:=T2^;
  Dispose(Xi);Dispose(Yi);Dispose(Zi);
  Dispose(T2);Dispose(T3);
END;

PROCEDURE PARAFAC(Var T : Tenseur;
  Var r : Integer;

```

```

        Var X,Y,Z : Matrice);
Calcule les matrices X,Y,Z telles que la reconstitution de tenseur Tens à
partir des coordonnées X1,Y1,Z1, colonnes des matrices X,Y,Z par un modèle
PARAFAC :  $Tens = \sum_i X_i \otimes Y_i \otimes Z_i$  soit optimale

```

```

VAR Dis1,Dis2,Test : Real;
    NombIter : Integer;
    TPermu2, TPermu3 : ^Tenseur;

```

```

PROCEDURE Init(VAR X,Y,Z : Matrice ; Var r : Integer);

```

```

    BEGIN de la procedure INIT < PARAFAC
        X.InitNombLign(T.NombPlan); X.InitNombCol(r); X.Aleatoire;
        Y.InitNombLign(T.NombCol) ; Y.InitNombCol(r); Y.Aleatoire;
        Z.InitNombLign(T.NombLign); Z.InitNombCol(r); Z.Aleatoire;
    END; de la procedure INIT < PARAFAC

```

```

PROCEDURE Iter(Var T : Tenseur ;
                Var X,Y,Z : Matrice;
                Var r : Integer);

```

```

    VAR i : Integer;
        TensXetY,TDual : ^Tenseur;

```

```

PROCEDURE Tranches(Var X,Y : Matrice ; Var TensXetY : Tenseur);

```

Construit le tenseur TensXetY dont les tranches T1 sont les matrices $X_1 \otimes Y_1$, X_1 & Y_1 étant vecteurs colonnes de X et Y

```

    VAR i : Integer;
        Xi,Yi : ^Vecteur;
        MatAux : ^Matrice;

```

```

    BEGIN de la procédure Tranches
        New(Xi);New(Yi);New(MatAux);
        TensXetY.InitDimT(Y.NombLign,X.NombLign,X.NombCol);
        For i:= 1 to TensXetY.NombPlan do Begin
            X.ExtractCol(i,Xi^);
            Y.ExtractCol(i,Yi^);
            ProTens2V(Xi^,Yi^,MatAux^);
            TensXetY.EcriPlan(i,MatAux^);
        End;
        Dispose(Xi);Dispose(Yi);Dispose(MatAux);
    END; de la procédure Tranches

```

```

PROCEDURE CalZ(Var T,T2 : Tenseur ; Var Z : Matrice);

```

Calcule $Z_i := T.T2_i$, où $T2_i$ est le i ème plan de T2

```

    VAR i : Integer;
        Zi : ^Vecteur;
        MatAux : ^Matrice;
    BEGIN de la procédure CalZ
        New(MatAux);New(Zi);
        Zi^.InitDim(T.NombLign);
        MatAux^.InitDimM(T2.NombLign,T2.NombCol);
        For i:=1 to T2.NombPlan do begin
            T2.ExtractPlan(i,MatAux^);
            ProdTensEtMat(T,MatAux^,Zi^);
            Z.EcriCol(i,Zi^);
        End;

```

```

        Dispose(Zi);Dispose(MatAux);
    END; de la procédure CalZ

BEGIN de la procedure ITER < PARAFAC
    New(TensXetY);New(TDual);
    Tranches(X,Y,TensXetY^);
    DualMat(TensXetY^,TDual^);
    CalZ(T,TDual^,Z);
    Dispose(TensXetY);Dispose(TDual);
END; de la procedure ITER

PROCEDURE Verifi(Var X,Y,Z : Matrice ; Var Tens : Tenseur;
                Var Dis : Real);
    VAR TConstru : ^Tenseur;
    BEGIN
        New(Tconstru);
        ConstruParamFac(X,Y,Z,Tconstru^);
        Dis := Distance(Tens,Tconstru^);
        GotoXY(20,12);
        Write('Distance : ',Dis :13:10);
        Dispose(Tconstru);
    END;

PROCEDURE Faire(Var Test,Dis1,Dis2 : Real);
    BEGIN
        If Dis2 > Dis1 Then begin
            WriteLn('Erreur dans le déroulement de la procédure PARAFAC');
            ReadLn;
            Exit;
        End;
        Test:= Dis1-Dis2;
    END;

BEGIN de la procédure PARAFAC
    New(TPermu2);New(TPermu3);
    TextBackGround(0);
    Init(X,Y,Z,r);
    Verifi(X,Y,Z,T,Dis1);WriteLn;
    NombIter:=0;
    Repeat
        NombIter:=NombIter+1;
        GotoXY(20,10);Write(' Iteration : ',NombIter :3,' ');
        Iter(T,X,Y,Z,r);
        T.Permu2(TPermu2^);
        Iter(TPermu2^,Y,Z,X,r);
        T.Permu3(TPermu3^);
        Iter(TPermu3^,Z,X,Y,r);
        Verifi(X,Y,Z,T,Dis2);
        Faire(Test,Dis1,Dis2);
        GotoXY(20,14);
        Write('Test : ',Test:13:10);
        Dis1:=Dis2;
    Until Test < Epsi;
    TextBackGround(7);
    Dispose(TPermu2);Dispose(TPermu3);
END; de la procédure PARAFAC

```

```

PROCEDURE PremParaFac(Var T : Tenseur;
                     Var Coef : Real;
                     Var X,Y,Z : Vecteur);
VAR NombIter : Integer;
    Test : Real;
    T2 : ^Tenseur;
    X1,Y1,Z1 : ^Vecteur;

PROCEDURE Init(VAR X,Y,Z : Vecteur);

    BEGIN de la procedure INIT < PremParaFac
        X.InitDim(T.NombPlan); X.Aleatoire; X.Normalisation;
        Y.InitDim(T.NombCol); Y.Aleatoire; Y.Normalisation;
        Z.InitDim(T.NombLign); Z.Aleatoire; Z.Normalisation;
    END; de la procedure INIT < PremParaFac

PROCEDURE Iter(Var T : Tenseur ;
               Var X,Y,Z : Vecteur);
VAR i,j : Integer;
    Valeur : Real;
    M : ^Matrice;
    V : ^Vecteur;
    BEGIN de la procedure ITER < PremParaFac
        New(M); New(V);
        ProdTensEtVec(T,X,M^);
        ProdMatEtVec(M^,Y,V^);
        Z:=V^; Z.Normalisation;
        Dispose(M); Dispose(V);
    END; de la procedure ITER < PremParaFac

PROCEDURE Recompo(Var T : tenseur ; Var X,Y,Z : Vecteur ;
                  Var NombIter : Integer);
Var Coef,Dis : Real;
    T2,T3 : ^Tenseur;
    Begin de la procedure Recompo < PremParaFac
        New(T2);New(T3);
        ProTens3V(X,Y,Z,T2^);
        Coef := ProScalTens(T,T2^);
        ProReelEtVec(Coef,T2^,T3^);
        Dis:=Distance(T,T3^)/T.Norme;
        GotoXY(20,10);
        Write('Iteration ',NombIter:3);
        GotoXY(20,12);
        Write('Approximation : ',Dis :13:10);
        Dispose(T2);Dispose(T3);
    End; de la procedure Recompo < PremParaFac

FUNCTION TestIter(Var x1,y1,z1,x2,y2,z2 : Vecteur) : Real;
    Begin
        TestIter:=Distance(x1,x2)+Distance(y1,y2)+Distance(z1,z2);

```



```

End;

BEGIN de la procedure PremParaFac
  New(T2);
  New(X1);New(Y1);New(Z1);
  TextBackGround(0);
  Init(X,Y,Z);
  NombIter:=0;
  Repeat
    NombIter := NombIter+1;
    X1^:=X;Y1^:=Y;Z1^:=Z;
    Iter(T,X,Y,Z);
    T.Permu2(T2^); Iter(T2^,Y,Z,X);
    T.Permu3(T2^); Iter(T2^,Z,X,Y);
    Recompo(T,X,Y,Z,NombIter);
    Test:=TestIter(X1^,Y1^,Z1^,X,Y,Z);
    GotoXY(20,14);
    Write('Test : ',Test:13:10);
  Until Test < Epsi;
  ProTens3V(X,Y,Z,T2^);
  Coef := ProScalTens(T,T2^);
  TextBackGround(7);
  Dispose(T2);
  Dispose(X1);Dispose(Y1);Dispose(Z1);
END; de la procedure PremParaFac

```

END.

```

PROGRAM Travail;
Permet de saisir un tenseur, et de réaliser trois calculs :
  1 - Recherche de la première Composante Principale
  2 - Recherche du Modèle CANDECOMP / PARAFAC
  3 - Recherche du rang
Est fourni comme illustration des unités DECLARE, ALGELINE, USUELS, ANAFAC &
ANATENS.
REPAS est une unité qui réalise des menus (quoique la logique eût voulu
l'inverse : un menu compose un repas)
USES Crt, Repas, Declare, Algeline, Usuels, AnaFac, AnaTens;
CONST NeMax = 50;
TYPE
  Mot      = String[35];
  MotMens  = Array[1..12] of Mot;
  MotTab   = Array[1..nemax] of Mot;

VAR r,n,cx : Integer;
    L : Real;
    X,Y,Z : ^Matrice;
    Vx,Vy,Vz : ^Vecteur;
    T : ^Tenseur;
    Men : MotTab;
LABEL MenP;

PROCEDURE Init;
BEGIN
  New(X);New(Y);New(Z);New(T);
  New(Vx);New(Vy);New(Vz);
  ClrScr;
END;

PROCEDURE Donnees;
BEGIN
  n:=2;
  Men[10]:='TENSEUR DE DONNEES';
  Men[1]:='Saisie';
  Men[2]:='Aléatoire';
  Menu(n,cx,Men);
  ClrScr;
  Case cx of
    1 : InitSaisieTens(T^);
    2 : InitAleaTens(T^);
  End;
END;

PROCEDURE Affiche;
VAR Format,Bande,Hauteur,Largeur : Byte;
    k1,k2,k3,k4 : Byte;
    M : ^Matrice;

PROCEDURE AfficheInit;
BEGIN
  New(M);
END;

PROCEDURE AfficheParam;

```

```

BEGIN
  TextBackGround(7);
  ClrScr;
  TextBackGround(1);
  Bande:=3;
  Format:=6;
  Hauteur:=T^.NombLign;
  Largeur:=T^.NombCol*Format;
END;

PROCEDURE AfficheConstru;
  VAR i : Integer;

PROCEDURE AfficheFenetre;
  PROCEDURE FenetreFormat;
    BEGIN
      k1:=Bande+(Bande+Largeur)*(i-1);
      k2:=Bande;
      k3:=(Bande+Largeur)*i;
      k4:=Bande+Hauteur-1;
    END;
  PROCEDURE FenetreTitre;
    BEGIN
      Window(k1,k2-2,k3,k2-2);
      ClrScr;
      Write('Tranche n° ',i);
    END;
  PROCEDURE FenetreEcrit;
    BEGIN
      Window(k1,k2,k3,k4);
      ClrScr;
    END;

  BEGIN
    FenetreFormat;
    FenetreTitre;
    FenetreEcrit;
  END;

PROCEDURE AfficheTitre;
  BEGIN
  END;

PROCEDURE AfficheEcrit;
  VAR Lign,Col : Integer;
  BEGIN
    T^.ExtractPlan(i,M^);
    For Lign:=1 to M^.NombLign do Begin
      For Col:=1 to M^.NombCol do Begin
        GotoXY(1+Format*(Col-1),Lign);
        Write(M^.Ind(Lign,Col):Format:2);
      End;
    End;
  END;

```

```

BEGIN
  For i:=1 to T^.NombPlan do Begin
    AfficheFenetre;
    AfficheTitre;
    AfficheEcrit;
  End;
END;

```

```

PROCEDURE AfficheSortie;
BEGIN
  Pause;
  Window(1,1,80,25);
  TextBackGround(7);
  ClrScr;
END;

```

```

PROCEDURE AfficheDone;
BEGIN
  Dispose(M);
END;

```

```

BEGIN { ***** de la procedure AFFICHE *****}
  AfficheInit;
  AfficheParam;
  AfficheConstru;
  AfficheSortie;
  AfficheDone;
END;

```

```

PROCEDURE Calcul;

```

```

PROCEDURE CalculParafac;
  VAR Rang : Integer;

```

```

PROCEDURE CalculParafacParam;
BEGIN
  TextBackGround(0);
  GotoXY(20,12);
  Write('Rang du tenseur du modèle : ');
  Read(Rang);
  TextBackGround(7);
  ClrScr;
  TextBackGround(0);
END;

```

```

BEGIN
  CalculParafacParam;
  GotoXY(20,5); Write('Itération ParaFac/CanDecomp en Cours');
  ParaFac(T^,Rang,X^,Y^,Z^);
  Pause;
END;

```

```

PROCEDURE RechercheRang;
CONST EpsiRang = 0.00001;
VAR Rang : Integer;

```

```

    n1,n2,n3 : Real;
    Dis : Real;
    T2 : ^Tenseur;

PROCEDURE InitRang(Var Rang : Integer);
    VAR Rang0 : Real;
    BEGIN
        n1:=T^.NombPlan;
        n2:=T^.NombLign;
        n3:=T^.NombCol;
        Rang0:=n1*n2*n3/(n1+n2+n3-2);
        If Int(Rang0) = Rang0
            Then Rang := Round(Rang0)
            Else Rang := Round(Rang0+0.5);
        Rang:=Rang-1;
    END;

PROCEDURE CalculRang;

    PROCEDURE EcriRang;
        BEGIN
            TextBackGround(0);
            GotoXY(20,8);
            Write('Rang : ');
            Write(Rang);
            TextBackGround(7);
        END;

    BEGIN
        Repeat
            Inc(Rang);
            EcriRang;
            ParaFac(T^,Rang,X^,Y^,Z^);
            ConstruaParaFac(X^,Y^,Z^,T2^);
            Dis:=Distance(T^,T2^);
        Until Dis < EpsiRang;
    END;

    BEGIN
        New(T2);
        InitRang(Rang);
        CalculRang;
        Pause;
        Dispose(T2);
    END;

PROCEDURE CalculPremCompo;
    BEGIN
        PremParaFac(T^,L,Vx^,Vy^,Vz^);
        Pause;
    END;

    BEGIN
        n:=5;
        Men[10]:='CALCUL';
        Men[1]:='ParaFac / Candecomp';

```

```

Men[2]:='Recherche de rang';
Men[3]:='Premiere Composante';
Men[4]:='Modèle de Tucker';
Men[5]:='Fin des Calculs';
Menu(n,cx,Men);
ClrScr;
Case cx of
    1 : CalculParafac;
    2 : RechercheRang;
    3 : CalculPremCompo;
    5 : Exit;
End;
END;

```

```

PROCEDURE Done;
BEGIN
    Dispose(Vz);Dispose(Vy);Dispose(Vx);
    Dispose(T);Dispose(Z);Dispose(Y);Dispose(X);
    ClrScr;
END;

```

```

BEGIN
    Init;
    Donnees;
    Affiche;
    Calcul;
    Done;
END.

```

CALCUL DE DETERMINANTS

1. CALCUL D'UN DÉTERMINANT D'ORDRE 2N

1. CALCUL D'UN DETERMINANT D'ORDRE 2N

1. Lemme■ Enoncé

Le calcul d'un déterminant général d'ordre $2n$ peut se ramener au calcul de la somme de C_{2n}^n produits de deux déterminants d'ordre n .

■ Démonstration

1 - Soit le déterminant d'ordre $2n$ général :

$$\Delta = \begin{vmatrix} a_1^1 & a_2^1 & \dots & a_j^1 & \dots & a_{2n}^1 \\ a_1^2 & a_2^2 & \dots & a_j^2 & \dots & a_{2n}^2 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_1^i & a_2^i & \dots & a_j^i & \dots & a_{2n}^i \\ \dots & \dots & \dots & \dots & \dots & \dots \\ a_1^{2n} & a_2^{2n} & \dots & a_j^{2n} & \dots & a_{2n}^{2n} \end{vmatrix}$$

2 - Appelons $(x_1, x_2, \dots, x_j, \dots, x_{2n})$ les vecteurs colonnes de Δ . Il vient alors : $\Delta = \text{Det}(x_1, x_2, \dots, x_j, \dots, x_{2n})$ est le déterminant du système des $2n$ vecteurs x_j .

3 - Posons $x_j = u_j + v_j$, avec :

- $u_j = (a_j^1, a_j^2, \dots, a_j^1, \dots, a_j^n, 0, \dots \dots \dots, 0)$
- $v_j = (0, \dots \dots \dots 0, a_j^{n+1}, a_j^{n+2}, \dots, a_j^{n+1}, \dots a_j^{2n})$.

4 - Comme Det est une application $2n$ -linéaire sur \mathcal{E} , il vient :

$$\begin{aligned} \Delta &= \text{Det } (x_j)_j \\ &= \text{Det } (u_j + v_j)_j \\ &= \sum_{I \subset \{1, 2n\}} \text{Det } (y_\gamma)_{\gamma \in I} \end{aligned}$$

où I décrit l'ensemble des parties de $\{1, 2n\} \subset \mathbb{N}$, et, dans le $2n$ -uplet des y_γ , $y_\gamma = u_\gamma$ si $\gamma \in I$ et $y_\gamma = v_\gamma$ si $\gamma \in \{1, 2n\} - I$. On utilise simplement le fait que Det est une application bilinéaire.

5 - Si $\text{Card}(I) = p$, il y a donc p vecteurs u_α et $2n-p$ vecteurs v_β dans le $2n$ -uplet des y_γ . Si $p \neq n$, il vient : $p > n$ ou $2n-p > n$. Donc, une des deux familles (u_α) ou (v_β) comprend plus de n vecteurs. Or, ces vecteurs sont dans un espace de dimension n par construction. Une des deux familles est donc liée. Le système (y_γ) est donc lié, et son déterminant est nul.

Nous avons donc montré :

$$p \neq n \Rightarrow \text{Det}(y_\gamma) = 0$$

6 - Il vient donc :

$$\Delta = \sum_{\substack{I \subset \{1, 2n\} \\ \text{Card}(I)=n}} \text{Det } (y_\gamma)$$

Or, il y a $C_{2n}^n = \frac{(2n)!}{(n!)^2}$ parties de $\{1, 2n\}$ à n éléments. La somme précédente

contient donc $\frac{(2n)!}{(n!)^2} = \frac{2n(2n-1)\dots(n+1)}{n(n-1)\dots 2}$ déterminants.

7 - Intéressons nous aux déterminants $\text{Det } (y_\gamma)$. Une permutation des vecteurs (u_α) et (v_β) dans (y_γ) en change le signe selon la signature de la permutation mais non la valeur absolue. Une permutation adéquate permet donc d'écrire ces déterminants, au signe près, tels que les n premiers vecteurs du $2n$ -uplet (y_γ) sont les (u_α) et les n derniers sont les v_β . Sous cette forme, il est clair que :

$$\text{Det}(y_\gamma) = (-1)^\sigma \cdot \text{Det}(u_\alpha) \cdot \text{Det}(v_\beta),$$

où les déterminants à calculer sont d'ordre n , et non $2n$.

8 - D'où il vient, finalement :

$$\Delta = \sum_{\substack{I \subset \{1, 2n\} \\ \text{Card}(I)=p}} (-1)^\sigma \cdot \text{Det}(u_\alpha)_{\alpha \in I} \cdot \text{Det}(v_\beta)_{\beta \in \{1, 2n\} - I}$$

9 - Remarquons que le calcul de Δ par le développement de Lagrange selon les mineurs mène au calcul de $(2n!)/(n!)$ déterminants d'ordre n , soit $n!$ fois plus.

2. Exemple

Des exemples pratiques de calcul seront donnés ci-dessous pour le calcul effectif de déterminants de Sylvester. Indiquons seulement :

□ Pour $n = 4$: cette décomposition conduit au calcul de $2 \cdot C_4^2 = 12$ déterminants d'ordre 2, au lieu des $6 \cdot 5 \cdot 4 \cdot 3 = 360$ par la décomposition de Lagrange.

□ Pour $n = 8$: cette décomposition conduit au calcul de $2 \cdot C_8^4 = 140$ déterminants d'ordre 4 [au lieu de $8 \cdot 7 \cdot 6 \cdot 5 = 1\,680$], ce qui se ramène au calcul de $140 \cdot 12 = 1\,680$ déterminants d'ordre 2 [au lieu de $20\,160$].

3. Remarque

La décomposition précédente est très proche de la décomposition de Lagrange selon les mineurs. On peut la généraliser pour englober comme cas particulier les deux calculs. En effet :

1 - Posons $\dim(\mathcal{E}) = n$ et calculons $\text{Det}(x_1, \dots, x_n)$.

2 - Soient deux espaces \mathcal{F} et \mathcal{G} tels que $\mathcal{E} = \mathcal{F} \oplus \mathcal{G}$.

3 - Tout vecteur x_i s'écrit d'une façon et d'une seule : $x_i = u_i + v_i$, avec

$u_i \in \mathcal{F}$ et $v_i \in \mathcal{G}$, $\dim(\mathcal{F}) = p$ et $\dim(\mathcal{G}) = q$; $p + q = n$.

4 - Il vient alors : $\text{Det}(x_i) = \text{Det}(u_i + v_i) = \sum_{I \subset \{1, n\}} \text{Det}(z_k)$, où $z_k = u_k$ si $k \in I$ et $z_k = v_k$ sinon. Cette somme contient donc 2^n termes.

5 - Posons : $\text{Card}(I) = r$. Alors, $\text{Det}(z_k)$ est nul dès que $r > p$ (car le système des u_k est lié) ou que $n-r > q$, soit $r < p$ car $q = n-p$ (car le système des v_k est lié). $\text{Det}(z_k)$ est donc non nul seulement si $r = p$.

6 - D'où : $\text{Det}(x_i) = \sum_{\substack{I \subset \{1, n\} \\ \text{Card}(I)=p}} \text{Det}(z_k) = \sum_{\substack{I \subset \{1, n\} \\ \text{Card}(I)=p}} (-1)^\sigma \text{Det}(u_k) \cdot \text{Det}(v_k)$. C'est donc

la somme de C_n^p produit d'un déterminant d'ordre p par un déterminant d'ordre $n-p$. On obtient la décomposition précédente en choisissant $p = n/2$ si n est pair, et la décomposition de Lagrange en choisissant $p = 1$.

2. CALCUL DU DETERMINANT DE SYLVESTER POUR $N = 2$

1. Résultat

$$\begin{aligned} \Delta(2) &= \begin{vmatrix} a & b \\ a' & b' \end{vmatrix} \cdot \begin{vmatrix} c & b \\ c' & b' \end{vmatrix} + \begin{vmatrix} a & c \\ a' & c' \end{vmatrix}^2 \\ &= (ab' - a'b)(cb' - c'b) + (ac' - a'c)^2 \end{aligned}$$

2. Démonstration

Nous allons appliquer le calcul précédent avec $n=2$.

1- Nous avons :

$$\Delta(2) = \begin{vmatrix} c & b & a & 0 \\ 0 & c & b & a \\ c' & b' & a' & 0 \\ 0 & c' & b' & a' \end{vmatrix}$$

2 - Soit, par une permutation sur les lignes :

$$\Delta(2) = - \begin{vmatrix} c & b & a & 0 \\ c' & b' & a' & 0 \\ 0 & c & b & a \\ 0 & c' & b' & a' \end{vmatrix}$$

3 - Posons donc :

$$u_1 = \begin{vmatrix} c \\ c' \\ 0 \\ 0 \end{vmatrix} ; u_2 = \begin{vmatrix} b \\ b' \\ 0 \\ 0 \end{vmatrix} ; u_3 = \begin{vmatrix} a \\ a' \\ 0 \\ 0 \end{vmatrix} ; u_4 = \vec{0}$$

$$v_1 = \vec{0} ; v_2 = \begin{vmatrix} 0 \\ 0 \\ c \\ c' \end{vmatrix} ; v_3 = \begin{vmatrix} 0 \\ 0 \\ b \\ b' \end{vmatrix} ; v_4 = \begin{vmatrix} 0 \\ 0 \\ a \\ a' \end{vmatrix}$$

4 - Les parties à deux éléments de $\{1,4\}$ sont :

$\{1,2\} ; \{1,3\} ; \{1,4\} ; \{2,3\} ; \{2,4\} ; \{3,4\}$.

Comme $u_4 = 0$, nécessairement $y_4 = v_4$.

De même, $y_1 = u_1$ puisque $v_1 = 0$.

Il reste donc :

$$\Delta(2) = - \text{Det}(u_1, u_2, v_3, v_4) - \text{Det}(u_1, v_2, u_3, v_4)$$

$$= - \begin{vmatrix} c & b & 0 & 0 \\ c' & b' & 0 & 0 \\ 0 & 0 & b & a \\ 0 & 0 & b' & a' \end{vmatrix} - \begin{vmatrix} c & 0 & a & 0 \\ c' & 0 & a' & 0 \\ 0 & c & 0 & a \\ 0 & c' & 0 & a' \end{vmatrix}$$

$$= - \begin{vmatrix} c & b & 0 & 0 \\ c' & b' & 0 & 0 \\ 0 & 0 & b & a \\ 0 & 0 & b' & a' \end{vmatrix} + \begin{vmatrix} c & a & 0 & 0 \\ c' & a' & 0 & 0 \\ 0 & 0 & c & a \\ 0 & 0 & c' & a' \end{vmatrix}$$

$$= - \begin{vmatrix} c & b \\ c' & b' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} + \begin{vmatrix} c & a \\ c' & a' \end{vmatrix}^2$$

$$= \begin{vmatrix} a & b \\ a' & b' \end{vmatrix} \cdot \begin{vmatrix} c & b \\ c' & b' \end{vmatrix} + \begin{vmatrix} a & c \\ a' & c' \end{vmatrix}^2$$

3. CALCUL DE DETERMINANT DE SYLVESTER POUR N = 4

1. Notations

1 - Soient deux polynômes en x du quatrième degré :

$$\begin{cases} a x^4 + b x^3 + c x^2 + d x + e = 0 \\ a' x^4 + b' x^3 + c' x^2 + d' x + e' = 0 \end{cases}$$

2 -Le résultant est donc :

$$\Delta(4) = \begin{vmatrix} e & d & c & b & a & . & . & . \\ . & e & d & c & b & a & . & . \\ . & . & e & d & c & b & a & . \\ . & . & . & e & d & c & b & a \\ e' & d' & c' & b' & a' & . & . & . \\ . & e' & d' & c' & b' & a' & . & . \\ . & . & e' & d' & c' & b' & a' & . \\ . & . & . & e' & d' & c' & b' & a' \end{vmatrix}$$

où les points <.> désignent des zéros <0>.

2. Résultat

Le calcul du déterminant de Sylvester précédent d'ordre 8 ne requiert que le calcul de 53 produits et 40 sommes de nombres réels.

3. Démonstration

1 - Par permutation des lignes, ce déterminant s'écrit :

$$\Delta(4) = \begin{vmatrix} e & d & c & b & a & . & . & . \\ e' & d' & c' & b' & a' & . & . & . \\ . & e & d & c & b & a & . & . \\ . & e' & d' & c' & b' & a' & . & . \\ . & . & e & d & c & b & a & . \\ . & . & e' & d' & c' & b' & a' & . \\ . & . & . & e & d & c & b & a \\ . & . & . & e' & d' & c' & b' & a' \end{vmatrix}$$

2 - Posons :

$$u_1 = \begin{vmatrix} e \\ e' \\ . \\ . \\ . \\ . \\ . \\ . \end{vmatrix} ; u_2 = \begin{vmatrix} d \\ d' \\ e \\ e' \\ . \\ . \\ . \\ . \end{vmatrix} ; u_3 = \begin{vmatrix} c \\ c' \\ d \\ d' \\ . \\ . \\ . \\ . \end{vmatrix} ; u_4 = \begin{vmatrix} b \\ b' \\ c \\ c' \\ . \\ . \\ . \\ . \end{vmatrix} ; u_5 = \begin{vmatrix} a \\ a' \\ b \\ b' \\ . \\ . \\ . \\ . \end{vmatrix} ; u_6 = \begin{vmatrix} . \\ . \\ a \\ a' \\ . \\ . \\ . \\ . \end{vmatrix} ; u_7 = \begin{vmatrix} . \\ . \\ . \\ . \\ . \\ . \\ . \\ . \end{vmatrix} ; u_8 = \begin{vmatrix} . \\ . \\ . \\ . \\ . \\ . \\ . \\ . \end{vmatrix}$$

et :

$$v_1 = \begin{vmatrix} . \\ . \\ . \\ . \\ . \\ . \\ . \\ . \end{vmatrix} ; v_2 = \begin{vmatrix} . \\ . \\ . \\ . \\ . \\ . \\ . \\ . \end{vmatrix} ; v_3 = \begin{vmatrix} . \\ . \\ . \\ e \\ e' \\ . \\ . \\ . \end{vmatrix} ; v_4 = \begin{vmatrix} . \\ . \\ . \\ d \\ d' \\ e \\ e' \end{vmatrix} ; v_5 = \begin{vmatrix} . \\ . \\ . \\ c \\ c' \\ d \\ d' \end{vmatrix} ; v_6 = \begin{vmatrix} . \\ . \\ . \\ b \\ b' \\ c \\ c' \end{vmatrix} ; v_7 = \begin{vmatrix} . \\ . \\ . \\ a \\ a' \\ b \\ b' \end{vmatrix} ; v_8 = \begin{vmatrix} . \\ . \\ . \\ . \\ . \\ a \\ a' \end{vmatrix}$$

3 - Comme $u_7 = u_8 = 0$, : $y_7 = v_7$ et $y_8 = v_8$.

De même, comme $v_1 = v_2 = 0$: $y_1 = u_1$ et $y_2 = u_2$.

4 - Comme dans (y_γ) il faut 4 vecteurs u_α et 4 vecteurs v_β , et que 2 u ($\gamma=1$ et $\gamma=2$) d'une part et 2 v ($\gamma=7$ et $\gamma=8$) d'autre part sont imposés pour que $\text{Det}(y_\gamma)$ soit non nul, il reste à choisir 2 vecteurs u (donc 2 vecteurs v) parmi les 4 y_γ , pour $\gamma=3, \dots, 6$; soit 6 solutions.

5 - Nous avons donc :

$$\begin{aligned} \Delta(4) = & \text{Det}(u_1, u_2, u_3, u_4, v_5, v_6, v_7, v_8) + \text{Det}(u_1, u_2, u_3, v_4, u_5, v_6, v_7, v_8) \\ & + \text{Det}(u_1, u_2, u_3, v_4, v_5, u_6, v_7, v_8) + \text{Det}(u_1, u_2, v_3, u_4, u_5, v_6, v_7, v_8) \\ & + \text{Det}(u_1, u_2, v_3, u_4, v_5, u_6, v_7, v_8) + \text{Det}(u_1, u_2, v_3, v_4, u_5, u_6, v_7, v_8) \end{aligned}$$

soit, en développant :

$$\Delta(4) = \begin{vmatrix} e & d & c & b & . & . & . & . \\ e' & d' & c' & b' & . & . & . & . \\ . & e & d & c & . & . & . & . \\ . & e' & d' & c' & . & . & . & . \\ . & . & . & . & c & b & a & . \\ . & . & . & . & c' & b' & a' & . \\ . & . & . & . & d & c & b & a \\ . & . & . & . & d' & c' & b' & a' \end{vmatrix} + \begin{vmatrix} e & d & c & . & a & . & . & . \\ e' & d' & c' & . & a' & . & . & . \\ . & e & d & . & b & . & . & . \\ . & e' & d' & . & b' & . & . & . \\ . & . & . & d & . & b & a & . \\ . & . & . & d' & . & b' & a' & . \\ . & . & . & e & . & c & b & a \\ . & . & . & e' & . & c' & b' & a' \end{vmatrix} + \begin{vmatrix} e & d & c & . & . & . & . & . \\ e' & d' & c' & . & . & . & . & . \\ . & e & d & . & . & a & . & . \\ . & e' & d' & . & . & a' & . & . \\ . & . & . & d & c & . & a & . \\ . & . & . & d' & c' & . & a' & . \\ . & . & . & e & d & . & b & a \\ . & . & . & e' & d' & . & b' & a' \end{vmatrix} + \begin{vmatrix} e & d & . & b & a & . & . & . \\ e' & d' & . & b' & a' & . & . & . \\ . & e & . & c & b & . & . & . \\ . & e' & . & c' & b' & . & . & . \\ . & . & e & . & . & b & a & . \\ . & . & e' & . & . & b' & a' & . \\ . & . & . & d & . & c & b & a \\ . & . & . & d' & . & c' & b' & a' \end{vmatrix} + \begin{vmatrix} e & d & . & . & a & . & . & . \\ e' & d' & . & . & a' & . & . & . \\ . & e & . & . & b & a & . & . \\ . & e' & . & . & b' & a' & . & . \\ . & . & e & d & . & . & a & . \\ . & . & e' & d' & . & . & a' & . \\ . & . & . & e & . & . & b & a \\ . & . & . & e' & . & . & b' & a' \end{vmatrix}$$

soit, par permutation des colonnes :

=	e	d	c	b
	e'	d'	c'	b'
	.	e	d	c
	.	e'	d'	c'
	c	b	a	.
	c'	b'	a'	.
	d	c	b	a
	d'	c'	b'	a'

-	e	d	c	a
	e'	d'	c'	a'
	.	e	d	b
	.	e'	d'	b'
	d	b	a	.
	d'	b'	a'	.
	e	c	b	a
	e'	c'	b'	a'

$$- \begin{vmatrix} e & d & b & . & . & . & . & . \\ e' & d' & b' & . & . & . & . & . \\ . & e & c & a & . & . & . & . \\ . & e' & c' & a' & . & . & . & . \\ . & . & . & . & e & c & a & . \\ . & . & . & . & e' & c' & a' & . \\ . & . & . & . & . & d & b & a \\ . & . & . & . & . & d' & b' & a' \end{vmatrix} + \begin{vmatrix} e & d & a & . & . & . & . & . \\ e' & d' & a' & . & . & . & . & . \\ . & e & b & a & . & . & . & . \\ . & e' & b' & a' & . & . & . & . \\ . & . & . & . & e & d & a & . \\ . & . & . & . & e' & d' & a' & . \\ . & . & . & . & . & e & b & a \\ . & . & . & . & . & e' & b' & a' \end{vmatrix}$$

soit, en développant :

$$= \begin{vmatrix} e & d & c & b \\ e' & d' & c' & b' \\ . & e & d & c \\ . & e' & d' & c' \end{vmatrix} \cdot \begin{vmatrix} c & b & a & . \\ c' & b' & a' & . \\ d & c & b & a \\ d' & c' & b' & a' \end{vmatrix} - \begin{vmatrix} e & d & c & a \\ e' & d' & c' & a' \\ . & e & d & b \\ . & e' & d' & b' \end{vmatrix} \cdot \begin{vmatrix} d & b & a & . \\ d' & b' & a' & . \\ e & c & b & a \\ e' & c' & b' & a' \end{vmatrix} \\ + \begin{vmatrix} e & d & c & . \\ e' & d' & c' & . \\ . & e & d & a \\ . & e' & d' & a' \end{vmatrix} \cdot \begin{vmatrix} d & c & a & . \\ d' & c' & a' & . \\ e & d & b & a \\ e' & d' & b' & a' \end{vmatrix} + \begin{vmatrix} e & d & b & a \\ e' & d' & b' & a' \\ . & e & c & b \\ . & e' & c' & b' \end{vmatrix} \cdot \begin{vmatrix} e & b & a & . \\ e' & b' & a' & . \\ . & c & b & a \\ . & c' & b' & a' \end{vmatrix} \\ + \begin{vmatrix} e & d & b & . \\ e' & d' & b' & . \\ . & e & c & a \\ . & e' & c' & a' \end{vmatrix} \cdot \begin{vmatrix} e & c & a & . \\ e' & c' & a' & . \\ . & d & b & a \\ . & d' & b' & a' \end{vmatrix} + \begin{vmatrix} e & d & a & . \\ e' & d' & a' & . \\ . & e & b & a \\ . & e' & b' & a' \end{vmatrix} \cdot \begin{vmatrix} e & d & a & . \\ e' & d' & a' & . \\ . & e & b & a \\ . & e' & b' & a' \end{vmatrix}$$

que l'on note :

$$\Delta(4) = \Delta_1 \Delta_2 - \Delta_3 \Delta_4 + \Delta_5 \Delta_6 + \Delta_7 \Delta_8 + \Delta_9 \Delta_{10} + \Delta_{11} \Delta_{12}$$

6 - Remarquons que les déterminants Δ_i d'ordre 4 concernés sont soit des déterminants de Sylvester, soit ils en sont très proches.

Posons :

$$D(x, y, z, a, b, c) = \begin{vmatrix} x & y & z & . \\ x' & y' & z' & . \\ . & p & q & r \\ . & p' & q' & r' \end{vmatrix}$$

et :

$$F(x, y, z, t, p, q) = \begin{vmatrix} x & y & z & t \\ x' & y' & z' & t' \\ . & p & q & . \\ . & p' & q' & . \end{vmatrix}$$

Il vient alors :

$$\begin{aligned}
 \circ D(x, y, z, a, b, c) &= \begin{vmatrix} x & y & z & . \\ x' & y' & z' & . \\ . & p & q & r \\ . & p' & q' & r' \end{vmatrix} \\
 &= \begin{vmatrix} x & . & z & . \\ x' & . & z' & . \\ . & p & q & r \\ . & p' & q' & r' \end{vmatrix} + \begin{vmatrix} x & y & z & . \\ x' & y' & z' & . \\ . & . & q & r \\ . & . & q' & r' \end{vmatrix} \\
 &= - \begin{vmatrix} x & z & . & . \\ x' & z' & . & . \\ . & q & p & r \\ . & q' & p' & r' \end{vmatrix} + \begin{vmatrix} x & y & z & . \\ x' & y' & z' & . \\ . & . & q & r \\ . & . & q' & r' \end{vmatrix} \\
 &= \begin{vmatrix} x & y & . \\ x' & y' & . \end{vmatrix} \begin{vmatrix} q & r \\ q' & r' \end{vmatrix} - \begin{vmatrix} x & z & . \\ x' & z' & . \end{vmatrix} \begin{vmatrix} p & r \\ p' & r' \end{vmatrix}
 \end{aligned}$$

$$\begin{aligned}
 \circ F(x, y, z, t, p, q) &= \begin{vmatrix} x & y & z & t \\ x' & y' & z' & t' \\ . & p & q & . \\ . & p' & q' & . \end{vmatrix} \\
 &= \begin{vmatrix} x & t & y & z \\ x' & t' & y' & z' \\ . & . & p & q \\ . & . & p' & q' \end{vmatrix} \\
 &= \begin{vmatrix} x & t & . \\ x' & t' & . \end{vmatrix} \begin{vmatrix} p & q \\ p' & q' \end{vmatrix}
 \end{aligned}$$

7 - Il vient alors :

$$\begin{aligned}
 \circ \Delta_1 &= \begin{vmatrix} e & d & c & b \\ e' & d' & c' & b' \\ . & e & d & c \\ . & e' & d' & c' \end{vmatrix} = D(e, d, c, e, d, c) \\
 &= \begin{vmatrix} e & d & . \\ e' & d' & . \end{vmatrix} \begin{vmatrix} d & c \\ d' & c' \end{vmatrix} - \begin{vmatrix} e & c & . \\ e' & c' & . \end{vmatrix} \begin{vmatrix} e & b \\ e' & b' \end{vmatrix}
 \end{aligned}$$

$$\circ \Delta_2 = \begin{vmatrix} c & b & a & . \\ c' & b' & a' & . \\ d & c & b & a \\ d' & c' & b' & a' \end{vmatrix} = \begin{vmatrix} c & b & a & . \\ c' & b' & a' & . \\ . & c & b & a \\ . & c' & b' & a' \end{vmatrix} + \begin{vmatrix} . & b & a & . \\ . & b' & a' & . \\ d & c & b & a \\ d' & c' & b' & a' \end{vmatrix}$$

$$= D(c, b, a, c, b, a) + F(d, c, b, a, b, a)$$

$$= \begin{vmatrix} c & b \\ c' & b' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} c & a \\ c' & a' \end{vmatrix} \cdot \begin{vmatrix} c & a \\ c' & a' \end{vmatrix} + \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix}$$

$$\circ \Delta_3 = \begin{vmatrix} e & d & c & a \\ e' & d' & c' & a' \\ . & e & d & b \\ . & e' & d' & b' \end{vmatrix} = \begin{vmatrix} e & d & c & . \\ e' & d' & c' & . \\ . & e & d & b \\ . & e' & d' & b' \end{vmatrix} + \begin{vmatrix} e & d & c & a \\ e' & d' & c' & a' \\ . & e & d & . \\ . & e' & d' & . \end{vmatrix}$$

$$= D(e, d, c, e, d, b) + F(e, d, c, a, e, d)$$

$$= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} d & b \\ d' & b' \end{vmatrix} - \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \cdot \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} + \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} e & d \\ e' & d' \end{vmatrix}$$

$$\circ \Delta_4 = \begin{vmatrix} d & b & a & . \\ d' & b' & a' & . \\ e & c & b & a \\ e' & c' & b' & a' \end{vmatrix} = \begin{vmatrix} . & a & b & d \\ . & a' & b' & d' \\ a & b & c & e \\ a' & b' & c' & e' \end{vmatrix} = \begin{vmatrix} a & b & c & e \\ a' & b' & c' & e' \\ . & a & b & d \\ . & a' & b' & d' \end{vmatrix}$$

$$= \begin{vmatrix} a & b & c & . \\ a' & b' & c' & . \\ . & a & b & d \\ . & a' & b' & d' \end{vmatrix} + \begin{vmatrix} a & b & c & e \\ a' & b' & c' & e' \\ . & a & b & . \\ . & a' & b' & . \end{vmatrix}$$

$$= D(a, b, c, a, b, d) + F(a, b, c, e, a, b)$$

$$= \begin{vmatrix} a & b \\ a' & b' \end{vmatrix} \cdot \begin{vmatrix} b & d \\ b' & d' \end{vmatrix} - \begin{vmatrix} a & c \\ a' & c' \end{vmatrix} \cdot \begin{vmatrix} a & d \\ a' & d' \end{vmatrix} + \begin{vmatrix} a & e \\ a' & e' \end{vmatrix} \cdot \begin{vmatrix} a & b \\ a' & b' \end{vmatrix}$$

$$\circ \Delta_5 = \begin{vmatrix} e & d & c & . \\ e' & d' & c' & . \\ . & e & d & a \\ . & e' & d' & a' \end{vmatrix}$$

$$= D(e, d, c, e, d, a)$$

$$= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} - \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \cdot \begin{vmatrix} e & a \\ e' & a' \end{vmatrix}$$

$$\circ \Delta_6 = \begin{vmatrix} d & c & a & . \\ d' & c' & a' & . \\ e & d & b & a \\ e' & d' & b' & a' \end{vmatrix} = \begin{vmatrix} d & c & a & . \\ d' & c' & a' & . \\ . & d & b & a \\ . & d' & b' & a' \end{vmatrix} + \begin{vmatrix} . & c & a & . \\ . & c' & a' & . \\ e & d & b & a \\ e' & d' & b' & a' \end{vmatrix}$$

$$= D(d, c, a, d, b, a) + F(e, a, c, a)$$

$$= \begin{vmatrix} d & c \\ d' & c' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} \cdot \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} + \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} c & a \\ c' & a' \end{vmatrix}$$

$$\circ \Delta_7 = \begin{vmatrix} e & d & b & a \\ e' & d' & b' & a' \\ . & e & c & b \\ . & e' & c' & b' \end{vmatrix} = \begin{vmatrix} e & d & b & . \\ e' & d' & b' & . \\ . & e & c & b \\ . & e' & c' & b' \end{vmatrix} + \begin{vmatrix} e & d & b & a \\ e' & d' & b' & a' \\ . & e & c & . \\ . & e' & c' & . \end{vmatrix}$$

$$= D(e, d, b, e, c, b) + F(e, a, e, c)$$

$$= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} c & b \\ c' & b' \end{vmatrix} - \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} \cdot \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} + \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} e & c \\ e' & c' \end{vmatrix}$$

$$\circ \Delta_8 = \begin{vmatrix} e & b & a & . \\ e' & b' & a' & . \\ . & c & b & a \\ . & c' & b' & a' \end{vmatrix}$$

$$= D(e, b, a, c, b, a)$$

$$= \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} c & a \\ c' & a' \end{vmatrix}$$

$$\circ \Delta_9 = \begin{vmatrix} e & d & b & . \\ e' & d' & b' & . \\ . & e & c & a \\ . & e' & c' & a' \end{vmatrix}$$

$$= D(e, d, b, e, c, a)$$

$$= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} c & a \\ c' & a' \end{vmatrix} - \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} \cdot \begin{vmatrix} e & a \\ e' & a' \end{vmatrix}$$

$$\circ \Delta_{10} = \begin{vmatrix} e & c & a & . \\ e' & c' & a' & . \\ . & d & b & a \\ . & d' & b' & a' \end{vmatrix}$$

$$= D(e, c, a, d, b, a)$$

$$= \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} d & a \\ d' & a' \end{vmatrix}$$

$$\begin{aligned}
\circ \Delta_{11} &= \begin{vmatrix} e & d & a & . \\ e' & d' & a' & . \\ . & e & b & a \\ . & e' & b' & a' \end{vmatrix} \\
&= D(e, d, a, e, b, a) \\
&= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} e & a \\ e' & a' \end{vmatrix}
\end{aligned}$$

$$\circ \Delta_{12} = \begin{vmatrix} e & d & a & . \\ e' & d' & a' & . \\ . & e & b & a \\ . & e' & b' & a' \end{vmatrix} = \Delta_{11}$$

8 - Les calculs font appel à $C_5^2 = 10$ déterminant d'ordre 2 du type :

$$d(x, y) = \begin{vmatrix} x & y \\ x' & y' \end{vmatrix}. \text{ Posons :}$$

$$\begin{aligned}
d_1 &= d(a, b) ; d_2 = d(a, c) ; d_3 = d(a, d) ; d_4 = d(a, e) ; d_5 = d(b, c) ; \\
d_6 &= d(b, d) ; d_7 = d(b, e) ; d_8 = d(c, d) ; d_9 = d(c, e) ; d_{10} = d(d, e)
\end{aligned}$$

9 - D'où :

$$\begin{aligned}
\circ \Delta_1 &= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} d & c \\ d' & c' \end{vmatrix} - \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \cdot \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \\
&= (de' - d'e)(cd' - c'd) - (c'e - ce')^2 \\
&= d_{10}d_8 - d_9^2
\end{aligned}$$

$$\begin{aligned}
\circ \Delta_2 &= \begin{vmatrix} c & b \\ c' & b' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} c & a \\ c' & a' \end{vmatrix} \cdot \begin{vmatrix} c & a \\ c' & a' \end{vmatrix} + \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} \\
&= (bc' - b'c)(ab' - a'b) - (ac' - a'c)^2 + (ad' - a'd)(ab' - a'b) \\
&= d_5d_1 - d_2^2 + d_3d_1
\end{aligned}$$

$$\begin{aligned}
\circ \Delta_3 &= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} d & b \\ d' & b' \end{vmatrix} - \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \cdot \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} + \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \\
&= (bd' - b'd)(de' - d'e) - (ce' - c'e)(be' - b'e) + (ae' - a'e)(de' - d'e)
\end{aligned}$$

$$= d_6 d_{10} - d_9 d_7 + d_4 d_{10}$$

$$\begin{aligned} \circ \Delta_4 &= \begin{vmatrix} a & b \\ a' & b' \end{vmatrix} \cdot \begin{vmatrix} b & d \\ b' & d' \end{vmatrix} - \begin{vmatrix} a & c \\ a' & c' \end{vmatrix} \cdot \begin{vmatrix} a & d \\ a' & d' \end{vmatrix} + \begin{vmatrix} a & e \\ a' & e' \end{vmatrix} \cdot \begin{vmatrix} a & b \\ a' & b' \end{vmatrix} \\ &= (ab' - a'b)(bd' - b'd) - (ac' - a'c)(ad' - a'd) + (ae' - a'e)(ab' - a'b) \\ &= d_1 d_6 - d_2 d_3 + d_4 d_1 \end{aligned}$$

$$\begin{aligned} \circ \Delta_5 &= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} - \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \cdot \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \\ &= (ad' - a'd)(de' - d'e) - (ae' - a'e)(ce' - c'e) \\ &= d_3 d_{10} - d_4 d_9 \end{aligned}$$

$$\begin{aligned} \circ \Delta_6 &= \begin{vmatrix} d & c \\ d' & c' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} \cdot \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} + \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} c & a \\ c' & a' \end{vmatrix} \\ &= (ab' - a'b)(cd' - c'd) - (ad' - a'd)^2 + (ae' - a'e)(ce' - c'e) \\ &= d_1 d_8 - d_3^2 + d_2 d_4 \end{aligned}$$

$$\begin{aligned} \circ \Delta_7 &= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} c & b \\ c' & b' \end{vmatrix} - \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} \cdot \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} + \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \\ &= (bc' - b'c)(de' - d'e) - (be' - b'e)^2 + (ae' - a'e)(ce' - c'e) \\ &= d_5 d_{10} - d_7^2 + d_4 d_9 \end{aligned}$$

$$\begin{aligned} \circ \Delta_8 &= \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} c & a \\ c' & a' \end{vmatrix} \\ &= (ab' - a'b)(be' - b'e) - (ac' - a'c)(ae' - a'e) \\ &= d_4 d_7 - d_2 d_4 \end{aligned}$$

$$\begin{aligned} \circ \Delta_9 &= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} c & a \\ c' & a' \end{vmatrix} - \begin{vmatrix} e & b \\ e' & b' \end{vmatrix} \cdot \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \\ &= (ac' - a'c)(de' - d'e) - (ae' - a'e)(be' - b'e) \\ &= d_2 d_{10} - d_4 d_7 \end{aligned}$$

$$\begin{aligned}
\circ \Delta_{10} &= \begin{vmatrix} e & c \\ e' & c' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} d & a \\ d' & a' \end{vmatrix} \\
&= (ab' - a'b)(ce' - c'e) - (ad' - a'd)(ae' - a'e) \\
&= d_1 d_9 - d_3 d_4
\end{aligned}$$

$$\begin{aligned}
\circ \Delta_{11} &= \begin{vmatrix} e & d \\ e' & d' \end{vmatrix} \cdot \begin{vmatrix} b & a \\ b' & a' \end{vmatrix} - \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \cdot \begin{vmatrix} e & a \\ e' & a' \end{vmatrix} \\
&= (ab' - a'b)(de' - d'e) - (ae' - a'e)^2 \\
&= d_1 d_{10} - d_4^2
\end{aligned}$$

10 - Le calcul d'un déterminant de Sylvester d'ordre 4 relève donc d'une procédure simple :

- 1) Donnée des 8 coefficients $a, b, c, d, e, a', b', c', d', e'$
- 2) Calcul des 10 déterminants d'ordre 2 : $d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9, d_{10}$
- 3) Calcul des 7 grandeurs $\Delta_1, \Delta_2, \Delta_3, \Delta_4, \Delta_5, \Delta_6, \Delta_7$
- 4) Calcul du déterminant de Sylvester.

2) requiert pour chaque déterminant le calcul de 2 produits et une somme, soit en tout 20 produit et 10 sommes.

3) requiert le calcul de 27 produits et 15 sommes

4) requiert le calcul de 6 produits et 5 sommes.

Le calcul total du déterminant de Sylvester d'ordre 4 requiert donc le calcul de 53 produits et 30 sommes.

SYSTEMES DE N EQUATIONS ALGEBRIQUES A N INCONNUES

La résolution des systèmes de n équations algébriques à n inconnues peut toujours se ramener à la résolution d'une équation algébrique à une inconnue. L'élimination d'une inconnue se réalise en utilisant un résultat classique en algèbre sur les déterminants de Sylvester, qui exprime une condition nécessaire et suffisante pour que deux polynômes aient une racine (éventuellement complexe) commune.

Les déterminants de Sylvester sont présentés dans les manuels d'algèbre, par exemple (CHARLES & ALLOUCH, 1984 ; LANG, 1984 ; MALLIAVIN M.-P., 1985). L'élimination d'une inconnue entre deux équations par cette technique est classique, et présentée dans (CHARLES & ALLOUCH, 1984)

1. RACINES COMMUNES A DEUX POLYNOMES A UNE INCONNUE

1. Lemme de Sylvester

■ Enoncé

Etant donné deux polynômes réels ou complexes à une inconnue de degré p et q , une condition nécessaire et suffisante pour qu'ils aient une racine complexe commune est la nullité du déterminant de Sylvester, qui est un polynôme homogène de degré $p+q$ des coefficients des deux polynômes, noté $\Delta(P, Q)$.

■ Démonstration

1-Soient $P[X]$ et $Q[X]$ deux polynômes réels ou complexes à une inconnue, de degré p et q . Ces polynômes s'écrivent :

$$P[X] = \alpha_0 + \alpha_1 X + \alpha_2 X^2 + \dots + \alpha_p X^p$$

$$Q[X] = \beta_0 + \beta_1 X + \beta_2 X^2 + \dots + \beta_q X^q$$

2-S'ils ont une racine α commune, ces polynômes s'écrivent :

$$P[X] = (X-\alpha)P'[X] ; d^\circ(P') = p-1$$

$$Q[X] = (X-\alpha)Q'[X] ; d^\circ(Q') = q-1.$$

3-Il vient donc, dans l'algèbre des polynômes : $P'Q - PQ' = 0$. Or, en le développant, $P'Q$ est une combinaison linéaire des polynômes $Q, XQ, X^2Q, \dots, X^{p-1}Q$. De même, PQ' est une combinaison linéaire des polynômes $P, XP, X^2P, \dots, X^{q-1}P$.

4-Les polynômes $Q, XQ, X^2Q, \dots, X^{p-1}Q, P, XP, X^2P, \dots, X^{q-1}P$ forment donc un système lié de $p+q$ polynômes dans un espace vectoriel de dimension $p+q$. Le déterminant du système est donc nul. Il est appelé déterminant de Sylvester des deux polynômes, et s'écrit :

$$\Delta(P, Q) = \begin{vmatrix} \alpha_0 & \alpha_1 & \dots & \alpha_{p-1} & \alpha_p & 0 & \dots & \dots & \dots & \dots & 0 \\ 0 & \alpha_0 & \alpha_1 & \dots & \alpha_{p-1} & \alpha_p & 0 & \dots & \dots & \dots & 0 \\ 0 & 0 & \alpha_0 & \alpha_1 & \dots & \alpha_{p-1} & \alpha_p & 0 & \dots & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & \dots & 0 & \alpha_0 & \alpha_1 & \dots & \dots & \alpha_{p-1} & \alpha_p \\ 0 & \dots & \dots & \dots & \dots & 0 & \alpha_0 & \alpha_1 & \dots & \alpha_{p-1} & \alpha_p \\ \beta_0 & \beta_1 & \dots & \dots & \dots & \dots & \beta_{q-1} & \beta_q & 0 & \dots & 0 \\ 0 & \beta_0 & \beta_1 & \dots & \dots & \dots & \dots & \beta_{q-1} & \beta_q & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \dots & 0 & \beta_0 & \beta_1 & \dots & \dots & \dots & \beta_{q-1} & \beta_q \\ 0 & \dots & \dots & \dots & 0 & \beta_0 & \beta_1 & \dots & \dots & \dots & \beta_{q-1} \end{vmatrix}$$

qui est une équation homogène de degré $p+q$ des coefficients des deux polynômes.

2. Deuxième lemme

■ Énoncé

Étant données les deux polynômes P et Q à une inconnue de degré n suivants :

$$\begin{cases} \alpha_n x^n + \alpha_{n-1} x^{n-1} + \dots + \alpha_2 x^2 + \alpha_1 x + \alpha_0 \\ \alpha'_n x^n + \alpha'_{n-1} x^{n-1} + \dots + \alpha'_2 x^2 + \alpha'_1 x + \alpha'_0 \end{cases}$$

une condition nécessaire pour qu'ils aient une racine complexe commune est :

$$\begin{cases} \begin{vmatrix} \alpha_{n-1} & \alpha_n \\ \alpha'_{n-1} & \alpha'_n \end{vmatrix} \cdot x^{n-1} + \dots + \begin{vmatrix} \alpha_1 & \alpha_n \\ \alpha'_1 & \alpha'_n \end{vmatrix} \cdot x^1 + \dots + \begin{vmatrix} \alpha_1 & \alpha_n \\ \alpha'_1 & \alpha'_n \end{vmatrix} \cdot x + \begin{vmatrix} \alpha_0 & \alpha_n \\ \alpha'_0 & \alpha'_n \end{vmatrix} = 0 \\ \begin{vmatrix} \alpha_n & \alpha_0 \\ \alpha'_n & \alpha'_0 \end{vmatrix} \cdot x^{n-1} + \dots + \begin{vmatrix} \alpha_{1+1} & \alpha_0 \\ \alpha'_{1+1} & \alpha'_0 \end{vmatrix} \cdot x^1 + \dots + \begin{vmatrix} \alpha_2 & \alpha_0 \\ \alpha'_2 & \alpha'_0 \end{vmatrix} \cdot x + \begin{vmatrix} \alpha_1 & \alpha_0 \\ \alpha'_1 & \alpha'_0 \end{vmatrix} = 0 \end{cases}$$

Cette condition n'est pas suffisante si $n > 2$.

■ Démonstration

Ce lemme est immédiat. En appelant $(1) = 0$ et $(2) = 0$ les deux polynômes, il résulte de $(1) \cdot \alpha'_n - (2) \cdot \alpha_n = 0$ et de $(1) \cdot \alpha'_0 - (2) \cdot \alpha_0 = 0$ après simplification par x si 0 n'est pas racine commune (donc si $\alpha_0 \neq 0$ et $\alpha'_0 \neq 0$).

■ Remarque

De proche en proche, avec le lemme précédent, on baisse d'une unité à chaque

pas le degré des polynômes ayant une racine commune.

2. RESOLUTION DES SYSTEMES ALGEBRIQUES DE N EQUATIONS A N INCONNUES

L'utilisation du lemme de Sylvester permet de résoudre des systèmes algébriques de n équations algébriques à n inconnues.

La démonstration se fait par récurrence sur n .

1-Supposons que l'on sache résoudre un système de $n-1$ équations algébriques à $n-1$ inconnues.

2-Soit :

$$\begin{cases} P_1(\xi_1, \dots, \xi_n) = 0 \\ \dots \\ P_n(\xi_1, \dots, \xi_n) = 0 \end{cases}$$

un système algébrique de n équations à n inconnues. Nous allons éliminer ξ_n .

3-Chacun des polynômes P_i peut être lu comme un polynôme en ξ_n , dont les coefficients sont des polynômes en ξ_1, \dots, ξ_{n-1} :

$$P_i(\xi_1, \dots, \xi_n) = Q_{0,i}(\xi_1, \dots, \xi_{n-1}) + \xi_n Q_{1,i}(\xi_1, \dots, \xi_{n-1}) + \dots + \xi_n^m Q_{m,i}(\xi_1, \dots, \xi_{n-1})$$

4 - Si $(\xi_{\alpha,1}, \dots, \xi_{\alpha,n})$ est une solution du système, $\xi_{\alpha,n}$ est une racine du polynôme P_i , les $\xi_{\alpha,j}$ étant fixé pour $j = 1, \dots, n-1$.

5 - On peut donc exprimer successivement que P_1 et P_i , pour $i = 2, \dots, n$

ont une racine commune en ξ_n , soit :

$$\left\{ \begin{array}{l} \Delta(P_1, P_2) = 0 \\ \dots \\ \Delta(P_1, P_1) = 0 \\ \dots \\ \Delta(P_1, P_n) = 0 \end{array} \right.$$

6 - Comme $\Delta(P, Q)$ est un polynôme des coefficients de P et Q , que les coefficients de P_1 sont des polynômes en $(\xi_1, \dots, \xi_{n-1})$, le système précédent est un système de $n-1$ équations algébriques à $n-1$ inconnues.

7 - Sachant résoudre un tel système (hypothèse de la récurrence), on sait résoudre un système de n équations à n inconnues en le ramenant à la recherche des racines d'un polynôme à une inconnue.

3. DETERMINANTS DE POLYNOMES

La démonstration précédente fait appel à des déterminants dont les éléments sont des polynômes, qu'il faut calculer.

1. Polynômes à une inconnue

■ Lemme

Tout déterminant d'ordre n dont les éléments sont des polynômes de degré p à une inconnue est un polynôme de degré p dont les coefficients sont des déterminants d'ordre n .

■ Démonstration

1 - Soit donc Δ un tel déterminant :

$$\Delta = \text{Det}[P_{ij}(\xi)]$$

c'est à dire que Δ est le déterminant d'une matrice dont les éléments sont des polynômes P_{ij} en ξ .

2 - Supposons que : $\forall i, j, d^{\circ}(P_{ij}) \leq p$. Ecrivons alors :

$$P_{ij}(\xi) = \sum_{k=0}^p \alpha_{ijk} \xi^k$$

et posons :

$$a_{jk} = \sum_{i=1}^n \alpha_{ijk} e_i$$

3 - On en déduit :

$$\begin{aligned} \Delta &= \text{Det} \left[\sum_{k_1} \xi^{k_1} a_{1k_1}, \dots, \sum_{k_n} \xi^{k_n} a_{nk_n} \right] \\ &= \sum_{k_1} \sum_{k_2} \dots \sum_{k_n} \xi^{k_1} \xi^{k_2} \dots \xi^{k_n} \text{Det} [a_{1k_1}, a_{2k_2}, \dots, a_{nk_n}] \\ &= \sum_{k_1} \sum_{k_2} \dots \sum_{k_n} \xi^{k_1+k_2+\dots+k_n} \text{Det} (a_{1,k_1}, a_{2,k_2}, \dots, a_{n,k_n}) \\ &= \sum_p \xi^p \left[\sum_{k_1+k_2+\dots+k_n=p} \text{Det} (a_{1,k_1}, a_{2,k_2}, \dots, a_{n,k_n}) \right] \\ &= \sum_p \beta_p \xi^p \\ &\text{avec } \beta_p = \sum_{k_1+k_2+\dots+k_n=p} \text{Det} (a_{1,k_1}, a_{2,k_2}, \dots, a_{n,k_n}) \end{aligned}$$

■ Calcul des coefficients

Le résultat précédent permet le calcul des coefficients β_k en faisant appel au calcul de p^n déterminants d'ordre n . La procédure suivante, qui peut être programmée, n'effectue pas le successif des β_k en cherchant les n -uplets $(k_i)_i$ tels que $\sum_i k_i = p$, ce qui serait fastidieux, mais calcule dans un ordre logique tous les $\text{Det}(a_{ij})$ et incrémente les β_k correspondant :

- Pour k de 1 à n : $\beta(k) \leftarrow 0$
- Pour i de 1 à n

Pour j de 1 à p : donnée des vecteurs a_{1j}

- Pour k_1 de 1 à p
 - Pour k_2 de 1 à p
 - ...
 - Pour k_n de 1 à p :
 - Donnée de M ayant pour vecteurs colonnes $(a_{1,k_1}, \dots, a_{n,k_n})$
 - Calcul de $\text{Det}(M)$
 - Calcul de $p = k_1 + k_2 + \dots + k_n$
 - $\beta(p) \leftarrow \beta(p) + \text{Det}(M)$

2. Polynômes à plusieurs inconnues

Les déterminants intervenant dans la résolution des équations algébriques à plusieurs inconnues sont des déterminants dont les coefficients sont des polynômes à plusieurs inconnues.

Nous supposerons dans la suite que les polynômes P sont des polynômes à q inconnues de degré p_1, \dots, p_q pour ces inconnues : $P \equiv P(\xi_1, \dots, \xi_q)$. Le calcul de ces déterminants est alors proche de celui des déterminants de polynômes à une inconnue, et la seule difficulté consiste à conserver la clarté de l'écriture des expressions : il en effet facile de se perdre dans le maniement des multi-indices.

■ Lemme

Tout déterminant d'ordre n dont les éléments sont des polynômes à q inconnues de degré p_i pour la i -ième inconnue est un polynôme à q inconnues de degré p_i pour la i -ième inconnue dont les coefficients sont des déterminants d'ordre n .

■ Démonstration

1 - Soit donc Δ un tel déterminant :

$$\Delta = \text{Det}[P_{ij}(\xi_1, \dots, \xi_q)]$$

c'est à dire que Δ est le déterminant d'une matrice dont les éléments sont des polynômes P_{ij} en ξ_1, \dots, ξ_q .

2 - Supposons que : pour ξ_k , $\forall i, j$, $d^0(P_{ij}) \leq p_k$. Ecrivons alors :

$$P_{ij}(\xi_1, \dots, \xi_q) = \sum_{k_1} \sum_{k_2} \dots \sum_{k_q} \alpha_{ij, k_1, k_2, \dots, k_q} \xi_1^{k_1} \xi_2^{k_2} \dots \xi_q^{k_q}$$

c'est à dire que, dans P_{ij} , $\alpha_{ij, k_1, k_2, \dots, k_q}$ est le coefficient réel du monôme $\xi_1^{k_1} \xi_2^{k_2} \dots \xi_q^{k_q}$, que nous écrivons pour simplifier l'écriture $\alpha(i, j, \vec{k})$ où \vec{k} est un vecteur de $\{1, p_1\} \times \{1, p_2\} \times \dots \times \{1, p_q\}$. Adoptons comme autre convention :

$$\xi^{\vec{k}} = \xi_1^{k_1} \xi_2^{k_2} \dots \xi_q^{k_q} \text{ où } \vec{k} = (k_1, \dots, k_q)$$

D'où :

$$P_{ij}(\xi_1, \dots, \xi_q) = \sum_{\vec{k}} \alpha(i, j, \vec{k}) \xi^{\vec{k}}$$

3 - Posons alors :

$$a(j, \vec{k}) = \sum_{i=1}^n \alpha(i, j, \vec{k}) e_i$$

4 - D'où :

$$\begin{aligned} \Delta &= \text{Det}(P_{ij}) \\ &= \text{Det} \left[\sum_{\vec{k}} \alpha(i, j, \vec{k}) \xi^{\vec{k}} \right] \\ &= \text{Det} \left[\sum_{\vec{k}_1} a(1, \vec{k}_1) \xi^{\vec{k}_1}, \dots, \sum_{\vec{k}_n} a(n, \vec{k}_n) \xi^{\vec{k}_n} \right] \\ &= \sum_{\vec{k}_1} \sum_{\vec{k}_2} \dots \sum_{\vec{k}_n} \xi^{\vec{k}_1} \xi^{\vec{k}_2} \dots \xi^{\vec{k}_n} \text{Det} \left[a(1, \vec{k}_1), \dots, a(n, \vec{k}_n) \right] \\ &= \sum_{\vec{k}_1} \sum_{\vec{k}_2} \dots \sum_{\vec{k}_n} \xi^{\vec{k}_1 + \vec{k}_2 + \dots + \vec{k}_n} \text{Det} \left[a(1, \vec{k}_1), \dots, a(n, \vec{k}_n) \right] \\ &= \sum_{\vec{l}} \xi^{\vec{l}} \left[\sum_{\vec{k}_1 + \vec{k}_2 + \dots + \vec{k}_n = \vec{l}} \text{Det} \left[a(1, \vec{k}_1), \dots, a(n, \vec{k}_n) \right] \right] \\ &= \sum_{\vec{l}} \beta_{\vec{l}} \xi^{\vec{l}} \end{aligned}$$

avec $\beta_{\vec{l}} = \sum_{\vec{k}_1 + \vec{k}_2 + \dots + \vec{k}_n = \vec{l}} \text{Det} \left[a(1, \vec{k}_1), \dots, a(n, \vec{k}_n) \right]$

■ Calcul des coefficients

Le résultat précédent permet le calcul des coefficients β_k en faisant appel au calcul de p^n déterminants d'ordre n , selon la procédure suivante où, par rapport au calcul successif des β_k en cherchant les n -uplets $(k_i)_1$ tels que $\sum_1 k_i = p$, on calcule dans un ordre logique tous les $\text{Det}(a_{ij})$ et incrémente les β_k correspondant :

- Pour k de 1 à n : $\beta(k) \leftarrow 0$
- Pour i de 1 à n
 - Pour j de 1 à p : donnée des vecteurs a_{ij}
- Pour k_1 de 1 à p
 - Pour k_2 de 1 à p
 - ...
 - Pour k_n de 1 à p :
 - Donnée de M ayant pour vecteurs colonnes $(a_{1,k_1}, \dots, a_{n,k_n})$
 - Calcul de $\text{Det}(M)$
 - Calcul de $p = k_1 + k_2 + \dots + k_n$
 - $\beta(p) \leftarrow \beta(p) + \text{Det}(M)$

□

We first present multilinear tools, thought to be of basic importance in data analysis of multiway arrays (MWA). We then recall results in Component Analysis of matrices. Generalisation leads either to TUCKER models or to PARAFAC/CANDECOMP models.

Although a great confusion may be noticed in litterature, we believe that those two families of models should be split apart.

We then develop the study of TUCKER-3 model, through a simplification of KROONENBERG & DE LEEUW (1981) proof of the (local) convergence algorithm. TUCKER model is easely generalisable to TUCKER-n, with metrics on the n spaces, and linear constraints on components.

Next chapter is devoted to the study of PARAFAC/CANDECOMP models, with a proposition of an algorithm to compute on a given MWA such a model with a given non trivial metric on each space and linear constraints on components. This may be looked at as a extension of CANDELINC to non trivial metrics.

Rank of a MWA is a crucial notion, and bibliographic overview of inequalities on rank is given, with an algorithm to improve most of these results.

Finally, special attention is given to MWA on \mathbb{R}^2 , because computation may go on to the end through easely solvable algebraic equations. A consequence is proposed to compute multiplicative polynomial models with independant factors. We conclude by opening the way to an easy study of MWA with algebraic constraints.

PRESENTATION OF MULTILINEAR TOOLS

None of these tools is new. Most of them go back to the development of linear algebra in XIXth century. We simply recall main definitions and properties thought to be of some help in increasing simplicity when solving numerical analysis problems on multiway arrays (MWA).

Tensor product

Tensor product and related multilinear and linear operators are the key algebraic tools in this study, which allow with a minimum of notation complications and computational difficulties the generalisation of component analysis to multiway arrays. Unfortunately, very useful concise notations for matrix algebraic computations are strictly related to matrix spaces, such as scalar product : $\langle A, B \rangle = \text{Tr}(A'B)$ and others.

Tensor product is introduced in a very simple way, as a multilinear form : $a \otimes b(x, y) = \langle a, x \rangle \langle b, y \rangle$, which can easily be generalized to p-linear forms : $a_1 \otimes \dots \otimes a_p(x_1, \dots, x_p) = \langle a_1, x_1 \rangle \dots \langle a_p, x_p \rangle$.

Any array is then looked at as a multilinear form on an adequate space, or a tensor. For example, a matrix is written as $\sum_{ij} \alpha_{ij} e_i \otimes e_j$, and so on. We note by \mathcal{A}_q the vector space of q-multilinear forms.

Metric properties of tensor spaces

A tensor is then a vector in a tensor space. All properties of finite dimensional vector spaces can be drawn, especially useful metric properties. Metric properties are important in our study because the main objective is to approximate a given MWA by another one restricted in a given submanifold of the set of MWA.

For a matrix, for example, the result naturally leads to the Frobenius norm : $\langle A, B \rangle = \sum_{ij} \alpha_{ij} \beta_{ij}$, which is easily generalizable : $\langle A, B \rangle = \sum_i \alpha_i \beta_i$.

One special feature is the canonical link between a set of metrics M_i on vector spaces \mathcal{E}_i and a metric T on $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_n$ as follow :

$$\langle A, x_1 \otimes \dots \otimes x_n \rangle_T = \langle A, T.(x_1 \otimes \dots \otimes x_n) \rangle = \langle A, (M_1.x_1) \otimes \dots \otimes (M_n.x_n) \rangle$$

Inner product

Inner product is the dual operator to tensor product. It is closely related to tensor contraction (but not identical, however, because we deliberately do not distinguish covariant and contravariant tensors). The definition is then : $\langle X, Y \otimes Z \rangle = \langle X.Y, Z \rangle$. This is an easy generalisation of product of a matrix by a vector : $A.x = y$.

Rapid computation rules may easely be drawn :

- if $X \in \mathcal{A}_{p+q}$ and $Y \in \mathcal{A}_p$, then $X.Y \in \mathcal{A}_q$
- $A.(B+C) = A.B + A.C$
- $A.(B \otimes C) = (A.B).C$
- $(A \otimes B).C = (A.C) \otimes B$

which are quite useful in concise computation and avoiding use of indices and coefficients.

Permutation

Permutation is another useful algebraic operator, which generalizes transposition of matrices. If σ is the following permutation : $\sigma(1,2,3) = (2,1,3)$, for example, then $\sigma(x \otimes y \otimes z) = y \otimes x \otimes z$, which can be generalized to \mathcal{A}_3 through linearity.

Conclusion : a combination of these operators

Most of computational problems require combinations of these operators. For example, CARROLL, PRUZANSKY & KRUSKAL (1981) while presenting CANDELINC model define an operator \circ , in which a combination of permutation and inner product can easely be recognized.

Our conclusion is that an operative and concise algebraic notation is unavoidable while seeking generalization of component analysis to MWA. Multilinear algebra is a good candidate.

COMPONENT ANALYSIS ON MATRICES

The problem that is to be solved

We recall that component analysis is the following problem : For a given A of $\mathcal{E} \otimes \mathcal{F}$, find out \hat{A} of $\mathcal{E} \otimes \mathcal{F}$ of rank r , with r chosen in advance, such as $\|A - \hat{A}\|$ minimum. There are several ways for writing \hat{A} , such as :

$$\square \hat{A} = XY'$$

$$\square \hat{A} = \sum_{i=1}^r x_i \otimes y_i \text{ or } \hat{A} = \sum_{i=1}^r \lambda_i x_i \otimes y_i \text{ with } \|x_i\| = \|y_i\| = 1$$

$$\square \hat{a}_{ij} = \sum_k \xi_{ik} \eta_{jk}$$

The solution

It is well known that the solution is given through the Singular Value Decomposition of A , as follow :

$$\hat{A} = \sum_{i=1}^r \lambda_i x_i \otimes y_i \text{ with } A'A.x_i = \lambda_i x_i \text{ and } y_i = A.x_i$$

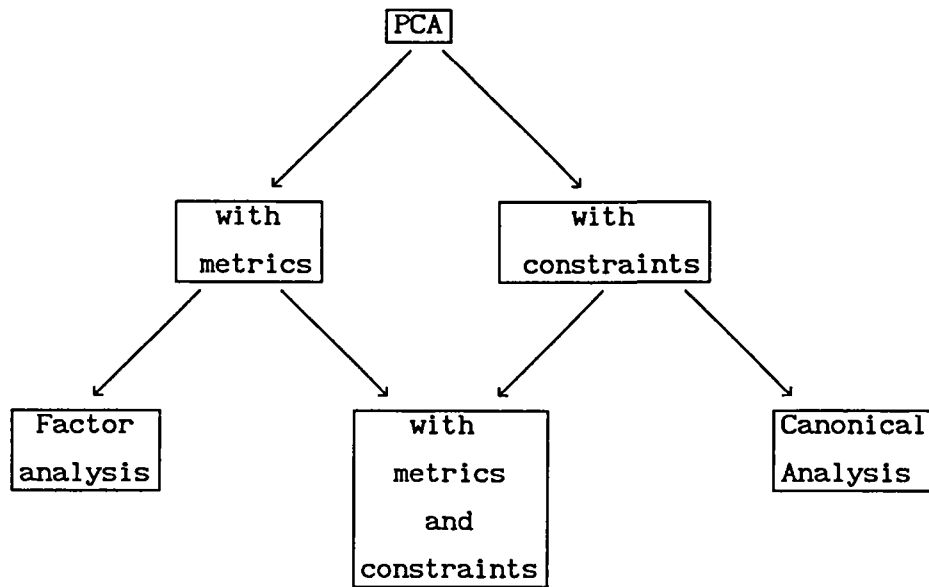
Extension of PCA

Two main extensions are well known.

First one takes metrics into account, with metrics on \mathcal{E} and \mathcal{F} defined by symmetric definite positive matrices M and N respectively. One special choice of adequate metrics related to datas leads naturally to Factor Analysis.

Secondly one takes linear constraints on component into account, due to belong to given subspace of \mathcal{E} and \mathcal{F} respectively. One special choice of constraints related to datas leads to canonical analysis.

We have the following diagramm :



Solutions of PCA with metrics and linear constraints are well known, and can easily be recovered through tensor computation.

EXTENSION FROM MATRICES TO MWA

Extension of PCA to MWA may start from two different angles of approach. Let us consider the following PCA model :

$$\hat{A} = \sum_{i=1}^r x_i \otimes y_i$$

□ First, we may consider the vector subspaces spanned by the $(x_i)_i$ in \mathcal{E} and the (y_i) in \mathcal{F} , noted \mathcal{E}' and \mathcal{F}' . \hat{A} simply is the orthogonal projection of A on the Subspace $\mathcal{E}' \otimes \mathcal{F}'$ of $\mathcal{E} \otimes \mathcal{F}$. This leads to TUCKER models : if p and q are given integers, find out \mathcal{E}' and \mathcal{F}' , subspaces of \mathcal{E} and \mathcal{F} of dimension p and q , such as $\|A - \mathcal{P}(A)\|$ is minimum, if \mathcal{P} denotes orthogonal projection on $\mathcal{E}' \otimes \mathcal{F}'$. Generalisation to MWA will be easy and natural.

□ Second, we may consider that \hat{A} is the sum of r rank one matrices. Hence the problem : for a given integer r , find out the best approximation of A as

the sum of r rank one matrices. This is PARAFAC/CANDECOMP problem, and is easily generalised to MWA too.

As a matter of fact, it is well known that TUCKER models and PARAFAC/CANDECOMP models are identical for matrices, and both represent PCA. But, the fact is that they are no more identical models for n -way arrays, with $n > 2$.

TUCKER MODELS

Definition

If A is a MWA from $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$, and (r_1, r_2, r_3) a set of integers, then a TUCKER model for A (3TM) is the best approximation of A as a MWA in $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$, where $\dim(\mathcal{F}_i) = r_i$. If \mathcal{P} denotes orthogonal projection of $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$ onto $\mathcal{F}_1 \otimes \mathcal{F}_2 \otimes \mathcal{F}_3$, then $\|A - \mathcal{P}(A)\|$ is minimal.

If \mathcal{E}_i is an euclidian vector space, with metrics defined by matrix M_i , then \mathcal{P} is the orthogonal projection with the canonically associated metrics on $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$.

If linear constraints are given through the fact that 3TM components are due to belong to given subspaces \mathcal{G}_i , then $\mathcal{F}_i \subset \mathcal{G}_i$.

Solution

The unknowns are not vectors, but subspaces \mathcal{F}_i . Solutions will be found through iterative alternative best approximation to subspaces. If, for example, \mathcal{F}_2 and \mathcal{F}_3 are given, and we are looking for the subspace \mathcal{F}_1 such that $\|A - \mathcal{P}(A)\|$ minimum, then \mathcal{F}_1 is spanned by the r_1 first components of PCA of A under linear constraint (LCPCA) given by $\mathcal{F}_2 \otimes \mathcal{F}_3$, when keeping in mind natural isomorphism between $\mathcal{L}(\mathcal{E}_1, \mathcal{E}_2 \otimes \mathcal{E}_3)$ and $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$. Iterative computation is then easy to set up, with help of permutations.

If \mathcal{E}_i have euclidian structures produced by matrices M_i , then each

iterative stage will be a LCPCA with metrics. If 3TM components are due to belong to a given \mathcal{F}_1 subspace, than LCPCA will be with double constraints.

If, for example, f_1 is an orthonormal basis for \mathcal{F}_1 , then $\mathcal{P}(A) = \sum_{ijk} \lambda_{ijk} f_{1i} \otimes f_{2j} \otimes f_{3k}$, (λ_{ijk}) being the core matrix (in fact the core MWA).

PARAFAC / CANDECOMP MODELS

Definition

If A is a MWA from $\mathcal{E}_1 \otimes \mathcal{E}_2 \otimes \mathcal{E}_3$, then a Parafac/Candecomp (PCM) model for A is the best approximation of A as a sum of r (given integer) rank one MWA. It can be written : $A = \sum_i x_i \otimes y_i \otimes z_i + R$, with $\|R\|$ minimum. Extensions of PCM proceed as follow : PCM with metrics if \mathcal{E}_1 is an euclidian vector space with metrics defined by matrix M_1 , and PCM with linear constraints if vectors x_i are due to belong to a subspace \mathcal{F}_1 , y_i to a subspace \mathcal{F}_2 and z_i to a subspace \mathcal{F}_3 .

Solution

Solution of PCM is well known to be obtained through alternative vectorial least square algorithm on the (x_i) , (y_i) and (z_i) vector sets. A new concise solution, easly generalizable to PCM extensions and n-way PCM (with $n > 3$) is presented. If, for example, we denote by D_1 the matrix $x_i \otimes y_i$, then the best approximation for z_i with given (x_i) and (y_i) vectors is as follow : $z_i = A.D_1^*$, D_1^* being the dual matrix of D_1 : $\langle D_1^*, D_j \rangle = \delta_{ij}$. This allows iterative computation, combined with permutations.

Unfortunately, this procedure does converge, but very slowly and may be unefficient.

If \mathcal{E}_1 has an euclidian structure determined by a matrix M_1 , iterative computation should be modified as follow : $z_i = A.(V(V.(x_i \otimes y_i)))^*$, where V is a matrix built from M_1 and M_2 . This allows computation of each iterative step.

A more easily computable procedure is then proposed from isometry between some spaces, which leads to computation of a model with metrics defined by unity matrices, which is simpler.

Taking into account linear constraints on the MWA A need only to project A on the space induced by the constraints : the solution is then the best approximation of this projected MWA.

Taking onto account linear constraints and metrics is then easily done.

Application

An application of first component of PCM can be found in multiplicative polynomial models (MPM). If a polynomial regression $\eta = f(\xi_1, \xi_2, \xi_3)$ does significantly exist, finding out the best MPM $\eta = P_1(\xi_1) \cdot P_2(\xi_2) \cdot P_3(\xi_3)$ is a consequence of finding out the first component of PCP on a MWA built from the polynomial regression, with dimension on each way equal to the degree of the polynomials plus one.

GENERALIZATION TO N-WAY ARRAYS

N-MWA is an array on $\mathcal{E}_1 \otimes \dots \otimes \mathcal{E}_n$, and such MWA are often encountered in applications (multiple contingency tables, for example).

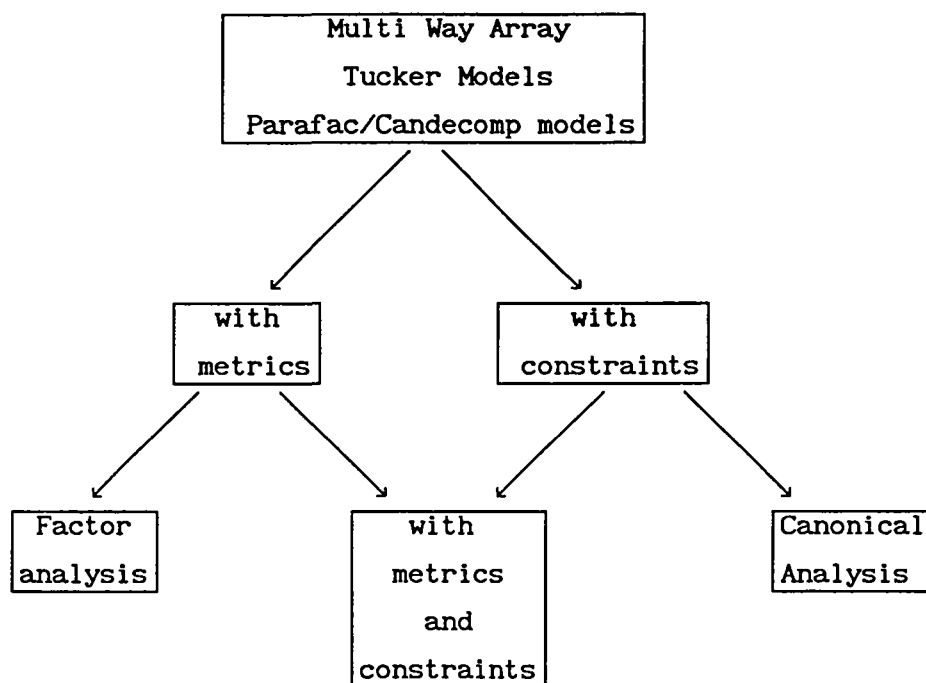
Constant use of concise algebraic tools allows easy generalization of previous results to MWA, with more than three ways. For example, such a generalization is proposed for PCM, with metrics and linear constraints. Proposed algorithms lead to computation of solutions.

For example, isometry allows computations of PCM models with given metrics with a procedure which requires only one computation of a PCM with identity metrics.

Such a generalization is also proposed for TUCKER models, with

iterative alternative algorithms fo TM with metrics and linear constraints which allow computation of solutions, too. Isometry is still very useful.

Finally, we can propose the following models for study of MWA :



RANK INEQUALITIES

Study of rank inequalities is not a compulsory stage of component analysis of MWA, but it enlightens non linearity of these analyses as well as deep differences with PCA on matrices.

HISTORICAL BACKGROUND

PCA has been developped since the beginning of the century. The algebraic solutions go back to ECKART & YOUNG (1936), although the result has been well known to mathematician since SCHMIDT's work in 1906 on kernels of linear fuctional operators. First publications taking linear constraints into account seem to be by RAO (1964) and CARROLL & Al. in 1978, although best approximation of eigenvalues in given subspaces has been known to

mathematicians too since Lord RALEIGH's work in 1899 on vibration modes and RITZE's work in 1906.

TUCKER models have been presented as decomposition models by TUCKER in the 1963-1966 years, with work by LEVIN too. Numerical solution of best approximation algorithm was first published by KROONENBERG & DE LEEUW in 1980. Generalization to n-way TUCKER models was proposed simultaneously by KAPTEYN & Al. in 1986, although in a purely algebraic presentation without link with numerical tools, and by POLIT, with FORTRAN programs. Taking metrics into account was designed by D'AUBIGNY & POLIT in 1989. We have not heard of algorithms with TUCKER models dealing with linear constraints (although some assumptions were made in CARROLL & Al. paper on CANDELINC).

PARAFAC model started with HARSHMANN papers in 1970, simultaneously with CARROLL & CHANG paper on CANDECOMP, both studying the same model with two closely related algorithms enlarging WOLDE & NILES's algorithm on matrices. This algorithm was known to ANDERSON in 1958. Extension of CANDECOMP model to n-way array was part of 1970's article. Linear constraints were taken into account in presentation of CANDELINC model, by CARROLL, PRUZANSKY & KRUSKAL in 1981. We have not noticed in literature studies with given non trivial metrics.

Comparison between TUCKER and PARAFAC/CANDECOMP models in literature is rather confusing, and the difference between them, and evidence that none can be reduced as a particular case of the other, was long to become clear.

Rank studies date back to the beginning of the century too, with work by HITCHCOCK in 1926. But most intensive research was made in the 70's, related to algebraic computational complexity, with many contributions by several authors, among whom ATKINSON, BROCKETT, DOBKIN, KRUSKAL, HOWELL, LAFON and others. Often cited pioneering publication is STRASSEN's in 1968. The only well known case is $2 \times n \times n$ case since JA' JA works in 1978, with matrix pencil tools.

Personal contribution

A great deal of the results proposed in this work deeply root in a

field of intensive research since the end of the sixties, and are already known to specialists. But, we do not know a general synthetic presentation of all these results, as complete as possible with survey of different research directions, such as non trivial metrics, linear constraints, and extensions to MWA with more than 3 ways, including a mathematical unifying tool allowing clear understanding of all the choices involved.

Use of isometries and projectors to study models with given metrics and linear constraints seems new to us, although it is a very natural procedure (once it has been written down !)

This complete and simple presentation is what we aimed at, proposing personal contributions especially in simplifying and unifying published algorithms, introducing isometries and projectors as useful tools, proposing new one for n way arrays and little progress in difficult field of rank inequalities.

BIBLIOGRAPHIE

Anderson T. W. : - 1958 - An introduction to multivariate statistical analysis. John Wiley & Sons, New York.

Atkinson M. D. & Lloyd S. : - 1980 - Bounds on the ranks of some 3-Tensors. *Linear Algebra and Appl.* 31 19-31.

Atkinson M. D. & Stephens N. M. : - 1979 - On the Multiplicative Complexity of a Family of Bilinear Forms. *Linear Algebra and Appl.* 27 1-8.

Benzecri J.-P. : - 1973 - L'analyse des données. T I : La Taxinomie ; T II : L'analyse des correspondances. Dunod, Paris.

Berger M. & Gostiaux M : - 1987 - Géométrie différentielle : variétés, courbes et surfaces. PUF, Paris. 511 p.

Bertin J.-E. & Bertin M.-J. : Algèbre linéaire et géométrie classique. Masson, 1981. 148 pp.

Bonifas L., Escoufier Y., Gonzales P.-L. & Sabatier R. : - 1984 - Choix de variables en analyse de composantes principales. *Rev. Stat. Appl.* XXXII(2), 5-15.

Bourbaki N. : - 1984 - Eléments d'histoire des mathématiques. Masson, Paris, 376 pp.

Brillouin L. : - 1938 - Les tenseurs en mécanique et en élasticité. Masson, Paris. Réimpression Jacques Gabay, Paris, 1987. 370 pp.

Brockett W & Dobkin D : - 1976 - On the number of multiplications required for matrix multiplication. *SIAM J. Comput.* 5(4), 624-628.

----- : - 1978 - On the optimal evaluation of a set of

bilinear forms. *Linear Algebra and Appl.*, **19**, 207-235.

Cailliez F. : - 1984 - Analyse des données. Presses de l'université de Montréal, **87**, 103 p.

Cailliez F. & Pagès J.-P. : - 1976 - Introduction à l'analyse des données. SMASH, Paris, 616 pp.

Carroll J. D. & Chang J. J. : - 1970 - Analysis of individual differences in multidimensional scaling via n-way generalization of "Eckart-Young" decomposition. *Psychometrika*, **35**, 283-319.

Carroll J. D., Pruzansky S. & Kruskal J. B. : - 1980 - CANDELINC : a general approach to multidimensional analysis of many-way arrays with linear constraints on parameters. *Psychometrika*, **45**, 3-23.

Cazes P. : - 1970 - Application de l'analyse des données à l'étude de problèmes géologiques. Thèse 3^{ème} cycle, Paris VI.

----- : - 1980 & 1981 - L'analyse de certains tableaux rectangulaires décomposés en blocs : généralisation des propriétés rencontrées dans l'étude des correspondances multiples. I : Définitions et applications à l'analyse canonique des variables qualitatives. *C.A.D.*, **V(2)**, 145-161 ; II : Questionnaires : variantes de codage et nouveaux calculs de contributions. *C.A.D.*, **V(4)**, 387-403 ; III : Codage simultané de variables qualitatives et quantitatives. *C.A.D.*, **VI(1)**, 9-18 ; IV : Cas modèle. *C.A.D.*, **VI(2)**, 135-143.

Chambadal L. & Ovaert J.-L. : - 1968 - Algèbre linéaire et algèbre tensorielle. Dunod, 544 p.

Charles B. & Allouch D. : - 1984 - Algèbre générale. PUF, Paris. 336 pp.

Chatelin F. : - 1988 - Valeurs propres de matrices. Masson, Paris. 223 pp.

Chen C. W. : - 1974 - An optimal property of principal components. *Comm.*

Stat., 3(10) 979-983.

Coppi R. & Bolasco S. (Ed.) : - 1989 - Multiway Data Analysis. Elsevier - North Holland, Amsterdam. 552 pp.

Darroch J. N. : - 1965 - An optimal property of principal components. *Ann. Math. Stat.*, 36, 1579-1582.

d'Aubigny G : - 1989 - L'analyse multidimensionnelle des données de dissimilarité. Thèse, Université Joseph Fourier, Grenoble. 485 pp.

----- & Polit E. : - 1989 - Some optimality properties of the generalization of the Tucker method to the analysis of n-way tables with specified metrics. in Multiway Data Analysis, Coppi & Bolasco Ed., Elsevier, North Holland. Amsterdam. p. 39-52.

Deheuvels R. : - 1981 - Formes quadratiques et groupes classiques. PUF, Paris. 456 p.

Denis J.-B. : - 1983 - Interaction entre deux facteurs. Thèse INA Paris Grignon. 223 p.

----- : - 1991 - Ajustement de modèles linéaires et bilinéaires sous contraintes linéaires avec données manquantes. *Rev. Stat. App.*, XXXIX(2) 5-25.

Denis J.-B. & Dhorne T. : - 1989 - Orthogonal Tensor Decomposition of 3-Way Tables. in Multiway Data Analysis, Coppi R. & Bolasco Ed., Elsevier, North Holland. Amsterdam. pp. 31-37.

Diday E., Lemaire J., Pouget J., Testu F. : - 1982 - Eléments d'analyse des données. Dunod, Paris, 462 pp.

Dieudonné J. : -1970 - Eléments d'analyse. Tome I, éléments d'algèbre linéaire, pp. 367-380 ; Tome III, compléments d'algèbre, pp. 337-358. Gauthier Villars, Paris.

----- : - 1974 - Cours de géométrie algébrique. PUF, Paris. Tome I, 234 pp.; tome II, 222 pp.

----- : - 1978 - Abrégé d'histoire des mathématiques. Hermann, Paris, 517 pp.

Dobkin D. : - 1973a - On the arithmetic complexity of a class of arithmetic computations, Res. Rep. # 23, Department of Computer Science, Yale University.

----- : - 1973b - On the optimal evaluation of a set of n-linear forms. Conference Record, 14th Annual Symposium on Switching and Automata theory. Iowa City. pp 92-102.

Dodson C. T. J. & Poston T. : - 1977 - Tensor geometry. Pitman, London. 598 p.

Dubrovin B., Novikov S. & Fomenko A. : - 1982 - Géométrie contemporaine. Deuxième partie : géométrie et topologie des variétés. Mir, Moscou. 371 p.

Eckart C. & Young G. : - 1936 - The approximation of one matrix by another of lower rank. *Psychometrika*, 1, 211-218.

Eddington A. S. : - 1922 - The mathematical theory of relativity. Cambridge University Press. Réimpression par Chelsea Publishing Company, New York. Sans date. 270 pp.

Escoufier Y. : - 1973 - Le traitement des variables vectorielles. *Biometrics*, 29, 751-760.

----- : - 1982 - L'analyse des tableaux de contingence simples et multiples. *Metron*, XL(1-2), 53-77.

----- : - 1985 - L'analyse des correspondances : ses propriétés et ses extensions. Institut International de Statistique. Amsterdam.

----- : - 1987 - The duality diagram. A means for better practical applications. in Developments in numerical ecology, Legendre Ed., Springer Verlag, p. 139-156.

----- : - 1988 - Beyond Correspondance analysis. In H. H. Bock, Ed., Classification and related methods of data analysis. Elsevier, 505-514.

Forsythe G. E. : - 1968 - On the asymptotic directions of the s-dimensional optimum gradient method. *Numerische Matematik*, 11 57-76.

Foucart T : - 1984 - Analyse factorielle de tableaux multiples. Masson, Paris, 185 pp.

Franc A. : - 1989 - Multiways arrays, some algebraic remarks ; in Multiway Data Analysis, Coppi & Bolasco Ed., Elsevier, North Holland. Amsterdam. p. 19-29.

Gastinel N. : - 1971 - Sur le calcul des produits de matrices. *Num. Math.* 17 222-229.

----- : - 1972 - Le rang tensoriel d'un ensemble de matrices. *Séminaire d'Analyse Numérique de l'Université de Grenoble.* 159.

Glacon F., : - 1981 - Analyse conjointe de plusieurs matrices de données. Comparaison de différentes méthodes. Thèse 3^e cycle, Université Scientifique et Médicale de Grenoble.

Grace J. H. & Young A. : - 1903 - The algebra of invariants. Cambridge University Press. Reimpression par Chelsea Publishing Company, New York. Sans date. 384 pp.

Grandmaître F. R. : - 1966 - Théorie des matrices. T.I : Théorie générale, 350 p ; T.II : Questions spéciales & applications, 268 p. Dunod.

Henderson H. V. & Searle S. R. : - 1981 - The Vec-permutation matrix, the Vec-operator and Kronecker product : a review. *Linear & multilinear algebra*, 9, 271-288.

Hopcroft J. E. & Kerr L. R. : - 1971 - On minimizing the numbers of multiplications necessary for matrix multiplication. *SIAM J. Appl. Math.* 20 30-36.

Householder A. S. : - 1964 - The theory of matrices in numerical analysis. Blaisdell, New York. Réédité par Dover Publications Inc (1975). 257 pp.

Howell T. D. : - 1976 - Tensor rank and complexity of bilinear forms. PhD Thesis, Cornell University (cité dans HOWELL, 1978).

----- : - 1978 - Global properties of tensor rank. *Linear Algebra and Appl.* 22, 9-23.

Ja' Ja' J. : - 1979 - Optimal evaluation of pairs of bilinear forms. *SIAM J. Comput.* 8, n°3, 443-462.

Jaffrenou P. A. : - 1978 - Sur l'analyse des familles finies de variables vectorielles. Bases algébriques et application à la description statistique. Thèse 3^e cycle, Université Claude Bernard, Lyon I, 97 pp.

Kapteyn A., Neudecker H. & Wansbeek T. : - 1986 - An approach to n-mode component analysis. *Psychometrika*, 51(2), 269-275.

Kiers H. A. L. : - 1988 - Comparaison of Anglo-Saxon and French three-way methods. *Statistique et Analyse de Données*, 13, 14-32.

----- : - 1989 - Three-way methods for the analysis of qualitative and quantitative two-way data. DSWO Press, Leiden, Pays-Bas. 185 p.

Kroonenberg P. M. : - 1983 - Three mode principal component analysis. Theory and applications. DSWO press, Leiden, Pays-Bas. 399 p.

----- : - 1989a - The analysis of multiple tables in factorial ecology. III. Thre-mode principal component analysis : "Analyse triadique complète". *Acta Oecologica, Oecol. Gener.*, 10, n°3, 245-256.

----- : - 1989b - Singular value decomposition of interactions in three way contingency tables. In *Multiway Data Analysis*, R. Coppi & S. Bolasco Ed., Elsevier, pp 169-184.

----- & de Leeuw J. : - 1980 - Principal component analysis of three-mode datas by means of alternating least square algorithms. *Psychometrika*, 45, 69-97.

Kruskal J. B. : - 1977 - Three way arrays : rank and uniqueness of trilinear decomposition, with applications to arithmetic complexity and statistics. *Linear Algebra and Appl.* 18 95-138.

----- : - 1984 - Multilinear Methods. In Law H. G. & Al, Ed., *Research methods for multimode data analysis*. Praeger, 1984, pp. 36-62.

----- : - 1989 - Rank, decomposition, and uniqueness for 3-way and N-way arrays. in *Multiway Data Analysis*, Coppi & Bolasco Ed., Elsevier, North Holland. Amsterdam. pp. 7-18.

Kurosh A. G. : - 1967 - Algèbre générale. Dunod, Paris, 325 p.

Laderman J. D. : - 1976 - A non commutative algorithm for multiplying 3x3 matrices using 23 multiplications. *Bull. AMS* 82, 126-128.

Lafon J. C. : - 1975 - Optimum computation of p bilinear forms. *Linear Algebra and Appl.*, 10, 225-240.

Lang S. : - 1984 - Algebra. Second edition. Addison Wesley, Reading. 713 p.

Lastovicka J. L. : - 1981 - The extension of component analysis to four-mode matrices. *Psychometrika*, 46, 47-57.

Lavit Ch. : - 1988 - Analyse conjointe de tableaux quantitatifs. Masson, Paris, 251 p.

Law H. G., Snyder C. W. Jr, Hattie J. A. & Mc Donald R. P. (Ed.) : - 1984 - Research methods for multimode data analysis. Praeger, 1984, 686 p.

Lebart L. & Fénelon J.-P. : - 1975 - Statistique et informatique appliquée. Dunod, Paris, 439 p.

----- , Morineau A. & Fénelon J.-P. : - 1982 - Traitement des données statistiques. Méthodes et Programmes. Dunod, Paris, 510 p.

Levi-Civita T. : - 1926 - Lezioni di calcolo differenziale assoluto. Traduit en anglais : The absolute differential calculus, Blackie & Son, London, 1926. Réédité par Dover Publications Inc., New York, 1977. 452 pp.

L'Hermier des Plantes H. : - 1976 - Structuration des tableaux à trois indices de la statistique. Thèse de 3^{ème} cycle, Montpellier II.

Lichnerowicz A : - 1946 - Eléments de calcul tensoriel. Armand Colin, Paris. 216p.

Lyttekens E. : - 1966 - On the fixed point property of Wold's iterative estimation method for principal components. in Krishnaiah P. R. Ed., Multivariate Analysis, Academic Press, New York. pp 335-350.

Luenberger D. G. : - 1973 - Introduction to linear and non linear programming. Addison Wesley.

Malliavin M.-P. : - 1985 - Algèbre commutative. Applications en géométrie et théorie des nombres. Masson, Paris. 250 p.

Malliavin P. : - 1972 - Géométrie différentielle intrinsèque. Hermann, Paris. 307 p.

Millsap R. E. : - 1988 - Component analysis in cross-sectional and

longitudinal data. *Psychometrika*, 53(1), 123-134.

Minoux M. : - 1983 - Programmation mathématique : théorie et algorithmes. Dunod, Paris. Tome 1. 294 p.

Mizère D. : - 1981 - Analyse d'un cube de données : décomposition tensorielle et liens entre procédures de comparaison de tableaux rectangulaires de données. *Thèse*. Université de Grenoble.

Mirsky L. : - 1960 - Symmetric gauge functions and unitarily Invariant norms. *Quarterly Journal of Mathematics*, 11, 50-59.

----- : - 1963 - Results and Problems in the Theory of Doubly Stochastic Matrices. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*. 1, 319-334.

Northcott D. G. : - 1968 - Lessons on Rings Modules and Multiplicities. Cambridge University Press, London. 444 pp.

Pagès J.-P., Escoufier Y., Cazes P. : - 1976 - Opérateurs et analyse des tableaux à plus de deux dimensions. *Cahiers du B.U.R.O.* 25 61-89.

Parlett B. N. : 1980 - The symmetric Eigenvalue Problem. Prentice Hall Series in Computational mathematics. Prentice Hall, Inc., Englewood Cliffs. 348 p.

Polit E. : - 1986 - Une n-ACP d'un hypercube de données. *Thèse*, Université de Grenoble.

Postnikov M. : - 1981 - Leçons de géométrie. Algèbre linéaire et géométrie différentielle. Editions MIR, Moscou, 263 p.

Press W. H., Flannery B. P., Teukolsky S. A. & Vetterling W. T. : - 1989 - Numerical recipes in PASCAL. The art of scientific computing. Cambridge University Press, 759 p.

- Rao C. R. : - 1964 - The use and the interpretation of principal component analysis in applied research. *Sankya*, A-26, 329-359.
- : - 1979 - Separation theorems for Singular Values of Matrices and their applications in Multivariate Analysis. *Journal of Multivariate Analysis*, 9, 362-377.
- : - 1980 - Matrix approximations and reduction of dimensionality in multivariate statistical analysis. In P. R. Krishnaiah Ed., *Multivariate Analysis* 5, 3-22.
- Robert P. & Escoufier Y. : - 1976 - A unifying tool for linear multivariate statistical methods : the RV coefficient . *Appl. Stat.* 15(3), 257-265.
- Ruthisauer H. & Bauer F. L. : - 1955 - Détermination des vecteurs propres d'une matrice par une méthode itérative avec convergence quadratique. *C. R. A. S.*, 240, 1680-1681.
- Sabatier R : - 1983 - Approximations d'un tableau de données. Application à la reconstitution des paléoclimats. Thèse 3^{ème} cycle. USTL, Montpellier.
- : - 1984 - Quelques généralisations de l'analyse en composantes principales de variables instrumentales. *Stat. & Ann. Donn.*, 9(3) 75-103.
- : - 1987 - Méthodes factorielles en analyse des données : approximations et prise en compte de variables concomitantes. Thèse d'Etat, USTL, 242 p.
- , Jean Y. & Escoufier Y. : - 1984 - Approximation d'applications linéaires et analyse en composantes principales. in *Data Analysis & Informatics*. North Holland. 569-580.
- , Lebreton J. D. & Chessel D. : - 1989 - Principal Component Analysis with Instrumental Variables as a tool for modelling composition data. In *Multiway Data Analysis*, Coppi & Bolasco Ed., Elsevier, North Holland. Amsterdam. p. 341-352.

Schachtel A. : - 1978 - A non commutative algorithm for multiplying 5x5 matrices using 103 multiplications, *IPL* 7 180-182.

Schwarz H. R. : - 1989 - Numerical Analysis, a comprehensive introduction. John Wiley & Sons, Chichester, 517 p.

Sibony M. & Mardon J.-C. : - 1982 - Analyse Numérique. T. I : Systèmes linéaires et non linéaires. Hermann, Paris.

Stewart G. W. & Ji-guang Sun : - 1990 - Matrix perturbation theory. Academic Press, Boston. 365 p.

Stoker J. J. : - 1969 - Differential Geometry. John Wiley & Son, New York. 404 p.

Strassen V. : - 1969 - Gaussian elimination is not optimal. *Num. Math.* 13, 354-356.

----- : - 1972 - Evaluation of rational functions. in Complexity in Computer Computation, IBM Research Symposia. 1-10.

Struik D. J : - 1987 - A concise history of mathematics. Dover, New York. 228 pp.

Tucker L. R. : - 1963 - Implications of factor analysis of three-way matrices for measurement of changes. In *Problems in measuring changes*, C. W. Harris Ed. University of Wisconsin Press, Madison, Wisconsin.

----- : - 1964 - The extension of factor analysis to three dimensional matrices. In *Contributions to mathematical psychology*, Frederiksen N. & Gulliksen H. Ed., Holt, Rinehart & Winston, New York.

----- : - 1966 - Some mathematical notes on three mode factor analysis. *Psychometrika*, 31, 279-311.

----- : - 1972 - Relations between multidimensional scaling and three-mode factot analysis. *Psychometrika*, 37, 3-27.

----- & Messick S. : - 1963 - An individual difference model for multidimensional scaling. *Psychometrika*, 28, 333-367.

Van der Burg E., de Leeuw J. & Verdegaal R. : - 1988 - : Homogeneity analysis with k sets of variables : an alternative least square method with optimal scaling features. *Psychometrika*, 53(2), 177-197.

Weyl H : - 1946 - The classical groups. Princeton University Press. 2nd Edition. 320p.

Wilkinson J. H. : - 1988 - The Algebraic Eigenvalue Problem. Oxford University Press, Oxford, 662 p.

Winograd S. : - 1970 - On multiplication of 2x2 matrices. *Linear Algebra Appl.* 4, 381-388.

Wold H. : - 1966 - Estimation of Principal Components and related models by iterative least squares. in Krishnaiah P. R. Ed., *Multivariate Analysis*, Academic Press, New York. pp 391-420.

Yoshizawa T. : - 1987 - Singular value decomposition of multi-array data and its applications. Technical report CS-87-038, Department of Computer Science, Yamanashi University, Takeda, Kofu 400, Japan.