Change-point detection with kernel methods

M. Pierre-Jean, G. Marot , G. Rigaill & A. Célisse

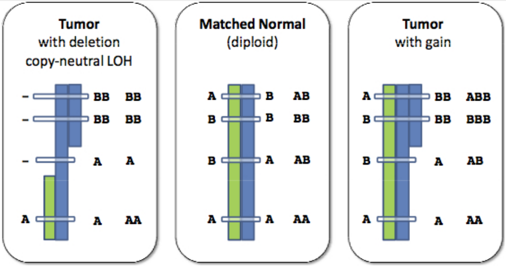
Laboratoire Statistique et Génome
Université d'Évry Val d'Éssonne
UMR CNRS 8071 USC INRA

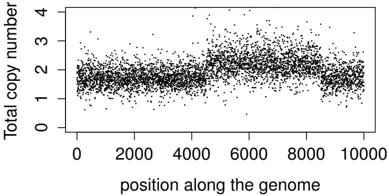
CERIM
Université Lille 2 Droit et Santé
Faculté de médecine

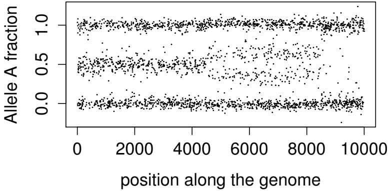
27 May 2013

Outline

Outline







$$N_{A,t} + N_{B,t}$$

$$\frac{N_{B,t}}{N_{A,t}+N_{B,t}}$$

Goal :

- Identification of change-points in the signal

Data characteristics

- High dimension $\rightarrow$ Complexity of n^K if K is the number of breakpoints
- High structuration of signal (piecewise constant , multimodality of allele B fraction)

Goal :

- Identification of change-points in the signal

Data characteristics

- High dimension $\rightarrow$ Complexity of n^K if K is the number of breakpoints
- High structuration of signal (piecewise constant , multimodality of allele B fraction)

State of the Art

- Dynamic programming to solve complexity problem
- Change-point detection in the mean and the variance
- Symetrisation of allele B fraction

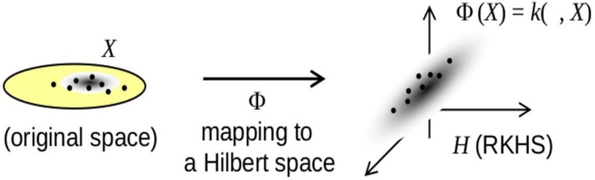
Proposition [Arlot et al. 2012] :
Change-point detection in the whole distribution of signal

$\rightarrow$ Using kernel methods

Outline

Modelisation

- $(X_1, \ldots X_n) \in \mathcal{X}$ initial data
- $k(.,.) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ reproducing kernel (RKHS : $\mathcal{H}$)
- $\phi(.) : \mathcal{X} \to \mathcal{H}$ tel que $\phi(x) = k(x,.)$: canonical feature map



Modelisation

Representation in RKHS

- $\forall 1 \leq i \leq n \quad Y_i = \phi(X_i) \in \mathcal{H}$

Mean element

- The mean element of $\mathbb{P}_{X_i}$ (distribution of X_i) : $\mu_i^\star$
  $\forall f \in \mathcal{H} \quad < \mu^\star, f >_{\mathcal{H}} = \mathbb{E}_{X_i}(< \phi(X_i), f >_{\mathcal{H}})$
- For characteristic kernels : $\mathbb{P}_{X_i} \neq \mathbb{P}_{X_j} \Rightarrow \mu_i^\star \neq \mu_j^\star$

Model in RKHS

- $\forall 1 \leq i \leq n \quad Y_i = \mu_i^\star + \epsilon_i$

$\Longrightarrow$ Change-point detection in the mean of Y_i

Change-point choice [Arlot et al. (2012)]

Goal : Estimate $\mu^\star$
Assumption : $\mu^\star = (\mu_1^\star, \ldots, \mu_n^\star) \in \mathcal{H}$ piecewise constant
Kernel used : Gaussian kernel : $k_\delta(x,y) = \exp\left(-\frac{\|x-y\|^2}{\delta}\right)$
Estimation of $\mu^\star$

- m a particular segmentation
- λ segment in m
- $\forall$ segment $\lambda \in m$, $\quad \hat{\mu}_m(\lambda) = \frac{1}{Card(\lambda)} \sum_{j\in\lambda} Y_j$

Best segmentation with fixed number of segment D

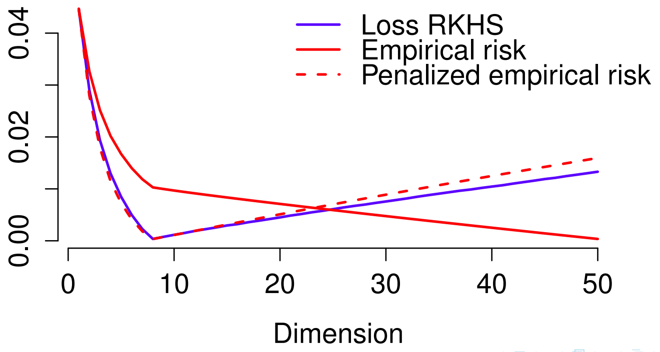
- $\hat{m}_D = \underset{m \in \mathcal{M}_n | D_m = D}{\arg\min} \|Y - \hat{\mu}_m\|^2_{\mathcal{H}_n}$

Best number of segment $\hat{D}$

- $\hat{D} = \underset{D \in \{1, \ldots D_{max}\}}{\arg\min} \|Y - \hat{\mu}_{\hat{m}_D}\|^2_{\mathcal{H}^n} + pen(D)$

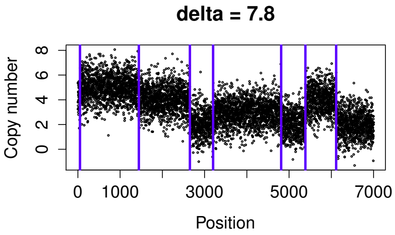
Illustration

Loss RKHS : $L(\hat{\mu}_m) = \|\hat{\mu}_m - \mu^\star\|^2_{\mathcal{H}^n}$
Empirical Risk RKHS : $R(\hat{\mu}_m) = \|Y - \hat{\mu}_m\|^2_{\mathcal{H}^n}$



Outline

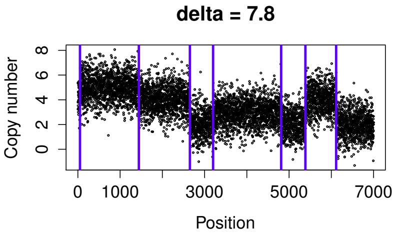
Influence of kernel in segmentation quality



- All change-points are recovered

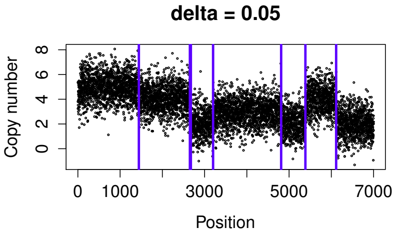
$$k_\delta(x, y) = \exp\left(\frac{\|x-y\|^2}{\delta}\right)$$

Influence of kernel in segmentation quality



- All change-points are recovered

$$k_\delta(x, y) = \exp\left(\frac{\|x-y\|^2}{\delta}\right)$$



- 2nd an 3rd change-points are very close
- One change point is missed

Choice of kernel

For each kernel : best segmentation $\hat{m}_{\hat{D}}(\delta)$
Comparison of segmentations ?

RKHS

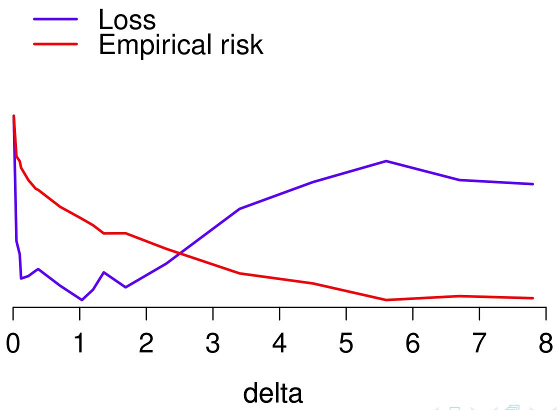
- Comparison of losses in RHKS
  $\rightarrow$ Problem : RKHS norms are not comparable

In initial space

- Estimation of loss by cross-validation (efficient)
- Comparison of quatities

Illustration

Loss in $\mathbb{R}$: $L(\hat{f}_m) = \|\hat{f}_m - f\|^2_{\mathbb{R}^n}$
Empirical Risk in $\mathbb{R}$: $R(\hat{f}_m) = \|Y - \hat{f}_m\|^2_{\mathbb{R}^n}$



Outline

- Results for other kernels
- Best kernel not only for a kernel type
- Combined total DNA copy number and allele B fraction :
  $k\left((x_1, y_1), (x_2, y_2)\right) = (1-\alpha)k_1(x_1, x_2) + \alpha k_2(y_1, y_2)$
- Other applications (audio, video)

Thanks for your attention