Identifying patterns in trajectories

Marie-Pierre Etienne

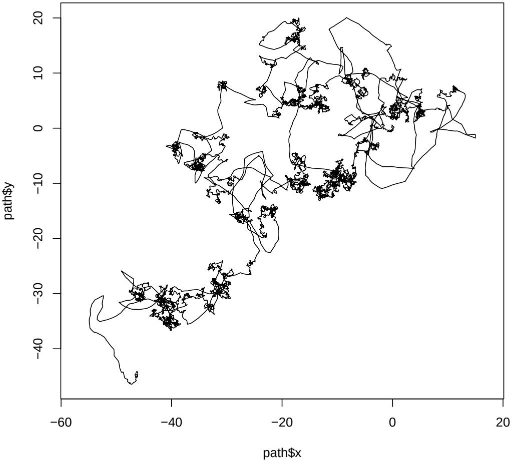
AgroParisTech / INRA

Movement Ecology Workshop 2015 - Port Elizabeth

Detection of homogenous region in trajectories

Why ?

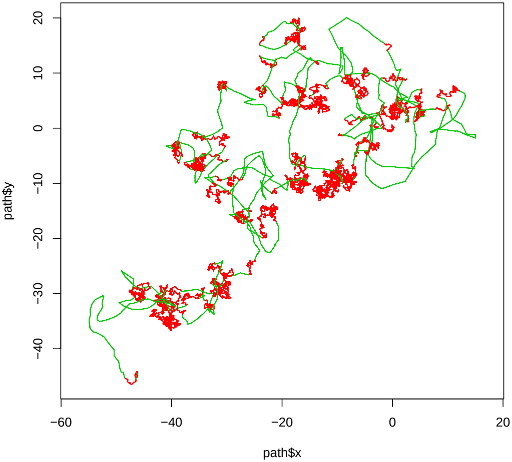
- Different behaviour
- Link with different activities
- Link with different environmental condition



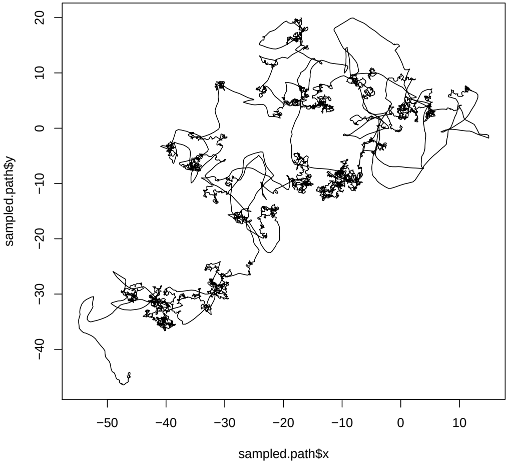
Detection of homogenous region in trajectories

Why ?

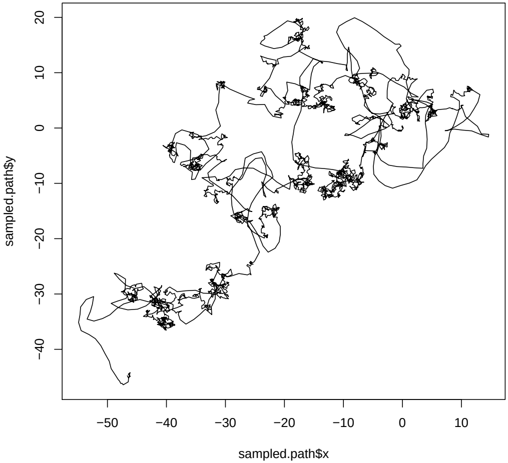
- Different behaviour
- Link with different activities
- Link with different environmental condition



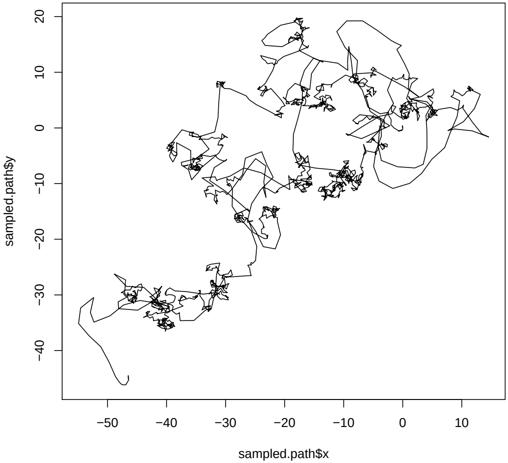
Effect of sampling step



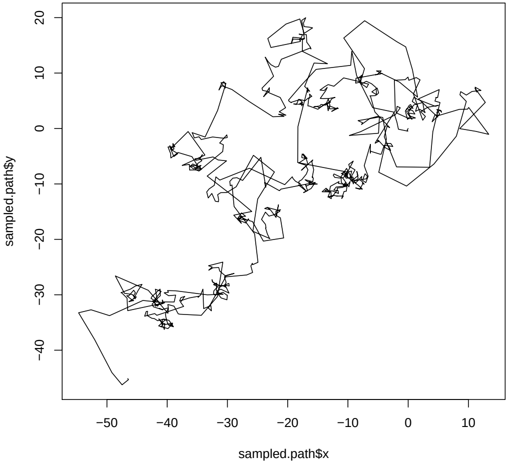
Effect of sampling step



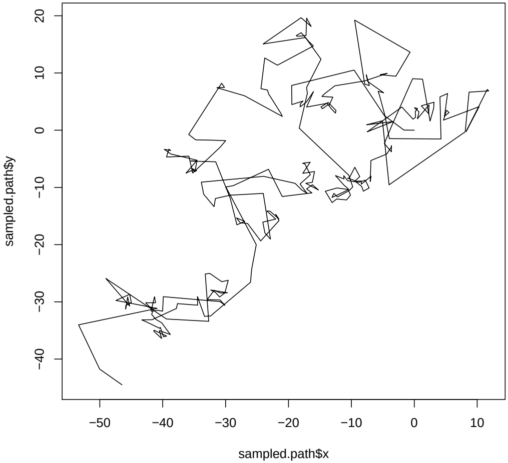
Effect of sampling step



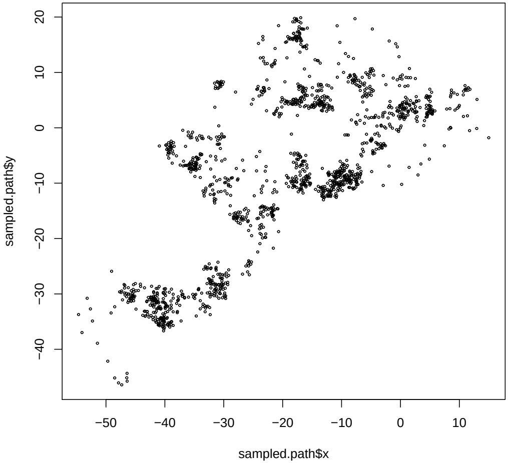
Effect of sampling step



Effect of sampling step



Effect of sampling step



Summarising trajectories

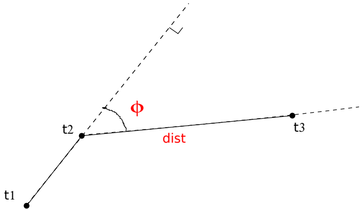
$(t_1, \ldots, t_N)$ denotes the time acquisition and $((x_1, y_1), \ldots, (x_N, y_N))$ the position at those times.

Trajetories as Turning angle and Speed

$$\mathbf{\Phi} = (\phi_2, \ldots, \phi_N)$$

$$\mathbf{S} = (S_2, \ldots, \S_N),$$

with $S_i = dist_i/(t_i - t_{i-1})$



Summarising trajectories

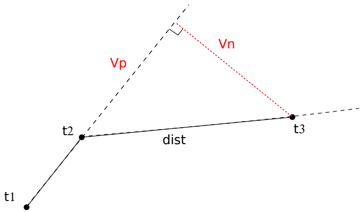
$(t_1, \ldots, t_N)$ denotes the time acquisition and $((x_1, y_1), \ldots, (x_N, y_N))$ the position at those times.

Trajetories as Persistent and Normal Velocity

$$\mathbf{V}^P = (V_2^P, \ldots, V_N^P)$$
$$\mathbf{V}^N = (V_2^N, \ldots, V_N^N)$$

with

$$V_i^P = S_i \, cos(\phi_i)$$
$$V_i^N = S_i \, sin(\phi_i)$$



Trajectories data

How trajectories data migh be considered?

- A sequence of (time, position)
- Turning angle and speed sequences
- Persistent and Normal Velocity sequences

Trajectories data

How trajectories data migh be considered?

- A sequence of (time, position)
- Turning angle and speed sequences
- Persistent and Normal Velocity sequences

What is affected by sampling?

- A sequence of (time, position)
- Turning angle and speed sequences
- Persistent and Normal Velocity sequences

Trajectories data

How trajectories data migh be considered?

- A sequence of (time, position)
- Turning angle and speed sequences
- Persistent and Normal Velocity sequences

What is affected by sampling?

- A sequence of (time, position)
- Turning angle and speed sequences
- Persistent and Normal Velocity sequences

Model approach:

- Most methods don't consider the two phenomena : movement and sampling process.
- Results will be closely dependent of the sampling step.

Convention

Notation:

- $\mathbf{Y} = (Y_1, \ldots, Y_n) =$ observed data (typically Speed)
- $\mathbf{Z}$ unobserved data (typically State, for mixture and Hidden Markov model)
- $\boldsymbol{\theta}=$ the unknown parameters of $\mathbf{Y}$ and $\mathbf{Z}$.

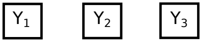
Convention

Notation:

- $\mathbf{Y} = (Y_1, \ldots, Y_n) =$ observed data (typically Speed)
- $\mathbf{Z}$ unobserved data (typically State, for mixture and Hidden Markov model)
- $\boldsymbol{\theta}=$ the unknown parameters of $\mathbf{Y}$ and $\mathbf{Z}$.

Graphical Representation (DAG):

Change point



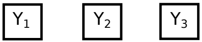
Convention

Notation:

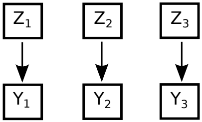
- $\mathbf{Y} = (Y_1, \ldots, Y_n)$ = observed data (typically Speed)
- $\mathbf{Z}$ unobserved data (typically State, for mixture and Hidden Markov model)
- $\boldsymbol{\theta}$= the unknown parameters of $\mathbf{Y}$ and $\mathbf{Z}$.

Graphical Representation (DAG):

Change point



Mixture

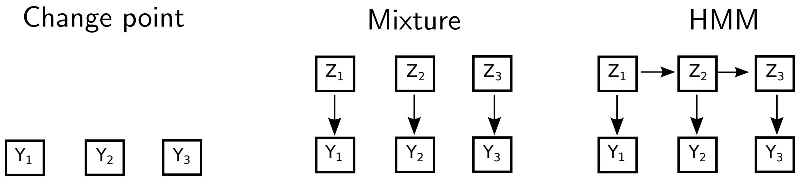


Convention

Notation:

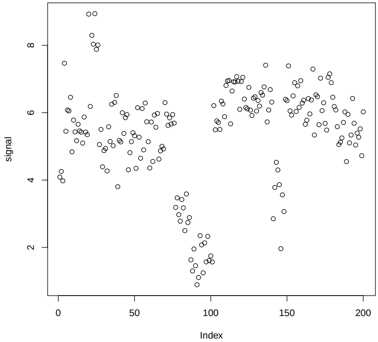
- $\mathbf{Y} = (Y_1, \dots, Y_n) =$ observed data (typically Speed)
- $\mathbf{Z}$ unobserved data (typically State, for mixture and Hidden Markov model)
- $\boldsymbol{\theta}=$ the unknown parameters of $\mathbf{Y}$ and $\mathbf{Z}$.

Graphical Representation (DAG):



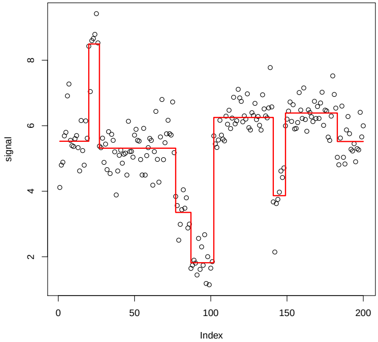
Plan

Change point detection context



Goal : Identifying homogenous region and abrupt changes in the signal.

Change point detection context



These regions may be interpreted afterwards.

Change point detection context

Two possible statistical point of view :

- Bayesian approach : the output is the posterior probability for each point to be a change point.
- *Frequentist* approach : the output is the best segmentation (according to a given critrion)

Underlying model

General framework :

- Data $Y_1, \ldots, Y_n$ are drawn from a given pdf, driven by unknown parameter θ
$$Y_t \sim f_\theta(.)$$
- θ values change at $K-1$ unknow instants, the change point : $t_1, \ldots, t_{K-1}$:
$$Y_t \sim f(\theta_k) \text{ if } t \text{ in region } I_k = [t_{k-1}+1, t_k]$$

Remark : $K-1$ change points $\Leftrightarrow$ K regions.

Underlying model

Change point detection in the trend :

- Data $Y_1, \ldots, Y_n$ are drawn from a given pdf, driven by unknown parameter θ
$$Y_t \sim f_\theta(.)$$
- θ values change at $K-1$ unknow instants, the change point : $t_1, \ldots, t_{K-1}$:
$$Y_t = \mu_k + E_t \quad \{E_t\}\text{i.i.d.} \sim \mathcal{N}(0, \sigma^2) \text{ if } t \text{ in portion } I_k,$$
for $k = 1, \ldots, K$.

Remark : $K-1$ change points $\Leftrightarrow$ K regions.

Estimation procedure

- Unknown parameters : $\mu_1, \ldots, \mu_K$, σ, and $T_1, \ldots, T_K$, but also K itself.

Estimation procedure

- Unknown parameters : $\mu_1, \ldots, \mu_K$, σ, and $T_1, \ldots, T_K$, but also K itself.
- Estimation Procedure
  - for a given K, $\mu_1, \ldots, \mu_K$, $\sigma_1, \ldots, \sigma_K$, and $T_1, \ldots, T_K$ are estimated using maximum likelihood (and dynanic programming)
  - K is chosen using penalized likelihood approach

Estimation procedure

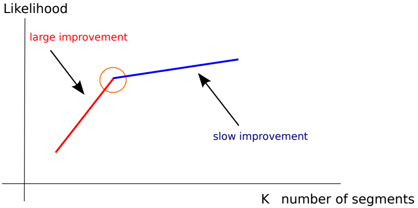
- Unknown parameters : $\mu_1, \ldots, \mu_K$, σ, and $T_1, \ldots, T_K$, but also K itself.
- Estimation Procedure
  - for a given K, $\mu_1, \ldots, \mu_K$, $\sigma_1, \ldots, \sigma_K$, and $T_1, \ldots, T_K$ are estimated using maximum likelihood (and dynanic programming)
  - K is chosen using penalized likelihood approach

Likelihood increases with the number of segment, but

Estimation procedure

- Unknown parameters : $\mu_1, \dots, \mu_K$, σ, and $T_1, \dots, T_K$, but also K itself.
- Estimation Procedure
  - for a given K, $\mu_1, \dots, \mu_K$, $\sigma_1, \dots, \sigma_K$, and $T_1, \dots, T_K$ are estimated using maximum likelihood (and dynanic programming)
  - K is chosen using penalized likelihood approach

Likelihood increases with the number of segment, but



Estimation procedure

Likelihood

$$
\begin{aligned}
2log(P_K(T,\theta)) &= 2\sum_{k=1}^{K} \log f(\{Y_t\}_{t\in I_k};\theta_k) = 2\sum_{k=1}^{K}\sum_{t\in I_k} \log f(Y_t;\theta_k) \\
&= -n\log\sigma^2 - \frac{1}{\sigma^2}\sum_{k=1}^{K}\sum_{t\in I_k}(Y_t-\mu_k)^2 + \text{cst.}
\end{aligned}
$$

Estimations

$$
(\hat{T},\hat{\theta}) = \underset{(T,\theta)}{\operatorname{argmax}}\, log(P_K(T,\theta))
$$

If the change points are known

$$
\widehat{\mu}_k = \frac{1}{n_k}\sum_{t\in I_k} Y_t \qquad \widehat{\sigma}^2 = \frac{1}{n}\sum_{k=1}^{K}\sum_{t\in I_k}(Y_t-\widehat{\mu}_k)^2
$$

Finding the K-1 change points

Considering all possible segmentations, the best segmentation minimizes

$$J_k(1, n) = \sum_{k=1}^{K} \sum_{t \in I_k} (Y_t - \widehat{\mu}_k)^2.$$

Finding the K-1 change points

Considering all possible segmentations, the best segmentation minimizes

$$J_k(1, n) = \sum_{k=1}^{K} \sum_{t \in I_k} (Y_t - \widehat{\mu}_k)^2.$$

But,

$$\binom{n-1}{K-1}$$

possible choices for the $K-1$ positions :

$\Rightarrow$ Practically impossible even for small K and n

Finding the K-1 change points

Considering all possible segmentations, the best segmentation minimizes

$$J_k(1,n) = \sum_{k=1}^{K} \sum_{t \in I_k} (Y_t - \widehat{\mu}_k)^2.$$

But,

$$\binom{n-1}{K-1}$$

possible choices for the $K-1$ positions :

$\Rightarrow$ Practically impossible even for small K and n

$K = 10,\ n = 200,\ \binom{K-1}{n-1} \approx 2.10^{16}$

Dynamic programming, with complexity $(\mathcal{O}(n^2))$.

Finding the K-1 change points

Dynamic programming, with complexity ($\mathcal{O}(n^2)$).

Sub-paths of the optimal path are themselves optimal,

Bellmann optimality

Finding the K-1 change points

Dynamic programming, with complexity ($\mathcal{O}(n^2)$).

Initialisation: Compute for $0 \leq i < j \leq n$, cost of portion I_{ij} :

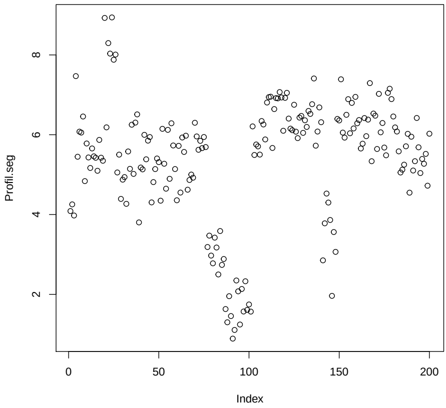
$$J_1(i,j) = \sum_{t=i+1}^{j} (Y_t - \widehat{\mu})^2$$

Etape k: Compute for $2 \leq k \leq K$, $J_k(i,j)$ the cost of the best segmentation in k segments between i and j.

$$J_k(i,j) = \min_{i<h<j} \left[J_{k-1}(i,h) + J_1(h+1,j) \right].$$

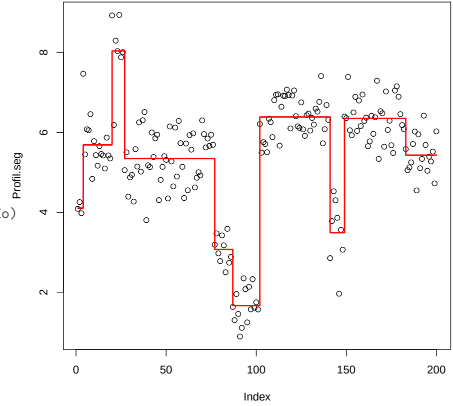
How to perform this segmentation approach ?

```
load("../Data/dataSegmentation.Rd")
summary(Profil.seg)


   Min. 1st Qu.
 0.8904  4.8720
 Median    Mean
 5.6990  5.3660
3rd Qu.    Max.
 6.3070  8.9400


plot(Profil.seg)
```



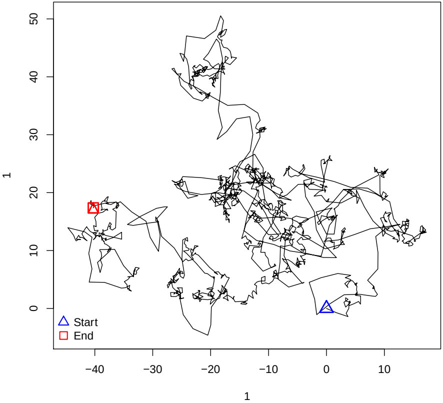
How to perform this segmentation approach ?

```r
library('cghseg')
## format data into CGHdata
signalCGH    <- new("CGHdata",Y=Profil.seg)
CGHo         <- new("CGHoptions")
calling(CGHo)<- FALSE ## no classification

segSignal   <- uniseg(.Object=signalCGH,CGHo=CGHo)
segSignalProf <- getsegprofiles(segSignal)
plot(Profil.seg)
lines(1:length(segSignalProf),
      segSignalProf, type="s", col=2, lwd=2)
```



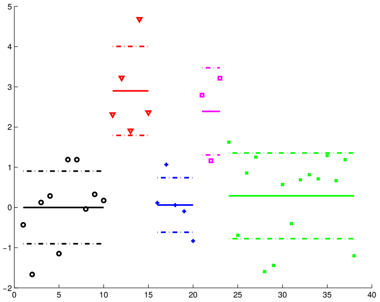
Do it yourself

```
load(file="../Data/trajEx.Rd")
plot(traj.ex, addpoints = F)
legend("bottomleft",pch=c(2, 0), col=c(4,2),
       legend=c("Start", "End"), bty = "n",
       pt.lwd = c(1.5,1.5), pt.cex = c(1.5,1.5))
```

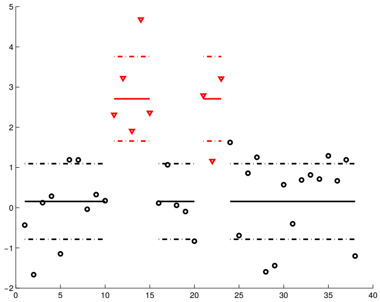


When Segmentation is not sufficient

Pure segmentation



Segmentation + classification



Segmentation-Clustering

- Assuming a secondary underlying structure of the segments into P groups with weights $\pi_1, ..., \pi_P (\sum_p \pi_p = 1)$.
- Let's define hidden variables Z_{kp}, indicators of the group to which segment k belongs.
- π_p denotes the proportion of group p.
- The distribution of the signal given the group of the segment is

$$t \in I_k, k \in p \qquad \Rightarrow \qquad Y_t \sim \mathcal{N}(m_p, \sigma^2)$$
$$Y^k | Z_{kp} = 1 \sim \mathcal{N}(m_p, \sigma^2).$$

- The parameters of this model are

$$\begin{aligned} \text{the breakpoint positions:} \quad T &= (t_1, ..., t_{K-1}), \\ \text{the mixture characteristics:} \quad \Theta &= (\pi_1, \ldots, \pi_P; \mu_1, \ldots, \mu_P, \sigma). \end{aligned}$$

Hybrid algorithm

2 levels of statistical units

- The inference of the breakpoints T is made at the position level t;
- The inference of the groups (status) (Θ, τ_{kp}) is made at the segment level k.

Hybrid algorithm

Alternate parameters estimation with K and P known

1. When T is fixed, the Expectation-Maximisation (EM) algorithm estimates Θ;

$$\hat{\Theta}^{(h+1)} = \arg\max_{\Theta} \left\{ \log \mathcal{L}_{KP} \left(\Theta, T^{(h)} \right) \right\}.$$

$$\log \mathcal{L}_{KP}(\hat{\Theta}^{(h+1)}; \hat{T}^{(h)}) \geq \log \mathcal{L}_{KP}(\hat{\Theta}^{(h)}; \hat{T}^{(h)})$$

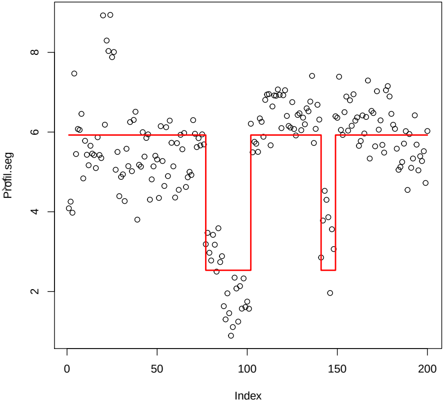
2. When Θ is fixed, dynamic programming estimates T;

$$\hat{T}^{(h+1)} = \arg\max_{T} \left\{ \log \mathcal{L}_{KP} \left(\hat{\Theta}^{(h+1)}, T \right) \right\}.$$

$$\log \mathcal{L}_{KP}(\hat{\Theta}^{(h+1)}; \hat{T}^{(h+1)}) \geq \log \mathcal{L}_{KP}(\hat{\Theta}^{(h+1)}; \hat{T}^{(h)})$$

How to perform this segmentation/clustering approach ?

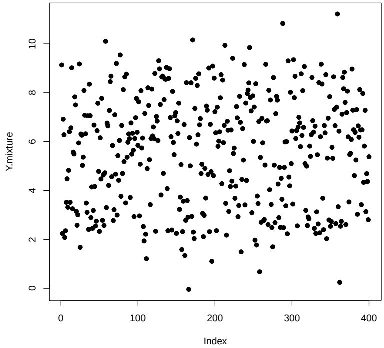
```r
## format data into CGHdata
calling(CGHo)<- TRUE ## no classification
CGHo@nblevels=2
segSignal   <- uniseg(.Object=signalCGH,CGHo=CGHo)
segSignalProf <- getsegprofiles(segSignal)
plot(Profil.seg)
lines(1:length(segSignalProf),
      segSignalProf, type="s", col=2, lwd=2)
```



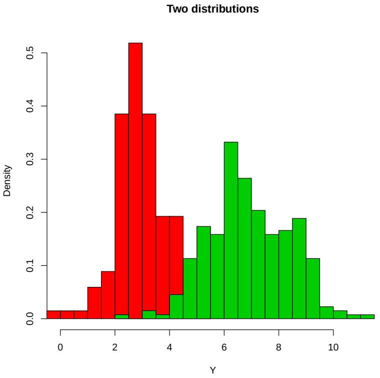
Do it yourself

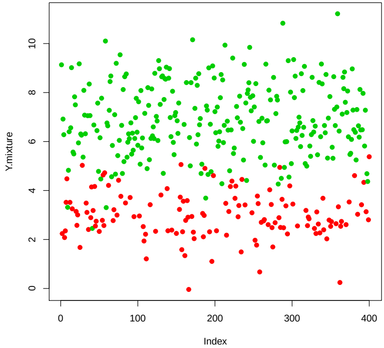
Plan

Problem presentation



Problem presentation





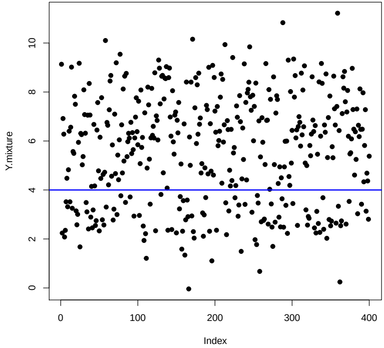
Problem presentation

Basic idea :

"Expert" threshold s

$$State_i = 1 \quad \text{if} Y_i < s$$

$$State_i = 2 \quad \text{if} Y_i \geq s$$



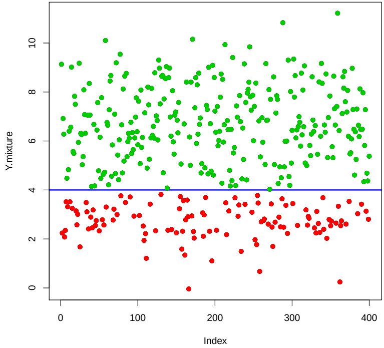
Problem presentation

Basic idea :

"Expert" threshold s

$$State_i = 1 \quad \text{if} Y_i < s$$

$$State_i = 2 \quad \text{if} Y_i \geq s$$



Problem presentation

Basic idea :

"Expert" threshold s

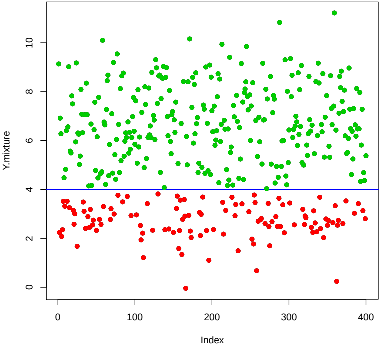
$$State_i = 1 \quad \text{if} Y_i < s$$

$$State_i = 2 \quad \text{if} Y_i \geq s$$

Improvement:

Estimating the threshold s and reconstruction of the hidden state (colour)

Compute the probability to belong to State 1 or 2.



Problem presentation

Basic idea :

"Expert" threshold s

$$State_i = 1 \quad \text{if} Y_i < s$$

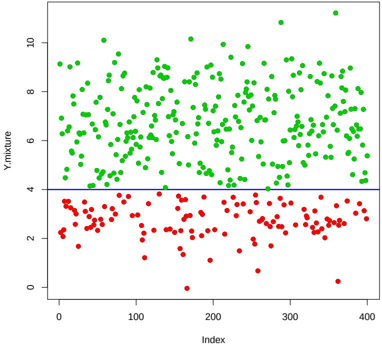
$$State_i = 2 \quad \text{if} Y_i \geq s$$

Improvement:

Estimating the threshold s and reconstruction of the hidden state (colour)

Compute the probability to belong to State 1 or 2.

$\Longrightarrow$ Mixture Model



Proposed model

Model For a given number of states K,

- Modelling Z: $\pi_k = \mathbb{P}(Z_i = k), \quad k = 1, \ldots, K, \quad \sum_k \pi_k = 1$
  $Z_i \overset{i.i.d}{\sim} \mathcal{M}(1, \boldsymbol{\pi}), \quad P(Z_{ik} = 1) = \pi_k$
- Modelling Y: The $Y_i' s$ are assumed to be independent conditionnaly to $\mathbf{Z}$: $(Y_i | Z_i = k) \overset{i.i.d}{\sim} f_{\theta_k}()$.

Proposed model

Model For a given number of states K,

- Modelling Z: $\pi_k = \mathbb{P}(Z_i = k), \quad k = 1, \dots, K, \quad \sum_k \pi_k = 1$
  $Z_i \overset{i.i.d}{\sim} \mathcal{M}(1, \boldsymbol{\pi}), \quad P(Z_{ik} = 1) = \pi_k$
- Modelling Y: The $Y_i's$ are assumed to be independent conditionnaly to $\mathbf{Z}$: $(Y_i | Z_i = k) \overset{i.i.d}{\sim} f_{\theta_k}()$.

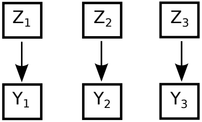
Model parameters are $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$

Proposed model

Model For a given number of states K,

- Modelling Z: $\pi_k = \mathbb{P}(Z_i = k), \quad k = 1, \ldots, K, \quad \sum_k \pi_k = 1$
  $Z_i \overset{i.i.d}{\sim} \mathcal{M}(1, \boldsymbol{\pi}), \quad P(Z_{ik} = 1) = \pi_k$
- Modelling Y: The $Y_i's$ are assumed to be independent conditionnaly to $\mathbf{Z}$: $(Y_i | Z_i = k) \overset{i.i.d}{\sim} f_{\theta_k}()$.

Model parameters are $\boldsymbol{\pi}$ and $\boldsymbol{\theta}$



Proposed model

Model For a given number of states K,

- Modelling Z: $\pi_k = \mathbb{P}(Z_i = k), \quad k = 1, \ldots, K, \quad \sum_k \pi_k = 1$
  $Z_i \overset{i.i.d}{\sim} \mathcal{M}(1, \boldsymbol{\pi})$
- Modelling Y: The $Y_i's$ are assumed to be independent conditionnaly to $\mathbf{Z}$: $(Y_i|Z_i = k) \overset{i.i.d}{\sim} f_{\theta_k}()$.

```
K <- 2; N <- 400; mu <- c(3, 7); sigma <- c(1,1.5)
Z <- sample(1:2, size = N, replace=T, prob=c(0.3, 0.7))
plot(Z, col=Z+1, pch=15)
Y.mixture <- rnorm(N, mean=mu[Z], sd=sigma[Z])
plot(Y.mixture, col=Z+1, pch=19)
```

Proposed model

Model For a given number of states K,

- Modelling Z: $\pi_k = \mathbb{P}(Z_i = k), \quad k = 1, \dots, K, \quad \sum_k \pi_k = 1$
  $Z_i \overset{i.i.d}{\sim} \mathcal{M}(1, \boldsymbol{\pi})$
- Modelling Y: The $Y_i's$ are assumed to be independent conditionnaly to $\mathbf{Z}$: $(Y_i | Z_i = k) \overset{i.i.d}{\sim} f_{\theta_k}()$.

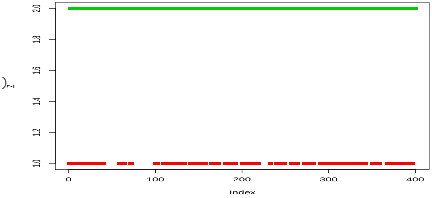
```
K <- 2; N <- 400; mu <- c(3, 7); sigma <- c(1,1.5)
Z <- sample(1:2, size = N, replace=T, prob=c(0.3, 0.7))
plot(Z, col=Z+1, pch=15)
Y.mixture <- rnorm(N, mean=mu[Z], sd=sigma[Z])
plot(Y.mixture, col=Z+1, pch=19)
```

Proposed model

Model For a given number of states K,

- Modelling Z: $\pi_k = \mathbb{P}(Z_i = k), \quad k = 1, \dots, K, \quad \sum_k \pi_k = 1$
  $Z_i \overset{i.i.d}{\sim} \mathcal{M}(1, \boldsymbol{\pi})$
- Modelling Y: The $Y_i's$ are assumed to be independent conditionnaly to $\mathbf{Z}$: $(Y_i | Z_i = k) \overset{i.i.d}{\sim} f_{\theta_k}()$.

```r
K <- 2; N <- 400; mu <- c(3, 7); sigma <- c(1,1.5)
Z <- sample(1:2, size = N, replace=T, prob=c(0.3, 0.7))
plot(Z, col=Z+1, pch=15)
Y.mixture <- rnorm(N, mean=mu[Z], sd=sigma[Z])
plot(Y.mixture, col=Z+1, pch=19)
```

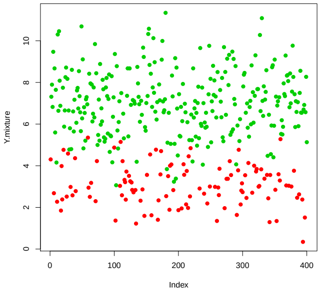


Proposed model

Model For a given number of states K,

- Modelling Z: $\pi_k = \mathbb{P}(Z_i = k), \quad k = 1, \dots, K, \quad \sum_k \pi_k = 1$
  $Z_i \overset{i.i.d}{\sim} \mathcal{M}(1, \boldsymbol{\pi})$
- Modelling Y: The $Y_i's$ are assumed to be independent conditionnaly to $\mathbf{Z}$: $(Y_i | Z_i = k) \overset{i.i.d}{\sim} f_{\theta_k}()$.

```
K <- 2; N <- 400; mu <- c(3, 7); sigma <- c(1,1.5)
Z <- sample(1:2, size = N, replace=T, prob=c(0.3, 0.7))
plot(Z, col=Z+1, pch=15)
Y.mixture <- rnorm(N, mean=mu[Z], sd=sigma[Z])
plot(Y.mixture, col=Z+1, pch=19)
```



Model Properties

- Couples $\{(Y_i, Z_i)\}$ are i.i.d.
- Label switching:
  the model is invariant for any permutation of the labels $\{1, \ldots, K\} \Rightarrow$ the mixture model has $K!$ equivalent definitions.
- Distribution of a Y_i:

$$P(Y_i) = \sum_{k=1}^{K} P(Y_i, Z_i = k) = P(Z_i = k)P(Y_i | Z_i = k)$$

- Distribution of $\mathbf{Y}$:

$$\begin{aligned} P(\mathbf{Y}; \boldsymbol{\theta}, \boldsymbol{\pi}) = \prod_{i=1}^{n} \sum_{k=1}^{K} P(Y_i, Z_i = k) &= \prod_{i=1}^{n} \sum_{k=1}^{K} P(Z_i = k)P(Y_i | Z_i = k) \\ &= \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k f_{\theta_k}(Y_i) \end{aligned}$$

Statistical inference of incomplete data models

Maximum likelihood estimate: We are looking for

$$(\widehat{\boldsymbol{\theta}}, \widehat{\boldsymbol{\pi}}) = \arg\max_{\boldsymbol{\theta}, \boldsymbol{\pi}} \log P(\mathbf{Y}; \boldsymbol{\theta}, \boldsymbol{\pi})$$

- Likelihood of the observed data (or observed likelihood):

$$\log P(\mathbf{Y}; \boldsymbol{\theta}, \boldsymbol{\pi}) = \sum_{i=1}^{n} \log \left[\sum_{k=1}^{K} \pi_k f_{\theta_k}(Y_i) \right]$$

- No analytical estimators.
- It is not always possible since this sum typically involves K^n terms : $2^{100} \approx 10^{30}$, the computation will take 10^{10} years on a 2014 computer.
- Brute force algorithm is not the way

And what if **Z** were observed ?

The complete likelihood is

$$\begin{aligned}\log P(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta}, \boldsymbol{\pi}) &= \log P(\mathbf{Z}; \boldsymbol{\pi}) + \log P(\mathbf{Y}|\mathbf{Z}; \boldsymbol{\theta}) \\ &= \sum_i \sum_k Z_{ik} \log \pi_k + \sum_i \sum_k Z_{ik} \log f_{\theta_k}(Y_i) \\ &= \sum_i \sum_k Z_{ik} [\log \pi_k + \log f_{\theta_k}(Y_i)].\end{aligned}$$

Now, the sum contains nK (200 if $n = 100$ and $K = 2$) terms. It is much easier.

And what if **Z** were observed ?

The complete likelihood is

$$\begin{aligned}\log P(\mathbf{Y},\mathbf{Z};\boldsymbol{\theta},\boldsymbol{\pi}) &= \log P(\mathbf{Z};\boldsymbol{\pi}) + \log P(\mathbf{Y}|\mathbf{Z};\boldsymbol{\theta})\\ &= \sum_i \sum_k Z_{ik}\log \pi_k + \sum_i \sum_k Z_{ik}\log f_{\theta_k}(Y_i)\\ &= \sum_i \sum_k Z_{ik}[\log \pi_k + \log f_{\theta_k}(Y_i)].\end{aligned}$$

Now, the sum contains nK (200 if $n = 100$ and $K = 2$) terms. It is much easier.

Unfortunately **Z** are unknown.

And what if **Z** were observed ?

The complete likelihood is

$$\begin{aligned}\log P(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta}, \boldsymbol{\pi}) &= \log P(\mathbf{Z}; \boldsymbol{\pi}) + \log P(\mathbf{Y}|\mathbf{Z}; \boldsymbol{\theta}) \\ &= \sum_i \sum_k Z_{ik} \log \pi_k + \sum_i \sum_k Z_{ik} \log f_{\theta_k}(Y_i) \\ &= \sum_i \sum_k Z_{ik} [\log \pi_k + \log f_{\theta_k}(Y_i)].\end{aligned}$$

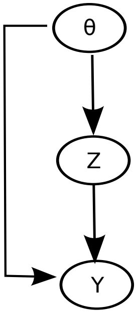
Now, the sum contains nK (200 if $n = 100$ and $K = 2$) terms. It is much easier.

Unfortunately **Z** are unknown.

Idea: Replace Z_i, by our best guess, that is :

$$\tau_{ik} := \mathbb{E}(Z_i = k|Y_i) = P(Z_i = k|Y_i)$$

More generally - EM algorithm



Bayes Formula

$$
\begin{aligned}
P(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta}) &= P(\mathbf{Y} | \mathbf{Z}; \boldsymbol{\theta}) P(\mathbf{Z}; \boldsymbol{\theta}), \\
&= P(\mathbf{Z} | \mathbf{Y}; \boldsymbol{\theta}) P(\mathbf{Y}; \boldsymbol{\theta}).
\end{aligned}
$$

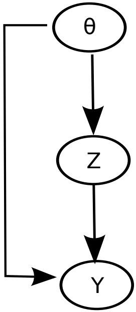
Therefore,

$$
\begin{aligned}
\log P(\mathbf{Y}; \boldsymbol{\theta}) &= \log \left\{ P(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta}) / P(\mathbf{Z} | \mathbf{Y}; \boldsymbol{\theta}) \right\} \\
&= \log P(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta}) - \log P(\mathbf{Z} | \mathbf{Y}; \boldsymbol{\theta})
\end{aligned}
$$

For a given $\boldsymbol{\theta}_0$, we may compute $P_{\boldsymbol{\theta}_0} = P(\mathbf{Z} | \boldsymbol{\theta}_0, \mathbf{Y})$ and

$$
\begin{aligned}
\log P(\mathbf{Y}; \boldsymbol{\theta}) &= \mathbb{E}_{\boldsymbol{\theta}_0}(\log P(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta})) - \mathbb{E}_{\boldsymbol{\theta}_0}(\log P(\mathbf{Z} | \mathbf{Y}; \boldsymbol{\theta})) \\
&= Q(\boldsymbol{\theta}, \boldsymbol{\theta}_0) - H(\boldsymbol{\theta}, \boldsymbol{\theta}_0)
\end{aligned}
$$

More generally - EM algorithm



Since

$$\log P(\mathbf{Y};\boldsymbol{\theta}) = Q(\boldsymbol{\theta},\boldsymbol{\theta}_0) - H(\boldsymbol{\theta},\boldsymbol{\theta}_0),$$

and $H(\boldsymbol{\theta},\boldsymbol{\theta}_0)$ achieves its maximum in $\boldsymbol{\theta}_0$,

$$\log P(\mathbf{Y};\boldsymbol{\theta}) - \log P(\mathbf{Y};\boldsymbol{\theta}_0) = (Q(\boldsymbol{\theta},\boldsymbol{\theta}_0) - Q(\boldsymbol{\theta},\boldsymbol{\theta}_0)) + (H(\boldsymbol{\theta}_0,\boldsymbol{\theta}_0) - H(\boldsymbol{\theta},\boldsymbol{\theta}_0)).$$

Expectation - Maximization algorithm

1. Phase E :
   Calculate $Q(\boldsymbol{\theta},\boldsymbol{\theta}^k)$ for every $\boldsymbol{\theta}$.
2. Phase M :
   Define $\boldsymbol{\theta}^{k+1} = argmax\, Q(\boldsymbol{\theta},\boldsymbol{\theta}^k)$

EM algorithm for independent mixture model

$$
\begin{aligned}
Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{(\ell)}) &= \mathbb{E}_{\boldsymbol{\theta}^{(\ell)}}(\log P(\mathbf{Y}, \mathbf{Z}; \boldsymbol{\theta})) \\
&= \mathbb{E}_{\boldsymbol{\theta}^{(\ell)}} \left\{ \sum_i \sum_k Z_{ik} [\log \pi_k + \log f_{\theta_k^{(\ell)}}(Y_i)] \right\} \\
&= \sum_i \sum_k \mathbb{E}_{\boldsymbol{\theta}^{(\ell)}}(Z_i = k | Y_i) \log \left[\pi_k f_{\theta_k^{(\ell)}}(Y_i) \right]
\end{aligned}
$$

Recall that $\tau_{ik}^{(\ell)} := P_{\boldsymbol{\theta}^{(\ell)}}(Z_i = k | Y_i)$

$$
Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(\ell)}) = \sum_i \sum_k \tau_{ik}^{(\ell)} \log \pi_k + \sum_i \sum_k \tau_{ik}^{(\ell)} \log f_{\theta_k^{(\ell)}}(Y_i)
$$

$\rightarrow$ Need to estimate $\tau_{ik}^{(\ell)}$

EM algorithm for independent mixture model

- Initialisation of $\boldsymbol{\theta}^{(0)} = (\pi_1, ..., \pi_K, \theta_1, ..., \theta_K)^{(0)}$.

EM algorithm for independent mixture model

- Initialisation of $\boldsymbol{\theta}^{(0)} = (\pi_1, ..., \pi_K, \theta_1, ..., \theta_K)^{(0)}$.
- Alternate

E-step Calculation of

$$\tau_{ik}^{(\ell)} = P(Z_i = k | y_i, \boldsymbol{\theta}^{(\ell-1)}) = \frac{\pi_k^{(\ell-1)} f_{\theta_k^{(\ell-1)}}(y_i)}{\sum_{k'} \pi_{k'}^{(\ell-1)} f_{\theta_{k'}^{(\ell-1)}}(x_i)}$$

M-step Maximization of

$$(\boldsymbol{\pi}, \gamma) \longmapsto \sum_i \sum_k \tau_{ik}^{(\ell)} [\log \pi_k + \log f(x_i; \gamma_k)]$$

In our situation

- $Z \in \{1,2\}$: $P(Z=1) = \pi_1$ and $P(Z=2) = 1 - \pi_1$
- For $k = 1$ or 2, $(X|Z=k) \sim \mathcal{N}(\mu_k, \sigma_k^2)$
- the parameter vector is $(\pi, \mu_1, \mu_2, \sigma_2^2, \sigma_2^2)$
  Assume that n observations $y_1, y_2, ..., y_n$ are available
- The parameter estimators at step $(\ell+1)$ of the EM algorithm are given by:

$$
\begin{aligned}
\widehat{\pi}_1^{(\ell+1)} &= \frac{1}{n}\sum_{i=1}^{n} \tau_{i1}^{(\ell)}, \\
\widehat{\mu}_k^{(\ell+1)} &= \frac{1}{\sum_{i=1}^{n} \tau_{ik}^{(\ell)}} \sum_{i=1}^{n} \tau_{ik}^{(\ell)} y_i \\
\widehat{\sigma}^2_{k^{(\ell+1)}} &= \frac{1}{\sum_{i=1}^{n} \tau_{ik}^{(\ell)}} \sum_{i=1}^{n} \tau_{ik}^{(\ell)} (y_i - \widehat{\mu}_k^{(\ell)})^2
\end{aligned}
$$

$\rightarrow$ They are a weighted version of the usual maximum likelihood estimates.

Back on earth - Practically speaking

```
#library('mclust')
library('mixtools')
Y.clustering <- normalmixEM (Y.mixture, lambda = NULL, mu = NULL, sigma = NULL, k = 2,
                             mean.constr = NULL, sd.constr = NULL,
                             epsilon = 1e-07, maxit = 1000, maxrestarts=20)


number of iterations= 62


summary(Y.clustering)


summary of normalmixEM object:
         comp 1
lambda 0.242871
mu     2.908037
sigma  0.896202
         comp 2
lambda 0.757129
mu     6.893074
sigma  1.617269
loglik at estimate:  -870.9465


Y.clustering$posterior


             comp.1
  [1,] 3.825020e-01
  [2,] 3.397032e-06
```

Plan

Markov chain model

Modelling dependence between time step : If an animal is feeding at time i, he has more chance to be feeding at time $i+1$ than if he was travelling at time i.

$$P(Z_{i+1} = 1 | Z_i = 1) \neq P(Z_{i+1} = 1 | Z_i = 2)$$

Markov Chain definition **Z** is a Markov chain if

$$P(Z_{i+1} | Z_{1:i}) = P(Z_{i+1} | Z_i)$$

Z is completely defined by the distribution $\mu_1 = P(Z_1)$ and the transition matrix

$$\Pi = \begin{bmatrix} \pi_{11} & 1-\pi_{11} \\ 1-\pi_{22} & 1-\pi_{22} \end{bmatrix}$$

Markov chain simulation

```r
#
N <- 200
pi11 <- 0.6
pi22 <- 0.7
## initial distribution
mu1 <- c(0.5, 0.5)
##transition matrix
PI <- matrix(c(pi11, 1-pi11, 1-pi22, pi22), ncol=2, byrow = T)

##initialisation of Z
Z <- rep(NA, N)
Z[1] <- sample(1:2, size=1, prob = mu1)
for( i in 1:(N-1))
{
  Z[i+1] <- sample(1:2, size=1, prob = PI[Z[i],])
}
plot(1:N, Z, "s")
points(1:N, Z, col=Z+1, pch=19)
```

Hidden Markov Chain model

Hidden State Modelling: $\mathbf{Z}$ is assumed to follow a Markov Chain model with unknown initial distribution $\boldsymbol{\mu}$ and transition matrix $\mathbf{\Pi}$.

Plan

Plan

Circular Distribution

If $Z \sim WC(\mu, \gamma)$,

$$f_{WC}(\theta; \mu, \gamma) = \sum_{n=-\infty}^{\infty} \frac{\gamma}{\pi(\gamma^2 + (\theta - \mu + 2\pi n)^2)}$$