



HAL
open science

Using DEXi functions

Jacques-Eric Bergez

► **To cite this version:**

| Jacques-Eric Bergez. Using DEXi functions. INRAE UMR AGIR. 2014, pp.1-14. hal-03860476

HAL Id: hal-03860476

<https://hal.inrae.fr/hal-03860476v1>

Submitted on 18 Nov 2022

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.

Using DEXi functions

JE Bergez

July 09, 2014

Current version 1.9

General information

This document explains how to use the DEXi_fct R code in order to work with DEXi-tree models under R (DEXi_fct) The code is in constant evolution. For more information, contact Jacques-Eric Bergez at INRA: jbergez@toulouse.inra.fr

In this code, the Tree is view as a set of Nodes.

A **Tree** is a R object made of:

- nbAttributes="numeric", #number of attributes
- nbLeaves="numeric", #number of leaves
- Depth="numeric", #Maximum depth of the tree
- Attributes="character", #List of names of attributes - Leaves="character", #List of names of leaves
- Aggregated="character", #List of names of aggregated nodes
- isMultiple="logical", #a simple tag to know if multiple leaves
- Multiple="data.frame", #List of the multiple leaves and number of occurrence
- isLeafAggregated="logical", #a simple tag to know if leaf-aggregated nodes
- LeafAggregated="character", #List of names of leaf-aggregated nodes
- Paths="list", #Path from root to leaf
- Nodes="list", #List of nodes
- EvalOrder="numeric", #Evaluation order in case of LeafAggregated nodes
- rootName="character" #name of the root node

A **Node** is a R object made of

- id="numeric", #unique sequential id of the node
- name="character", #name of the node
- isLeaf="logical", #is it a leaf?
- isLeafAndAggregated="logical", #is this leaf also an aggregated node !
- children="character", #list of the names of the node's children
- sisters="character", #list of the names of the node's sisters
- mother="character", #name of the node's mother
- aggregation="matrix", #if aggregated node, table of aggregation
- Proba="numeric", #Estimated weight of aggregation. If Leaf set basically to uniform
- Depth="numeric", #Depth of the node
- Twin="numeric", #In case of multiple leaves, give the id of the other leaves
- CondiProbaList="list",
- rangeScale="numeric", #range scale
- scaleLabel="character", #Labels of the different scales
- nodePath="character" #Node path from root to leaf

To access an attribute Y of the object X under R use "@" : `X@Y` e.g. `aNode@name` give you the name of the node aNode

Basic functions

To create a DEXi tree under R

```
modelName<-"MASC2.0.dxi" #The name of the DEXi file
version <- "DEXi_fct_1.9.r" #The name of the current file of function
#Load the functions to read and manipulate the DEXi model
source(paste("functions",version,sep="/")) #Depends of the structure of
your directory

## Creating a generic function for 'print' from package 'base' in the
global environment

#read the structure of the model
MT<-xmlTreeParse(paste("data",modelName,sep="/"),useInternalNodes=T)
listDEXi<-createTree(MT)
DEXi <- listDEXi[[1]]
```

The function createTree returns a list in case of multiple trees (for example when integrating the satellite trees in the model). Then one choose the tree he wants to work with

To view the tree: three possibilities: print, show, describe

- **print(aTree)**: gives a short summary of the tree

```
print(DEXi)
```

```
## Root name: Contribution au developpement durable
## Number of attributes: 65
## Number of aggregated attributes: 26
## Number of true leaves (no multiple, no aggregated): 39
## Maximum depth: 6
## List of repeated aggregated nodes: Non
## No multiple leaves
## No Leaf-Aggregated Leaf
```

- **show(aTree)**: represents the hierarchy of the tree and give information on repeated leaves. In case of multiple leaves or leaf-aggregated nodes ID of similar nodes are provided

```
show(DEXi)
```

```
## < 1 > Z: Contribution au developpement durable
## < 2 > - Y : Dimension economique
## < 3 > - - Y : Resultats economiques
## < 4 > - - - X : Rentabilite
## < 5 > - - - Y : Autonomie economique
## < 6 > - - - - X : Independance economique
## < 7 > - - - - X : Efficience economique
## < 8 > - - - X : Surcout en materiel
## < 9 > - - Y : Capacite productive a long terme
## < 10 > - - - Y : Maitrise de la fertilite physico-chimique
## < 11 > - - - - X : Maitrise du statut acido-basique du sol
## < 12 > - - - - X : Maitrise de l etat structural du sol
## < 13 > - - - - X : Maitrise de la fertilite phosphopotassique
## < 14 > - - - Y : Maitrise des bioagresseurs
## < 15 > - - - - X : Maitrise des maladies et ravageurs
```

```

## < 16 > - - - - X : Maitrise des adventices
## < 17 > - - Y : Contribution au developpement economique
## < 18 > - - - Y : Qualite des produits
## < 19 > - - - - X : Qualite sanitaire
## < 20 > - - - - X : Qualite technonologique et esthetique des produits
## < 21 > - - - X : Contribution a l emergence de filieres
## < 22 > - Y : Dimension sociale
## < 23 > - - Y : Satisfaction des attentes de la societe
## < 24 > - - - X : Contribution a l emploi
## < 25 > - - - X : Fourniture de matieres premieres
## < 26 > - - Y : Satisfaction des attentes de l agriculteur
## < 27 > - - - Y : Facilite de mise en oeuvre
## < 28 > - - - - X : Complexite des itineraires techniques
## < 29 > - - - - X : Temps de veille technico-economique
## < 30 > - - - Y : Qualite des conditions de travail
## < 31 > - - - - X : Surcharge de travail
## < 32 > - - - - X : Risque pour la sante de l applicateur
## < 33 > - - - - X : Difficulte physique
## < 34 > - Y : Dimension environnementale
## < 35 > - - Y : Contribution a la qualite du milieu
## < 36 > - - - Y : Contribution a la qualite de l eau
## < 37 > - - - - Y : Maitrise des pertes de pesticides Eaux
## < 38 > - - - - - X : Maitrise pertes dans les eaux profondes
## < 39 > - - - - - X : Maitrise pertes dans les eaux superficielles
## < 40 > - - - - X : Maitrise des pertes de NO3
## < 41 > - - - - X : Maitrise des pertes de P
## < 42 > - - - Y : Contribution a la qualite air
## < 43 > - - - - X : Maitrise des emissions de NH3
## < 44 > - - - - X : Maitrise des emissions de N2O
## < 45 > - - - - X : Maitrise des emissions de pesticides Air
## < 46 > - - - Y : Preservation de la qualite du sol
## < 47 > - - - - X : Maitrise de l accumulation d elements toxiques
## < 48 > - - - - X : Maitrise du statut organique
## < 49 > - - - - X : Maitrise de l erosion
## < 50 > - - Y : Pression sur les ressources abiotiques
## < 51 > - - - Y : Pression Eau
## < 52 > - - - - X : Conso. en eau d irrigation en periode critique
## < 53 > - - - - X : Dependance vis a vis de la ressource en eau
## < 54 > - - - Y : Pression Energie
## < 55 > - - - - X : Consommation en energie
## < 56 > - - - - X : Efficience energetique
## < 57 > - - - X : Pression Phosphore
## < 58 > - - Y : Conservation de la biodiversite
## < 59 > - - - Y : Conservation de la macrofaune
## < 60 > - - - - X : Conservation des insectes volants
## < 61 > - - - - X : Conservation de la macrofaune du sol
## < 62 > - - - Y : Conservation de la flore
## < 63 > - - - - X : Abondance floristique
## < 64 > - - - - X : Diversite floristique
## < 65 > - - - X : Conservation des micro-organismes du sol

```

- **describe(aTree)**: gives full information on the tree and nodes

- To view a given node: **print(aNode)**

```
print(DEXi@Nodes[[1]])

## Node name: Contribution au developpement durable
## ID: 1
## Node depth: 1
## From root to node: Contribution au developpement durable ->
## Is it a leaf: FALSE
## Is is a leaf-aggregated: FALSE
## Mother: Root
## Sisters: None
## Children: Dimension economique Dimension sociale Dimension
environnementale
## Estimated weights: 33.33 33.33 33.33
```

To manipulate the tree

You can create subtree: createSubTree <- function(aTree,nodeName)
- aTree: the name of the main tree
- nodeName: the name of the node to cut the tree

```
subTree <- createSubTree(DEXi,"Dimension economique")
print(subTree)

## Root name: Dimension economique
## Number of attributes: 20
## Number of aggregated attributes: 8
## Number of true leaves (no multiple, no aggregated): 12
## Maximum depth: 4
## List of repeated aggregated nodes: Non
## No multiple leaves
## No Leaf-Aggregated Leaf
```

You can get the estimated weights of an aggregation table: getEstimatedWeights <- function(aNode)

```
getEstimatedWeights(DEXi@Nodes[[1]])

##          xDimension economique          xDimension sociale
##                   0.3358                   0.3321
## xDimension environnementale
##                   0.3321
```

You can change the aggregation table: createAggregationMatrix <- function(aNode,expectedWeight,nbTables=1,popSize=50,ifers=50)
- aNode: the node that will be modified
- expectedWeight: an array of weights
- nbTables: the number of matrix to be returned
- popSize and ifers: parameters for the algogen

```
myWeights <- c(0.2,0.2,0.6)
newAggregation <- createAggregationMatrix(DEXi@Nodes[[1]],myWeights,5)
newTree <- DEXi
newTree@Nodes[[1]]@aggregation <- newAggregation[[1]]
getEstimatedWeights(newTree@Nodes[[1]])
```

```
##      xDimension economique      xDimension sociale
##      0.2                        0.2
## xDimension environnementale
##      0.6
```

You can change the uniformity of the leaf modalities: `changeLeavesUniformity<-function(aTree,aDistributionList)`

```
DEXi@Nodes[[4]]@Proba
## [1] 0.25 0.25 0.25 0.25

DEXi@Nodes[[6]]@Proba
## [1] 0.25 0.25 0.25 0.25

myDistributionList <- list()
myDistributionList[[1]] <- c(0.5,0.2,0.2,0.1) ;
names(myDistributionList)[1] <- "Rentabilite"
myDistributionList[[2]] <- c(0.1,0.2,0.3,0.4) ;
names(myDistributionList)[2] <- "Indepondance economique"
newTree<-changeLeavesUniformity(DEXi,myDistributionList)
```

To simulate

You can create options: `createOptions <- function(aTree,nbOptions=1,aSeed=-1)`

- aTree: the name of the Tree
- nbOptions: the number of option to be created
- aSeed: in case one wants to create the same series of option unless random values are choses

```
option <- createOptions(DEXi,nbOptions=3)
head(option)

##      [,1] [,2] [,3]
## Rentabilite      2      3      4
## Indepondance economique      3      4      2
## Efficiencie economique      2      1      3
## Surcout en materiel      2      2      1
## Maitrise du statut acido-basique du sol      2      1      1
## Maitrise de l etat structural du sol      3      3      1
```

You can read options from DEXi output: `loadOptions <- function(aFileName)`

- aFileName: the name of the file created by DEXi using ...

The file structure is as

```
"" ,option
Rentabilite,1
Indepondance economique,3
Efficiencie economique,3
Surcout en materiel,2
Maitrise du statut acido-basique du sol,4
```

You can save options to be read by DEXi (File/Import/Import options/type csv):

`saveOptions <- function(options,file="options.csv")`

The file structure is then

```
,V1,V2,V3
Rentabilite,4,1,3
Independance economique,3,2,4
Efficience economique,4,3,1
Surcout en materiel,1,2,2
Maitrise du statut acido-basique du sol,2,1,3
```

You can run the simulation: EvaluateScenario<-function(aTree,anOption)

```
option <- createOptions(DEXi,nbOptions=10)
#Evaluate an option: function EvaluateScenario
scenarios <-
sapply(1:dim(option)[2],function(x)EvaluateScenario(DEXi,as.matrix(option[,
x])))
dim(scenarios)

## [1] 65 10

head(scenarios)

##           [,1] [,2] [,3] [,4] [,5] [,6] [,7]
## Contribution au developpement durable      2    5    3    4    1    1    2
## Dimension economique                      4    4    4    3    1    1    3
## Resultats economiques                     2    3    4    3    1    1    3
## Rentabilite                               3    3    4    4    1    1    2
## Autonomie economique                      1    3    3    2    2    2    3
## Independance economique                   2    2    2    3    1    1    4
##           [,8] [,9] [,10]
## Contribution au developpement durable      2    3    1
## Dimension economique                      2    5    1
## Resultats economiques                     1    3    1
## Rentabilite                               2    3    1
## Autonomie economique                      1    2    2
## Independance economique                   1    3    1
```

You can save the scenarios: saveScenarios <- function(scenarios,file="scenarios.csv")

The structure of the created file is:

```
"" "V1" "V2" "V3" "V4" "V5" "V6" "V7" "V8" "V9" "V10"
"Contribution au developpement durable" 1 1 2 1 1 2 3 4 2 2
"Dimension economique" 2 2 3 2 2 2 3 5 2 3
"Resultats economiques" 3 4 2 3 2 1 3 4 2 2
"Rentabilite" 3 4 2 3 3 1 3 4 1 1
"Autonomie economique" 2 3 3 3 1 2 3 3 4 4
"Independance economique" 1 4 2 4 2 1 2 3 4 4
```

You can view a scenario (barplot): showScenario<-function(aScenario,aTree,isLabelY=TRUE,isPar=T)

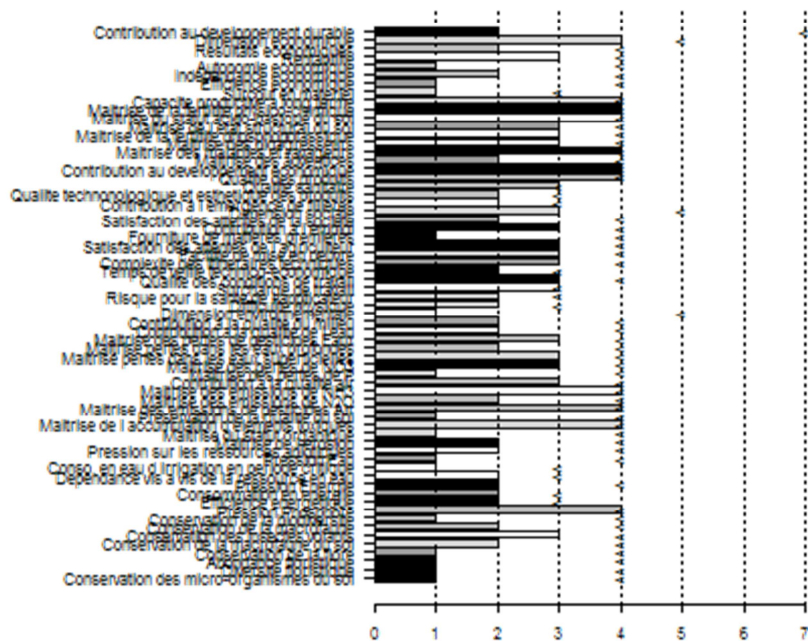
- aScenario: scenario to be shown

- aTree: the tree

- isLabelY: a boolean to draw the Y-axis

- isPAR: a boolean to format the graph

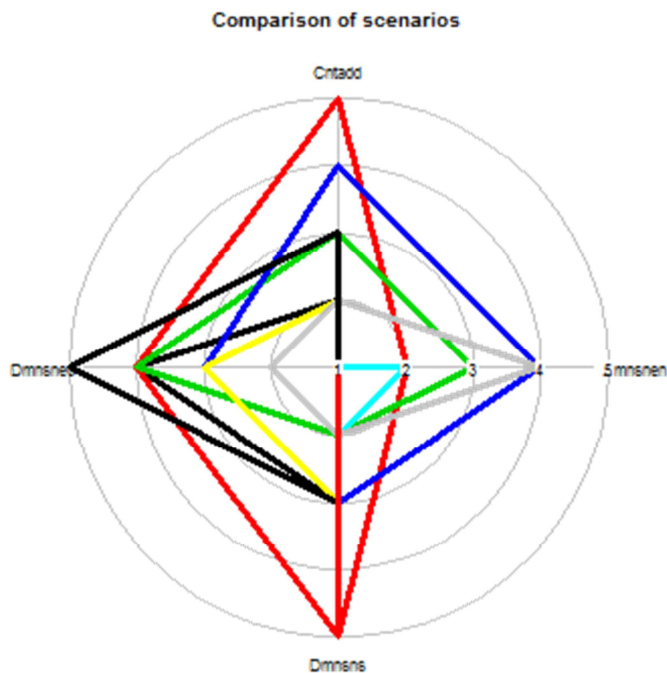
```
showScenario(as.matrix(scenarios[,1]),DEXi)
```



You can compare some scenario (radar plot): `compareScenario <- function(aTree,theScenarios,listNodes)`

- aTree: the tree
- theScenarios: the scenarios to be compared
- listNodes: the variables to be represented

```
compareScenario(DEXi,scenarios,c("Contribution au développement durable","Dimension économique","Dimension sociale","Dimension environnementale"))
```

To perform sensitivity analysis of the tree

You can run an ANOVA

- First check the requested time: `infoAOV <- function(aTree,iTest=50)`
- Then if "adequate" run the anova `AOV_DEXi<-function(aTree):` two models are tested: additif and power 2
- And then show results: `showAOV <- function(aAOV_DEXi, main.show=T, nb.plot=8, beside=T,las=1,...)`

```
infoAOV(DEXi)

##
## 39 factors
## Approximative required time to run the 9.574e+21 modalities 2.158e+19
minutes

petitArbre<-createSubTree(DEXi,"Resultats economiques")
infoAOV(petitArbre)

##
## 4 factors
## Approximative required time to run the 192 modalities 0.004544
minutes

AOV_out<-AOV_DEXi(petitArbre)
AOV_out

## [[1]]
##      df      ss ss.ratio main.ss.ratio      cm      F
## Rntb  3 131.69   0.618      0.618 43.896 311.00
## Inde  3  26.19   0.123      0.123  8.729  61.84
## Effe  3  26.19   0.123      0.123  8.729  61.84
```

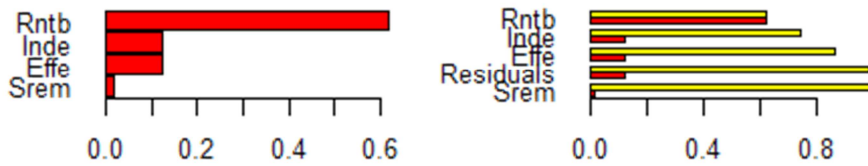
```

## Srem 2 3.51 0.016 0.016 1.755 12.44
##
## [[2]]
## df ss ss.ratio cm F
## Rntb 3 131.69 0.618 43.896 311.00
## Inde 3 26.19 0.123 8.729 61.84
## Effe 3 26.19 0.123 8.729 61.84
## Residuals 180 25.41 0.119 0.141 1.00
## Srem 2 3.51 0.016 1.755 12.44
##
## [[3]]
## df ss ss.ratio main.ss.ratio cm F
## Rntb 27 141.510 0.664 0.618 5.241 58.963
## Inde 27 32.885 0.154 0.123 1.218 13.702
## Effe 27 32.885 0.154 0.123 1.218 13.702
## Srem 20 7.104 0.033 0.016 0.355 3.996
##
## [[4]]
## df ss ss.ratio cm F
## Rntb 3 131.688 0.618 43.896 493.828
## Inde 3 26.187 0.123 8.729 98.203
## Effe 3 26.187 0.123 8.729 98.203
## Residuals 135 12.000 0.056 0.089 1.000
## Srem 2 3.510 0.016 1.755 19.746
## Rntb:Srem 6 3.281 0.015 0.547 6.152
## Rntb:Inde 9 3.271 0.015 0.363 4.089
## Rntb:Effe 9 3.271 0.015 0.363 4.089
## Inde:Effe 9 3.271 0.015 0.363 4.089
## Effe:Srem 6 0.156 0.001 0.026 0.293
## Inde:Srem 6 0.156 0.001 0.026 0.293

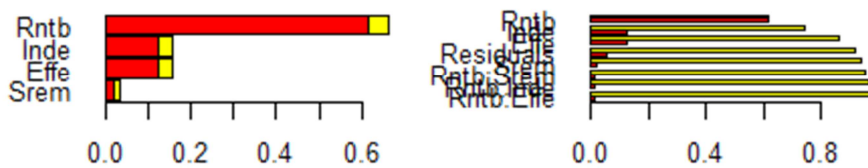
```

`showAOV(AOV_out)`

lain-effect and total Sums of Sq **Sums of Squares proportion:**



lain-effect and total Sums of Sq **Sums of Squares proportion:**



##		df	ss	ss.ratio	cm	F
##	Rntb	3	131.688	0.618	43.896	493.828
##	Inde	3	26.187	0.123	8.729	98.203
##	Effe	3	26.187	0.123	8.729	98.203
##	Residuals	135	12.000	0.056	0.089	1.000
##	Srem	2	3.510	0.016	1.755	19.746
##	Rntb:Srem	6	3.281	0.015	0.547	6.152
##	Rntb:Inde	9	3.271	0.015	0.363	4.089
##	Rntb:Effe	9	3.271	0.015	0.363	4.089
##	Inde:Effe	9	3.271	0.015	0.363	4.089
##	Effe:Srem	6	0.156	0.001	0.026	0.293
##	Inde:Srem	6	0.156	0.001	0.026	0.293

You can run a Monte-Carlo analysis

- First check the requested time: `infoMC<- function(aTree,nbRuns,iTest=50)`
- Then if adequate run the MC: `MonteCarlo<-function(aTree,nbRuns,isFile=F)`
- Then show results: `ShowMC<-function(aNode,aMC)`

```
nbRuns<-100
infoMC(DEXi,nbRuns)

##
## Approximative required time to run MC with 100 simulations 0.2202
minutes

MC<-MonteCarlo(DEXi,nbRuns)

##
## Time in: Mon Sep 29 14:46:45 2014
## Time out: Mon Sep 29 14:46:59 2014
```

```

par(mfrow=c(2,2))
ShowMC(DEXi@Nodes[[1]],MC)

##      1      2      3      4      5      6      7
## 0.40 0.27 0.18 0.08 0.04 0.03 0.00

ShowMC(DEXi@Nodes[[2]],MC)

##      1      2      3      4      5
## 0.17 0.35 0.22 0.17 0.09

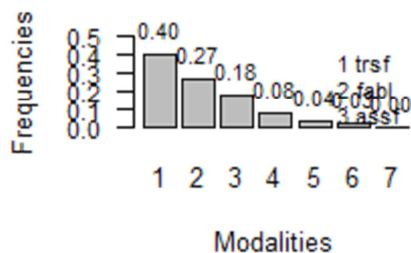
ShowMC(DEXi@Nodes[[19]],MC)

##      1      2      3
## 0.39 0.28 0.33

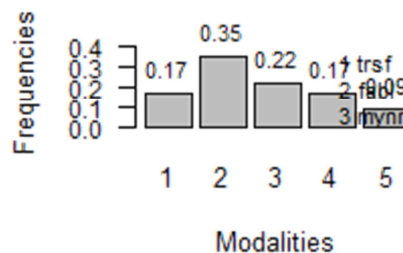
#Write a file that contain all the results obtained analysing the random
options selected for the MC analysis
write.table(MC,file=paste("output/","MC
results_.csv",sep=""),sep="," ,row.names=T,col.names=NA)

```

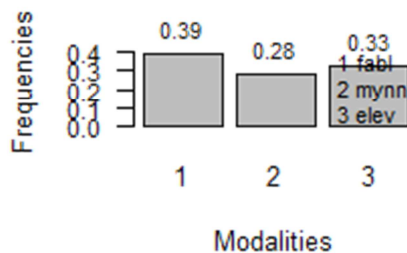
Contribution au developpement durab



Dimension economique [A]



Qualite sanitaire [L]



You can run a Sensitivity index computation

- Perform the analysis and save the file (to be used with Excel macro):`SI_DEXi <- function(aTree,fileName="SI_out.csv",isFile=T)`
- And then show the results for a given aggregated node `showSI<-function(aTree,aSI)`

```

SI <- SI_DEXi(DEXi,paste("output","SI_out.csv",sep="/"),TRUE)
SI[[1]]

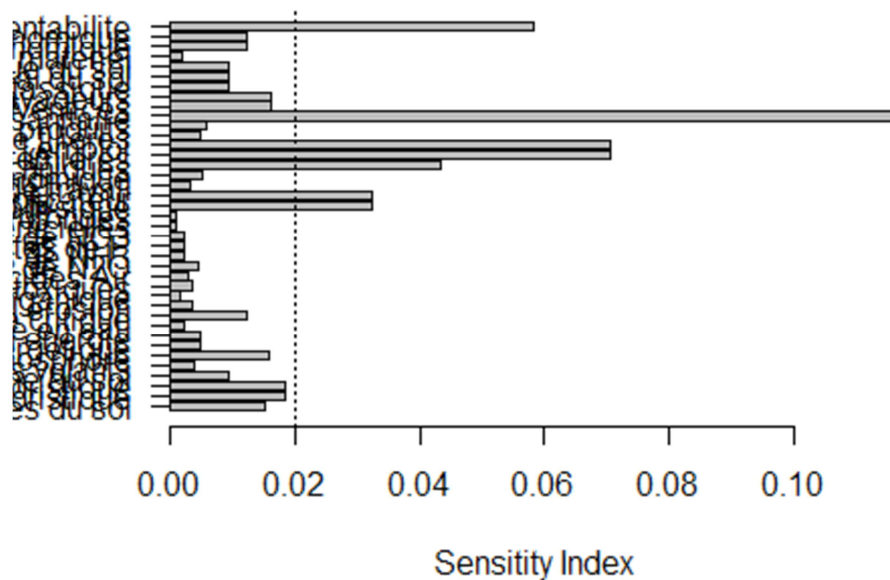
##
## Dimension economique                0.3431348    0    2
## Resultats economiques              0.0967172    0    3
## Rentabilite                        0.0582635    1    4

```

## Autonomie economique	0.0257173	0	4
## Independance economique	0.0120880	1	5
## Efficience economique	0.0120880	1	5
## Surcout en materiel	0.0018553	1	4
## Capacite productive a long terme	0.0717316	0	3
## Maitrise de la fertilite physico-chimique	0.0313068	0	4
## Maitrise du statut acido-basique du sol	0.0092107	1	5
## Maitrise de l etat structural du sol	0.0092107	1	5
## Maitrise de la fertilite phosphopotassique	0.0092107	1	5
## Maitrise des bioagresseurs	0.0349783	0	4
## Maitrise des maladies et ravageurs	0.0160998	1	5
## Maitrise des adventices	0.0160998	1	5
## Contribution au developpement economique	0.1398459	0	3
## Qualite des produits	0.1288367	0	4
## Qualite sanitaire	0.1169551	1	5
## Qualite technonologique et esthetique des produits	0.0057667	1	5
## Contribution a l emergence de filieres	0.0048905	1	4
## Dimension sociale	0.3230152	0	2
## Satisfaction des attentes de la societe	0.1538876	0	3
## Contribution a l emploi	0.0704336	1	4
## Fourniture de matieres premieres	0.0704336	1	4
## Satisfaction des attentes de l agriculteur	0.1587331	0	3
## Facilite de mise en oeuvre	0.0558515	0	4
## Complexite des itineraires techniques	0.0432873	1	5
## Temps de veille technico-economique	0.0050795	1	5
## Qualite des conditions de travail	0.0911074	0	4
## Surcharge de travail	0.0030127	1	5
## Risque pour la sante de l applicateur	0.0323120	1	5
## Difficulte physique	0.0323120	1	5
## Dimension environnementale	0.1994813	0	2
## Contribution a la qualite du milieu	0.0400211	0	3
## Contribution a la qualite de l eau	0.0075263	0	4
## Maitrise des pertes de pesticides Eaux	0.0017726	0	5
## Maitrise pertes dans les eaux profondes	0.0008203	1	6
## Maitrise pertes dans les eaux superficielles	0.0008203	1	6
## Maitrise des pertes de NO3	0.0022530	1	5
## Maitrise des pertes de P	0.0022530	1	5
## Contribution a la qualite air	0.0119380	0	4
## Maitrise des emissions de NH3	0.0023570	1	5
## Maitrise des emissions de N2O	0.0045663	1	5
## Maitrise des emissions de pesticides Air	0.0028191	1	5
## Preservation de la qualite du sol	0.0123782	0	4
## Maitrise de l accumulation d elements toxiques	0.0034428	1	5
## Maitrise du statut organique	0.0016544	1	5
## Maitrise de l erosion	0.0034428	1	5
## Pression sur les ressources abiotiques	0.0506309	0	3
## Pression Eau	0.0155722	0	4
## Conso. en eau d irrigation en periode critique	0.0123105	1	5
## Dependance vis a vis de la ressource en eau	0.0022086	1	5
## Pression Energie	0.0109334	0	4
## Consommation en energie	0.0047787	1	5
## Efficience energetique	0.0047787	1	5
## Pression Phosphore	0.0158774	1	4
## Conservation de la biodiversite	0.0829204	0	3

```
## Conservation de la macrofaune          0.0150686    0    4
## Conservation des insectes volants       0.0039362    1    5
## Conservation de la macrofaune du sol   0.0093427    1    5
## Conservation de la flore                0.0396258    0    4
## Abondance floristique                   0.0185345    1    5
## Diversite floristique                   0.0185345    1    5
## Conservation des micro-organismes du sol 0.0151833    1    4
```

```
showSI(DEXi,SI[[1]])
```



You can perform a One-At-a-Time analysis

- Perform the analysis: `OAT<-function(aTree,option)`
- And then show the results: `showOAT<-function(nodeName,aResults,aTree)`

```
optionOAT<-createOptions(DEXi,1)
results<-OAT(DEXi, optionOAT)
showOAT("Contribution au developpement durable",results,DEXi)
```

