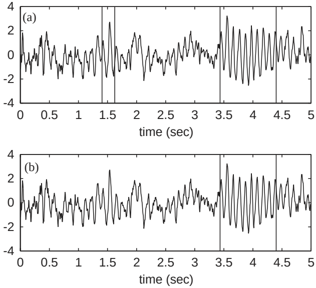
Segmentation models and applications with R

Franck Picard*

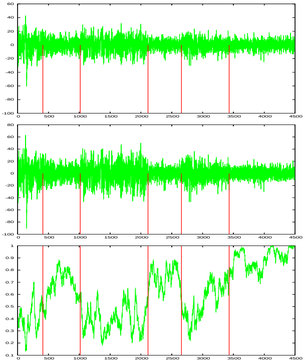
*UMR 5558 UCB CNRS LBBE, Lyon, France
franck.picard@univ-lyon1.fr
http://pbil.univ-lyon1.fr/members/fpicard/

INED-28/04/11

Segmentation is everywhere !

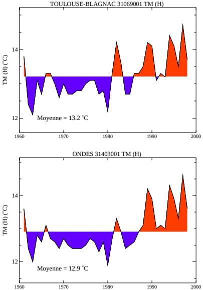


EEG segmentation [2]

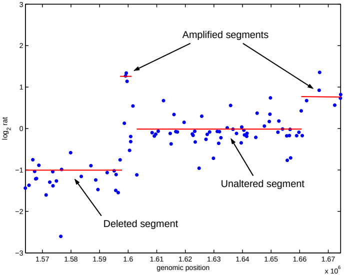


Market prices segmentation [3]

Segmentation is everywhere !



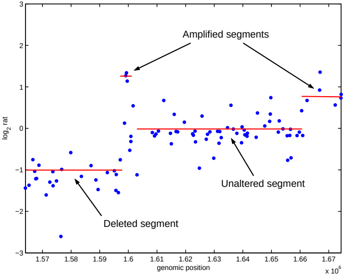
Climate series segmentation [5]



Array CGH segmentation [8]

Segmentation to detect copy number variations

- Comparative Genomic Hybridization is used to measure gene copy number variations between genomes.
- The number of genes is measured by fluorescence at given positions
- The logratio of signals shows jumps and segments
- Detect segments that correspond to regions that share the same copy number on average



Baseline at 0 for no difference

Outline of the presentation

- Explain the statistical developments associated with segmentation models
- Give an overview of the subject (with bibliography)
- Provide an R package dedicated to the analysis of CGH data by segmentation models
- Explain the choices relative to the construction of the package
- Introduce the generalization to multiple series segmentation

The cghseg package

- Idea: develop a package for segmentation in the context of CGH data analysis
- The community of Bioinformaticians uses `R` extensively
- The size of the data can be a problem (discussion)
- Use `S4` classes with 3 main classes:
  - `CGHData` (`CGHd`),
  - `CGHOptions` (`CGHo`),
  - `CGHResults` (`CGHr`).

The CGHData class

- Raw data are in the data.frame() format
- They are stored in a list() format in a CGHd object

```
> Y[1:5,1:5]
        Ind1       Ind2        Ind3        Ind4       Ind5
1  0.15218335  0.1741900  0.03386524  0.14293254  0.2639392
2  0.46361794 -0.6023429 -0.12644954 -0.27317036 -0.3458813
3  0.07078370  0.3880629  0.83691230  0.19800776  0.8538934
4  0.21176834  0.1623984  0.12279919 -0.39214814  0.1802575
5  0.35821410 -0.1347911 -0.11833753 -0.00863382 -0.4885733
>
> CGHd = new("CGHdata",Y)
> CGHd
****** CGHdata show ******
[CGHd show] Data are in the list format [[patient]]
[CGHd show] Data sample:
Y[[Ind1]]
[1] 0.1521834 0.4636179 0.0707837 0.2117683 0.3582141
Y[[Ind2]]
[1]  0.1741900 -0.6023429  0.3880629  0.1623984 -0.1347911
[CGHd show] probeID sample:
NULL
[CGHd show] genomic positions sample:
NULL
[CGHd show] GC content sample:
NULL
```

A piece-wise constant regression

- We observe a Gaussian process (iid) $\mathbf{Y} = \{Y_1, \ldots, Y_n\}$ with

$$Y_t \sim \mathcal{N}(\mu_t, \sigma^2).$$

- We suppose that there exists $K+1$ change-points $t_0 < \ldots < t_K$ such that the mean of the signal is constant between two changes and different from a change to another.
- $I_k =]t_{k-1}, t_k]$: interval of stationarity, μ_k the mean of the signal between two changes:

$$\forall t \in I_k, \;\; Y_t = \mu_k + E_t, \;\; E_t \sim \mathcal{N}(0, \sigma^2).$$

- In its generalization, the parameter subject to changes could be the variance, the spectrum...

Parameters and estimation strategy

- The parameters: $\mathbf{t} = \{t_0, \ldots, t_K\}$, $\boldsymbol{\mu} = \{\mu_1, \ldots, \mu_K\}$ and σ^2.
- The estimation is done for a given K which is estimated afterwards.
- The log-likelihood of the model is:

$$\log \mathcal{L}_K(\mathbf{Y}; \mathbf{t}, \boldsymbol{\mu}, \sigma^2) \;=\; \sum_{k=1}^{K} \sum_{t=t_{k-1}+1}^{t_k} f(y_t; \mu_k, \sigma^2).$$

- When K and $\mathbf{t}$ are known, how to estimate $\boldsymbol{\mu}$?
- When K is known, how to estimate $\mathbf{t}$?
- How to choose K ?

Parameter estimation

- When K and $\mathbf{t}$ are known the estimation of $\boldsymbol{\mu}$ is straightforward:

$$\begin{aligned}\widehat{\mu}_k &= \frac{1}{\widehat{t}_k - \widehat{t}_{k-1}} \sum_{t=\widehat{t}_{k-1}+1}^{\widehat{t}_k} y_t, \\ \widehat{\sigma}^2 &= \frac{1}{n} \sum_{k=1}^{K} \sum_{t=\widehat{t}_{k-1}+1}^{\widehat{t}_k} (y_t - \widehat{\mu}_k)^2.\end{aligned}$$

- Find $\widehat{\mathbf{t}}$ such that:

$$\widehat{\mathbf{t}} = \arg\max_{\mathbf{t}} \left\{ \log \mathcal{L}_K(\mathbf{Y}; \mathbf{t}, \boldsymbol{\mu}, \sigma^2) \right\}.$$

Dynamic Programming to optimize the log-likelihood

- Partition n data points into K segments: complexity $\mathcal{O}(n^K)$.
- DP reduces the complexity to $\mathcal{O}(n^2)$ when K is fixed.
- Analogy with the shortest path problem (Bellman principle)
- $\text{RSS}_k(i,j)$ cost of the path connecting i to j in k segments:

$$\begin{aligned}
\forall 0 \leq i < j \leq n, \;\; \text{RSS}_1(i,j) &= \sum_{t=i+1}^{j} (y_t - \bar{y}_{ij})^2, \\
\forall 1 \leq k \leq K-1, \;\; \text{RSS}_{k+1}(1,j) &= \min_{1 \leq h \leq j} \left\{ \text{RSS}_k(1,h) + \text{RSS}_1(h+1,j) \right\}.
\end{aligned}$$

Dynamic Programming on very large signals ?

- Even if DP reduces the computational burden to $\mathcal{O}(n^2)$ it may be problematic when $n \sim 10^6$
- Constraint the length of segments (lmin, lmax)
- Find a trick to the trick to decrease the complexity of DP [9]
- Use `C++` to externalize heavy computations

Model selection

- The number of segments K should be estimated:

$$\widehat{K} = \arg\max_{K} \left\{ \log \mathcal{L}_K(\mathbf{Y}; \widehat{\mathbf{t}}, \widehat{\boldsymbol{\mu}}, \widehat{\sigma}^2) - \beta \text{pen}(K) \right\}.$$

- Main difficulty: breakpoints are discrete parameters
  - the likelihood is not differentiable wrt $\mathbf{t}$
  - C_{n-1}^{K-1} possible segmentations for a model with K segments.
  - how to define the dimension of the model ?
- How to define pen(K),β ?
- modified BIC criterion [10], non asymptotic criterion [4], L-curve criterion [2].

uniseg() and the CGHResults class

- From a CGHd object and a CGHo object
- Use uniseg() such that CGHr = uniseg(CGHd,CGHo)
- uniseg() performs automatic model selection

```
> CGHr["loglik"]
$Ind1
 [1] -85.64 -50.72 -46.49 ...
$Ind2
 [1] -95.43 -58.53 -56.68 ...
> CGHr["mu"]
$Ind1
  begin end        mean
1     1  77 -0.03122034
2    78 100 -0.99103873
```

```
> CGHr["mu"]
$Ind2
  begin end        mean
1     1  43  0.02545556
2    44  56 -0.92030745
3    57 100 -0.17527263
...

> CGHr["from"]
[1] "uniseg"
```

Different functions to get many informations on the model

- Given the size of the data `CGHr` stores results in a sparse format
- Small functions are implemented to retrieve the desired information
- `bp = getbp(CGHr)` to retrieve breakpoints in a 0/1 format
- `seg = getsegprofiles(CGHr)` to retrieve predictions of the model

Joint segmentation of multiple profiles

- When analyzing multiple profiles (or *series*), one may want to perform a joint analysis [7, 1]
- $Y_i(t)$: the signal for individual $i = 1, ...I$ with segments $\{\mathcal{I}_k^i\}$

$$\forall t \in \mathcal{I}_k^i,\ Y_i(t) = \mu_{ik} + \varepsilon_i(t),\ \varepsilon_i(t) \sim \mathcal{N}(0, \sigma^2).$$

- $\boldsymbol{\mu}_i$ specific levels of segments
- $\mathbf{T}_i$ specific incidence matrix of the breaks

$$\mathbf{Y}_i = \mathbf{T}_i \boldsymbol{\mu}_i + \mathbf{E}_i$$

Power of the S4 classes

- We can still use the `CGHd` class for the data
- Use a new function adapted to the multi-series setting:

```
CGHr = multiseg(CGHd,CGHo)
```

- The format of the output is the same but the computational procedure is different
- `multiseg()` also uses C++ code to compute the breakpoint positions and the number of segments per series

Conclusions

- Segmentation models are used in many application fields
- Other packages exist like CBS [6] for sequential analysis
- Algorithmic considerations are central when using such models
- Developing a R package dedicated to segmentation requires the use of a more efficient language (like C++)
- The use of such strategy becomes a standard in computational biology (ultra-high dimensional)
- The submission to the CRAN is made more difficult by the different languages
- Check on http://pbil.univ-lyon1.fr/members/fpicard/ for more detailed presentations on the subject

F. Picard an E. Lebarbier, E. Budinska, and S. Robin.
Joint segmentation of multivariate gaussian processes using mixed linear models.
CSDA, 55(2), 2011.

M. Lavielle.
Using penalized contrasts for the change-point problem.
Signal Processing, 85(8):1501–1510, 2005.

M. Lavielle and Teyssière G.
Detection of multiple change-points in multiple time-series.
Lithuanian Mathematical Journal, 46(4):351–376, 2006.

E. Lebarbier.
Detecting multiple change-points in the mean of Gaussian process by model selection.
Signal Processing, 85:717–736, 2005.

O. Mestre.
Methodes statistiques pour l'homogeneisation de longues series climatiques.
PhD thesis, Université Paul Sabatier, Toulouse, 2000.

AB. Olshen, ES. Venkatraman, R. Lucito, and M. Wigler.
Circular binary segmentation for the analysis of array-based DNA copy number data.
Biostatistics, 5(4):557–572, 2004.

F. Picard, E. Lebarbier, M. Hoebeke, B. Thiam, and S. Robin.
Joint segmentation, calling, and normalization of multiple CGH profiles.
Biostatistics, 2011.

F. Picard, S. Robin, M. Lavielle, C. Vaisse, and J-J. Daudin.
A statistical approach for CGH microarray data analysis.
BMC Bioinformatics, 6:27, 2005.

Guillem Rigaill.
Pruned dynamic programming for optimal multiple change-point detection.

Technical report, arXiv:1004.0887v1, 2010.

NR. Zhang and DO. Siegmund.
A modified bayes information criterion with applications to the analysis of comparative genomic hybridization data.
Biometrics, 63(1):22–32, 2007.