Mélange de modèles et la librairie FLEXMIX

Amine OUNAJIM

PRISMATICS Lab

Predictive **R**esearch **I**n **S**pine/Neuromodulation **M**anagement **A**nd **T**horacic **I** nnovation/**C**ardiac **S**urgery

CHU de Poitiers

AXE SCALE-EPI

METHODS IN CLINICAL RESEARCH AND EPIDEMIOLOGY

C’EST QUOI UN MÉLANGE (FINI) ?

Supposons que nous voulions générer deux composantes / clusters également probables, nous décomposons le processus de génération en plusieurs étapes :

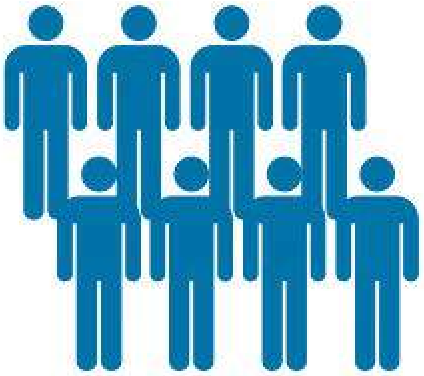
1. Tirer à pile ou face (avec une pièce équilibrée)

2. S'il s'agit de pile : générer un nombre aléatoire à partir d'une loi normale de moyenne 1 et de variance 0,25.

3. S'il s'agit de face : générer un nombre aléatoire à partir d'une loi normale de moyenne 3 et de variance 0,25.

4. Répéter 1 à 3.

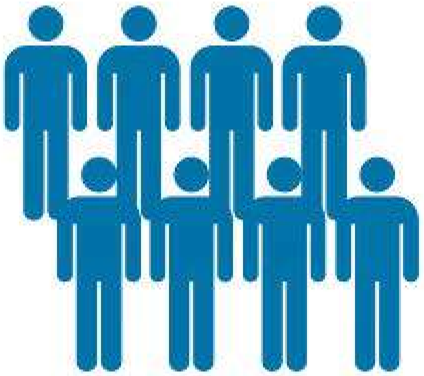
C’EST QUOI UN MÉLANGE ?



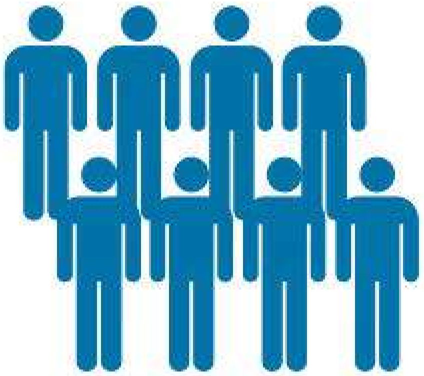
$X \sim \mathcal{N}(1, 0.25)$

$X \sim \mathcal{N}(3, 0.25)$

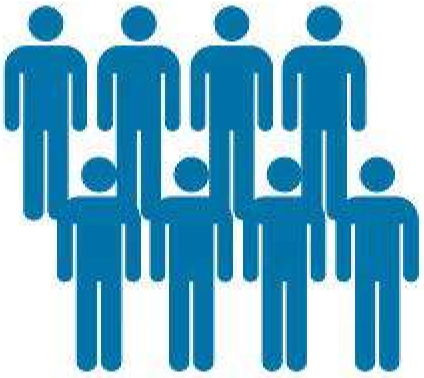
C’EST QUOI UN MÉLANGE ?



C’EST QUOI UN MÉLANGE ?



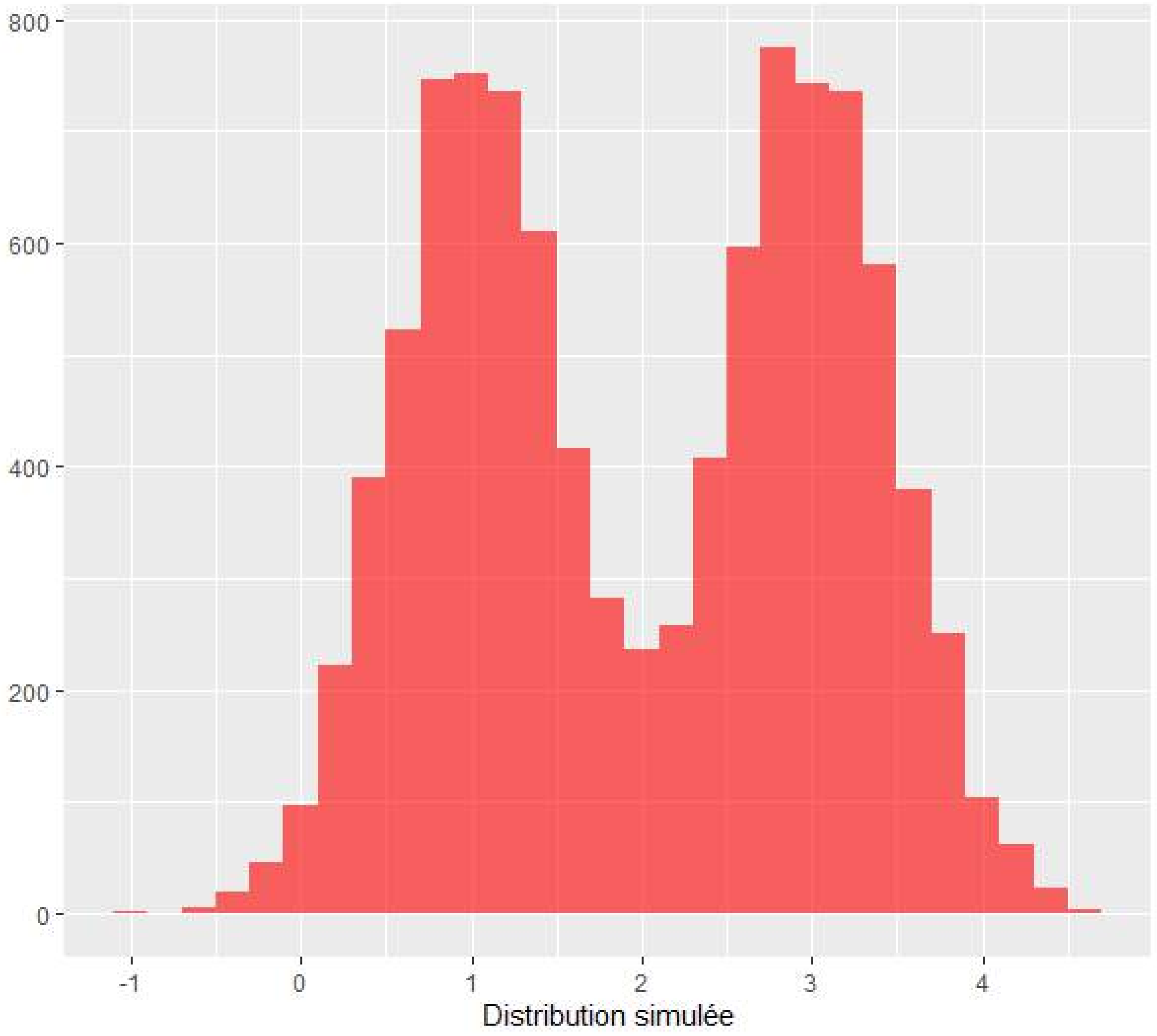
C’EST QUOI UN MÉLANGE ?



Hétérogène

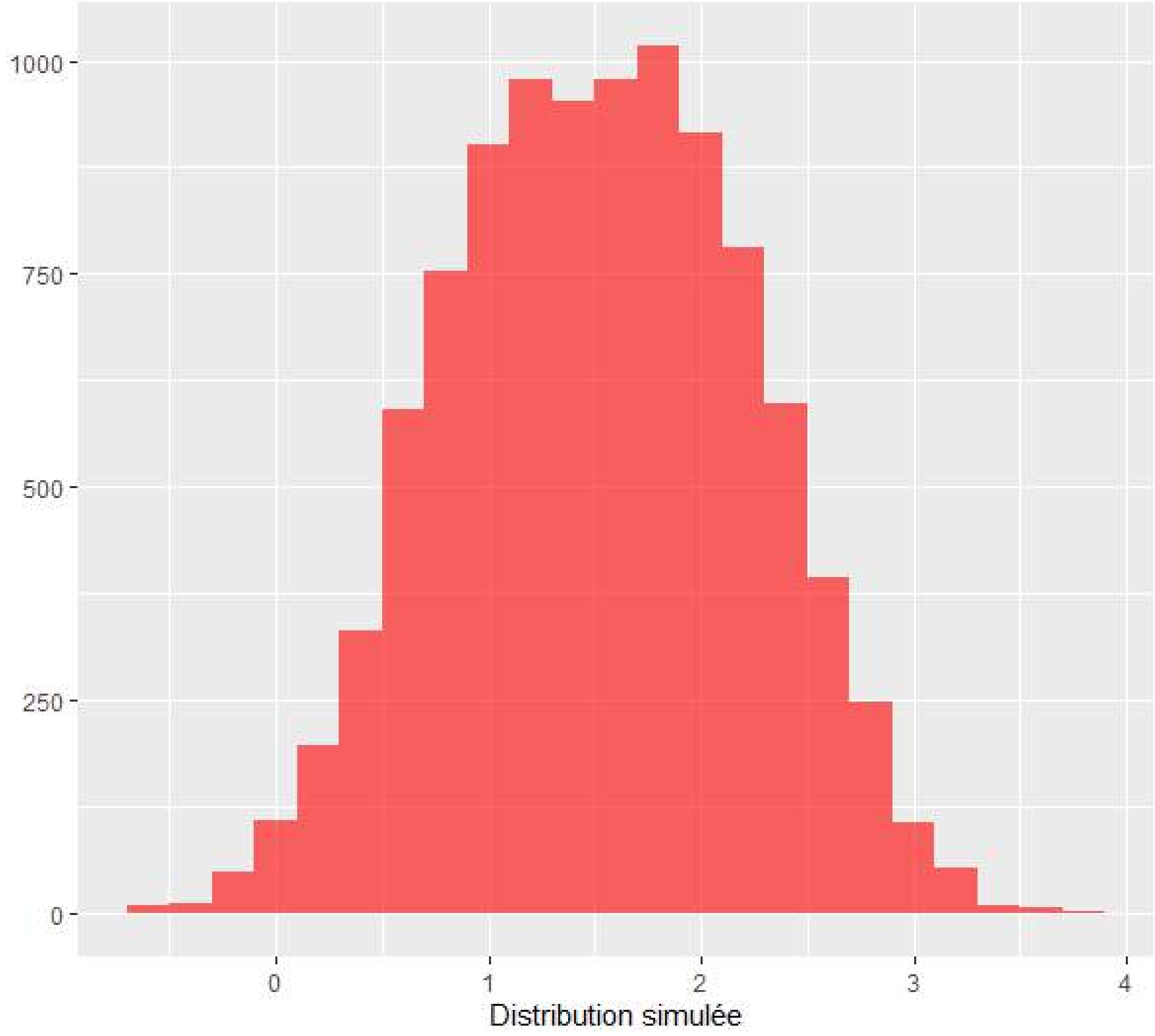
CODE R POUR SIMULER CET EXEMPLE

```
library(ggplot2)
coinflips= 1*(runif(10000)>0.5)
table(coinflips)

## coinflips
## 0       1
## 5024 4976

output=rep(0,10000)
sd1=0.5;sd2=0.5;mean1=1;mean2=3
for (i in 1:10000){
if (coinflips[i]==0)
output[i]=rnorm(1,mean1,sd1)
else
output[i]=rnorm(1,mean2,sd2) }
group=coinflips+1
do=data.frame(output)
qplot(output, data=do, geom="histogram",
fill=I("red"), binwidh=0.2, alpha=I(0.6),
xlab="Distribution simulée")
```



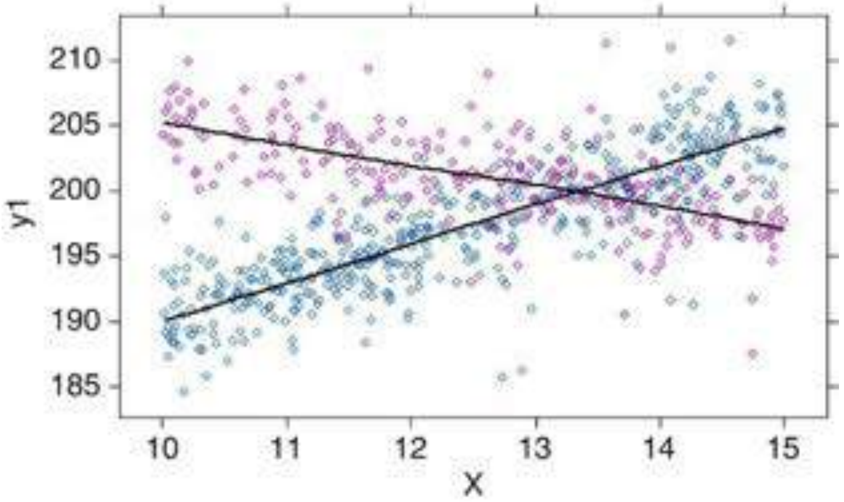
CODE R POUR SIMULER CET EXEMPLE

```
library(ggplot2)
coinflips= 1*(runif(10000)>0.5)
table(coinflips)

## coinflips
## 0       1
## 5024 4976

output=rep(0,10000)
sd1=0.5;sd2=0.5;mean1=1;mean2=2
for (i in 1:10000){
if (coinflips[i]==0)
output[i]=rnorm(1,mean1,sd1)
else
output[i]=rnorm(1,mean2,sd2) }
group=coinflips+1
do=data.frame(output)
qplot(output, data=do, geom="histogram",
fill=I("red"), binwidh=0.2, alpha=I(0.6),
xlab="Distribution simulée")
```

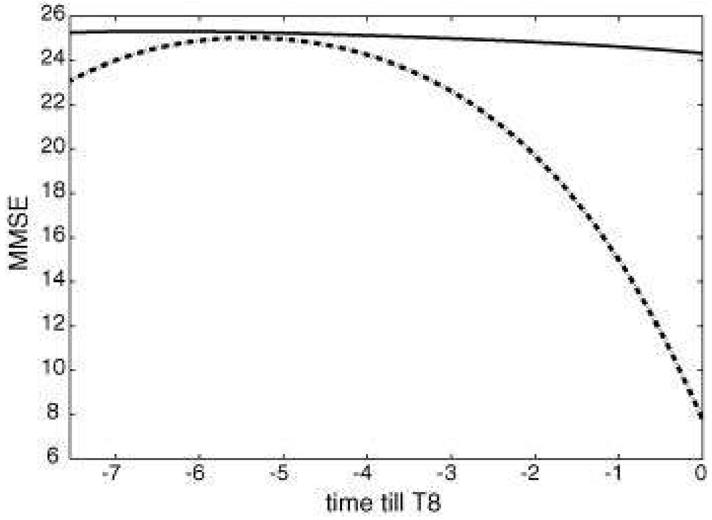


QUELQUES EXEMPLES DE MODÉLES DE MÉLANGE

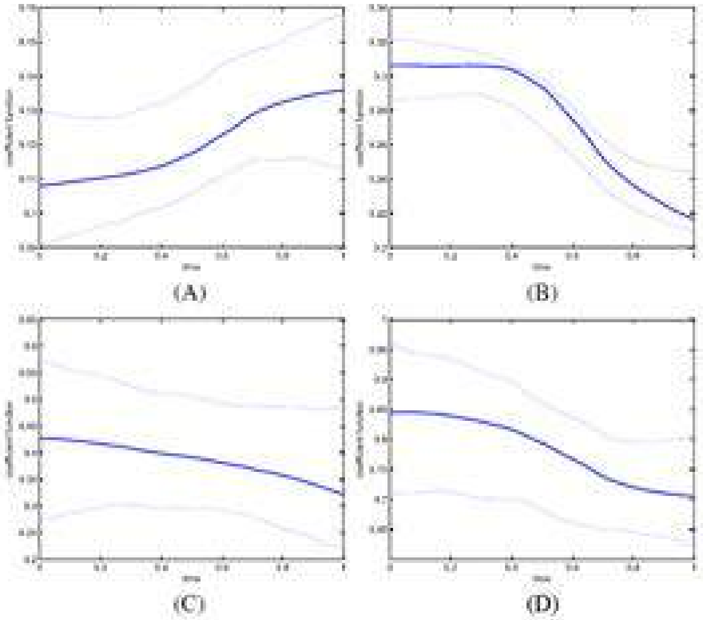
Mélange de régressions linéaires



Mélange de régressions polynomiales à effets mixtes*



Mélange de régressions à coefficients variables



*Proust, C., & Jacqmin-Gadda, H. (2005). Estimation of linear mixed models with a mixture of distribution for the random effects. Computer methods and programs in biomedicine, 78(2), 165–173. https://doi.org/10.1016/j.cmpb.2004.12.004

MODÉLE DE MÉLANGE

- Un modèle de mélange représente une distribution dans laquelle nous échantillonnons d'abord la variable latente z, puis les observations x à partir d'une distribution qui dépend de z.

$$h(x, \theta) = \sum_{c=1}^{C} \pi_c h_c(x, \theta_c)$$

$$p(z, \mathbf{x}) = p(z)\, p(\mathbf{x} \mid z)$$

$$\sum_{c=1}^{C} \mathbb{1}_{z=c} X_c$$

Avec $X_c \sim \mathcal{N}(\mu_c, \sigma_c^2)$ ou $X_c \sim \mathcal{N}(Y\beta_c, \sigma_c^2)$ ou $X_c \sim \mathcal{N}(f_c(Y), \sigma_c^2)$ ou ...

- Les probabilités utilisées pour échantillonner z sont appelées proportions de mélange

ESTIMATION : ALGORITHME EM

- La variable latente z (les clusters) est inconnue (non-observable) et les paramètres sont à estimer (e.g. les moyennes et écart-types des différents clusters dans un mélange gaussien).

- **Espérance-maximisation (EM)** est une méthode itérative qui alterne entre deux étapes :

  - **Étape d'espérance (étape E) :** Calcul des espérances a posteriori des variables latentes z
  - **Étape de maximisation (étape M) :** Estimer les paramètres en maximisant l'espérance de la vraisemblance des paramètres, en z sachant x et θ à l'itération précédente.

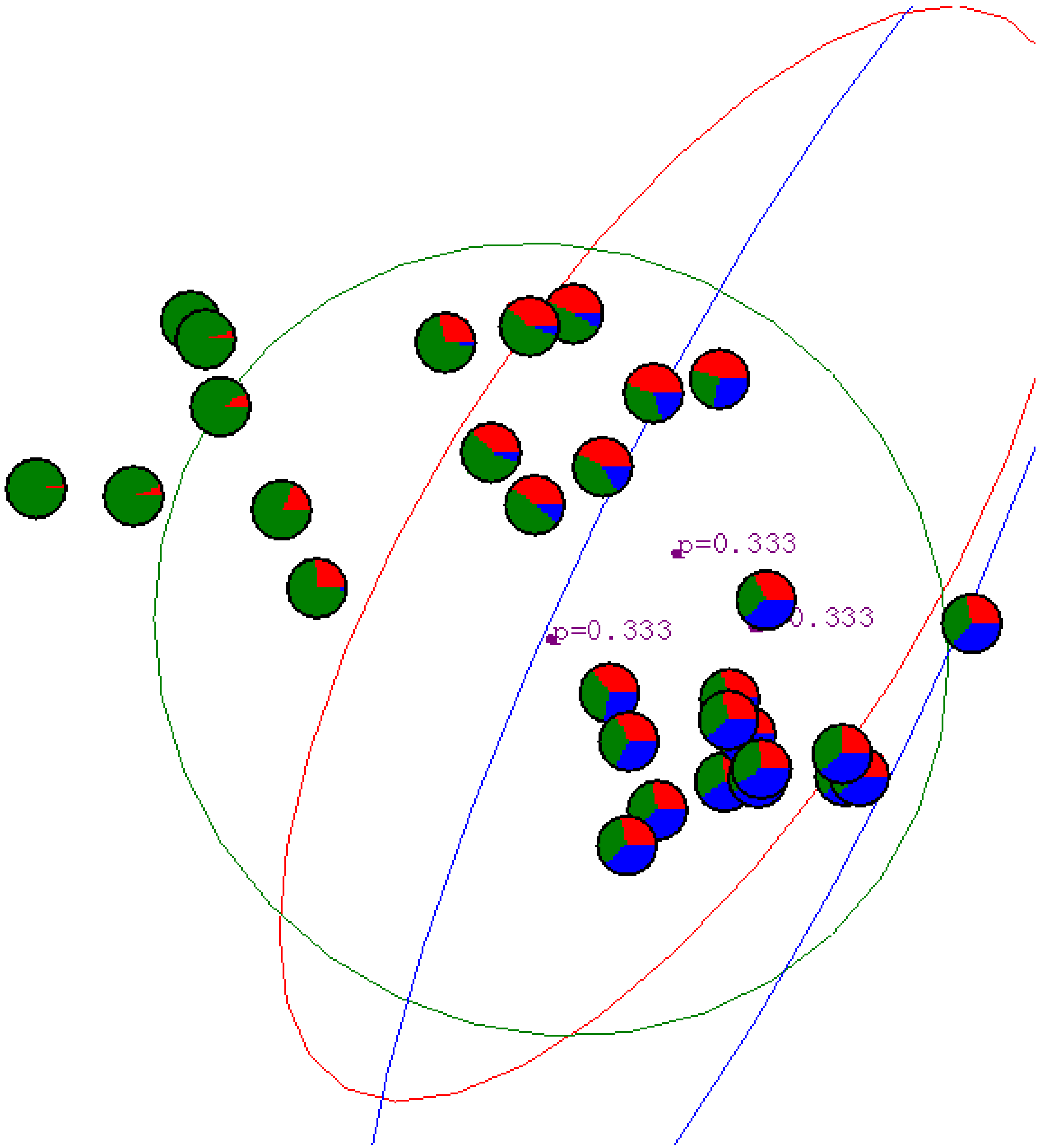
Étape E

$$\begin{aligned} r_k^{(i)} &\leftarrow \Pr(z^{(i)} = k \mid x^{(i)}) \\ &\propto \pi_k \cdot \mathcal{N}(x^{(i)}; \mu_k, \sigma_k) \end{aligned}$$

Étape M

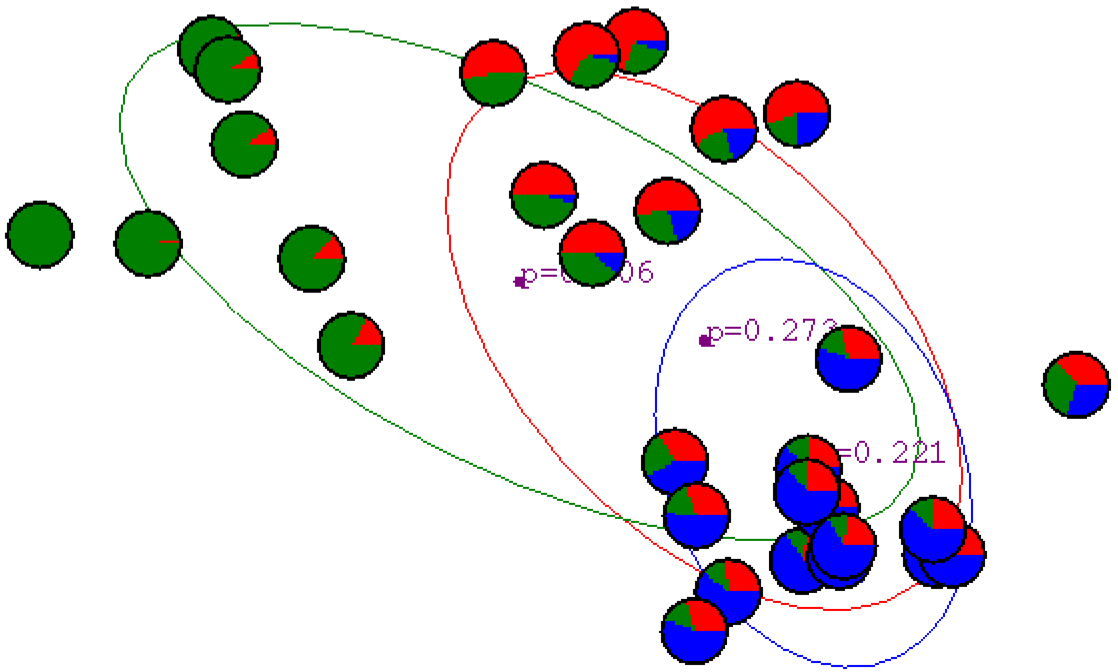
$$\theta \leftarrow \arg\max_{\theta} \sum_{i=1}^{N} \sum_{k=1}^{K} r_k^{(i)} \left[\log \Pr(z^{(i)} = k) + \log p(\mathbf{x}^{(i)} \mid z^{(i)} = k) \right]$$

Exemple de mélange de 3 gaussiennes: initialisation



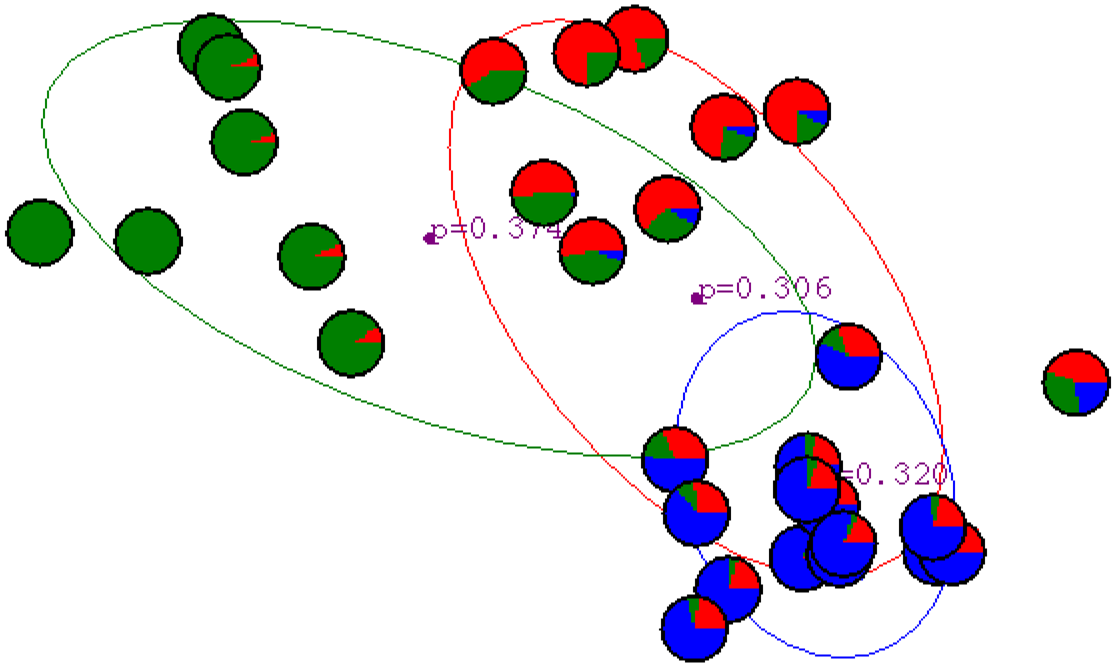
Andrew W. Moore (2001). Clustering with Gaussian mixtures. CMU school of computer science.

1ère iteration



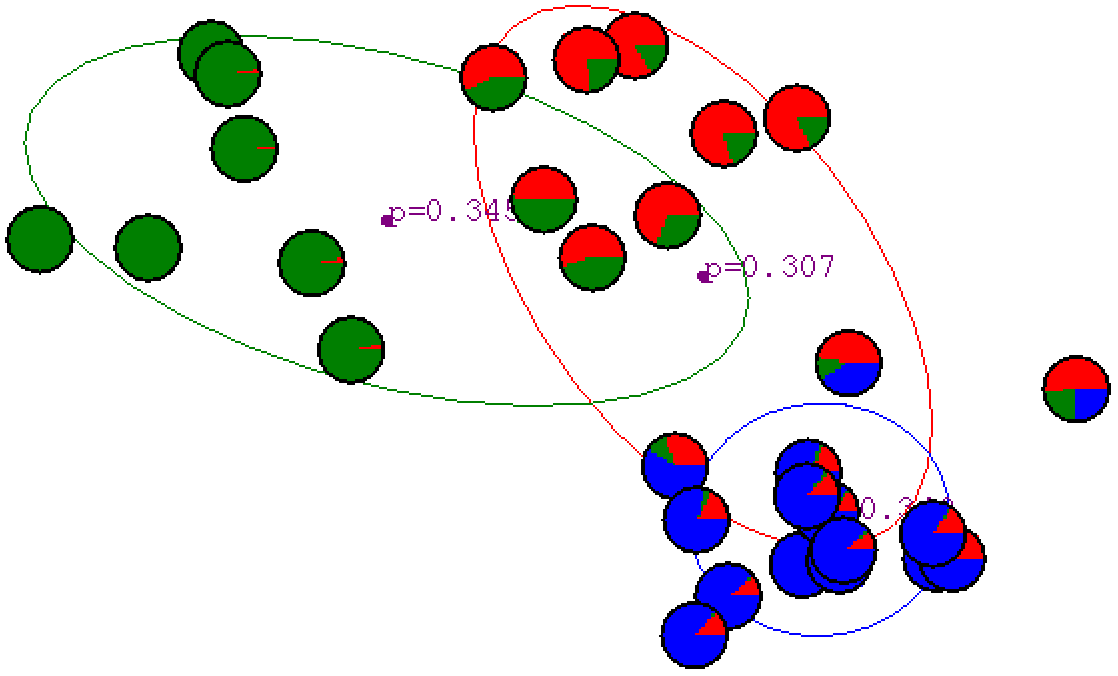
Andrew W. Moore (2001). Clustering with Gaussian mixtures. CMU school of computer science.

2ème iteration



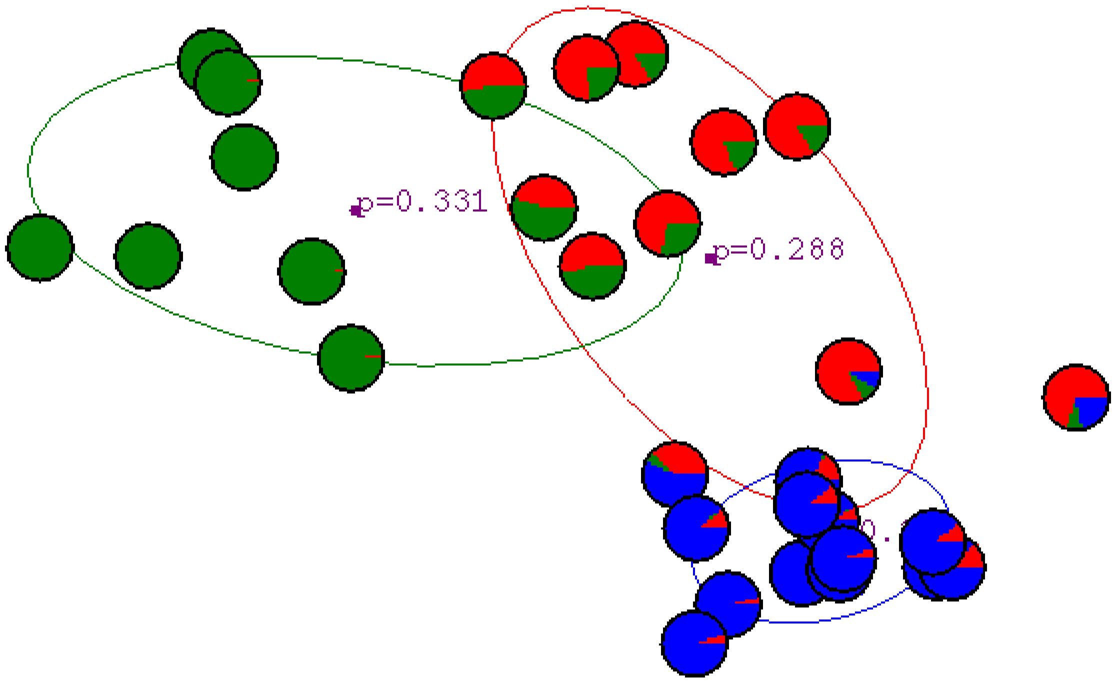
Andrew W. Moore (2001). Clustering with Gaussian mixtures. CMU school of computer science.

3ème iteration



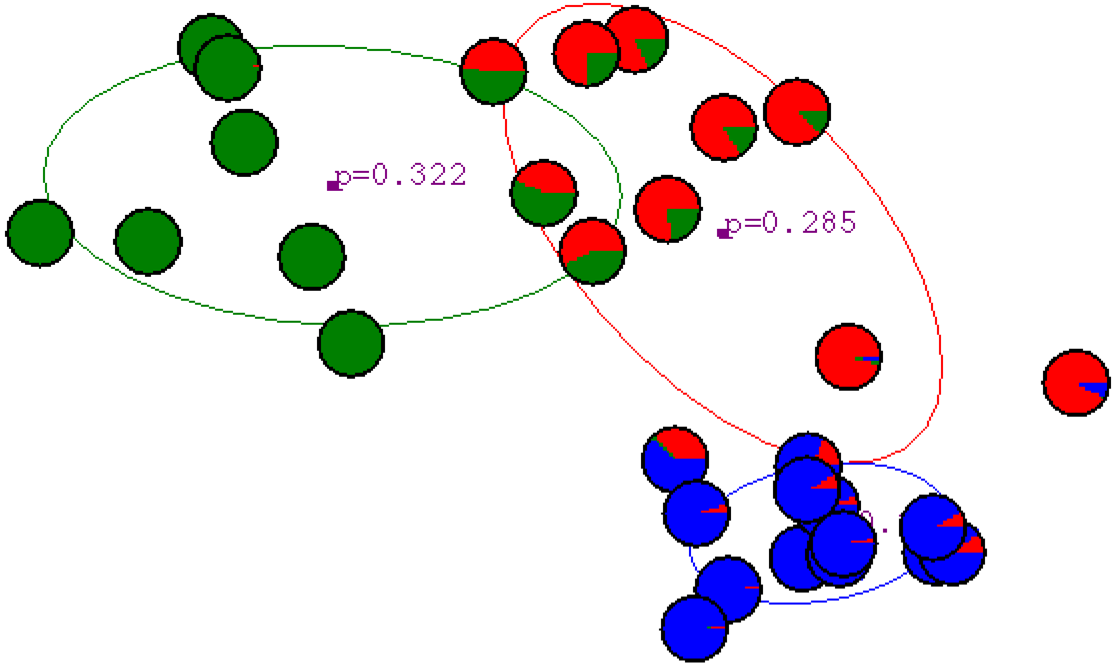
Andrew W. Moore (2001). Clustering with Gaussian mixtures. CMU school of computer science.

4ème iteration



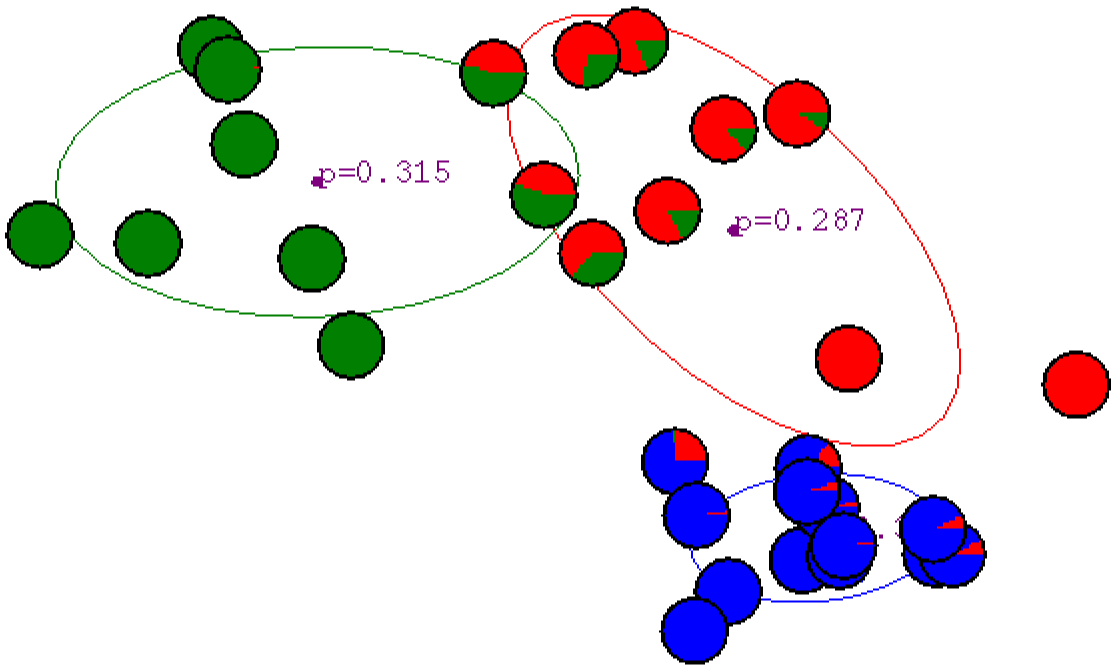
Andrew W. Moore (2001). Clustering with Gaussian mixtures. CMU school of computer science.

5ème iteration



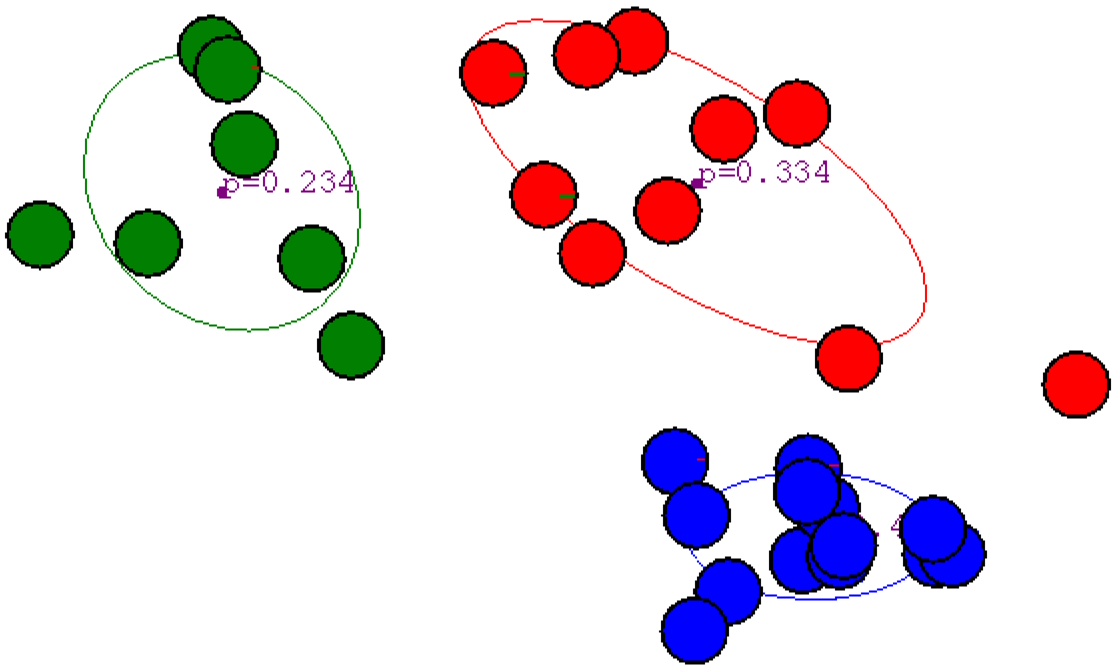
Andrew W. Moore (2001). Clustering with Gaussian mixtures. CMU school of computer science.

6ème iteration



Andrew W. Moore (2001). Clustering with Gaussian mixtures. CMU school of computer science.

20ème iteration



Andrew W. Moore (2001). Clustering with Gaussian mixtures. CMU school of computer science.

DEUX ASPECTS À CONSIDÉRER EN PRATIQUES

- Sélection du nombre de composantes :
  Sur quel critère(s) se baser ?

- Initialisation
  Comment initialiser les paramètres à l'itération 0 ?

SÉLÉCTION DU NOMBRE DE COMPOSANTES

- Nombre de cluster connu apriori.
- Critères d'information : par exemple AIC, BIC, ICL :

$$AIC = -2\log L(\theta) + 2\,(C\#parameters)$$

$$BIC = -2\log L(\theta) + \log(N)\,(C\#parameters)$$

$$ICL = BIC_{MLFA} - 2\sum_{c=1}^{C}\sum_{i=1}^{n}\hat{v}_{ic}\log(\hat{v}_{ic})$$

- Test du rapport de vraisemblance : dans un cadre ML.

INITIALISATION DES PARAMÉTRES

- Construire un vecteur de paramètres approprié θ.
  - Aléatoire.
  - Autres méthodes d'estimation.

- Classer les observations/attribuer des probabilités a posteriori à chaque observation.
  - Aléatoire.
  - Résultats de l'analyse des clusters : par exemple, clustering hiérarchique, k-means.

- Utiliser de courtes séries de EM ou SEM avec différentes initialisations (Biernacki et al., 2003).

DES LIBRAIRIES R POUR LES MODÉLES DE MÉLANGE

Package	Version	Regression	Implemented models	Downloads per day	Last update	Imports	Recursive dependencies	Language
mclust	5.4.7	✗	✗	5,223	31/10/2022	R (≥ 3.0)	0	Fortran
flexmix	2.3-17	✓	Poisson, binary, non-parametric, semi-parametric	3,852	07/06/2022	R (≥ 2.15.0), modeltools, nnet, stats4	3	R
mixtools	1.2.0	✓	multinomial, gamma, Weibull, non-parametric, semi-parametric	178	05/02/2022	R (≥ 3.5.0), kernlab, segmented, survival	6	C

MÉLANGE DE RÉGRESSIONS LINÉAIRES

$$Y = \sum_{c=1}^{2} \mathbb{1}_{z=c}(\beta_{0c} + \beta_{1c}X + \beta_{2c}X^2 + \epsilon)$$

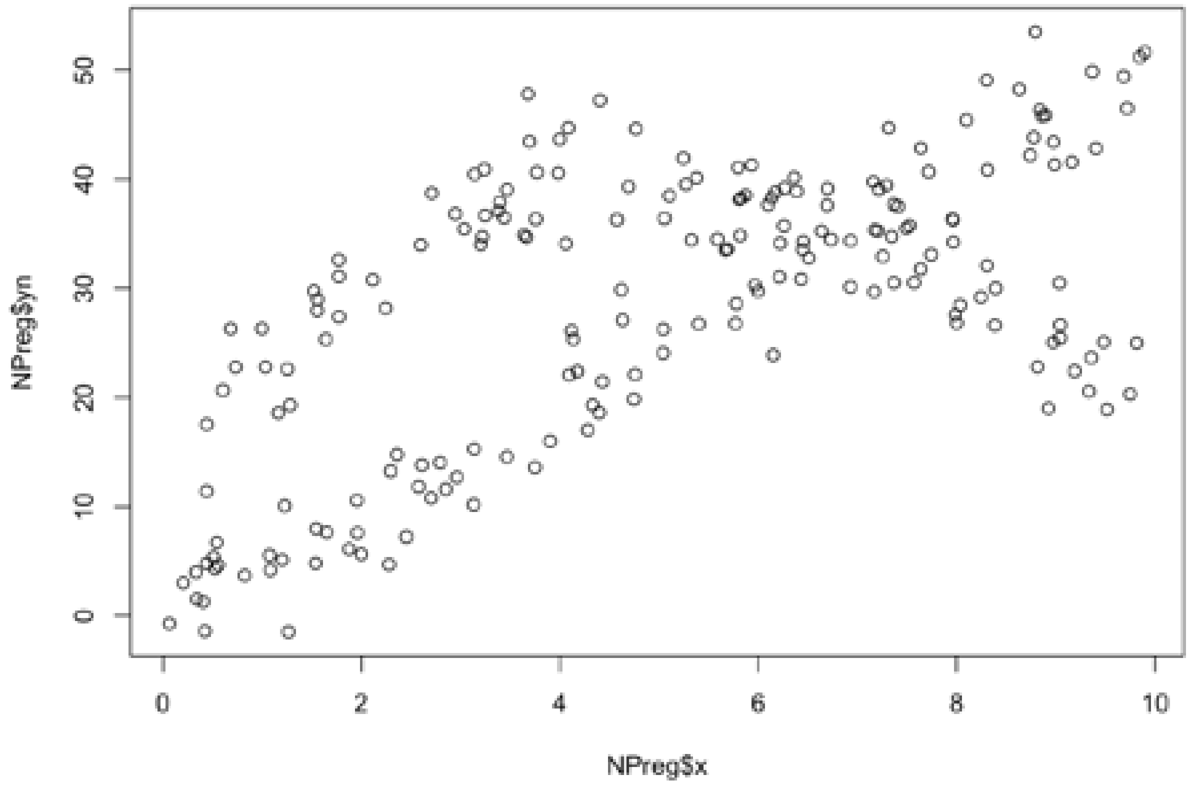
$$N = 200$$

C1: $$Y = 5X + \epsilon$$

C2: $$Y = 15 + 10X - X^2 + \epsilon$$

$$\pi_1 = \pi_2 = 0.5$$

$$\epsilon \sim \mathcal{N}(0, 3^2)$$



MÉLANGE DE RÉGRESSIONS LINÉAIRES

```r
library(flexmix)
data(NPreg)

m1 = flexmix(yn ~ x + I(x^2), data = NPreg, k = 1)

m2 = flexmix(yn ~ x + I(x^2), data = NPreg, k = 2)

m3 = flexmix(yn ~ x + I(x^2), data = NPreg, k = 3)

## OU

m = initFlexmix(yn ~ x + I(x^2), data = NPreg, k = 1:3, nrep=10)

m = unique(m) #pour garder que le modèle avec la Vrai Max pour chaque k

m2=getModel(m,which="BIC") #ou ICL ou AIC
```

```
Summary(m1)

             prior size post>0 ratio
Comp.1        1   200    200     1

'log Lik.' -728.3149 (df=4)
AIC: 1464.63   BIC: 1477.823

Summary(m2)

               prior size post>0 ratio
Comp.1 0.506  100    141 0.709
Comp.2 0.494  100    145 0.690

'log Lik.' -642.5454 (df=9)
AIC: 1303.091   BIC: 1332.776

Summary(m3)

                 prior size post>0  ratio
Comp.1 0.4090   87    130 0.6692
Comp.2 0.4921  101    145 0.6966
Comp.3 0.0989   12    122 0.0984

'log Lik.' -638.4451 (df=14)
AIC: 1304.89   BIC: 1351.067
```

MÉLANGE DE RÉGRESSIONS LINÉAIRES

$$Y = 15 + 10X - X^2 + \epsilon$$

```
parameters(m2, component = 1)
```

```
## Comp.1
## coef.(Intercept) 14.7171315
## coef.x            9.8462869
## coef.I(x^2)      -0.9683139
## sigma             3.4801398
```

$$Y = 5X + \epsilon$$

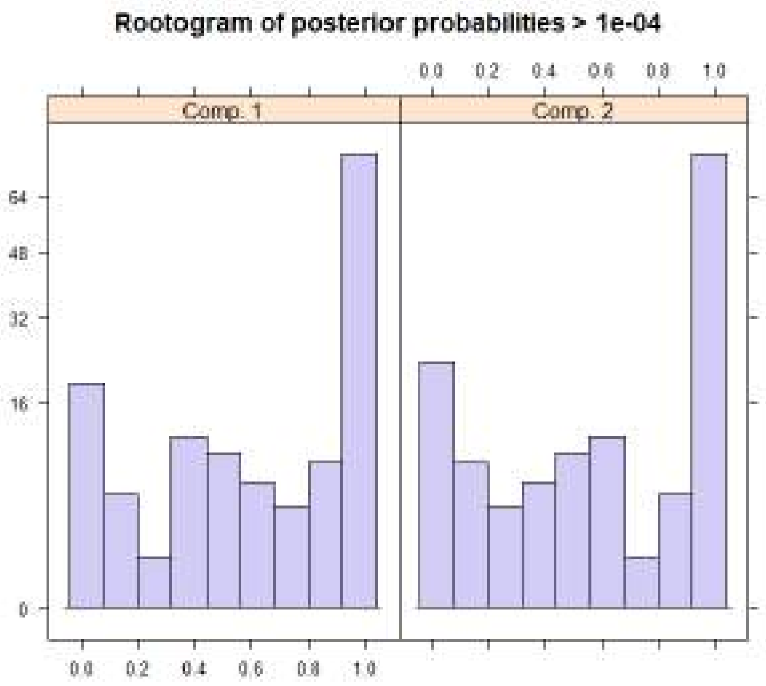
```
parameters(m2, component = 2)
```

```
## Comp.2
## coef.(Intercept) -0.20945380
## coef.x            4.81724681
## coef.I(x^2)       0.03621418
## sigma             3.47590252
```

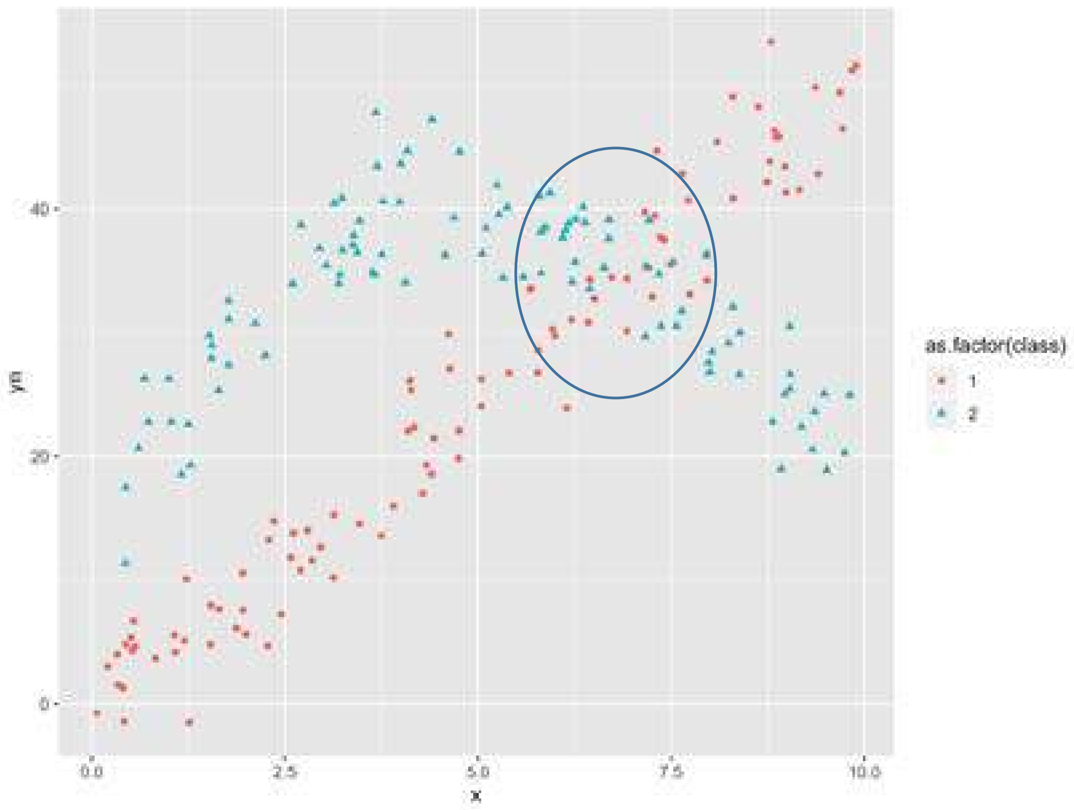
MÉLANGE DE RÉGRESSIONS LINÉAIRES

```
table(NPreg$class, clusters(m1))
```

	1	2
1	5	95
2	95	5




```
ggplot(NPreg,aes(x,yn)) +geom_point(aes(colour =
as.factor(class),shape=as.factor(class)))
```



MÉLANGES DE MODÉLES À EFFETS MIXTES

$$Y_i = \sum_{c=1}^{C} \mathbb{1}_{z_i=c}(X_i\beta_c^T + Z_i\mu_{ci}^T + \epsilon_{ci})$$

$$\sum_{c=1}^{C} \pi_c f_c(Y_i|X_i\beta_c^T, Z_iG_cZ_i^T + \sigma_c I_{n_i})$$

Y_i : La variable à expliquer / variable sortie / variable dépendante

X_i & Z_i : Les matrices des covariables pour les effets fixes (X) et aléatoires (Z)

β_c : Les coefficients associés au effets fixes pour les individus du cluster c

μ_{ci} : Les effets aléatoires de l’individu i du cluster c qui suivent une loi $\sim \mathcal{N}(0_{q\times n_i}, G_c)$

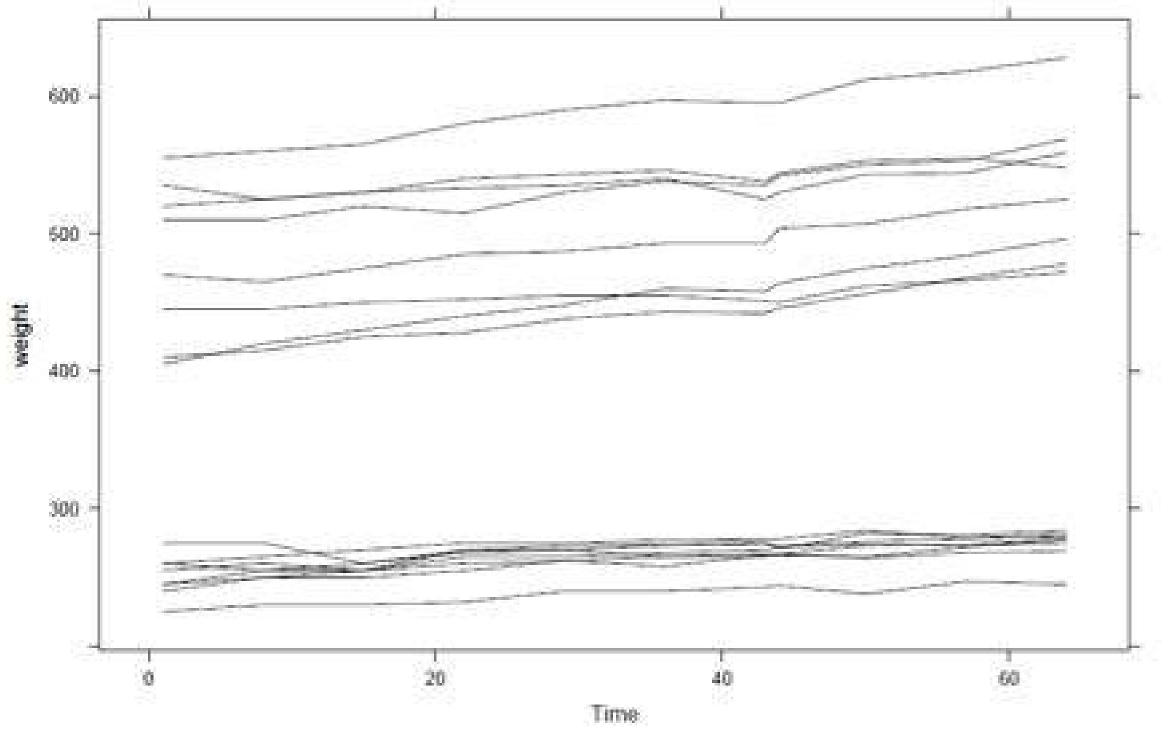
ϵ_{ci} : les erreurs aléatoires qui suivent une loi $\sim \mathcal{N}(0_{n_i}, \sigma_c I_{n_i})$

MÉLANGES DE MODÉLES À EFFETS MIXTES : EX1

- Données *BodyWeight* provenant du package *nlme*.

- N=176 observations sur le poids de n=16 rats à 11 moments différents.

- La variable *Diet* est une variable catégorielle indiquant ce que les rats ont mangé. Afin d’appliquer un modèle de mélange, nous supposerons que cette variable n’a pas été observée.

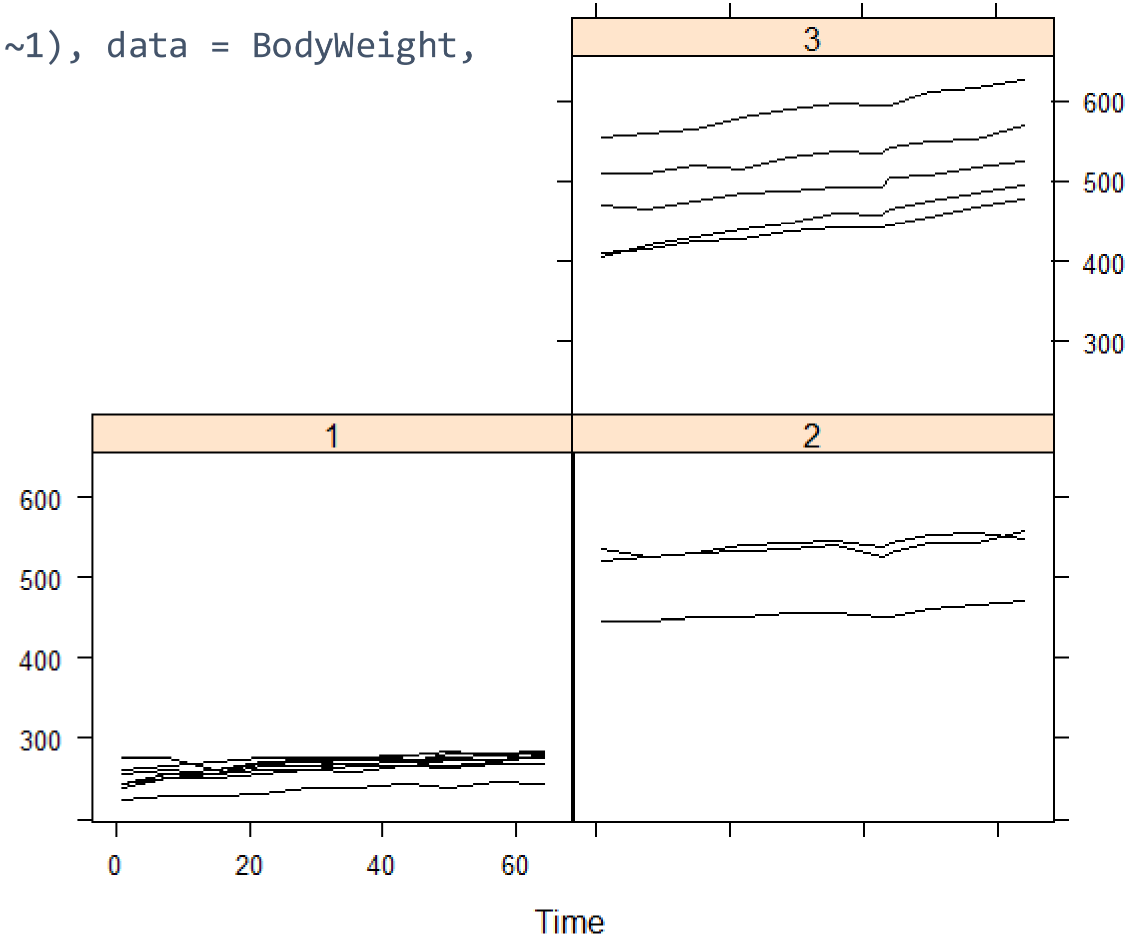


MÉLANGES DE MODÉLES À EFFETS MIXTES : EX1

```
data(BodyWeight)
mod_mix = stepFlexmix(weight ~ Time | Rat, k = 1:3, nrep = 10, model = FLXMRlmm(random = ~ 1), data = BodyWeight)

unique(mod_mix)
Call:
stepFlexmix(weight ~ Time | Rat, model = FLXMRlmm(random = ~1), data = BodyWeight,
    k = 1:3, nrep = 10, unique = TRUE)

  iter converged k k0     logLik      AIC      BIC      ICL
1    2      TRUE 1  1 -1101.2622 2210.524 2223.206 2223.206
2   36      TRUE 2  2  -610.8742 1239.748 1268.283 1268.283
3   29      TRUE 3  3  -596.0559 1220.112 1264.499 1264.503

mod = getModel(mod_mix)

mod

Call:
stepFlexmix(weight ~ Time | Rat, model = FLXMRlmm(random = ~1), data = BodyWeight,
    k = 3, nrep = 10)
Cluster sizes:
 1  2  3
66 88 22
convergence after 29 iterations
```

MÉLANGES DE MODÉLES À EFFETS MIXTES : EX1

```
summary(mod)

Call:
stepFlexmix(weight ~ Time | Rat, model = FLXMRlmm(random = ~1), data = BodyWeight,
    k = 3, nrep = 10)

       prior size post>0 ratio
Comp.1 0.500   88     88     1
Comp.2 0.188   33     33     1
Comp.3 0.312   55     55     1

'log Lik.' -582.1838 (df=14)
AIC: 1192.368   BIC: 1236.754

parameters(mod)

                      Comp.1       Comp.2      Comp.3
coef.(Intercept) 251.6516561  529.8851675  429.161465
coef.Time          0.3596391    0.3901581    1.064669
sigma2.Random    121.6707336 2607.4358991 4406.054847
sigma2.Residual   14.6091597   35.4136986   28.887654
```

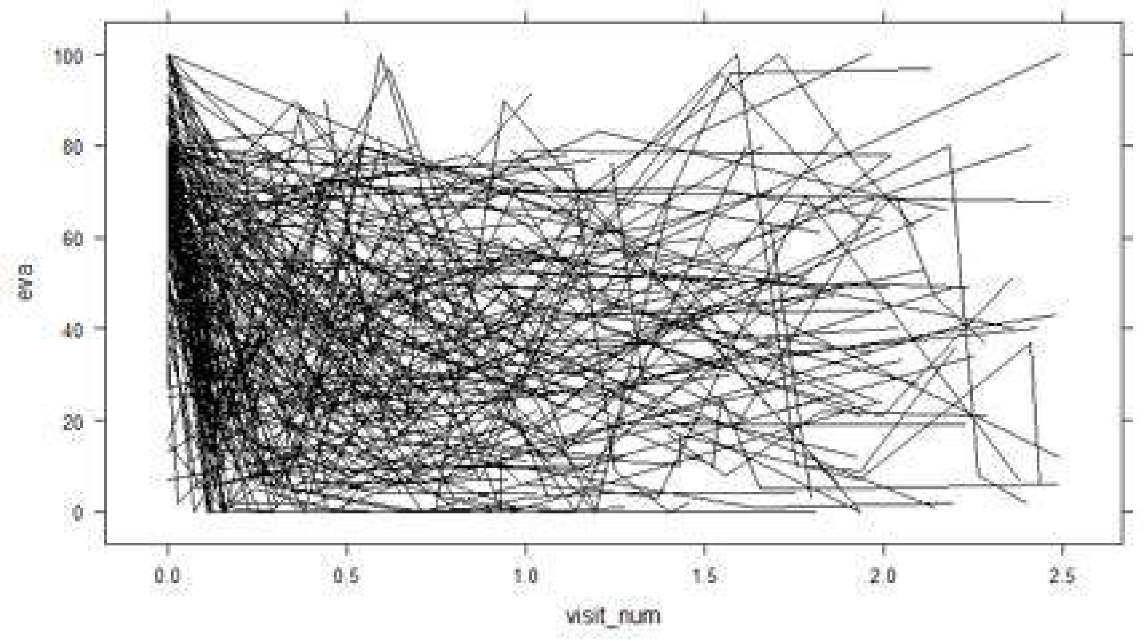


MÉLANGES DE MODÉLES À EFFETS MIXTES : EX2

- Données *PRISMAP, une base de données des patients avec des douleurs chroniques implanté d'un neurostimulateur*.

- N=864 observations des EVA (0-100) de n=281 patients à plusieurs visits (durée mediane de suivi = 10 mois IQR = 2mois – 20 mois).

- Objectif : extraire les trajectoires (avec un polynôme cubique du temps) de l'intensité de la douleur des patients douloureux chroniques implanté d'un neurostimulateur.



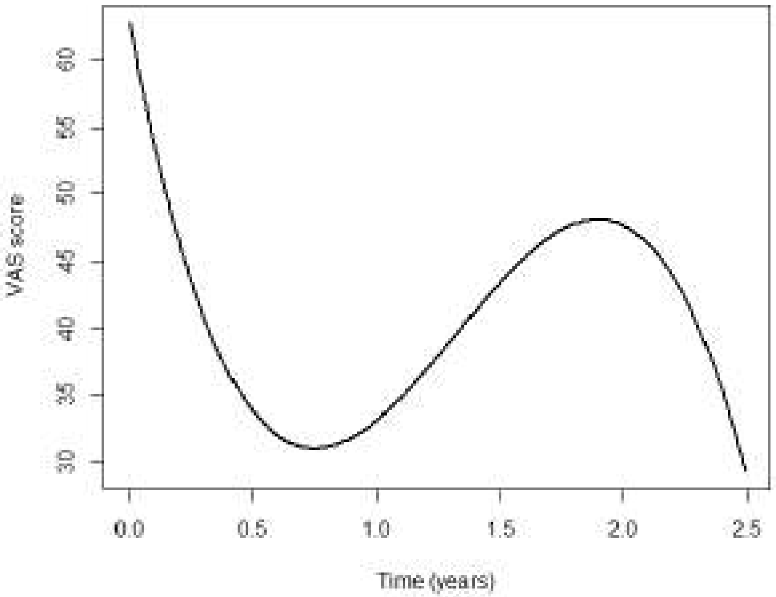
MÉLANGES DE MODÉLES À EFFETS MIXTES : EX2

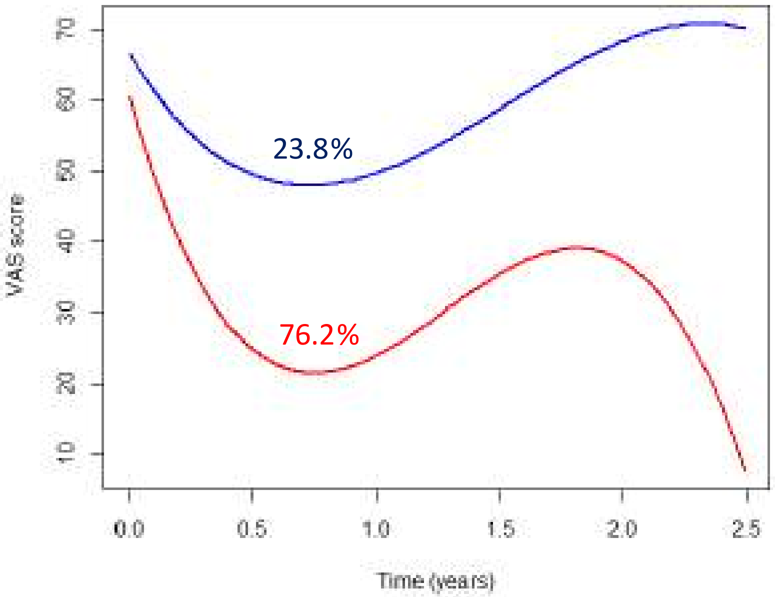
```
m1 = hlme(eva ~ visit_num + I(visit_num^2) +
I(visit_num^3), random= ~ 1, subject='name_pat',
ng=1, data=data_PRISM)
summary(m1)

Statistical Model:
     Dataset: data_PRISM
     Number of subjects: 281
     Number of observations: 864
     Number of latent classes: 1
     Number of parameters: 6

Goodness-of-fit statistics:
     maximum log-likelihood: -3990.16
     AIC: 7992.32
     BIC: 8014.15
Maximum Likelihood Estimates:
Fixed effects in the longitudinal model:
                    coef        Se    Wald p-value
intercept       62.79933 1.55022  40.510 0.00000
visit_num      -97.45952 7.71045 -12.640 0.00000
I(visit_num^2)  90.56464 9.48318   9.550 0.00000
I(visit_num^3) -22.80503 2.93857  -7.761 0.00000
Variance-covariance matrix of the random-effects:
          intercept
intercept   130.896
                              coef       Se
Residual standard error:  22.41484 0.63218
```

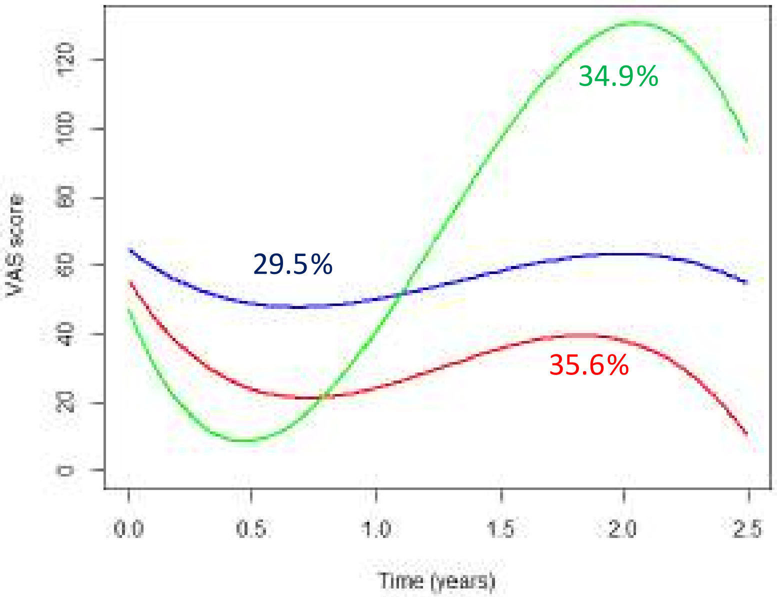
```
m2=hlme(eva ~ visit_num+I(visit_num^2)+I(visit_num^3)
,mixture = ~ visit_num + I(visit_num^2) +
I(visit_num^3), random=~1, subject='name_pat', ng=2,
data=data_PRISM, B=m1)
summary(m2)

Goodness-of-fit statistics:
     maximum log-likelihood: -3980.49
     AIC: 7982.98
     BIC: 8023
Maximum Likelihood Estimates:
Fixed effects in the class-membership model:
(the class of reference is the last class)
                       coef        Se    Wald p-value
intercept class1    0.57463   0.49129   1.170 0.24215
Fixed effects in the longitudinal model:
                       coef        Se    Wald p-value
intercept class1   60.41710   2.56390 23.565 0.00000
intercept class2   66.51776   3.75342 17.722 0.00000
visit_num class1 -120.60400 12.60174 -9.570 0.00000
visit_num class2  -57.20378 19.41497 -2.946 0.00322
I(...) class1     113.70993 14.27183  7.967 0.00000
I(...) class2      51.50554 22.53833  2.285 0.02230
I(...) class1     -29.60842  4.21567 -7.023 0.00000
I(...) class2     -11.22920  6.72774 -1.669 0.09510
Variance-covariance matrix of the random-effects:
          intercept
intercept  50.69708
                               coef        Se
Residual standard error:   21.91857  0.63678
```

```
m3=hlme(eva ~ visit_num+I(visit_num^2)+I(visit_num^
,mixture = ~ visit_num + I(visit_num^2) +
I(visit_num^3), random=~1, subject='name_pat', ng=3,
data=data_PRISM, B=m1)
summary(m3)

Goodness-of-fit statistics:
     maximum log-likelihood: -3977.23
     AIC: 7986.45
     BIC: 8044.66
Maximum Likelihood Estimates:
Fixed effects in the class-membership model:
(the class of reference is the last class)
                       coef        Se   Wald p-value
intercept class1   -0.00212   0.19023 -0.011 0.99111
intercept class2    0.09616   0.67095  0.143 0.88603
Fixed effects in the longitudinal model:
                       coef        Se   Wald p-value
intercept class1   59.75254   3.83872 15.566 0.00000
intercept class2   62.64761   4.16066 15.057 0.00000
intercept class3   66.56412   4.39744 15.137 0.00000
visit_num class1 -168.73790 25.24275 -6.685 0.00000
visit_num class2  -80.80090 22.41627 -3.605 0.00031
visit_num class3  -55.60214 22.01388 -2.526 0.01154
I(...) class1     183.89129 33.71561  5.454 0.00000
I(...) class2      58.77975 25.90731  2.269 0.02328
I(...) class3      50.70763 25.86859  1.960 0.04997
I(...) class1     -51.00017 10.61797 -4.803 0.00000
I(...) class2     -13.54854  7.49921 -1.807 0.07082
I(...) class3     -11.04346  7.54028 -1.465 0.14303
Variance-covariance matrix of the random-effects:
          intercept
intercept  44.78505
```

MÉLANGES DE MODÉLES À EFFETS MIXTES : EX2





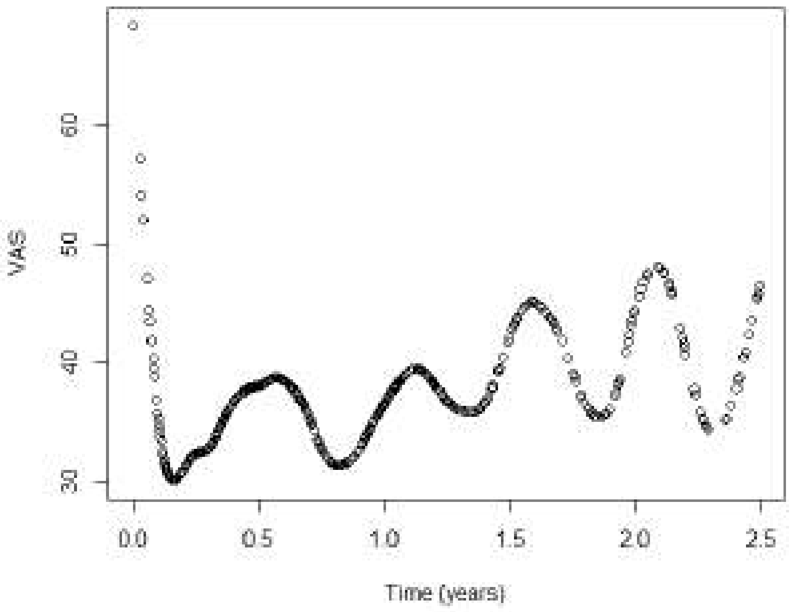



```
postprob(m2)
Posterior classification:
  class1 class2
N 214.00  67.00
%  76.16  23.84
Posterior classification table:
     --> mean of posterior probabilities in each
class
        prob1  prob2
class1 0.7597 0.2403
class2 0.2569 0.7431
```

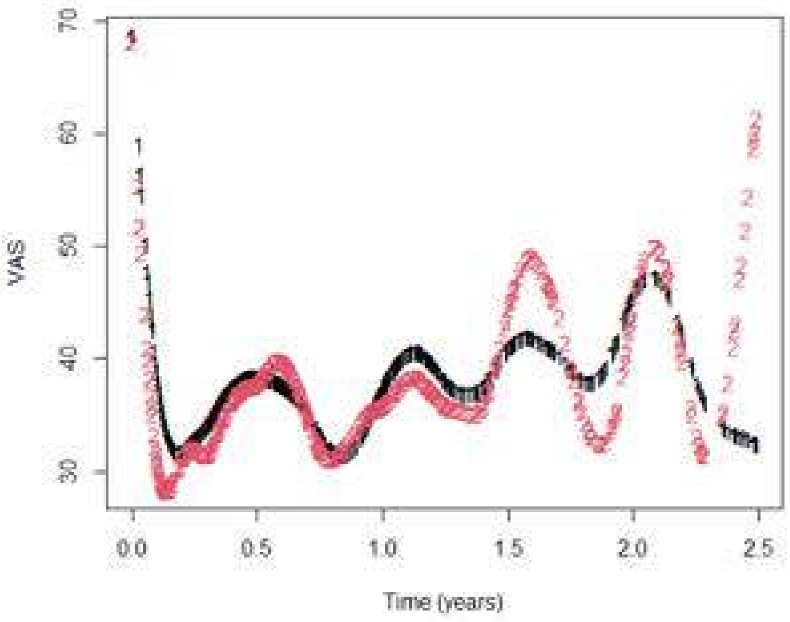
```
postprob(m3)
Posterior classification:
  class1 class2 class3
N  83.00 100.00  98.00
%  29.54  35.59  34.88
Posterior classification table:
     --> mean of posterior probabilities in each
class
        prob1  prob2  prob3
class1 0.6206 0.2620 0.1173
class2 0.2464 0.5267 0.2269
class3 0.1463 0.2594 0.5944
```

MÉLANGES DE MODÉLES À EFFETS MIXTES AVEC DES SPLINES: EX2

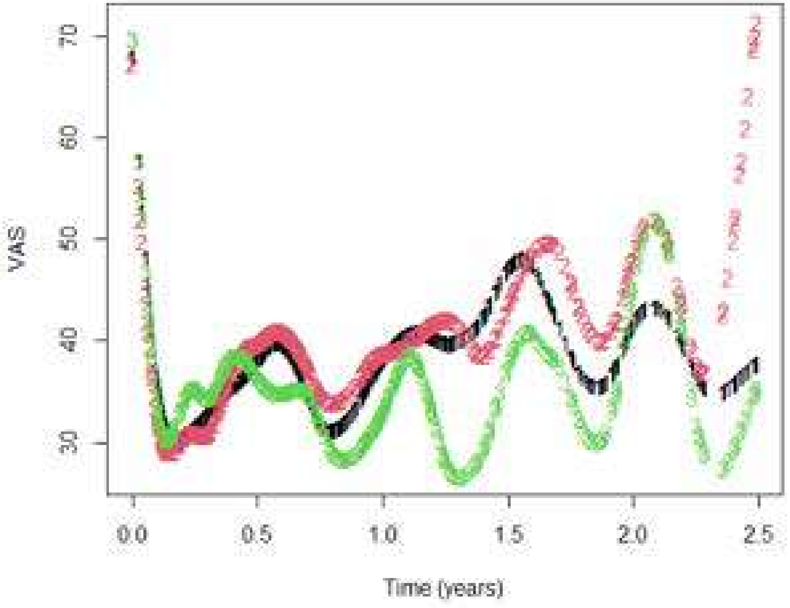
```
mod1 = flexmix(.~.|name_pat, k = 1,
model = FLXMRlmm(eva ~ 0 + visit_num, random = ~ 1,
lm.fit = "smooth.spline"), data = data_PRISM, control
= list(tolerance = 10^-3))

preds=fitted(mod1)

plot(data_PRISM$visit_num,unlist(preds),xlab="Time
(years)",ylab="VAS")
```




```
mod2 = flexmix(.~.|name_pat, k = 2,
model = FLXMRlmm(eva ~ 0 + visit_num, random = ~ 1,
lm.fit = "smooth.spline"), data = data_PRISM,
control = list(tolerance = 10^-3), cluster =
sample(1:2,864,replace=T))

preds=fitted(mod2)

matplot(data_PRISM$visit_num, preds,xlab="Time
(years)",ylab="VAS")
```




```
mod3 = flexmix(.~.|name_pat, k = 3,
model = FLXMRlmm(eva ~ 0 + visit_num, random = ~ 1,
lm.fit = "smooth.spline"), data = data_PRISM,
control = list(tolerance = 10^-3), cluster =
sample(1:3,864,replace=T))

preds=fitted(mod3)

matplot(data_PRISM$visit_num, preds,xlab="Time
(years)",ylab="VAS")
```



MÉLANGE DE MODÉLES NON-PARAMÉTRIQUES :
Mélange de modèles additifs généralisés

$$Y = \sum_{c=1}^{C} \mathbb{1}_{z=c}(\beta_{0c} + f_{1c}(X_1) + f_{2c}(X_2) + \cdots + f_{pc}(X_p) + \epsilon_c)$$

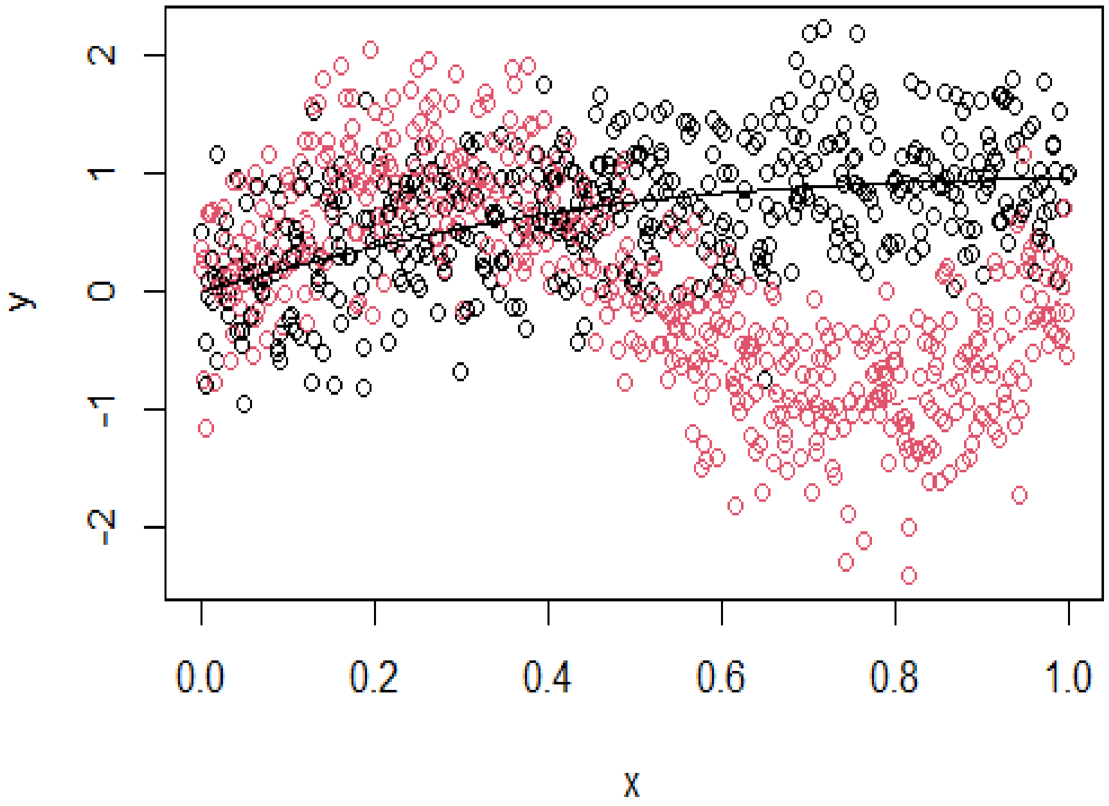
$$N = 200$$

C1 : $Y = sin(2\pi X)$

C2 : $Y = tanh(2x)$

$$\pi_1 = \pi_2 = 0.5$$

$$\epsilon \sim \mathcal{N}(0, 0.5^2)$$

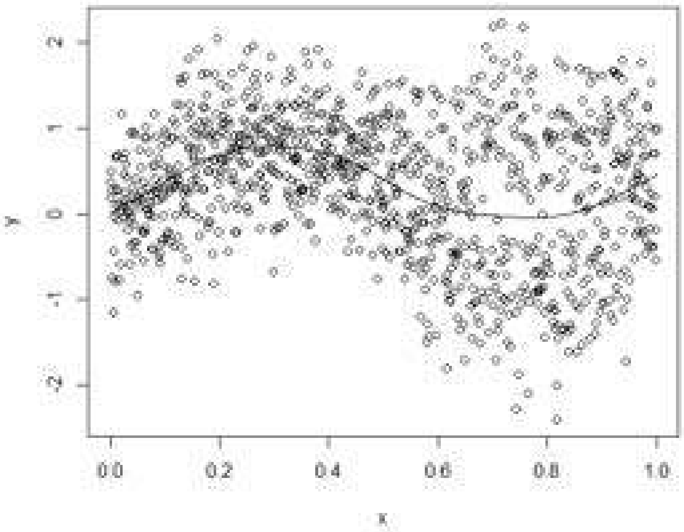


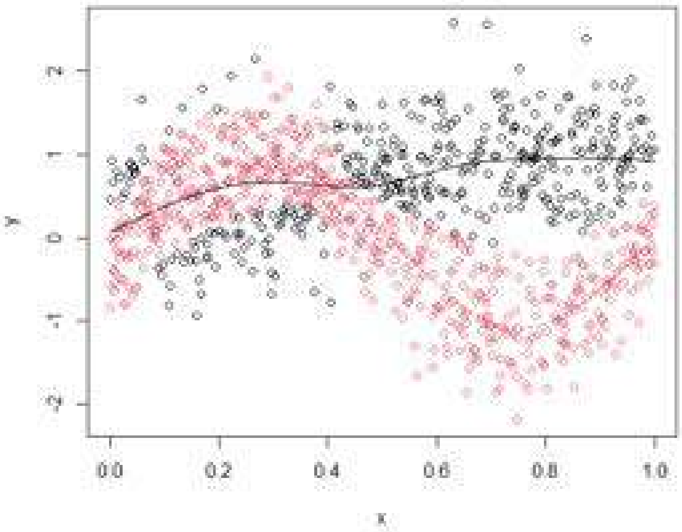
MÉLANGE DE MODÉLES NON-PARAMÉTRIQUES :
Mélange de modèles additifs généralisés

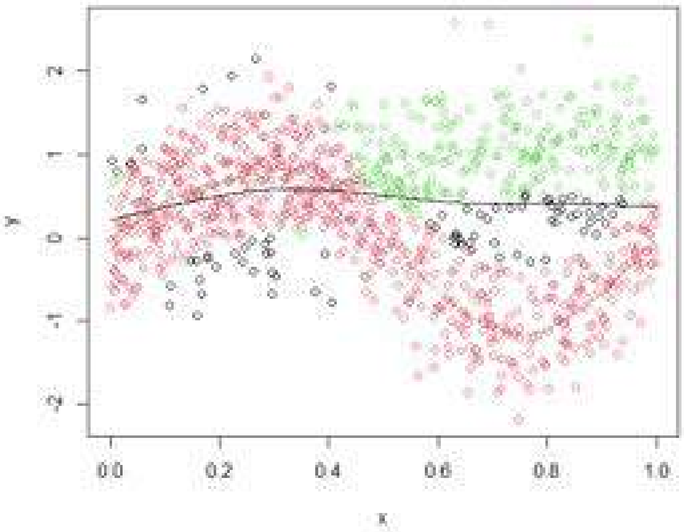
```
mod1=flexmix(y ~ s(x), model = FLXMRmgcv(), k=1,
control = list(tolerance = 10^-3))

plot(y ~ x, col = clusters(mod1))

matplot(x, fitted(mod1), type = "l", add = TRUE)

summary(mod1)

        prior size post>0 ratio
Comp.1      1 1000   1000     1

'log Lik.' -1150.984 (df=7.400336)
AIC: 2316.769   BIC: 2353.088
```

```
mod2=flexmix(y ~ s(x), model = FLXMRmgcv(), k=2,
control = list(tolerance = 10^-3))

plot(y ~ x, col = clusters(mod2))

matplot(x, fitted(mod2), type = "l", add = TRUE)

summary(mod2)

        prior size post>0 ratio
Comp.1 0.488  386    978 0.395
Comp.2 0.512  614    949 0.647

'log Lik.' -998.0276 (df=17.64656)
AIC: 2031.348   BIC: 2117.953
```

```
mod3=flexmix(y ~ s(x), model = FLXMRmgcv(), k=3,
control = list(tolerance = 10^-3))

plot(y ~ x, col = clusters(mod3))

matplot(x, fitted(mod3), type = "l", add = TRUE)

summary(mod3)

        prior size post>0 ratio
Comp.1 0.442  660    915 0.721
Comp.2 0.223   80   1000 0.080
Comp.3 0.336  260    931 0.279

'log Lik.' -998.6012 (df=25.92793)
AIC: 2049.058   BIC: 2176.306
```





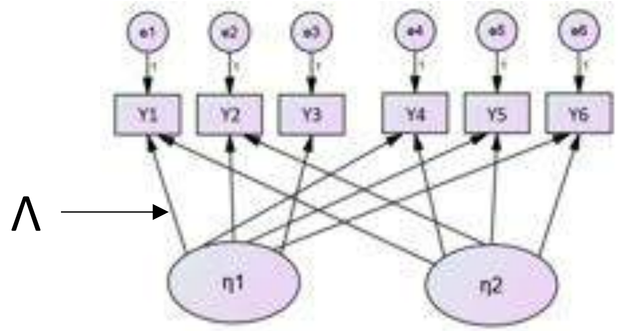


ANALYSE FACTORIELLE

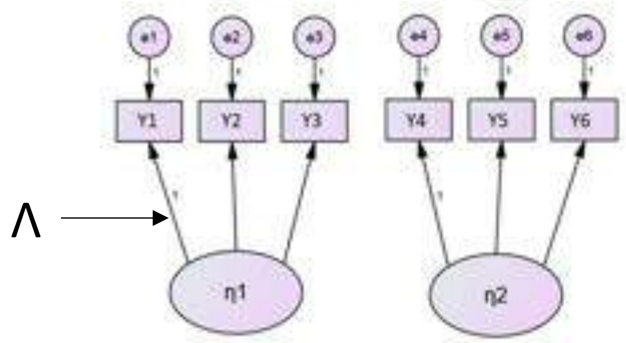
- Modèle d’analyse factorielle : Découvrir m < p facteurs sous-jacents (variables latentes η) à partir des covariances ou corrélations entre les p variables observées dans Y.

$$Y = \Lambda\eta + e$$

  - Réduction des données
  - Développement d'échelles
  - Évaluation de la qualité psychométrique d'un questionnaire
  - Évaluation de la dimensionnalité d'un ensemble de variables



- Analyse factorielle exploratoire : Utilisation des réponses observées pour obtenir une structure factorielle



- Analyse factorielle confirmatoire : Uses new data to determine whether hypothesized Factor structure is appropriate.

Williams, B. A., Onsman, A., & Brown, G. T. (2010). Exploratory factor analysis: A five-step guide for novices. Journal of Emergency Primary Health Care, 8(3), 1 - 13.
Hurley, A. E., Scandura, T. A., Schriesheim, C. A., Brannick, M. T., Seers, A., Vandenberg, R. J., & Williams, L. J. (1997). Exploratory and confirmatory factor analysis: Guidelines, issues, and alternatives. *Journal of Organizational Behavior, 18*(6), 667–683. https://doi.org/10.1002/(SICI)1099-1379(199711)18:6<667::AID-JOB874>3.0.CO;2-T

ANALYSE FACTORIELLE LONGITUDINALE

Problématique : On souhaite réduire J indicateurs/critères à un nombre plus petit K de scores latents évaluant des concepts plus globaux, en utilisant des données longitudinales.

Modèle d'analyse factorielle longitudinale s'écrit :

$$y_{ijt} = \Lambda_j \eta_{it} + \epsilon_{ijt}$$

$$\eta_{ikt} = X_{ikt} \beta_k + Z_{ikt} \xi_{ik} + \omega_{ikt}$$

- Λ_j : les saturations factorielles associée à l'item j .
- η_{it} : les scores latents du patients i à l'instant t.
- X_{ikt} : covariables des eets xes expliquant le score latent k.
- Z_{ikt} : covariables des eets aléatoires expliquant le score latent k.

MÉLANGE D'ANALYSES FACTORIELLES LONGITUDINALES

$$y_{ijt} = \sum_{c=1}^{C} \mathbb{1}_{\{v_i=c\}}(\Lambda_{jc}\eta_{i.tc} + \epsilon_{ijtc}),$$

$$\eta_{iktc} = X_{iktc}\beta_{kc} + Z_{iktc}\xi_{ikc} + \omega_{itc}$$

où v_i est une variables catégorielle représentant la classe latente du patient i. La variable v prend valeur dans $\{1, \ldots, C\}$ avec les probabilités $\{\pi_1, \ldots, \pi_C\}$ $(\sum_{c=1}^{C} \pi_c = 1)$.

Estimation des paramètres $\longrightarrow$ Algorithme EM

Ounajim, A., Slaoui, Y., Louis, P. Y., Billot, M., Frasca, D., & Rigoard, P. (2023). Mixture of longitudinal factor analyzers and their application to the assessment of chronic pain. Statistics in medicine, 42(18), 3259–3282. https://doi.org/10.1002/sim.9804

CHOIX DU NOMBRE DE CLASSE

Le modèle contient 2 hyperparamètres :

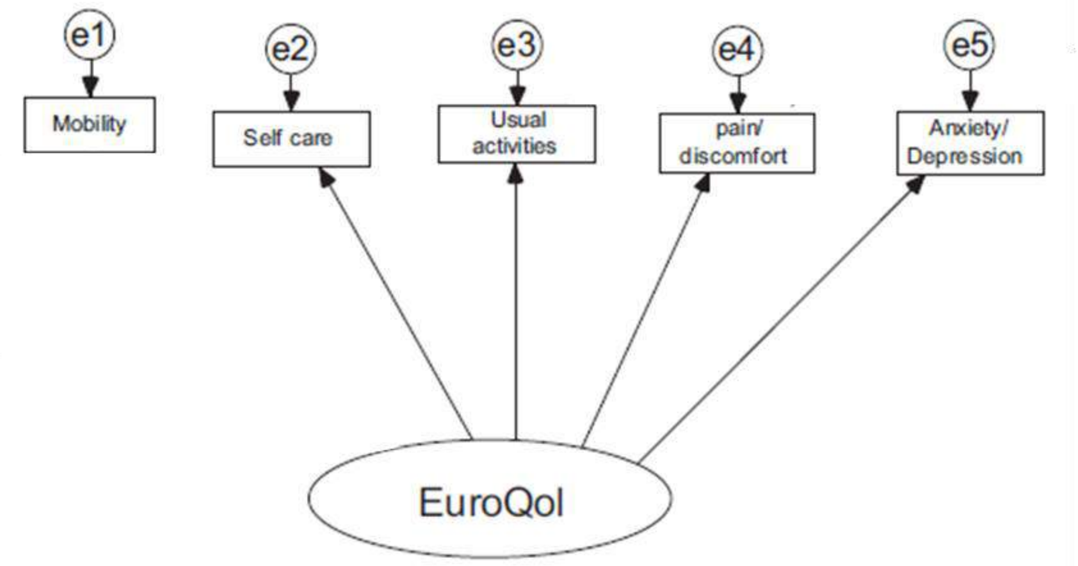
- Le choix du nombre de composantes dans le mélange.
- Le nombre de facteurs latents K à inclure dans le modèle.

$$BIC_{MLFA} = -2 \log L(\theta) + (C(\#parameters) + C - 1) \log(N)$$

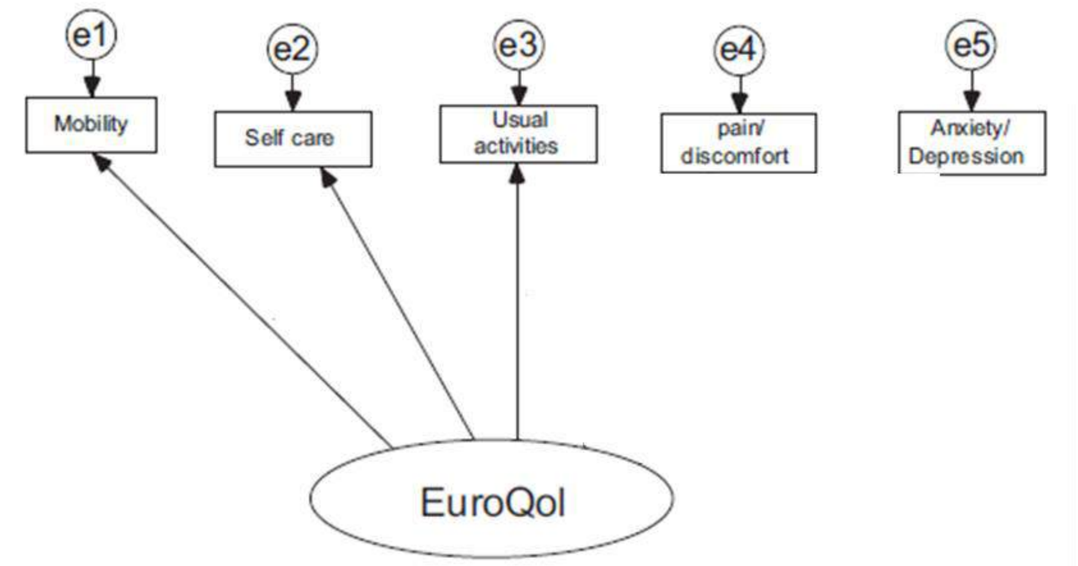
Avec :

$$\#parameters = (JK + J + \frac{K(K+1)}{2} + K(p+1) + \frac{(q+1)K((q+1)K+1)}{2})$$

EXEMPLE



Groupe 1



Groupe 2

ÉXEMPLE D'APPLICATION

Objectifs :

- Identifier K = 1 facteur latent permettant de résumer les 9 variables mesurées.
- Estimer l'évolution dans le temps de ce facteur latent.

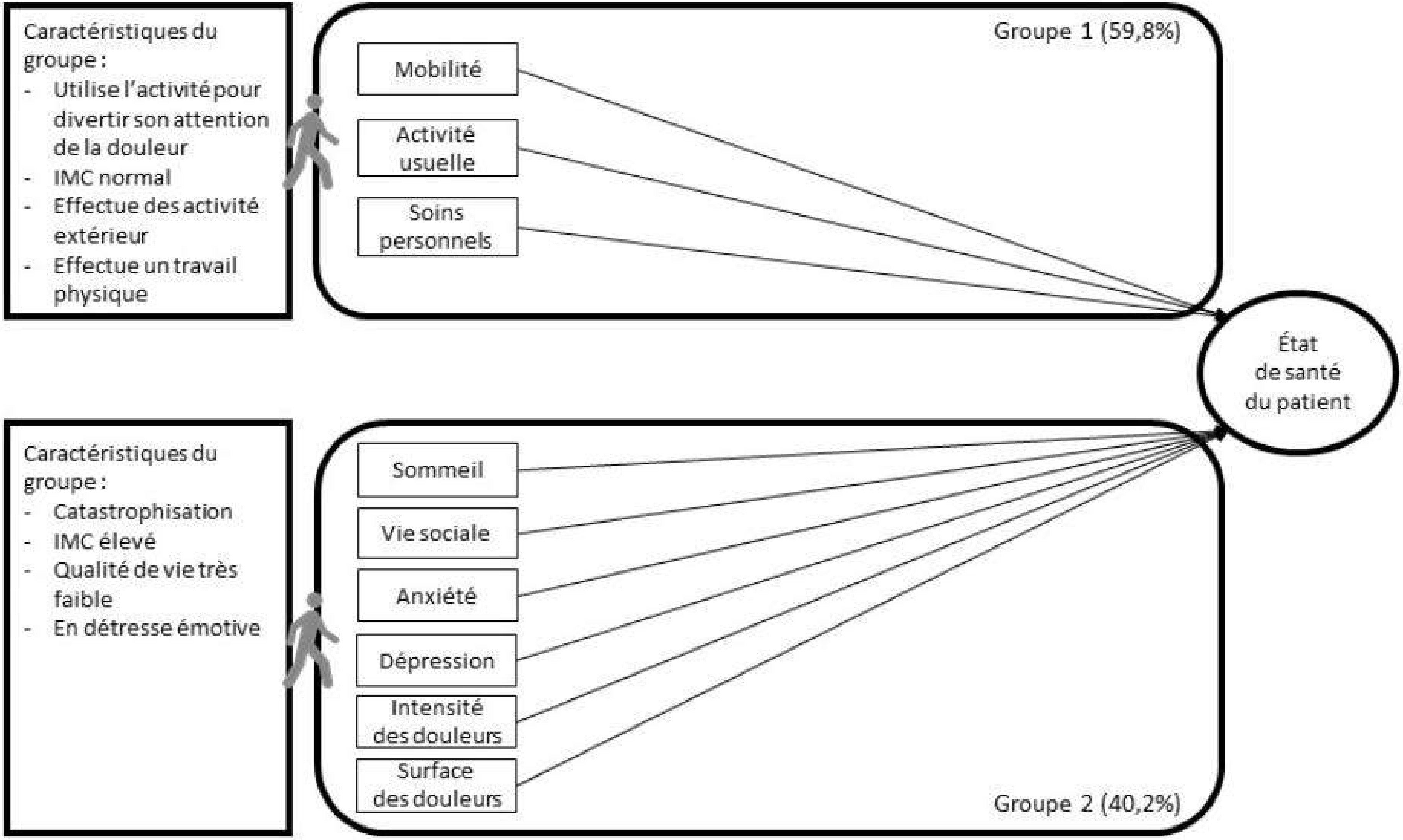
Données :

- n = 194 patients avec données complètes.
- Nombre total d'observations : N = 779 observations.
- J = 9 items qui mesurent la qualité de vie, la capacité fonctionnelle, l'état psychologique, la surface douloureuse et l'intensité de la douleur.

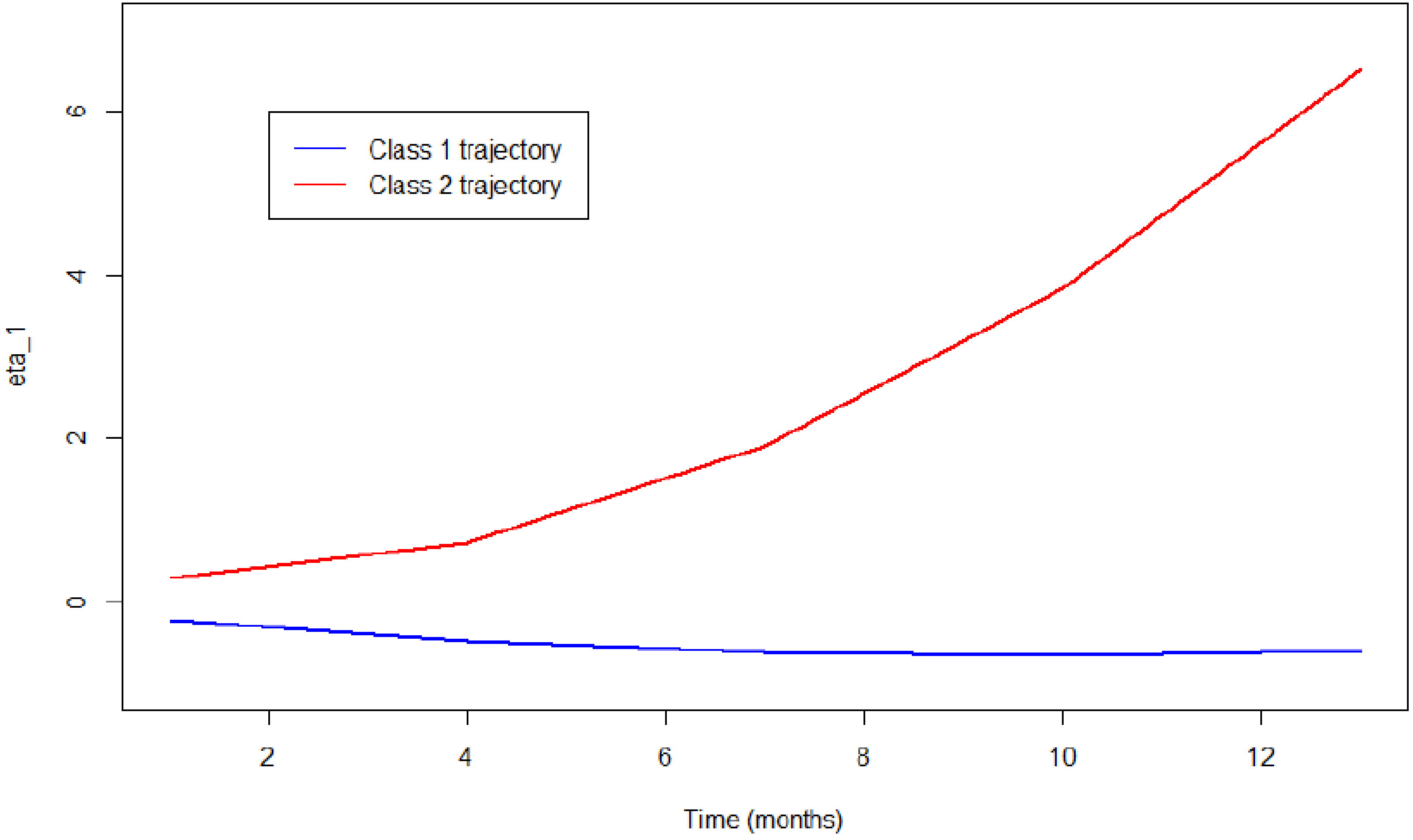
ÉXEMPLE D'APPLICATION

		Component 1 factors ($n = 116$)	Component 2 factors ($n = 78$)
Variable name	Parameter	η_{11}	η_{12}
Mobility	$\lambda_{1,.}$	**1.00**	0.66
Usual activities	$\lambda_{2,.}$	**0.96**	0.64
Personal care	$\lambda_{3,.}$	**0.78**	0.54
Sleeping	$\lambda_{4,.}$	0.35	**0.49**
Social life	$\lambda_{5,.}$	0.56	**0.84**
Depression	$\lambda_{6,.}$	0.48	**1.05**
Anxiety	$\lambda_{7,.}$	0.27	**1.00**
Pain surface	$\lambda_{8,.}$	0.22	**0.44**
Pain intensity	$\lambda_{9,.}$	0.63	**0.71**

ÉXEMPLE D'APPLICATION



ÉXEMPLE D’APPLICATION



AUTRES MÉLANGES POSSIBLES SUR FLEXMIX

- Mélange de modèles linéaires généralisés (Poisson, Multinomial, Gamma, ...)
- Mélange de modèles linéaires généralisés régularisés (glmnet).
- Mélange de modèles linéaires à effets mixtes avec censure à gauche.
- ...

Merci pour votre attention