



HAL
open science

UpDown -An R Package to Identify and Characterize Disturbances from Longitudinal Observations

Ingrid David, Vincent Le, Tom Rohmer

► **To cite this version:**

Ingrid David, Vincent Le, Tom Rohmer. UpDown -An R Package to Identify and Characterize Disturbances from Longitudinal Observations. 2023. hal-04246616

HAL Id: hal-04246616

<https://hal.inrae.fr/hal-04246616>

Preprint submitted on 17 Oct 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.

UpDown - An R Package to Identify and Characterize Disturbances from Longitudinal Observations

Ingrid David¹, Vincent Le^{1,2}, and Tom Rohmer^{*1}

¹GenPhySE, Université de Toulouse, INRAE, ENVT, F-31326 Castanet Tolosan

²Alliance R&D, 35650 Le Rheu, France

October 17, 2023

Abstract

The R package **UpDown** provides an algorithm to detect and characterize (start, end dates, intensity) disturbances that can occur at different hierarchical levels by studying the dynamics of longitudinal observations at individual and group scales. An Rshiny application is also provided in **UpDown** in order to visualize the longitudinal data at the different hierarchical scales associated with the information related to the detected disturbances.

1 Introduction

Longitudinal observations organized in hierarchical groups are studied in many contexts. For instance, in educational systems, students' progress can be observed for students distributed in classrooms belonging to different schools. In market analysis, the evolving price of a property depends on its locality as well as on its city or country. In pig farming systems, animals are spread out in pens that belong to different batches. In these three examples, the observations can then be considered at different levels: individual or group scales. For instance, we may be interested in obtaining information about the evolution of the price of a given house or about the median/average price of houses in a city. The feeding behavior of animals can be individually analyzed or studied at a pen or a herd scale.

Many R packages (R Core Team 2023) and methodologies make it possible to detect individual disturbances for dynamic observations (see, for example, **changepoint** Killick & Eckley (2014) for parametric and non-parametric tests, **npcp** Kojadinovic (2019) for non-parametric CUSUM tests, or **npcopTest** Rohmer (2018) for a non-parametric test for detecting change in the copula of multivariate observations with known changes in the marginal distributions). The most of the proposed methodologies in these packages make it possible to detect multiple disturbances.

Nevertheless, disturbances that alter observations can also occur at these different levels. They may only affect an observation (longitudinal) or a group of observations. For example, in the case of farm animals, an animal may be sick due to a disease that is not contagious. In

*Corresponding author: tom.rohmer@inrae.fr

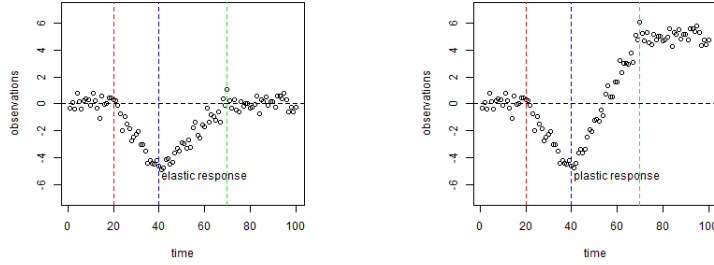


Figure 1: Elastic response (left) and plastic response (right) from a disturbance. The red dotted line is the start of the reaction, the blue dotted line is the end of the disturbance, and the green dotted line is the end of the reaction. The black dotted line is the expected evolution without disturbance.

that case, the disturbance (the disease) occurs at the individual level only. If the disease is contagious, all of the animals raised together in the same pen (same group) will be exposed to the disease, i.e., the disturbance occurs at the pen level, although not all of the animals will develop symptoms depending on their immune capacity. In this context, identifying whether a particular animal has been subjected to a disease is more effective using observations of all the animals in a pen (some animals will develop visible symptoms) than by observing only the particular animal that may not develop symptoms.

The UpDown algorithm was developed around this idea. It consists of studying observations at different scales in order to facilitate the detection and characterization of disturbances. When observations are collected repeatedly over time (longitudinal data), disturbances may be identified by observing changes in the dynamic of the observations over time. Other similarly methodologies have been proposed. For example Moreno-Muñoz et al. (2021) also use an approach based on a Gaussian mixture model for latent variable representation in a Bayesian context. The UpDown algorithm is able to detect and characterize disturbances that lead to an elastic or plastic response (Sauvant & Perez 2010, see Figure 1 for an illustration).

The algorithm captures deviations from the theoretical trajectory (the one that would have been observed in the absence of disturbances) by identifying abnormal evolutions compared to the expected one. The UpDown algorithm consists of two parts: (1) the Up-step, which identifies elements facing a disturbance from the bottom (unit) to the top level. The classification of the disturbed elements is done by fitting Gaussian mixture models; and (2) the Down-step, which validates elements detected in the Up-step, from the top level to the bottom level (unit) and that also allows us to identify elements that experience more than one disturbance (regardless of the level).

Remarks:

1. Note that before using **UpDown**, the longitudinal observations have to be corrected by their natural evolution. A median-based correction is proposed in the main function of the package. This type of correction makes it possible to be less impacted by the disturbance than a standard linear model.
2. The **UpDown** package is based on the assumption of a decrease in the evolution of the observations (corrected by their natural evolution) when a disturbance occurs, as illustrated in Figure 1. To consider disturbances that cause an increase in the evolution, use **UpDown** on the opposite of the observations. It goes without saying that the methodology cannot simultaneously detect many kinds of disturbances (that cause an

increase in some cases and a decrease in others).

2 Up-step

The Up-step consists in identifying disturbances at the different levels of the data organization based on the empirical distribution of the minimum slopes of the trajectories of the observations. To do that, firstly, for $i = 1, \dots, N$ where N is the total number of units, Nadaraya-Watson's smoothing curves f_i Nadaraya (1964), Watson (1964) are applied to the longitudinal observations $\mathbf{y}_i = (y_{i,1}, \dots, y_{i,n_i})$, with time values $t_{i,1}, \dots, t_{i,n_i}$, where n_i is the number of observations of the i th unit:

$$f_i(x|\mathbf{y}_i) = \frac{\sum_{j=1}^{n_i} y_{i,j} K\left(\frac{x-t_{i,j}}{h_n^{1/2}}\right)}{\sum_{j=1}^{n_i} K\left(\frac{x-t_{i,j}}{h_n^{1/2}}\right)}, \quad i = 1, \dots, N. \quad (1)$$

The chosen kernel K is the Gaussian density, and the bandwidth h_n is chosen depending on the largest length of observations per unit $n = \max_i(n_i)$. An excessively small value of h_n may not be sufficient to correct for the noise associated with the observations leading to wrong disturbance detection. An excessively high value of h_n will lead to an under-smoothing, and the algorithm might not detect the disturbances. The default choice in the function `UpDown` is $h_n = \sqrt{n}$. It can be modified by the user. The number of equally-spaced points over which the smoothing curves are estimated is $\lfloor nh_n \rfloor$, where for a real x , $\lfloor x \rfloor$ is the integer part of x . The first derivatives are obtained by differentiation on the basis of the smoothing curves,

$$\dot{f}_i(x|\mathbf{y}_i) = \frac{\lfloor nh_n \rfloor}{2(n_i - 1)} \left(f_i\left(x + \frac{n_i - 1}{\lfloor nh_n \rfloor} | \mathbf{y}_i\right) - f_i\left(x - \frac{n_i - 1}{\lfloor nh_n \rfloor} | \mathbf{y}_i\right) \right), \quad x \in [t_2, t_{n_i-1}]. \quad (2)$$

The minimum value of the first derivative (minimum of the slopes of the smoothing curve) is then extracted for each individual curve, $S_i = \min_x \dot{f}_i(x|\mathbf{y}_i)$. A mixture of two Gaussian distributions (`mixtools::normalmixEM`, Benaglia et al. 2009) is fitted to the sample consisting of the minimum values S_{i,n_i} using an Expectation-Maximization algorithm. Ids are then classified into "disturbed" and "non-disturbed", based on the posterior distributions. The method (smoothing, clustering) is then reapplied to each level above (one by one in a bottom-up manner) to identify disturbances acting at the group level. For the level $k \geq 2$ consisting of N_k groups, the median trajectories of the groups are recursively calculated as follows:

$$y_{\ell,j}^{(k)} = \text{median} \left(y_{i,j}^{(k-1)}; i \text{ such that } X_{ik} = \ell \right), \quad \ell = 1, \dots, N_k, \quad \text{with } y_{\ell,j}^{(1)} = y_{\ell,j}, \quad j \geq 1 \quad (3)$$

where X_{ik} corresponds to the group in level k to which group i in level $k - 1$ belongs. Median trajectories are illustrated in Figure 3. Note that for any group ℓ , if at least 50% of the trajectories (or median trajectories) constituting the group do not have observations at time step j , the median $y_{\ell,j}^{(k)}$ is not evaluated. The previous smoothing method (1) is then used again on the $\mathbf{y}_\ell^{(k)} = \left(y_{\ell,j}^{(k)} \right)_{j \geq 1}$. Finally, the minimum values, expressed as:

$$S_\ell^{(k)} = \min_x \dot{f}_\ell(x|\mathbf{y}_\ell^{(k)}), \quad \ell = 1, \dots, N_k, \quad k \geq 2, \quad (4)$$

are extracted and classifications are used again on the $\left(S_\ell^{(k)} \right)_\ell$.

Hence, for a given level, the mixture models make it possible to classify the elements with at least one disturbance for this level or for a higher level. Starting from the first level (unit,

i.e. $k = 1$) to the highest level minus one, if an element of the k th level is classified among the disturbed one and the group to which it belongs as well (at the $k + 1$ th level), this element will be considered to have experienced (at least) one group disturbance at level $k + 1$; hence, the disturbance at level k will not be considered.

3 Down-step

It should be emphasized that for any group ℓ and any time step j , if more than 50% of the trajectories constituting the underlying level undergo a disturbance (or if initial trajectory has not been recovered) a time j , the median trajectory in (3) may be impacted.

Hence in the Up-step, the group level of disturbances (at the hierarchical level $k \geq 2$) can be misclassified due to a high number of disturbed elements at the preceding level $k - 1$ which compose the group level k . It can also be due to a high variability in the dynamics of the observations. Moreover, the Up-step detects elements that experience at least one disturbance, but does not provide any information about the number and characterization of the disturbances experienced. To reduce the risk of misclassification at the group scale (i.e., $k \geq 2$), the Down-step thus proposes a post-validation of the detected groups in the Up-step by comparing the reaction start times to the disturbance.

3.0.1 Multiple disturbance detection

The Down-step makes it possible to detect multiple disturbances. Let τ_k the intersection of the two Gaussian curves of the mixture model at the hierarchical level k in the Up-step, i.e., the theoretical value of (4) corresponding to a posterior probability to be classified as disturbed of 0.5 by the mixture model at level k . Any local minima of the first derivative of the smoothing curves whose values are smaller than τ_k are considered as other potential disturbances at level k . The thresholds τ_k are illustrated by the dotted blue lines in Figure 4 for the case of three hierarchical levels.

A first evaluation of the start times for the disturbances corresponds to the times for which the local minima are reached.

3.0.2 Validation

If a disturbance occurs for a given group scale (i.e., level $k \geq 2$) and for a given time, this disturbance should affect all or part of the elements of level $k - 1$ at 'close times'.

Starting from the highest hierarchical level to the lowest group level (that is $k = 2$), a non-supervised clusterization of the identified starting points is done using a multi-modal Gaussian mixture model (`mclust::Mclust`, Scrucca et al. 2016). Each of the time clusters consisting of a sufficient percentage of the time observations (depending on the number of elements of the group) validates the corresponding group's disturbance. We suggest a validation threshold of 0.5, i.e., at least 50% of the elements constituting the group should have a disturbance in a close time to validate the group disturbance. This value can be changed by the user.

When a group disturbance is not validated at level k , the elements of the group classified among the disturbed ones in the mixture model at level $k - 1$ are again subject to validation at level $k - 1$.

3.0.3 Characterization

Regardless of the hierarchical level $k \geq 1$, in the case of a single disturbance, the starting points correspond to the times x for which the local minima (lower than a threshold value τ_k) of the derivative smoothing curve $\dot{f}_\ell(x|\mathbf{y}_\ell^{(k)})$ in (2) are reached.

The time corresponding to the first local minimum of the smoothing curve f after the starting point is considered as the end of the disturbance. If no local minimum of the smoothing curve is observed after a starting point, the date of the last observation is considered as the end of the disturbance. Finally, the intensity of the disturbance is evaluated by the slope between the start and end time points.

When multiple disturbances occur, it could create some confusion in the estimated starting/ending points between the two disturbances, especially for the close times or in the event that one of the two disturbances is not validated. To avoid such a source of confusion, in the case of multiple disturbances, the starting time and ending time are recalculated using the median of the starting and ending time based on each of the clusters of the multi-modal Gaussian mixture at level $k - 1$ of the validation step.

Because the method can validate the same disturbance at many levels, an indicator of concordance (Cohen's kappa; see Landis & Koch 1977, for details), is used to eliminate redundancy disturbance between two distinct levels, based on the rounded estimated starting and end points. Cohen's kappa is between 0 and 1. It evaluates the overlapping between two considered disturbances. An excessively high value of kappa can lead to not removing redundancy disturbances. At contrary an excessively small value of kappa can lead to not validating the multiple scale disturbances. Note that if kappa is not considered and the previous validation threshold is 0, the Down-step will do nothing.

4 Overview of UpDown

The dependencies for **UpDown** are: **mixtools** Benaglia et al. (2009) and **mclust** Scrucca et al. (2016) for classification and clusterization; **reshape2** Wickham (2007) and **dplyr** Wickham et al. (2021) for data manipulation; and **shiny** Chang et al. (2021) and **ggplot2** Wickham (2016) for data visualization. All these packages are available on the CRAN at <http://CRAN.R-project.org/>.

You can install the **UpDown** package from CRAN with the following R code

```
install.packages("UpDown")
library(UpDown)
```

or the from latest development version:

```
#install.packages("remotes")
remotes::install_github("TomRohmer/UpDown")
library(UpDown)
```

4.1 UpDown function

The **UpDown** function is available in the **UpDown** library :

```
UpDown(data, levels, vtime, obs, h.int=10, mixplot=FALSE, correction=NULL,
kappa=NULL, thr_va=0.5, options=list())
```

The input `data` should be a dataframe arranged like the `PigFarming` dataset, with one row per unit and observed time. Columns consist at least of the following: hierarchical levels (one column per level), time (integer) for which observations are measured, and observations. The `levels` input is the vector of the considered labels of the hierarchical levels ordered from the highest level to the unit level. The `vtime` input is the considered name of the time variable. Finally, the `obs` input is the name of the considered observations appearing in the dataframe.

The parameter `h.int` is the smoothing parameter used in the kernel smoothing method (1). The binary `mixplot` input makes it possible to specify whether or not the mixture model plots are desired. Such plots make it possible to detect any problem in the mixture fitting, for example due to abnormal observations.

The argument `correction` allows to correct the longitudinal observations by their natural evolution over time beforehand, subtracting the median observation relative to the modalities of the variable.

`Kappa` and `thr.va` are the values of the kappa and the threshold validation parameter used in the Down-step, respectively. When `Kappa` is not specified, no disturbances are removed. Moreover when `thr.va` takes the value of 0, all disturbances are validated.

Finally, `options` is a list of options including the classical options of the `mixtools::normalmixEM` R function. Because considering trajectories with too few observations might not make sense, units with less than 20 observations are automatically removed. This can be changed used the optional parameter `minobs` in the `options` parameter.

4.2 A shiny App function

The `UpDownApp` function is available in the `UpDown` library .

It makes possible to visualize the longitudinal intra-group observations and the detected disturbances, as well as the median-group observations and the corresponding smoothing curve. It allows to check if some detected elements correspond to registered disturbances (listed disease for example) and if the smoothing is well adapted to the data.

```
UpDownApp(UpDown.out,obs=NULL,width=1000,height=1000)
```

The `UpDown.out` input have to be the global output of the `Updown` function. `obs` is then an optional output. The default value is the observation's name specified in `UpDown`. In the case of multiple dynamic observations, it may be desirable to visualize each of the observations and to see if the detected disturbances affect the other traits or not. Hence, the `obs` input makes it possible to specify a vector of many observation names that appear in the initial dataset. The optional `width` and `height` options makes it possible to modify the dimension of the plots that appear in the Rshiny application.

5 Application on a simulated dataset

The `PigFarming` dataset is available in `UpDown`. It consists of simulated hierarchical data that mimics a pig farming system dataset based on Le et al. (2022). The weight (in kg) of 6,000 animals was simulated over a period of 93 to 100 days and was considered at three different hierarchical levels. Animals (first level) were raised in 40 batches (highest hierarchical level) and in 15 pens within each batch (second hierarchical level), leading to 15 animals per pen. The identifier of the animal ('id'), the identifier of the batch ('batch') and the identifier of the

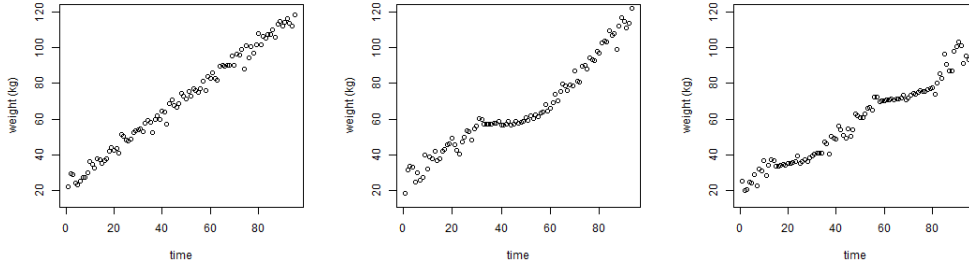


Figure 2: Plot of the weights of id 1 with no disturbance (left), id 6 with an individual disturbance (middle), and id 5405 with both pen disturbance and batch disturbance (right).

pen ('pen') are listed in `PigFarming` as well as the age of the animal, the time observation and the value of the simulated weight (e.g., Table 1).

In this simulated dataset, at the third level, 25% of the batches were subject to a disturbance. At the second level, 21% of the pens were exposed to a disturbance. Finally, 19% of the individuals also experienced an individual disturbance. he starts, the duration and the intensities of the disturbances were uniformly sampled respectively on $\{2, \dots, 99\}$, $\{1, \dots, 25\}$ and $(0.5, 2.5)$. The two components of the individual robustness, the resistance and resilience (Le et al. 2022) was sampled from a logit-normal distribution with variance 1.

```
data<-get(data(PigFarming))
head(data)
```

id	batch	pen	age	time	weight
1	b101	p10101	6	1	22.14
1	b101	p10101	7	2	29.76
1	b101	p10101	8	3	28.84
1	b101	p10101	9	4	24.18
1	b101	p10101	10	5	23.28
1	b101	p10101	11	6	25.20

Table 1: First part of the `PigFarming` dataset

In Figure 2, the longitudinal observations of the animals' identifiers (id) 1, 6 and 5405 were plotted. For id 1, no disturbance occurs. For id 6, an individual disturbance occurs at time 33 and with a duration of 25 days. For id 5405, a level 2 disturbance occurs at time 60 with a duration of 20 days, and a level 3 disturbance occurs at time 14 with a duration of 21 days. This non-observable information, on the intensity, starting and ending point of the disturbance for each level can be read in the complementary dataset `PigDisturbance`, e.g.:

```
TrueDist<-get(data(PigDisturbance))
subset(TrueDist,id %in% c(1,6,5405))
```

In Figure 3, the median trajectory of the groups designated as p10101 and p100101 of level 2 were plotted as well as the median trajectory of the group designated as b1001 of level 3.

Note that group p10101 contains id 6 where an individual disturbance occurs. The median trajectory is not impacted. Group p100101 contains id 5405 on which two disturbances

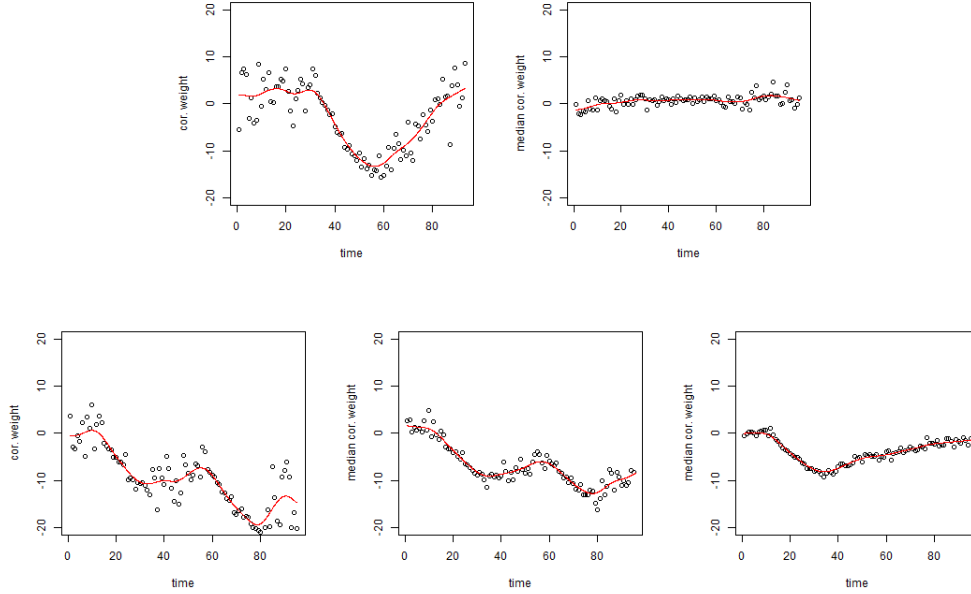


Figure 3: Plot of the age-corrected weight of id 6 with an individual disturbance (top left) and the median phenotype of pen p10101 that contains id 6 (top right). Plot of the weight of id 5405 with both pen disturbance and batch disturbance (bottom left), the median phenotype of pen p100101 that contains id 5405 (bottom middle) and the median weight of batch b1001 that contains pen p1001 (bottom right). The red curves are Nadaraya-Watson's smoothing curves.

affecting level 2 and level 3 simultaneously occur. The median trajectory is impacted by these two disturbance levels. Group b1001 contains group p100101. The median weight is only impacted by the disturbance affecting level 3.

We now compute the UpDown method on the PigFarming dataset.

```
levels=c("batch","pen","id")
UD_pig<- UpDown(data,levels=levels, vtime="time", obs="weight", kappa=0.75,
+ thr_va=0.5, h.int=10,mixplot=TRUE, correction="age")
```

```
level batch: number of elements=40; number of iterations= 8
level pen: number of elements=400; number of iterations= 26
level id: number of elements=6000; number of iterations= 60
```

```
subset(UD_pig$Up, id %in% c(1,6,5405))
```

batch	pen	id	det
b101	p10101	1	0
b101	p10101	6	id
b1001	p100101	5405	batch

Table 2: Results of the Up-step for animals with id 1, id 6 and id 5405.

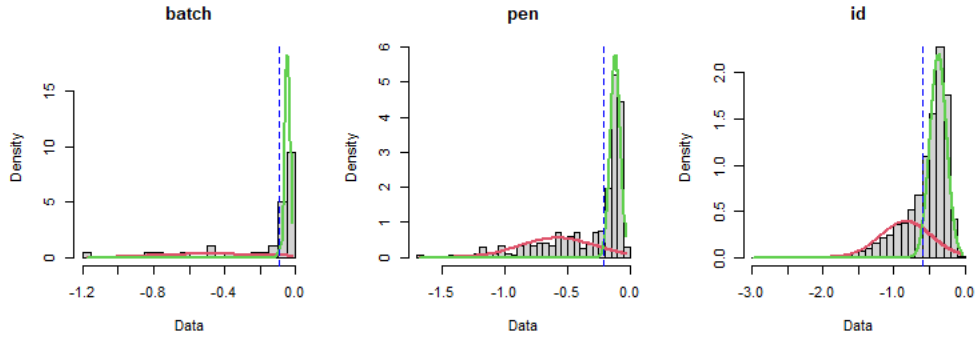


Figure 4: Plot of the mixture models for individuals, pens and batches levels

UD_pig\$Down\$batch

batch	start	end	intensity
b1001	16.49	34.24	0.32
b1002	74.82	82.63	0.17
b104	67.59	84.92	0.84
b401	69.54	76.29	0.47
b504	52.55	68.12	0.43
b603	95.86	96.00	0.17
b703	77.28	95.67	0.18
b704	87.54	96.00	0.35
b803	22.76	36.04	0.53
b804	72.91	82.07	0.08

Table 3: Results of the Down-step for the batch level.

The estimated starting and end points are shown in Figure 5. In group p100101 of level 2, the intensity of the disturbance was 0.46. In group b1001 of level 3, the intensity of the disturbance was 0.32.

Using the true values on the disturbances available on the `PigDisturbance` dataset, we can evaluate the performances of the algorithm in terms of sensitivity (percentage of well detected disturbances) and specificity (1- percentage of wrong detected disturbances). These performances are summarized in Table 4 and Figure 6. Of the 40 batches, 10 underwent disturbances and all were detected. Moreover, no batch disturbance was wrongly detected. The starting points were estimated with a median gap of 3.2 days and the end points were estimated with a median gap of 1.87 days with the true values. The correlation between the measurements made for the intensity and the true value of intensity was 0.93. Of the 400 pens, 84 underwent pen disturbances and 74% were detected. Moreover, only 2% were wrongly detected. The starting points of the well-detected disturbances were estimated with a median gap with the true values of 2.86 days, and the end points were estimated with a median gap with the true values of 1.37 days. The correlation between the measurement made for the intensity of the disturbance and the true value of intensity was 0.74. Of the 6,000 individuals, 1127 underwent individual disturbances and only 33% were detected. Nevertheless, only 2% were wrongly detected. The starting points of the well-detected disturbances were estimated

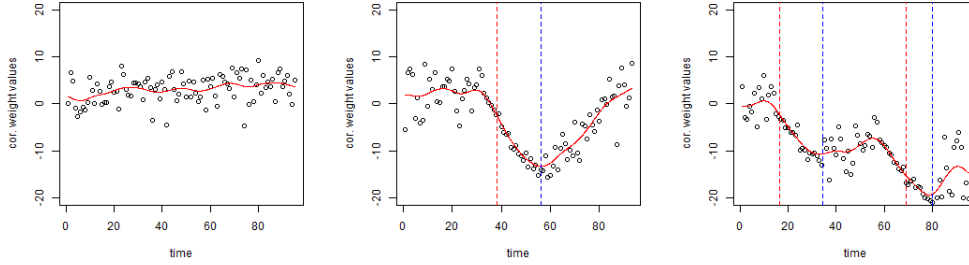


Figure 5: Plot of the age-corrected weights of id 1 with no disturbance (left) and id 6 with an individual disturbance (middle), and id 5405 with both pen disturbance and batch disturbance (right). The red curves are Nadaraya-Watson’s smoothing curves, the dotted red lines are the estimated starting points and the dotted blue lines are the estimated end points.

with a median gap of 3.03 days and the end points were estimated with a median gap of 2.00 days. The correlation between the measurement made for the intensity and the true value of intensity was only 0.55.

In fact, in the simulation model Le et al. (2022), the ‘robustness’ of the animal is a major part of the reaction of the animal exposed to disturbance, and the intensity of the disturbance is not sufficient to explain the reaction. Then, if the units (animals) are randomly spread out in the hierarchical level regardless of their ‘robustness’, these individual sources of variations do not affect the median trajectories.

	batch	pen	id
sensitivity (%)	1.00	0.74	0.33
specificity (%)	1.00	0.98	0.98
starting point	3.20	2.86	3.03
end point	1.87	1.37	2.00
intensity (corr)	0.93	0.74	0.55

Table 4: Table of the sensitivity and specificity of the detected elements, median difference between the true and estimated starting and end points and correlation between the true intensity and the proposed measurement of intensity

More exhaustive Monte Carlo simulations (based on 1000 simulations of such a dataset) can be found in Le (2022).

Finally, the dynamic trajectories can be visualized through the Rshiny application as well as the detected elements and the smoothing curve on the units and on the median trajectories in a second panel:

```
if(interactive()){
  UpDownApp(UD_pig)
}
```

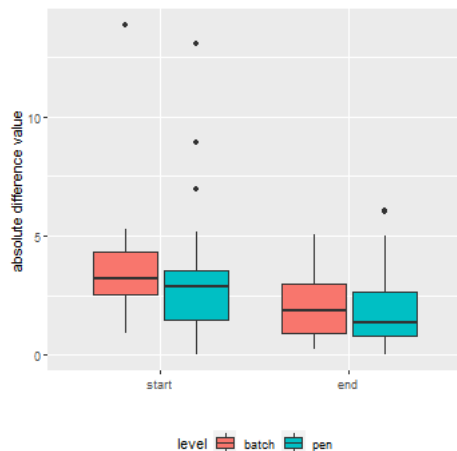


Figure 6: Boxplot of the difference between the true and estimated starting and end points

5.1 Calculation times

Using the `PigFarming` dataset, we evaluated the average calculation time based on $B = 10$ runs for a number N of ids of $N \in \{2400, 3000, 3600, 4200, 4800, 5400, 6000\}$ and two and three hierarchical levels. The corresponding number of elements in level 2 was $N_2 \in \{160, 200, 240, 280, 320, 360, 400\}$ and the corresponding number of elements in level 3 was $N_3 \in \{16, 20, 24, 28, 32, 36, 40\}$. The results are summarized in Table 5. For such a dataset, the calculation times are reasonably low, with 34 seconds on average for $N = 6000$ id with 93 to 100 observations per id and two considered levels, and depend very little on the number of considered levels.

6 Conclusion and future developments

The R package `UpDown` provides an algorithm to detect and characterize group disturbances from hierarchical systems and longitudinal data. It is based on unsupervised classification using a mixture model approach and working at different hierarchical levels, i.e., using information from the group. A validation procedure at the group levels, based on the estimated starting point of the detected elements is then proposed to guarantee a low ratio of wrong detection as well as a characterization of the detected disturbance by an estimation of the starting and end points of the reaction and the intensity of the disturbance.

In the simulations carried out, the `UpDown` method presents very good performances in terms of sensitivity (group scale) and specificity for reasonable computation times. Nevertheless the intensities were uniformly sampled in the simulation model Le et al. (2022). In practice, heavy-tailed distribution for the intensities could lead to not detecting the disturbances corresponding to the smallest intensities, i.e., the Gaussian mixture model may not be appropriate. In the same way, the quality of detection can be strongly affected by outliers. These outliers could be observed in the mixture results and removed from the dataset.

In future developments, we will consider the case of multivariate longitudinal observations for which, disturbances can affect one of the margins, all (or a part) of the margins, or the dependence structure between the margins. To do that, a multivariate mixture model would have to be considered in the Up-step.

```

library(microbenchmark)
for(g in seq(2400,6000,600){

  tt=microbenchmark(
  {
  UpDown(subset(data,id%in% 1:g),levels=c("pen","id"), vtime="time", obs="weight",
    kappa=0.75, thr_va=0.5, h.int=10,mixplot=FALSE, correction="age")
  },
  {
  UpDown(subset(data,id%in% 1:g),levels=c("batch","pen","id"),
    vtime="time", obs="weight",
    kappa=0.75, thr_va=0.5, h.int=10,mixplot=FALSE, correction="age")
  },
  times=10)
print(tt)
}

```

N	2400	3000	3600	4200	4800	5400	6000
2 levels	14.34	17.32	19.96	23.15	27.01	30.03	33.74
3 levels	13.61	16.49	19.43	21.11	25.09	26.97	30.31

Table 5: Average computation time (s) for UpDown method with 2 and 3 hierarchical levels.

7 Acknowledgement

This study was funded 50% by the Animal Genetics division of INRAE and 50% by the Alliance R&D (Axiom, Choice Genetics, Nucleus and IFIP). We particularly thank Alliance R&D for reviewing the manuscript and testing the **UpDown** package on real data from pig farms.

References

- Benaglia, T., Chauveau, D., Hunter, D. R. & Young, D. (2009), ‘mixtools: An R package for analyzing finite mixture models’, *Journal of Statistical Software* **32**(6), 1–29.
URL: <http://www.jstatsoft.org/v32/i06/>
- Chang, W., Cheng, J., Allaire, J., Sievert, C., Schloerke, B., Xie, Y., Allen, J., McPherson, J., Dipert, A. & Borges, B. (2021), *shiny: Web Application Framework for R*. R package version 1.6.0.
URL: <https://CRAN.R-project.org/package=shiny>
- Killick, R. & Eckley, I. A. (2014), ‘changept: An R package for changepoint analysis’, *Journal of Statistical Software* **58**(3), 1–19.
URL: <https://www.jstatsoft.org/index.php/jss/article/view/v058i03>
- Kojadinovic, I. (2019), *npcp: Some Nonparametric CUSUM Tests for Change-Point Detection in Possibly Multivariate Observations*. R package version 0.1-11.
URL: <https://CRAN.R-project.org/package=npcp>

- Landis, J. R. & Koch, G. G. (1977), ‘The measurement of observer agreement for categorical data’, *biometrics* pp. 159–174.
- Le, V. (2022), Nouvelle mesure de la robustesse des animaux d’élevage par utilisation des données de phénotypage haut-débit, Theses, INPT Toulouse.
URL: <https://hal.inrae.fr/tel-03967884>
- Le, V., Rohmer, T. & David, I. (2022), ‘Impact of environmental disturbances on estimated genetic parameters and breeding values for growth traits in pigs’, *Animal* **16**(4), 100496.
- Moreno-Muñoz, P., Ramírez, D. & Artés-Rodríguez, A. (2021), ‘Change-point detection in hierarchical circadian models’, *Pattern Recognition* **113**, 107820.
- Nadaraya, E. A. (1964), ‘On estimating regression’, *Theory of Probability & Its Applications* **9**(1), 141–142.
- R Core Team (2023), *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria.
URL: <https://www.R-project.org/>
- Rohmer, T. (2018), *npcopTest: Non Parametric Test for Detecting Changes in the Copula*. R package version 1.03.
URL: <https://CRAN.R-project.org/package=npcopTest>
- Sauvant, D. & Perez, J.-M. (2010), ‘Robustesse, rusticité, flexibilité, plasticité... les nouveaux critères de qualité des animaux et des systèmes d’élevage. avant propos’, *INRA Productions Animales* **23**(1), 3.
- Scrucca, L., Fop, M., Murphy, T. B. & Raftery, A. E. (2016), ‘mclust 5: clustering, classification and density estimation using Gaussian finite mixture models’, *The R Journal* **8**(1), 289–317.
URL: <https://doi.org/10.32614/RJ-2016-021>
- Watson, G. S. (1964), ‘Smooth regression analysis’, *Sankhyā: The Indian Journal of Statistics, Series A* pp. 359–372.
- Wickham, H. (2007), ‘Reshaping data with the reshape package’, *Journal of Statistical Software* **21**(12), 1–20.
URL: <http://www.jstatsoft.org/v21/i12/>
- Wickham, H. (2016), *ggplot2: Elegant Graphics for Data Analysis*, Springer-Verlag New York.
URL: <https://ggplot2.tidyverse.org>
- Wickham, H., François, R., Henry, L. & Müller, K. (2021), *dplyr: A Grammar of Data Manipulation*. R package version 1.0.7.
URL: <https://CRAN.R-project.org/package=dplyr>