Mixtures and Hidden Markov Models for genomics data

M.-L. Martin-Magniette
marie_laure.martin@agroparistech.fr

Researcher at the French National Institut for Agricultural Research

Plant Science Institut of Paris-Saclay (IPS2)
Group leader of the team Genomic networks
Applied Mathematics and Informatics Unit at AgroParisTech
Member of the team Statistique and Genome

Research interest

Statistics

- Mixture models
- Hidden Markov Models
- Gaussian Graphical models and regularization methods for network inference

Applied projects in plant genomics for

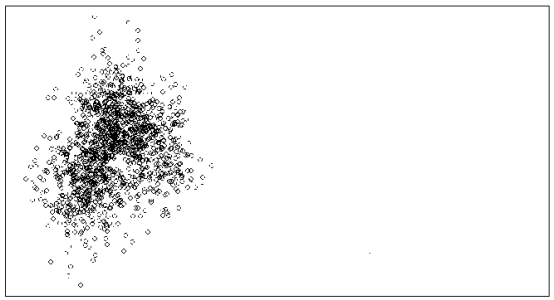
- Transcriptomic data (microarray and RNAseq data)
- Protein-DNA interaction study (ChIP-chip data)

Project of Genomic networks: Bioinformatics approach and statistical models

- for improving the functional and relational annotation of Arabidopsis thaliana
- based on global analysis of micraorray data, RNaseq data, study of the transcription factors ...

Introduction

Observations described by 2 variables
The objective is to model the distribution

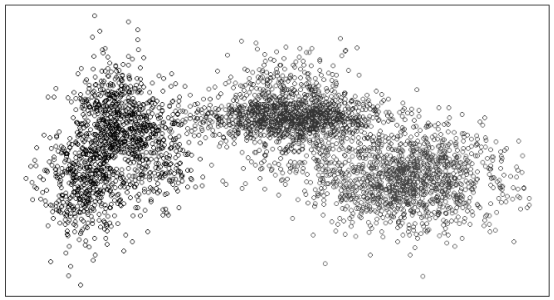


One Gaussian seems a good solution

This is an underlying structure observed through the data

Introduction

Observations described by 2 variables
The objective is to model the distribution



Data are scattered and subpopulations are observed

According to the experimental design, there exists no external information about them

This is an underlying structure observed through the data

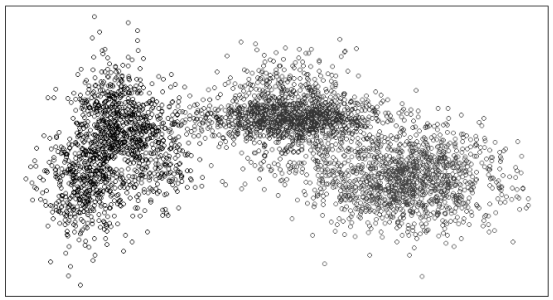
Mixture for identifying this underlying structure

Definition of a mixture model

It is a probabilistic model for representing the presence of subpopulations within an overall population.

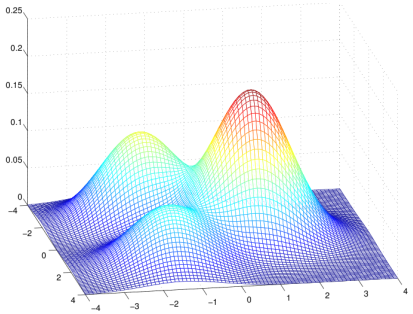
- Introduction of a latent variable Z indicating the subpopulation of each observation

what we observe

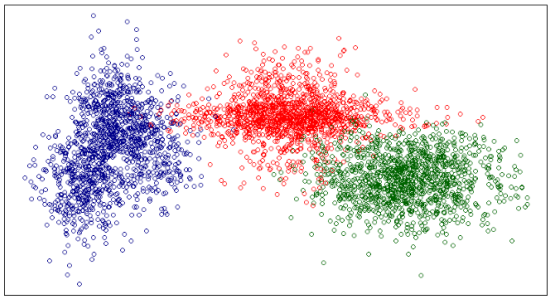


$Z = ?$

the model



the expected results

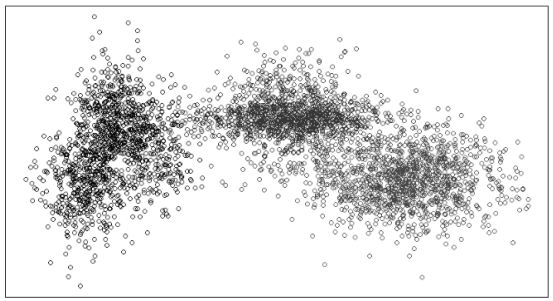


$Z : 1 = \bullet, 2 = \bullet, 3 = \bullet$

$\rightarrow$ It is an unsupervised classification method

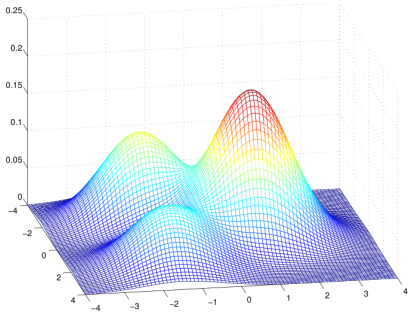
Key ingredients of a mixture model

what we observe

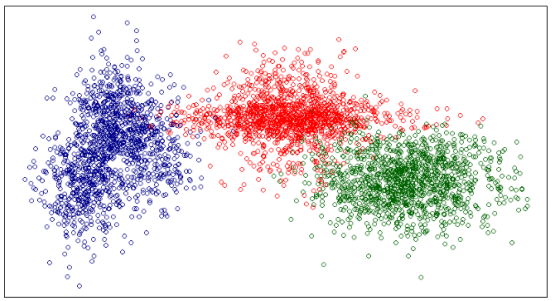


$Z = ?$

the model



the expected results



Z: 1 = ●, 2 = ●, 3 = ●

Let $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ denote n observations with $\mathbf{x}_i \in \mathbb{R}^Q$ and let $\mathbf{Z} = (Z_1, \ldots, Z_n)$ be the latent vector.

1) Distribution of Z: $\{Z_i\}$ are assumed to be independent and

$$P(Z_i = k) = \pi_k \quad \text{with} \quad \sum_{k=1}^{K} \pi_k = 1 \qquad \rightarrow \mathbf{Z} \sim \mathcal{M}(n; \pi_1, \ldots, \pi_K)$$

K is the number of components of the mixture

2) Distribution of $(\mathbf{X}_i | Z_i = k)$ is a parametric distribution $f(\bullet; \gamma_k)$

Questions around the mixtures

- Modeling: what distribution for each component ?
  $\rightsquigarrow$ it depends on observed data.

- Inference: how to estimate the parameters ?
  $\rightsquigarrow$ it is usually done with an iterative algorithm

- Model selection: how to choose the number of components ?
  - A collection of mixtures with **a varying number of components** is usually considered
  - A criterion is used to select the best model of the collection

Table of contents

Table of contents

Functional annotation is the new challenge

- It is now relatively easy to sequence an organism and to localize its genes
- But between 20% and 40% of the genes have an unknown function
- For *Arabidopsis thaliana*, 16% of the genes have a validated function and 20 % of the genes are orphean genes i.e. without any information on their function

$\rightarrow$ with the high-throughput technologies, it is now possible to improve the functional annotation

Co-expression analysis

- Co-expressed genes are good candidates to be involved in a same biological process (Eisen et al, 1998)
- Pearson correlation values are often used to measure the co-expression, but it is a local point of view
- Co-expression analysis can be recast as a research of an underlying structure in a whole dataset

Proposed model for a co-expression study

Denoting

- $\mathbf{X}_i = (X_{i1}, ..., X_{iQ})$: a random vector of Q expression differences for gene i
- Z_i represents the unknown cluster of co-expression

We assume that

- Z_i's are i.i.d

$$Z_i \sim \mathcal{M}(1; \boldsymbol{\pi}), \qquad \pi_k = \Pr\{Z_i = k\} = \Pr\{Z_{ik} = 1\}$$

- $\mathbf{X}_i$'s are independent conditionally to the Z_i's:

$$(\mathbf{X}_i | Z_i = k) \sim f(\cdot; \gamma_k), \qquad \text{e.g. } f(\cdot; \gamma_k) = \mathcal{N}(\boldsymbol{\mu}_k, \Sigma_k).$$

No information exists on the number of components K
We have to estimate

$$K \qquad \{\pi_k, \boldsymbol{\mu}_k, \Sigma_k\}_{k=1,...,K} \qquad \text{and} \qquad Pr\{Z_i = k | \mathbf{X}_i = \mathbf{x}_i\}$$

Results obtained with microarray data

- Gene expression measured with a two-color microarray
- Data are 45 expression differences characterizing the response of Arabidopsis thaliana when the plant is exposed to a stress
- Distribution modeled with a Gaussian mixture, number of component chosen with a model selection criterion

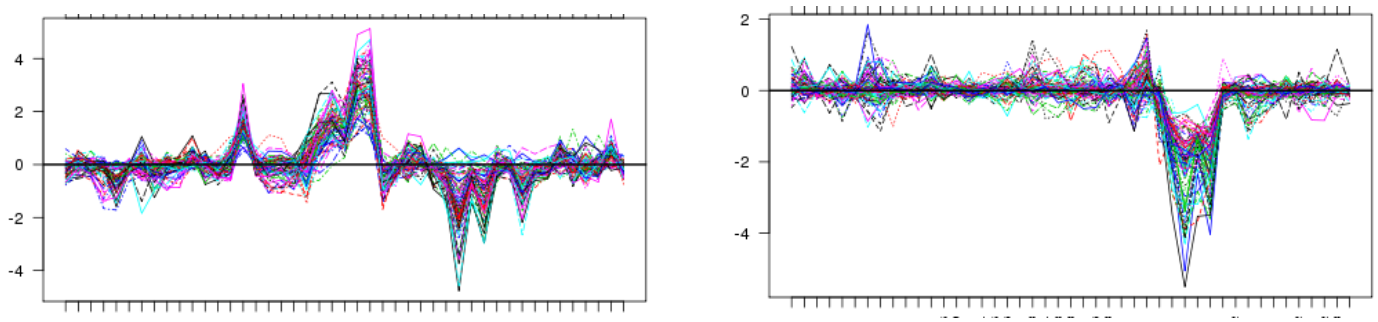


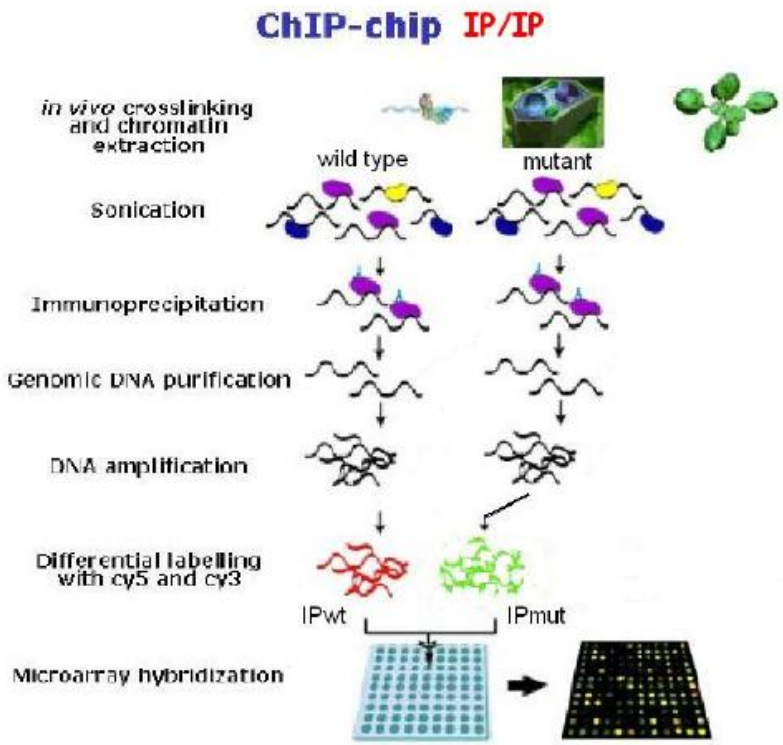
Table : Example of 2 clusters of genes

Table of contents

ChIP on chip

ChIP: Chromatin Immuno-Precipitation, aims at detecting protein-DNA interactions.
ChIP-chip IP vs IP: two IP samples are hybridized on a microarray where probes cover the whole genome.

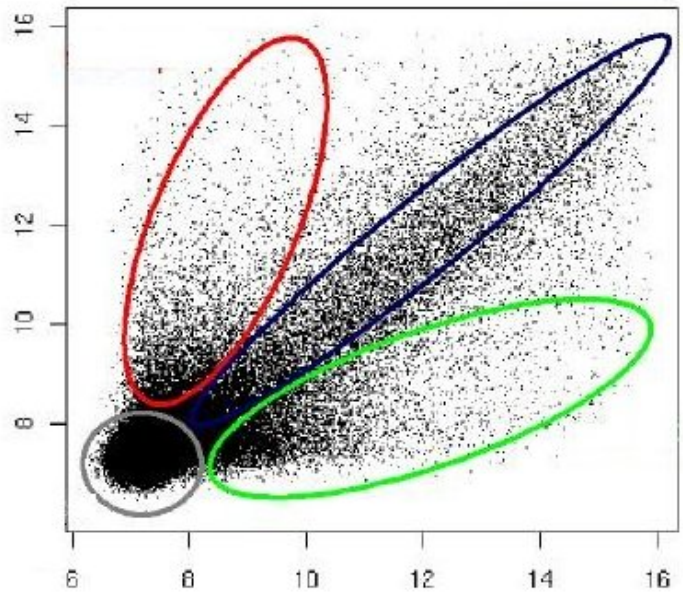
- Most methods relying on log-ratio = log(IP1/IP2)
- Non-zero log-ratio reveals differential protein-DNA interaction between samples 1 and 2.
- But the question of differential protein-DNA interaction can be recast as a research of underlying structure



Expected form of the dataset

- Consider the couple $(\log(IP1_i), \log(IP2_i))$
- An underlying structure is observed
- From a biological point of view, four subpopulations are expected

- a noise group: probes are not hybridized
- an identical group: probes hybridized similarly in both samples
- an overexpressed group: probe signal in Sample 1 is higher than in Sample 2
- an underexpressed group: probe signal in Sample 1 is lower than in Sample 2



Proposed model to compare two IP samples

Denoting

- $\mathbf{X}_i = (\log \mathrm{IP1}_i, \log \mathrm{IP2}_i)$ a random variable representing the two signals for probe i
- Z_i its unknown group

We assume that

- the Z_i's are i.i.d

$$Z_i \sim \mathcal{M}(1; \boldsymbol{\pi}), \qquad \pi_k = \Pr\{Z_i = k\} = \Pr\{Z_{ik} = 1\}$$

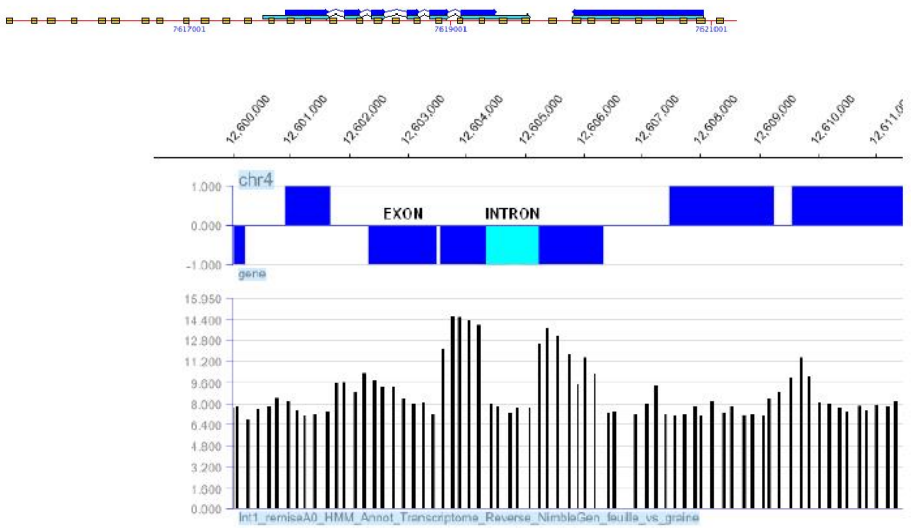
- the $\mathbf{X}_i$'s are independent conditionally to the Z_i's:

$$(\mathbf{X}_i | Z_i = k) \sim f(\cdot; \gamma_k), \qquad \text{e.g. } f(\cdot; \gamma_k) = \mathcal{N}(\boldsymbol{\mu}_k, \Sigma_k)$$

We have to estimate

$$\{\pi_k, \boldsymbol{\mu}_k, \Sigma_k\}_{k=1,\ldots,4} \qquad \text{and} \qquad Pr\{Z_i = k | \mathbf{X}_i = \mathbf{x}_i\}$$

Accounting for the genomic localisation



- Probes are (almost) equally spaced along the genome
- Probes tend to be clustered

We can propose an Hidden Markov model (Baum and Petrie (1966), Churchill (1992))

- The $\mathbf{X}_i$'s are still independent conditionally to the Z_i's:

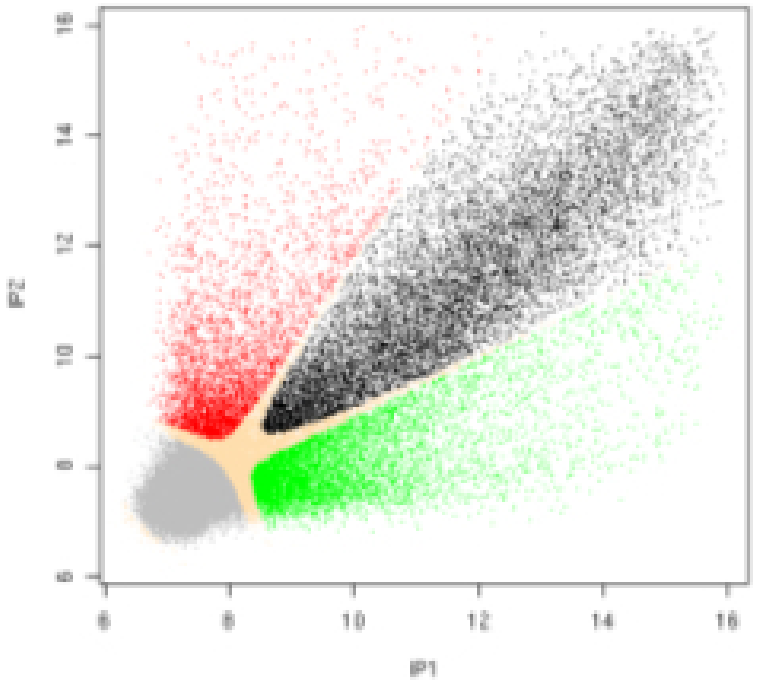
$$(\mathbf{X}_i | Z_i = k) \sim \mathcal{N}(\boldsymbol{\mu}_k, \Sigma_k)$$

- The $\{Z_i\} \sim MC(\boldsymbol{\pi})$

$$\pi_{k\ell} = \Pr\{Z_i = k | Z_{i-1} = \ell\}$$

Example of analysis on Arabidopsis thaliana

- Tiling array of Arabidopsis thaliana
- probes cover the whole genome
- Analysis is done per chromosome
- Comparison of a mutant to a wild-type plant (histone mark)



Regulatory network

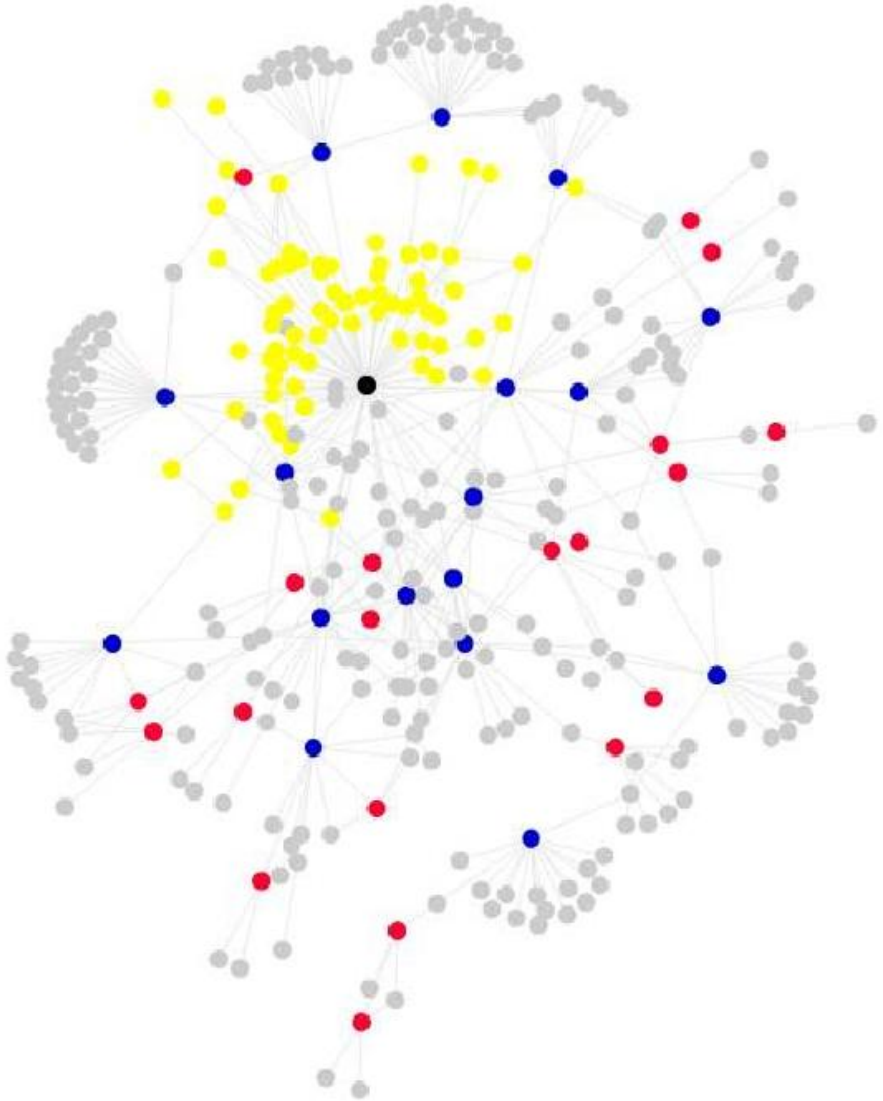
Regulatory network = directed graph where

- Nodes = genes (or groups of genes)
- Edges = regulations:

$$\{i \to j\} \quad \Leftrightarrow \quad i \text{ regulates } j$$

Typical questions are

- Do some nodes share similar connexion profiles?
- Is there a 'macroscopic' organisation of the network?



Proposed model

Denoting

- X_{ij} the presence of regulation from gene i to gene j,
- Z_i the unknown status of gene i,

we can assume that [Daudin *et al.* (2008)]

- the Z_i's are i.i.d

$$Z_i \sim \mathcal{M}(1; \boldsymbol{\pi}), \qquad \pi_k = \Pr\{Z_i = k\} = \Pr\{Z_{ik} = 1\}$$

- the X_{ij}'s are independent conditionally to the Z_i's:

$$(X_{ij}|Z_i = k, Z_j = \ell) \sim \mathcal{B}(\gamma_{k\ell}).$$

We want to estimate

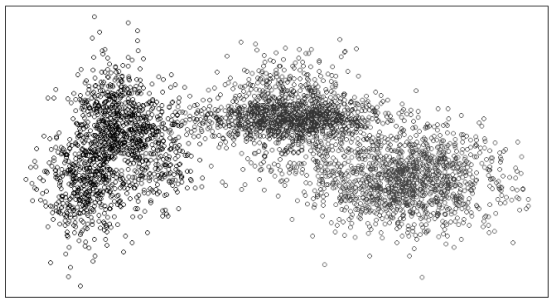
$$\boldsymbol{\theta} = (\boldsymbol{\pi}, \boldsymbol{\gamma}) \qquad \text{and} \qquad Pr\{Z_i = k|\{X_j\}\}$$

Table of contents

Table of contents

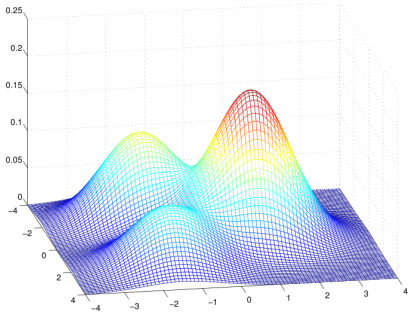
Key ingredients of a mixture model

what we observe

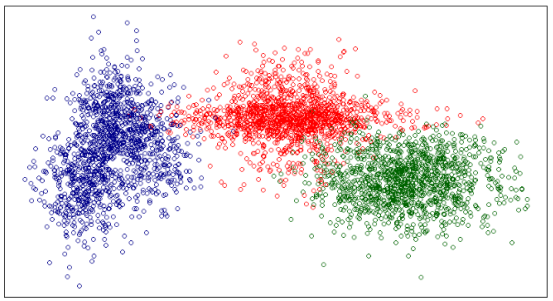


$Z = ?$

the model



the expected results



Z: 1 = ●, 2 = ●, 3 = ●

Let $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ denote n observations with $\mathbf{x}_i \in \mathbb{R}^Q$ and let $\mathbf{Z} = (Z_1, \ldots, Z_n)$ be the latent vector.

1) Distribution of Z: $\{Z_i\}$ are assumed to be independent and

$$P(Z_i = k) = \pi_k \quad \text{with} \quad \sum_{k=1}^{K} \pi_k = 1 \qquad \rightarrow \mathbf{Z} \sim \mathcal{M}(n; \pi_1, \ldots, \pi_K)$$

and where K is the number of components of the mixture

2) Distribution of $(\mathbf{X}_i | Z_i = k)$: a parametric distribution $f(\bullet; \gamma_k)$

Some properties:

- $\{Z_i\}$ are independent
- $\{\mathbf{X}_i\}$ are independent conditionally to $\{Z_i\}$
- Couples $\{(\mathbf{X}_i, Z_i)\}$ are i.i.d.
- The model is invariant for any permutation of the labels $\{1, \ldots, K\}$ $\Rightarrow$ the mixture model has $K!$ equivalent definitions.

Distribution of $\mathbf{X}$:

$$P(\mathbf{X}; \boldsymbol{\theta}) = \prod_{i=1}^{n} \sum_{k=1}^{K} P(\mathbf{X}_i, Z_i = k) = \prod_{i=1}^{n} \sum_{k=1}^{K} P(Z_i = k) P(\mathbf{X}_i | Z_i = k)$$

$$= \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k f(\mathbf{X}_i; \gamma_k)$$

$\rightarrow$ It is a weighted sum of parametric distributions known up to the parameter vector $\boldsymbol{\theta} = (\pi_1, \ldots, \pi_{K-1}, \gamma_1, \ldots, \gamma_K)$

Table of contents

Statistical inference of incomplete data models

Maximum likelihood estimate: We are looking for

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

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

$$\log P(\mathbf{x}; \boldsymbol{\theta}) = \sum_{i=1}^{n} \log \left[\sum_{k=1}^{K} \pi_k f(x_i; \gamma_k) \right]$$

- Explicit expressions of the estimators do not exist
- It is not always possible since this sum typically involves K^n terms.... untractable !!
- Maximization is usually done with an iterative algorithm

Complete likelihood

If Z were observed, we would calculate the complete likelihood

$$\begin{aligned}
\log P(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta}) &= \log P(\mathbf{Z}; \boldsymbol{\theta}) + \log P(\mathbf{X}|\mathbf{Z}; \boldsymbol{\theta}) \\
\log P(\mathbf{x}, \mathbf{Z}; \boldsymbol{\theta}) &= \sum_i \sum_k Z_{ik} \log \pi_k + \sum_i \sum_k Z_{ik} \log f(X_i; \gamma_k) \\
&= \sum_i \sum_k Z_{ik} [\log \pi_k + \log f(X_i; \gamma_k)].
\end{aligned}$$

It is much easier ... except that **Z** is unknown.
The key idea is to replace Z_i with what we expect to observe

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

Expectation-Maximization algorithm

Definition (Dempster, 1977)

It is an iterative algorithm based on the expectation of the complete likelihood conditionally to the observations and the current parameter $\boldsymbol{\theta}^{(l)}$

$$\boldsymbol{\theta}^{(\ell+1)} = \arg\max_{\boldsymbol{\theta}} \mathbb{E}\left\{\log P(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta}) | \mathbf{x}, \boldsymbol{\theta}^{(\ell)}\right\} := \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(\ell)})$$

Properties

- At each step, the observed likelihood increases
- Convergence is reached but not always toward a global maximum
- EM algorithm is sensitive to the initialization step

EM algorithm exists in all good statistical sotfwares

- EM algorithm is available in MCLUST and RMIXMOD packages of the R software.
- RMIXMOD proposes the best strategy of initialization

Expression of the completed likelihood

- The conditional expectation of the complete likelihood is named the completed likelihood.
- At step ℓ, it is defined by

$$\begin{aligned} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(\ell)}) &= \mathbb{E}\left\{ \log P(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta}) | \mathbf{x}, \boldsymbol{\theta}^{(\ell)} \right\} \\ &= \mathbb{E}\left\{ \sum_i \sum_k Z_{ik} [\log \pi_k + \log f(x_i; \gamma_k)] | \mathbf{x}, \boldsymbol{\theta}^{(\ell)} \right\} \\ &= \sum_i \sum_k \mathbb{E}(Z_i = k | x_i, \boldsymbol{\theta}^{(\ell)}) \log [\pi_k f(x_i; \gamma_k)] \end{aligned}$$

Recall that $\tau_{ik}^{(\ell)} := P(Z_i = k | X_i = x_i, \boldsymbol{\theta}^{(\ell)})$

$$Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(\ell)}) = \sum_i \sum_k \tau_{ik}^{(\ell)} \log \pi_k + \sum_i \sum_k \tau_{ik}^{(\ell)} \log f(x_i; \gamma_k)$$

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

Description of an EM algorithm

- Initialisation of $\boldsymbol{\theta}^{(0)} = (\pi_1, ..., \pi_K, \gamma_1, ..., \gamma_K)^{(0)}$.
- While the convergence is not reached

E-step Calculation of

$$\tau_{ik}^{(\ell)} = P(Z_i = k | x_i, \boldsymbol{\theta}^{(\ell-1)}) = \frac{\pi_k^{(\ell-1)} f(x_i; \gamma_k^{(\ell-1)})}{\sum_{k'} \pi_{k'}^{(\ell-1)} f(x_i; \gamma_{k'}^{(\ell-1)})}$$

M-step Maximization of

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

Description of an EM algorithm

- Initialisation of $\boldsymbol{\theta}^{(0)} = (\pi_1, ..., \pi_K, \gamma_1, ..., \gamma_K)^{(0)}$.
- While the convergence is not reached

E-step Calculation of

$$\tau_{ik}^{(\ell)} = P(Z_i = k | x_i, \boldsymbol{\theta}^{(\ell-1)}) = \frac{\pi_k^{(\ell-1)} f(x_i; \gamma_k^{(\ell-1)})}{\sum_{k'} \pi_{k'}^{(\ell-1)} f(x_i; \gamma_{k'}^{(\ell-1)})}$$

M-step Maximization of

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

Initialization: how to choose $\theta^{(0)}$?

- EM algorihtm is very sensitive to the initial values
- The best strategy is a small-EM
- Choose randomly several values
- From each of them, run some steps of the EM algorithm
- Choose the initial value maximizing the observed likelihood

Description of an EM algorithm

- Initialisation of $\boldsymbol{\theta}^{(0)} = (\pi_1, ..., \pi_K, \gamma_1, ..., \gamma_K)^{(0)}$.
- While the convergence is not reached

E-step Calculation of

$$\tau_{ik}^{(\ell)} = P(Z_i = k | x_i, \boldsymbol{\theta}^{(\ell-1)}) = \frac{\pi_k^{(\ell-1)} f(x_i; \gamma_k^{(\ell-1)})}{\sum_{k'} \pi_{k'}^{(\ell-1)} f(x_i; \gamma_{k'}^{(\ell-1)})}$$

M-step Maximization of

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

Examples of convergence criteria

- Fix ε at very small value e.g $\varepsilon = 10^{-8}$
- Max $|\boldsymbol{\theta}^{(\ell+1)} - \boldsymbol{\theta}^{(\ell)}| < \varepsilon$
- $|\log P(\mathbf{x}; \boldsymbol{\theta}^{(\ell+1)}) - \log P(\mathbf{x}; \boldsymbol{\theta}^{(\ell)})| < \varepsilon$

  Alternative method when EM is initialized with a small-EM
- Iterate a large number of times (e.g 1000 times)

Exercice

- Let n observations of a random variable X in $\mathbb{R}$ of unknown density g
- We propose to model g by a mixture of two univariate Gaussian distributions
- Give the distribution of the latent variable
- Give an expression of the conditional distribution of X
- Define the parameter vector of the model
- Give the expressions of the estimators in step ℓ of the EM algorithm

EM for an univariate Gaussian mixture

- $Z \in \{1, 2\}$: $P(Z = 1) = \pi_1$ 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 $x_1, x_2, ..., x_n$ are available
- The parameter estimators at step $(\ell + 1)$ of the EM algorithm are given by:

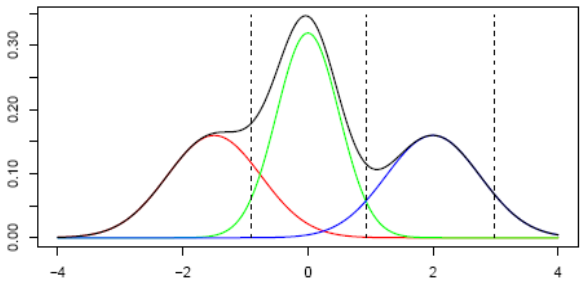
$$\begin{aligned}
\widehat{\pi}_1^{(\ell+1)} &= \frac{1}{n} \sum_{i=1}^{n} \tau_{i1}^{(\ell)}, \\
\widehat{\mu}_k^{(\ell+1)} &= \frac{1}{\sum_{i=1}^{n} \tau_{ik}^{(\ell)}} \sum_{i=1}^{n} \tau_{ik}^{(\ell)} x_i \\
\widehat{\sigma}_{k(\ell+1)}^2 &= \frac{1}{\sum_{i=1}^{n} \tau_{ik}^{(\ell)}} \sum_{i=1}^{n}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

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

Outputs of the model and data classification

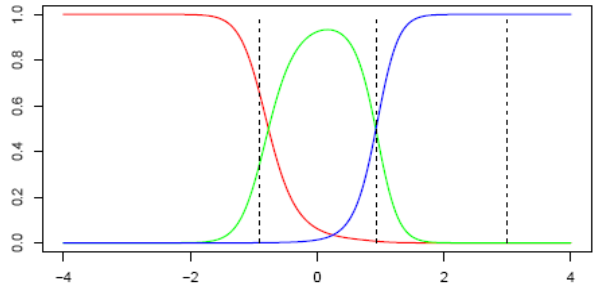
Distribution:

$$g(x_i) = \pi_1 f(x_i;\gamma_1) + \pi_2 f(x_i;\gamma_2) + \pi_3 f(x_i;\gamma_3)$$



Conditional probabilities:

$$\tau_{ik} = P(Z_i = k|x_i) = \frac{\pi_k f(x_i;\gamma_k)}{g(x_i)}$$



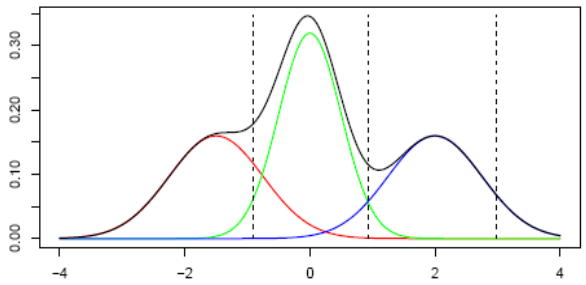
τ_{ik} (%)	$i=1$	$i=2$	$i=3$
$k=1$	0.658	0.007	0.0
$k=2$	0.342	0.478	0.0
$k=3$	0.0	0.515	1.0

$\rightarrow$ These probabilities enables the classification of the observations into the subpopulations

Outputs of the model and data classification

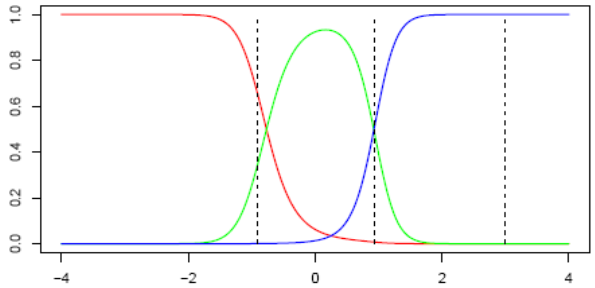
Distribution:

$$g(x_i) = \pi_1 f(x_i; \gamma_1) + \pi_2 f(x_i; \gamma_2) + \pi_3 f(x_i; \gamma_3)$$



Conditional probabilities:

$$\tau_{ik} = P(Z_i = k|x_i) = \frac{\pi_k f(x_i; \gamma_k)}{g(x_i)}$$



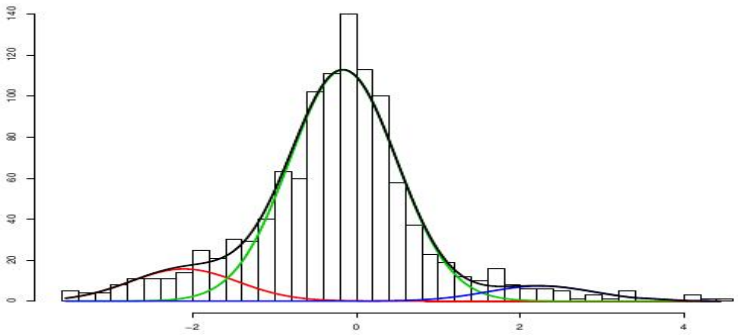
τ_{ik}	$i = 1$	$i = 2$	$i = 3$
$k = 1$	0.658	0.007	0.0
$k = 2$	0.342	0.478	0.0
$k = 3$	0.0	0.515	1.0

Maximum A Posteriori rule: Classification in the component for which the conditional probability is the highest.

Table of contents

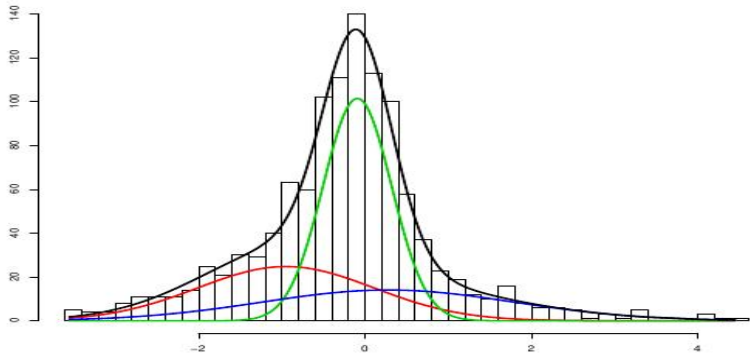
Homogeneous vs heterogeneous mixture (1/2)

Common variance



k	1	2	3
$\widehat{\pi}_k$	0.12	0.83	0.05
$\widehat{\mu}_k$	-2.11	-0.16	2.25
$\widehat{\sigma}_k^2$		0.43	

Different variances

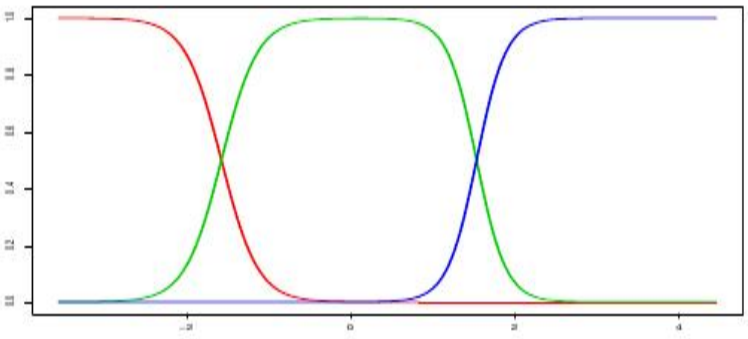


k	1	2	3
$\widehat{\pi}_k$	0.29	0.47	0.24
$\widehat{\mu}_k$	-0.95	-0.09	0.28
$\widehat{\sigma}_k^2$	1.12	0.17	2.29

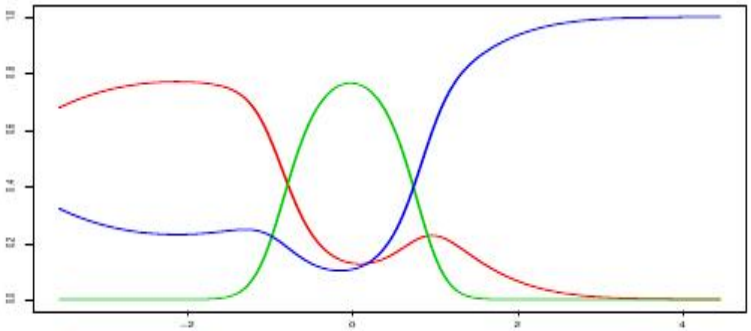
$\rightarrow$ The two variance modellings lead to different mixtures

Homogeneous vs heterogeneous mixture (2/2)

Common variance



Different variances



- Heterogenous variances provide a better fit to the distribution
- But the classification rule is not very convenient...
- In pratice we prefer homogeneous mixture models

Table of contents

Model selection

- The number of components of the mixture is often unknown and a collection of model $\mathcal{M}$ is considered.
- From a Bayesian point of view, the model maximizing the posterior probability $P(m|X)$ is to be chosen.
- With a non informative uniform prior distribution $P(m)$

$$P(m|X) \propto P(X|m)$$

.

Thus the chosen model satisfies

$$\tilde{m} = \underset{m \in \mathcal{M}}{\arg\max}\, P(\mathbf{X}|m) = \int P(\mathbf{X}|m,\theta)\pi(\theta|m)d\theta.$$

- $P(\mathbf{X}|m)$ is the integrated likelihood.
- $\pi(\theta|m)$ is the prior distribution of θ in the model m.

Bayesian Information Criterion: Schwarz (1978)

- $P(\mathbf{X}|m)$ is typically difficult to calculate
- an asymptotic approximation of $2\ln\{P(\mathbf{X}|m)\}$ is generally used
- This approximation is the Bayesian Information Criterion (BIC)

$$BIC(m) = \log P(\mathbf{x}|m, \widehat{\theta}) - \frac{\nu_m}{2}\log(n).$$

where

- ν_m is the number of free parameters of the model m
- $P(\mathbf{X}|m, \widehat{\theta})$ is the maximum likelihood under this model.

- The selected model $\widehat{m}$ maximizes the BIC criterion.
- BIC consistently estimates the number of components (Keribin C., 2000).
- BIC is expected to mostly select the dimension according to the global fit of the model.

Integrated Information Criterion (ICL)

- Biernacki *et al.* (2000) proposed a criterion based on the integrated complete likelihood

$$P(X, Z|m) = \int P(\mathbf{X}, Z|m, \theta)\pi(\theta|m)d\theta$$

- Using a BIC-like approximation of this integral

$$ICL(m) = \log P(\mathbf{X}, \widehat{Z}|m, \widehat{\theta}) - \frac{\nu_m}{2}\log(n),$$

where $\widehat{Z}$ stands for posterior mode of Z.

- McLachlan & Peel (2000) proposed to replace $\widehat{Z}$ with the conditional expectation of Z given the observation

$$ICL(m) = \log P(\mathbf{x}|m, \widehat{\theta}) - \mathcal{H}_X(Z) - \frac{\nu_m}{2}\log(n),$$

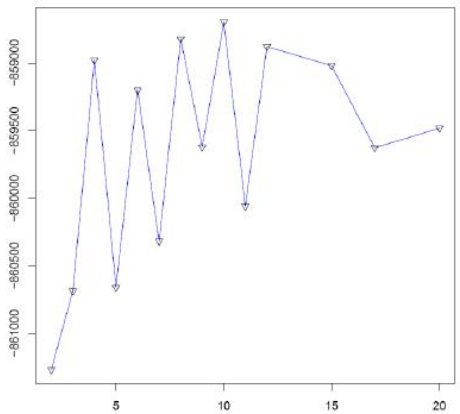
where

$$\mathcal{H}_X(Z) = -\mathbb{E}\left[\log P(Z|\mathbf{X}, m, \widehat{\theta})\right] = -\sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}\log\tau_{ik}$$

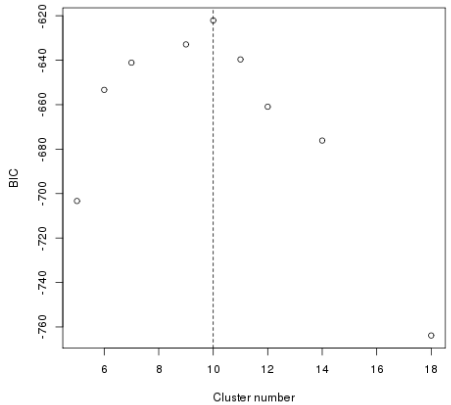
Conclusions for the model selection

- BIC aims at finding a good number of components to a global fit of the data distribution
- ICL is dedicated to a classification purpose. The penality has an entropy term that penalizes stronger models for which the classification is uncertain.
- For both criteria, they must be a convex function of the model dimension

Bad behavior



Correct behavior



$\rightarrow$ a non-convex function may indicate an issue of modeling

Slope heuristics (Birgé and Massart, 2006)

- Non-asymptotic framework: construct a penalized criterion such that the selected model has a risk close to the oracle model
- Theoretically validated in Gaussian framework, but encouraging applications in other contexts (Baudry et al., 2012)

$$SH(m) = \log P(\mathbf{X}|m, \hat{\theta}) + \kappa pen_{shape}(m)$$

In large dimensions:

- Linear behavior of $\frac{D}{n} \mapsto -\gamma_n(\hat{s}_D)$
- $\Rightarrow$ Estimation of slope to calibrate $\hat{\kappa}$ in a data-driven manner (Data-Driven Slope Estimation = DDSE), `capushe` R package

Table of contents

Mixture of multidimensional Gaussian

Let n observations of a random variable $X \in \mathbb{R}^Q$ of unknown density g.
Assume that a good proxy of g is a mixture

$$g(.) = \sum_{k=1}^{K} p_k \Phi(.|\mu_k, \Sigma_k)$$

with

- $\theta = (\mathbf{p}, \mu_1, \ldots, \mu_K, \Sigma_1, \ldots, \Sigma_K)$ where $\mathbf{p} = (p_1, \ldots, p_K)$, $\sum_{k=1}^{K} p_k = 1$
- $\Phi(.|\boldsymbol{\mu}_k, \Sigma_k)$ the density of $\mathcal{N}_Q(\boldsymbol{\mu}_k, \Sigma_k)$

Recall: $\boldsymbol{\mu}_k \in \mathbb{R}^Q$ and Σ_k is described by $\frac{Q(Q+1)}{2}$ parameters
Without assumptions, a mixture is described by $K \times Q + K \times \frac{Q(Q+1)}{2}$
$\rightarrow$ it becomes rapidly untractable

Mixture forms

Using the eigenvalue decomposition of each variance matrix:

$$\Sigma_k = \lambda_k D_k' A_k D_k$$

where

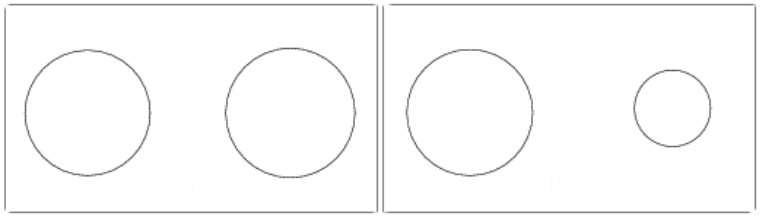
- $\lambda_k = |\Sigma_k|^{1/Q}$ (volume)
- D_k the matrix of eigenvectors of Σ_k (orientation)
- A_k the diagonal matrix of normalized eigenvalues of Σ_k (shape)

$\Rightarrow$ allow one to consider parsimonous and interpretable models.

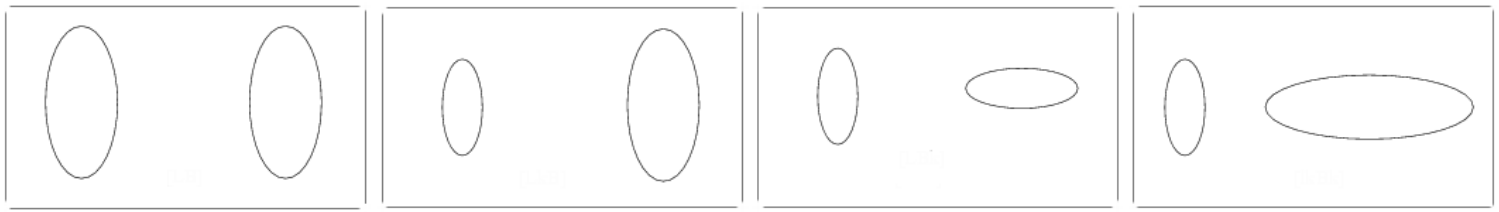
$$\begin{cases} \text{The spherical family} \\ \text{The diagonal family} \\ \text{The general family} \end{cases}$$

Mixture forms

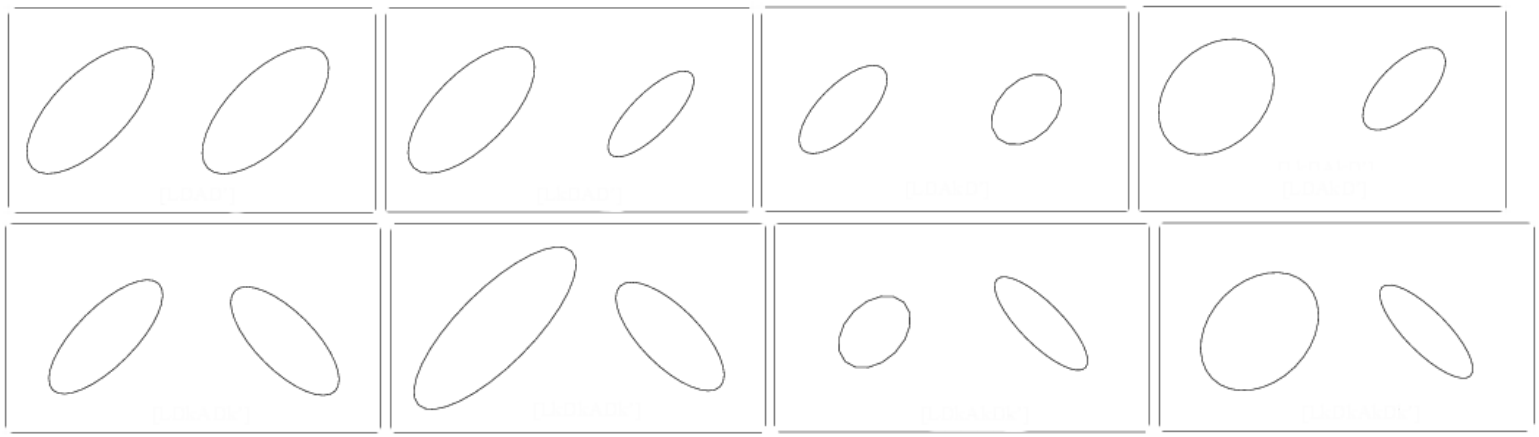
Spherical family (2 models)



Diagonal family (4 models)



General family (8 models)



Mixture forms

Using the eigenvalue decomposition of each variance matrix:

$$\Sigma_k = \lambda_k D_k' A_k D_k$$

where

- $\lambda_k = |\Sigma_k|^{1/Q}$ (volume)
- D_k the matrix of eigenvectors of Σ_k (orientation)
- A_k the diagonal matrix of normalized eigenvalues of Σ_k (shape)

$$\begin{cases} \text{The spherical family} \\ \text{The diagonal family} \\ \text{The general family} \end{cases}$$

$\Rightarrow$ 14 forms of mixture

Proportions can be assumed equal or not $\Rightarrow$ 28 forms of mixture

Hence a model is specified by $\begin{cases} K \text{ the number of clusters} \\ m \text{ the mixture form} \\ \text{if } \pi_k\text{'s are equal or not} \end{cases}$

Table of contents

Example of co-expression analysis

- Transcriptomic data from the database CATdb
- 4616 genes of Arabidopsis thaliana described by 33 biotic stress experiment
- Each gene is described by a vector $x_i \in \mathbb{R}^{33}$, where x_{ij} is the test statistic of Experiment j when a differential analysis is performed.
- The goal is to find groups of co-expressed genes
- Observations with missing values must be removed (or replace them with imputed values)
- Estimation of mixtures with a number of components varying from 2 to 70

```
>library(Rmixmod)
>Data<-read.table("StatOfTestSansDMANQ.txt",header=FALSE)
>dim(Data)
[1] 4616   33

>model<-mixmodGaussianModel(listModels=c("Gaussian_pk_L_C",
                        "Gaussian_pk_Lk_C","Gaussian_pk_L_Ck",
                        "Gaussian_pk_Lk_Ck"))

>strat <- mixmodStrategy(algo = "EM", nbTry = 5,
                      initMethod = "smallEM", nbTryInInit = 50,
              nbIterationInInit = 10, nbIterationInAlgo = 500,
              epsilonInInit = 0.001, epsilonInAlgo = 0.001)

>xem1<-mixmodCluster(Data,nbCluster=2:70,models=model,
                      strategy=strat,criterion=c("BIC","ICL"))
> save.image(file="savexem.RData")
```

Results analysis

- Plot the BIC and ICL curve as a function of the model dimension
- Look at the histogram of the maximum conditional probabilities
- Look at the maximum conditional probabilities of cluster membership by cluster with a boxplot

Some useful commands

- ICL1=sortByCriterion(xem1,"ICL") gives the values of the criterion ICL
- ICL1["bestResult"]@nbCluster and ICL1["bestResult"]@model describe the selected model by the criterion
- ICL1["bestResult"]@partition gives $\widehat{Z}$

For the j^{th} model of the collection

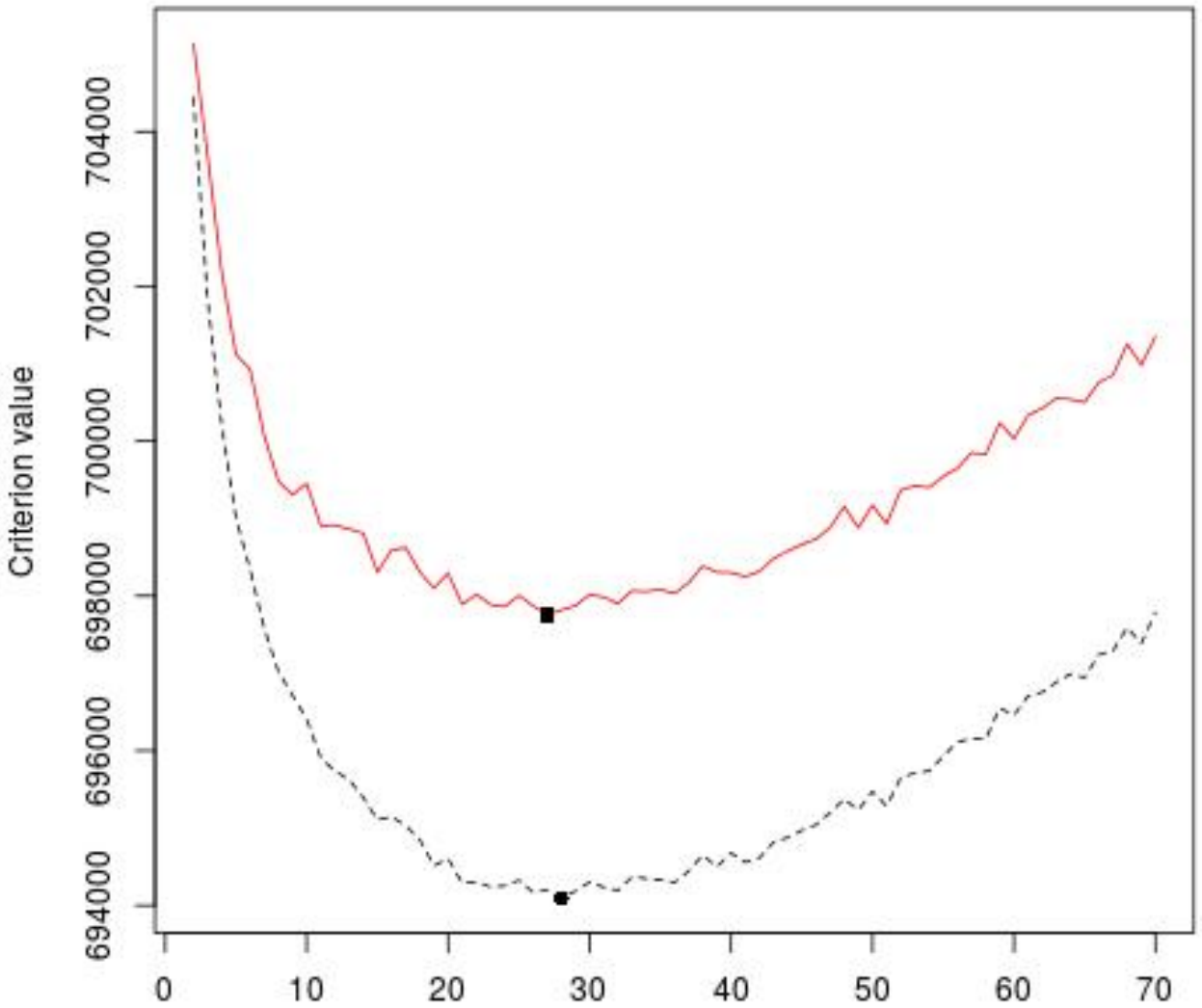
- xem1["results"][[j]]@model gives the mixture form
- xem1["results"][[j]]@nbCluster gives the component number
- xem1["results"][[j]]@proba is the matrix of the conditional probabilites
- xem1["results"][[j]]@criterionValue is a vector with BIC and ICL values

```
>ICL1<-sortByCriterion(xem1,"ICL")
>BIC1<-sortByCriterion(xem1,"BIC")
> ICL1["bestResult"]@nbCluster
[1] 27
> ICL1["bestResult"]@model
[1] "Gaussian_pk_Lk_C"
> BIC1["bestResult"]@nbCluster
[1] 28
> BIC1["bestResult"]@model
[1] "Gaussian_pk_Lk_C"
```

```
# to create a vector with criterion values
#for a given mixture form
critICL1<-NULL
critBIC1<-NULL
valK1<-NULL

for (j in 1:length(xem1["results"]))
{
 if(xem1["results"][[j]]@model == "Gaussian_pk_Lk_C")
  {
    critICL1<-c(critICL1,
                xem1["results"][[j]]@criterionValue[2])
    critBIC1<-c(critBIC1,
                xem1["results"][[j]]@criterionValue[1])
    valK1<-c(valK1,xem1["results"][[j]]@nbCluster)
  }
}
```

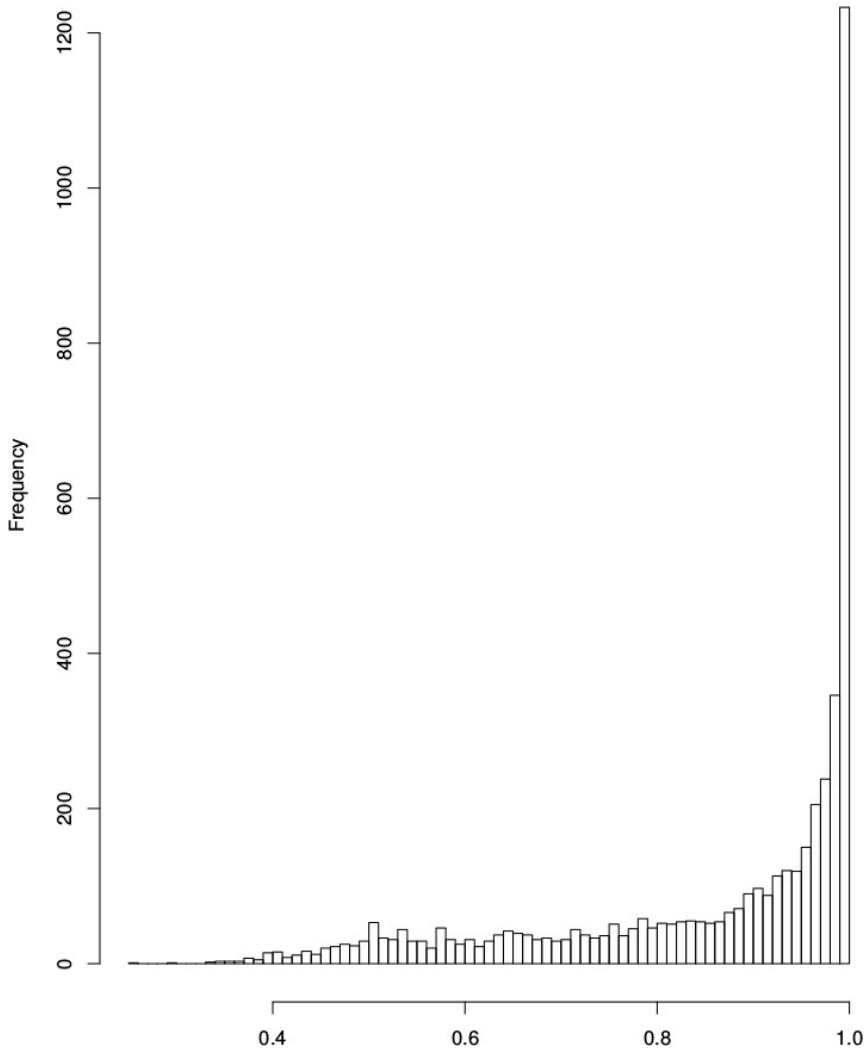
```
# to plot the criterion as a function of the model dimension


bornm1<-min(critBIC1,critICL1)
bornM1<-max(critBIC1,critICL1)
plot(valK1[order(valK1)],critBIC1[order(valK1)]
            ,type="l",ylim=c(bornm1-10,bornM1+10)
            ,main=" ",xlab=" ",ylab="Criterion value"
            ,lty = "dashed")
points(valK1[order(valK1)],critICL1[order(valK1)]
            ,type="l",col="red")
points(valK1[which.min(critICL1)],min(critICL1),pch=15)
points(valK1[which.min(critBIC1)],min(critBIC1),pch=16)
```

Model selection behaviour



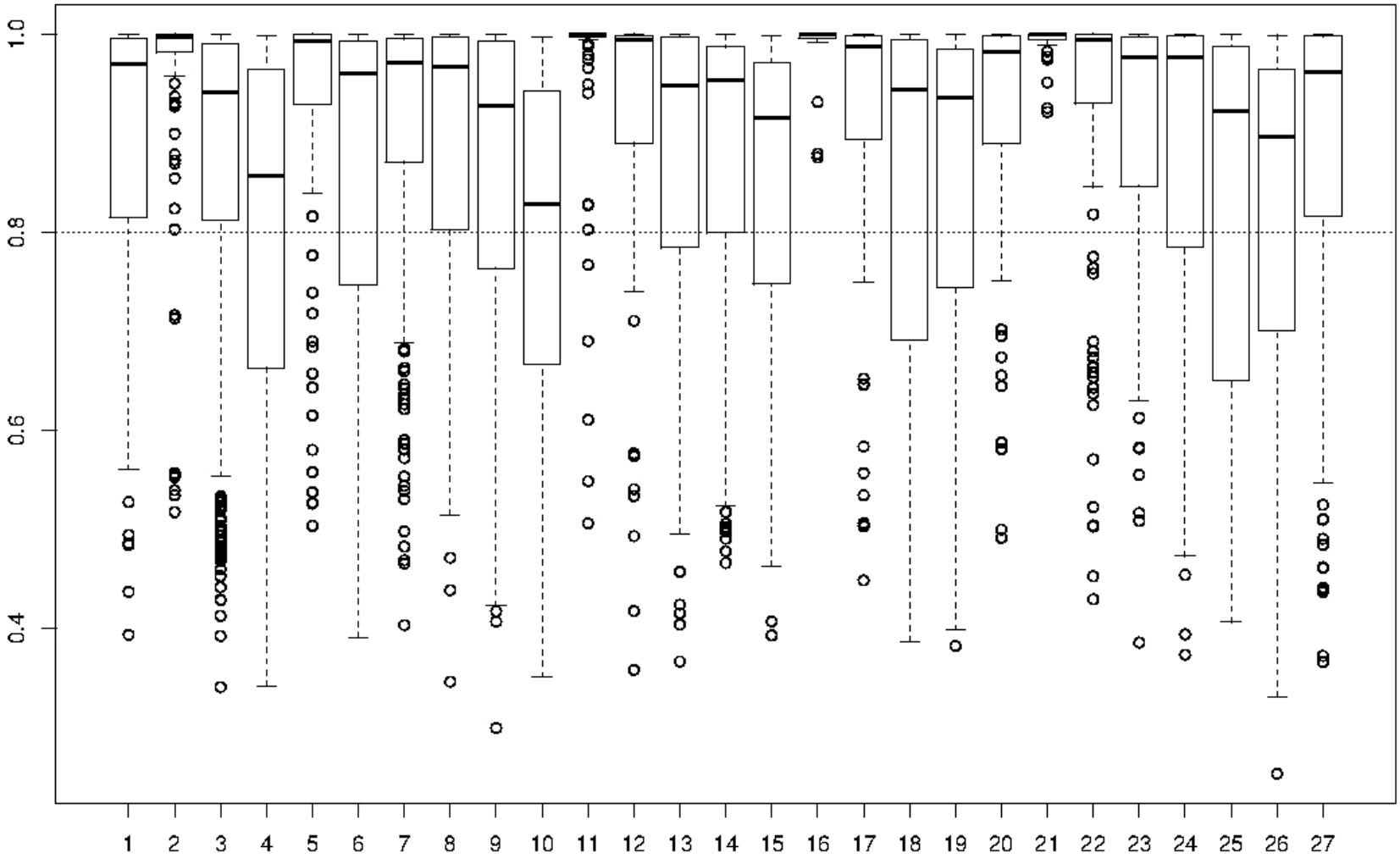
```
# Histogram of maximum conditional probabilities
# of cluster membership for all genes

max.cond.proba<-apply(ICL1["bestResult"]@proba,1,max)

hist(max.cond.proba,breaks=100,main=" ",xlab="")
```

Histogram of maximum conditional probabilities of cluster membership for all genes



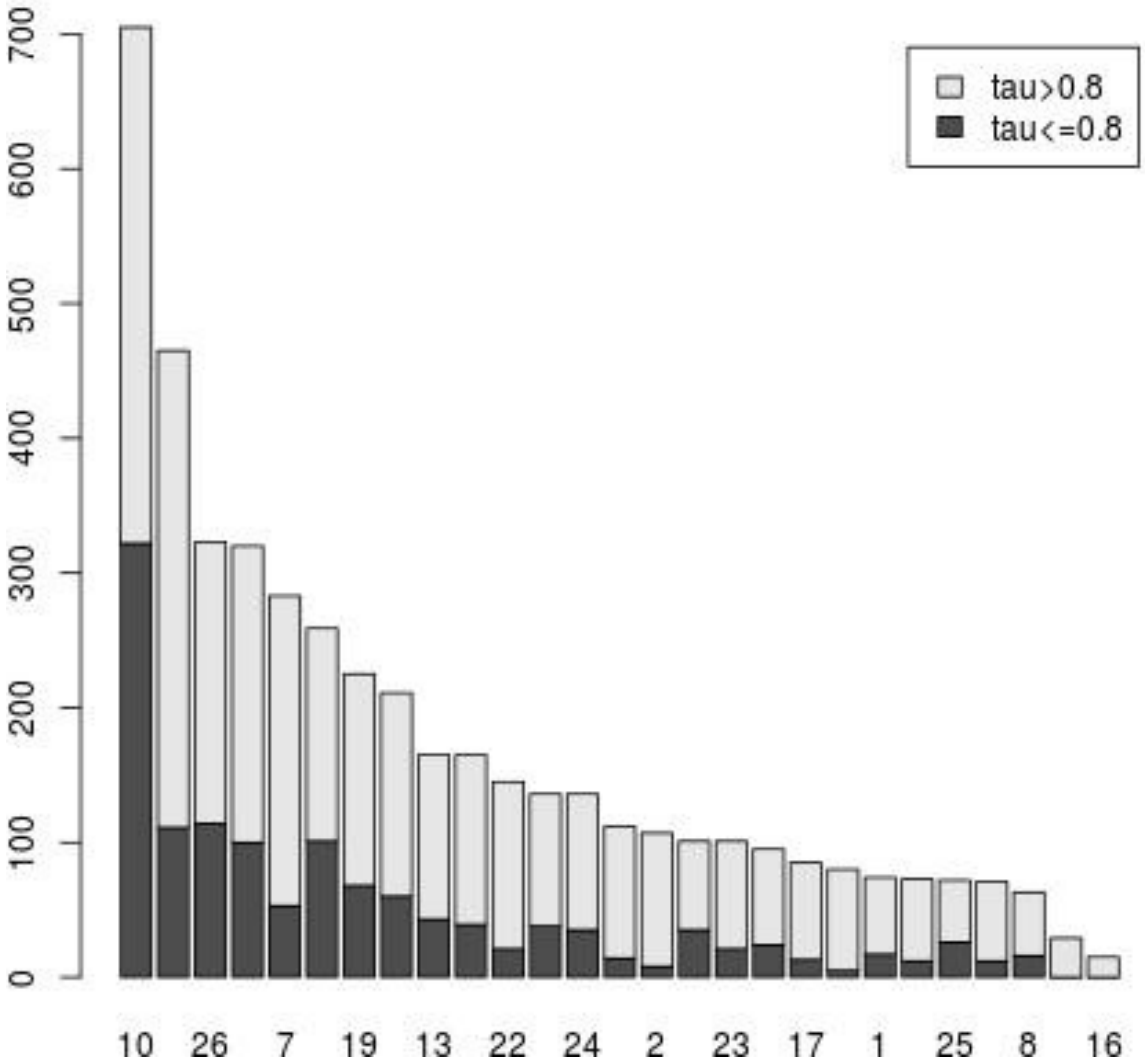
```
#Boxplot of maximum conditional probabilities
# of cluster membership by cluster

boxplot(max.cond.proba~ICL1["bestResult"]@partition)

abline(h = 0.8, col = "blue", lty = "dotted")
```

Boxplot of maximum conditional probabilities of cluster membership by cluster



```
# barplot

B<-cbind(tabulate(ICL1["bestResult"]@partition)
         ,tabulate(ICL1["bestResult"]@partition * (max.cond.p

II<-sort.int(tabulate(ICL1["bestResult"]@partition)
         ,index.return=TRUE,decreasing=TRUE)$ix

B<-B[II,]

B[,1]<-B[,1]-B[,2]

rownames(B)<-II

barplot(t(B),legend.text=c("tau<=0.8","tau>0.8"))
```

Cluster sizes and proportion of maximum conditional probabilities $t_{\max} > 0.80$ for each cluster



```
par(mfrow=c(2,2))
for (k in c(10,15,5,16))
{
I<-which(ICL1["bestResult"]@partition==k)
boxplot(Data[I,],ylim=c(min(Data[I,])-10,max(Data[I,])+10)
      ,main=paste("Cluster ",k,sep=""),xlab="",ylab="")
abline(h=0,lty=1)
}
```

Examples of four profiles

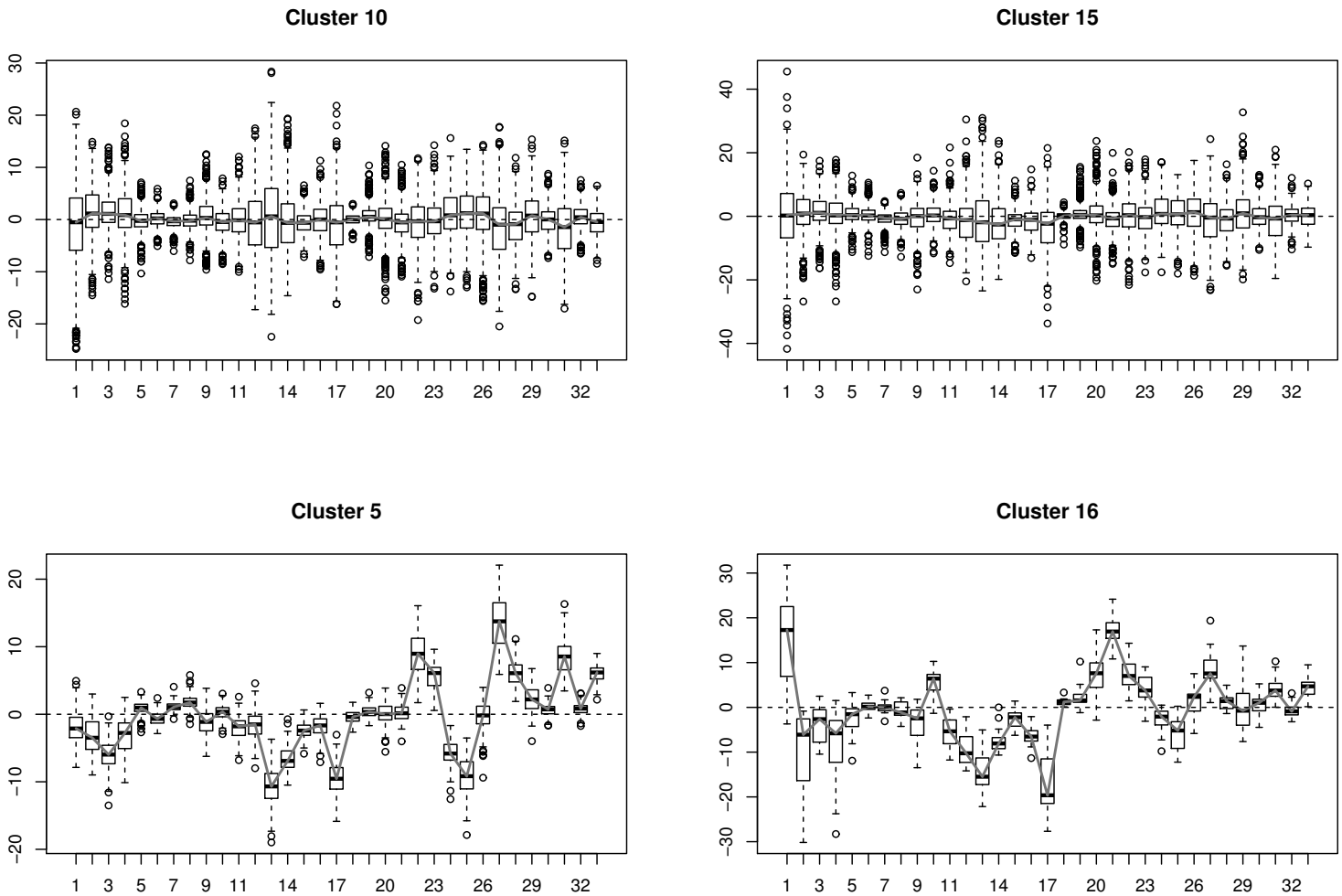


Table of contents

Conclusions

- Mixtures are useful to identify underlying structure

  How to model a mixture ?
- To define a mixture, specify the latent variable Z and its distribution
- Specify the conditional distribution of the observations

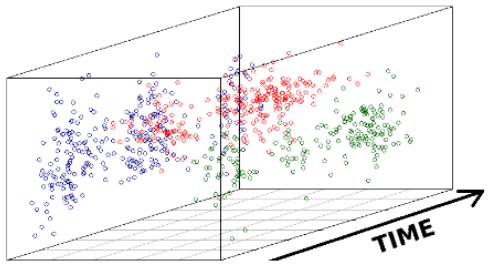
  How to estimate the parameters ?
- Parameters are estimated with an iterative algorithm
- EM algorithm is the most popular
- Other algorithm exist as well as SEM, CEM, SAEM

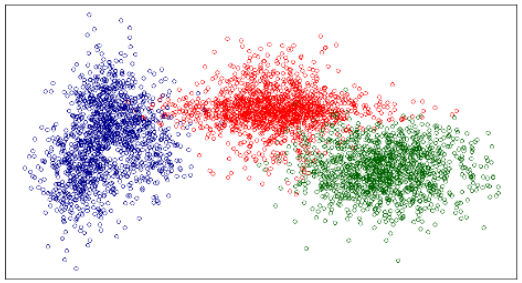
  How to select the number of components ?
- BIC and ICL are relevant in most cases

Table of contents

Table of contents

Motivation





- A spatial dependence may exist in the underlying structure
- It is possible to take it into account with an HMM
- Here an HMM is viewed as a generalization of a mixture model

Hidden Markov model

- $\{Z_i\} \sim MC(\boldsymbol{\pi})$, $\quad \pi_{k\ell} = \Pr\{Z_i = \ell | Z_{i-1} = k\}$;
- $Z_1 \sim \mathcal{M}(1; \boldsymbol{\nu})$ $\quad$ (e.g. $\nu =$ stationary distribution of $\boldsymbol{\pi}$);
- the X_i's are independent conditionally to $\mathbf{Z}$: $(X_i | Z_i = k) \sim f(\cdot; \gamma_k)$.

Distribution of the observed data

$$P(X_i; \boldsymbol{\theta}) = \sum_k \nu_k^i f(x; \gamma_k)$$

since

$$Z_i \sim \mathcal{M}(1; \boldsymbol{\nu}^i) \qquad \text{where} \qquad \boldsymbol{\nu}^i = \boldsymbol{\nu}\boldsymbol{\pi}^{i-1}$$

Dependency structure

Some properties:

- (Z_{i-1}, Z_i) are not independent,
- (X_{i-1}, X_i) are not independent,
- (X_{i-1}, X_i) are independent conditionally on Z_i,

Graphical representation

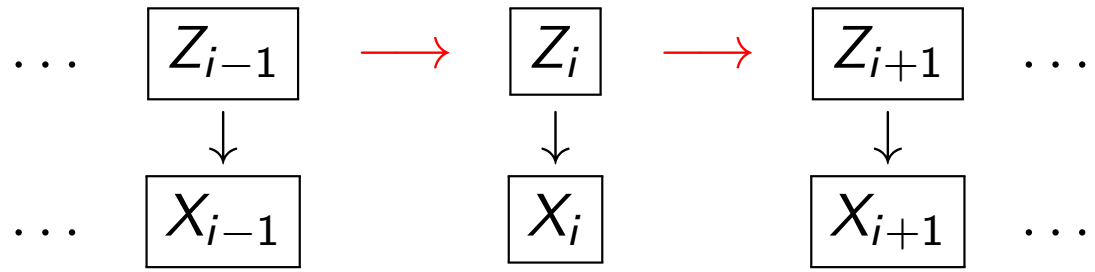


Table of contents

Inference

- As for an independent mixture model, an iterative algorithm is used
- It is based on the complete likelihood $P(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta})$

Recall on the EM algorithm

$$\boldsymbol{\theta}^{(\ell+1)} = \arg\max_{\boldsymbol{\theta}} \mathbb{E}\left\{\log P(\mathbf{X}, \mathbf{Z}; \boldsymbol{\theta}) | \mathbf{X}, \boldsymbol{\theta}^{(\ell)}\right\}$$

Complete and completed Likelihoods

$$
\begin{aligned}
P(\mathbf{X},\mathbf{Z}) &= P(\mathbf{Z})P(\mathbf{X}|\mathbf{Z}) \\
&= \left\{\prod_k \nu_k^{Z_{1k}} \prod_{i>1}\prod_{k,\ell} \pi_{k\ell}^{Z_{i-1,k}Z_{i,\ell}}\right\}\left\{\prod_i\prod_k f(X_i;\gamma_k)^{Z_{ik}}\right\} \\
\log P(\mathbf{X},\mathbf{Z}) &= \sum_k Z_{1k}\log\nu_k + \sum_{i>1}\sum_{k,\ell} Z_{i-1,k}Z_{i,\ell}\log\pi_{k\ell} \\
&\quad + \sum_i\sum_k Z_{ik}\log f(X_i;\gamma_k)
\end{aligned}
$$

By definition Completed likelihood $:= \mathbb{E}[\log P(\mathbf{X},\mathbf{Z})|X_1^n]$
We get

$$
\begin{aligned}
&\sum_k \mathbb{E}[Z_{1k}|X_1^n]\log\nu_k + \sum_{i>1}\sum_{k,\ell}\mathbb{E}[Z_{i-1,k}Z_{i,\ell}|X_1^n]\log\pi_{k\ell} \\
&\quad + \sum_i\sum_k \mathbb{E}[Z_{ik}|X_1^n]\log f(X_i;\gamma_k)
\end{aligned}
$$

Baum's forward-backward algorithm revisited (Devijvers 1985 Pattern Recogn Lett)

- Initialisation of $\boldsymbol{\theta}^{(0)} = (\Pi, \gamma_1, ..., \gamma_K)^{(0)}$.
- While the convergence is not reached

E-step Calculation of

$$\begin{aligned} \tau_{ik}^{(\ell)} &= P(Z_i = k | X_i, \boldsymbol{\theta}^{(\ell-1)}) \\ \eta_{ikh}^{(\ell)} &= \mathbb{E}[Z_{i-1,k} Z_{ih} | X_1^n, \boldsymbol{\theta}^{(\ell-1)}] \end{aligned}$$

M-step Maximization in $\boldsymbol{\theta} = (\boldsymbol{\pi}, \boldsymbol{\gamma})$ of

$$\sum_k \tau_{1k}^{(\ell)} \log \nu_k + \sum_{i>1} \sum_{k,h} \eta_{ikh}^{(\ell)} \log \pi_{kh} + \sum_i \sum_k \tau_{ik}^{(\ell)} \log f(x_i; \gamma_k)$$

Calculation of τ_{ik} and η_{ikh}

It is done by a Forward-backward algorithm

Forward equation: Denoting $F_{i\ell} = \Pr\{Z_i = \ell | X_1^i\}$,

$$F_{i\ell} \propto \sum_k F_{i-1,k} \pi_{k\ell} f(X_i; \gamma_\ell)$$

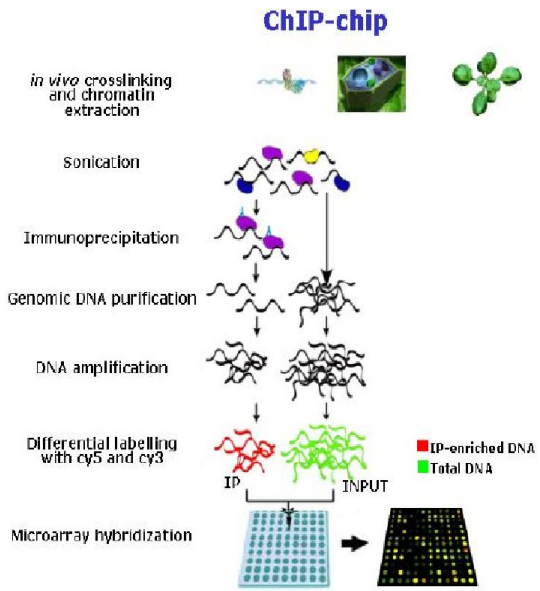
Backward equation: Once we get all F_{ik}, we get the τ_{ik} as

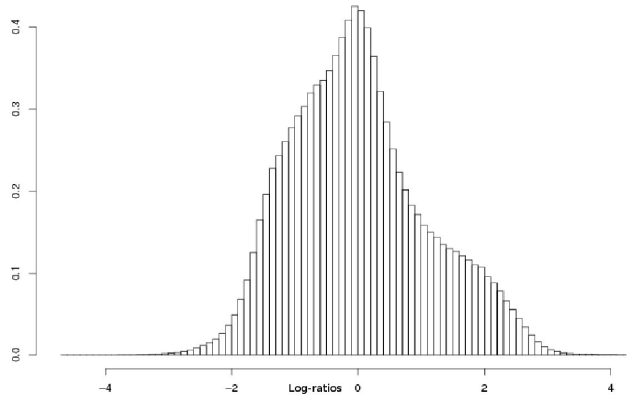
$$\tau_{ik} = \sum_\ell \eta_{ik\ell} \qquad \text{with} \quad \eta_{ik\ell} = \pi_{k\ell} \frac{\tau_{i+1,\ell}}{G_{i+1,\ell}} F_{ik} \quad \text{where} \quad G_{i+1,\ell} = \sum_k \pi_{k\ell} F_{ik}$$

Table of contents

Case of IP/Input experiments

Complete DNA is often used as a reference ('Input') to detect protein-DNA interaction.

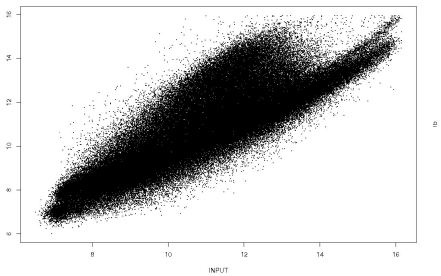




Question: What are the enriched probes, i.e. probes with an IP signal higher than an Input signal ?

Available data

- Let us consider n observations of a random variable $X = (\log Input, \log IP)$ (observations are probes of a microarray)
- We would like to model the IP signal as a function of the Input signal



- What mixture should be relevant ?
- Imagine that you have several biological replicates of this experiment, how to include this information in the model ? Write the model and give the parameter vector.
- How to take into account that signals of two adjacent probes are quite similar ? Write the model and give the parameter vector.

Mixture of regressions

The conditional density distribution of IP is modeled by a mixture of two linear regressions

$$\begin{aligned} IP_i &= a_0 + b_0 Input_i + \varepsilon_i \qquad \text{if } Z_i = 0 \text{ (normal)} \\ &= a_1 + b_1 Input_i + \varepsilon_i \qquad \text{if } Z_i = 1 \text{ (enriched)} \end{aligned}$$

where ε_i is a Gaussian random variable with mean 0 and variance σ^2.
The distribution of IP conditionally to Input is

$$g(IP_i|Input_i) = (1-\pi)\phi_0(IP_i|Input_i) + \pi\phi_1(IP_i|Input_i),$$

where

- π is the proportion of enriched probes
- $\phi_k(\cdot|x)$ stands for the probability density function of a Gaussian distribution with mean $a_k + b_k x$ and variance σ^2.

Inference using an EM algorithm

The mixture parameters (proportion, intercepts, slopes and variance) are estimated using the EM algorithm. Estimators are only weighted versions of the standard OLS intercept and slope estimates.

$$\hat{b}_k = \frac{\sum_i (Input_i - \overline{Input}_k)(IP_i - \overline{IP}_k)}{\sum_i \tau_{ik}(Input_i - \overline{Input}_k)^2}$$

$$\hat{a}_k = \overline{IP}_k - \hat{b}_k \overline{Input}_k$$

$$\hat{\sigma}^2 = \frac{1}{n} \sum_i \sum_k \hat{\tau}_{ik} \left[IP_i - (\hat{a}_k + \hat{b}_k Input_i) \right]$$

		constante a	pente b	variance σ^2	probabilité π
REP1	Groupe Normal	1.47	0.82	0.42	0.74
	Groupe Enrichi	-0.47	1.17	0.42	0.26
REP2	Groupe Normal	2.29	0.75	0.38	0.76
	Groupe Enrichi	0.47	1.07	0.38	0.24

MultiChIPmix: Mixture of two linear regressions

- Let Z_i the status of the probe i: $P(Z_i = 1) = \pi$
- The linear relation between IP and Input depends on the probe status

$$IP_{ir} = \begin{cases} a_{0r} + b_{0r} Input_{ir} + E_{ir} & \text{if } Z_i = 0 \text{ (normal)} \\ a_{1r} + b_{1r} Input_{ir} + E_{ir} & \text{if } Z_i = 1 \text{ (enriched)} \end{cases} \qquad V(IP_{ir}) = \sigma_r^2$$

The distribution of IP conditionally to Input is

$$g(IP_{ir}|Input_{ir}) = (1-\pi)\phi_{0r}(IP_{ir}|Input_{ir}) + \pi\phi_{1r}(IP_{ir}|Input_{ir}),$$

where

- π is the proportion of enriched probes
- $\phi_{kr}(\cdot|x)$ stands for the probability density function of a Gaussian distribution with mean $a_{kr} + b_{kr}x$ and variance σ^2.

MultiChIPmix: Mixture of two linear regressions

- Let Z_i the status of the probe i: $P(Z_i = 1) = \pi$
- The linear relation between IP and Input depends on the probe status

$$IP_{ir} = \begin{cases} a_{0r} + b_{0r} Input_{ir} + E_{ir} & \text{if } Z_i = 0 \text{ (normal)} \\ a_{1r} + b_{1r} Input_{ir} + E_{ir} & \text{if } Z_i = 1 \text{ (enriched)} \end{cases} \qquad V(IP_{ir}) = \sigma_r^2$$

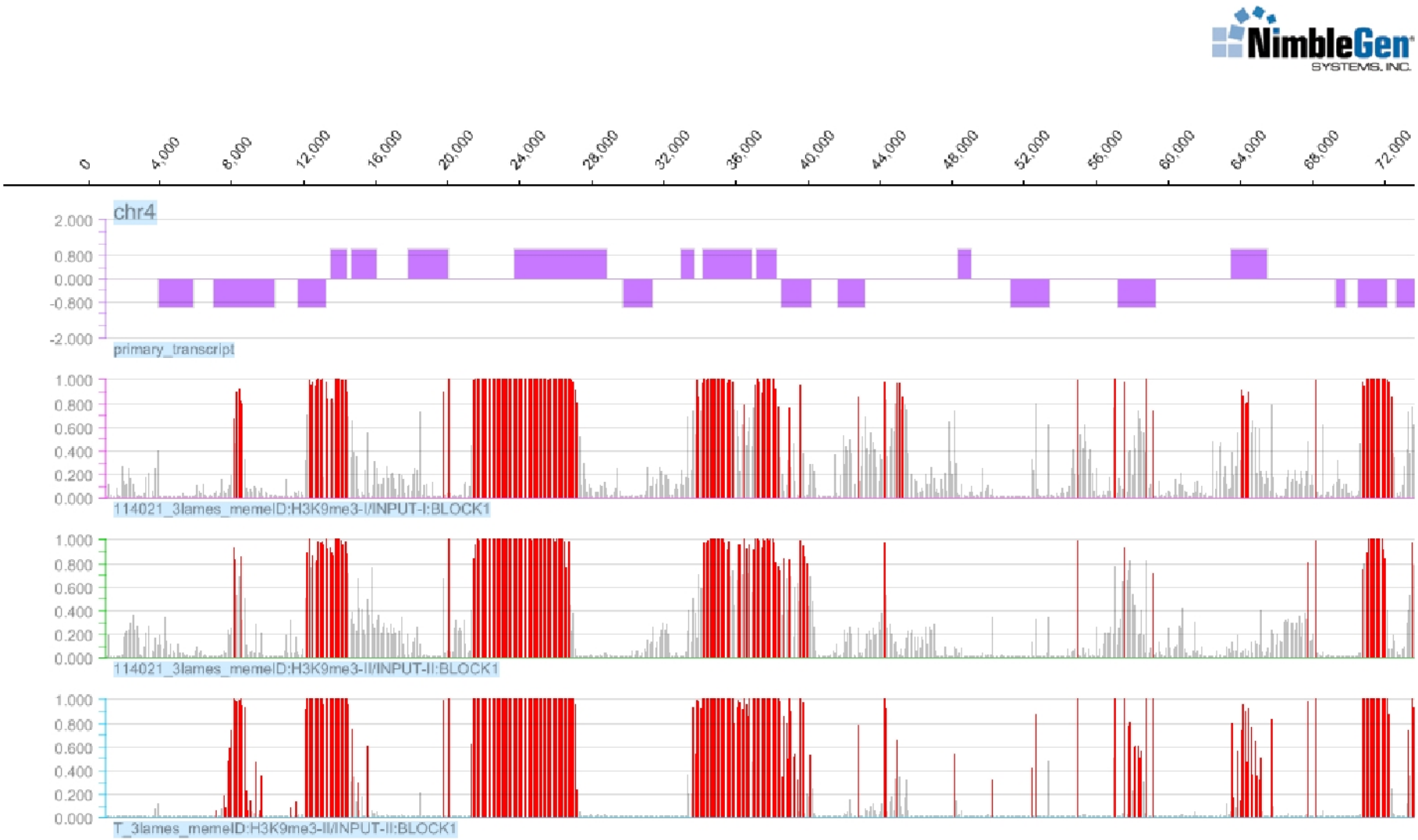
The distribution of IP conditionally to Input is

$$g(IP_{ir}|Input_{ir}) = (1-\pi)\phi_{0r}(IP_{ir}|Input_{ir}) + \pi\phi_{1r}(IP_{ir}|Input_{ir}),$$

where

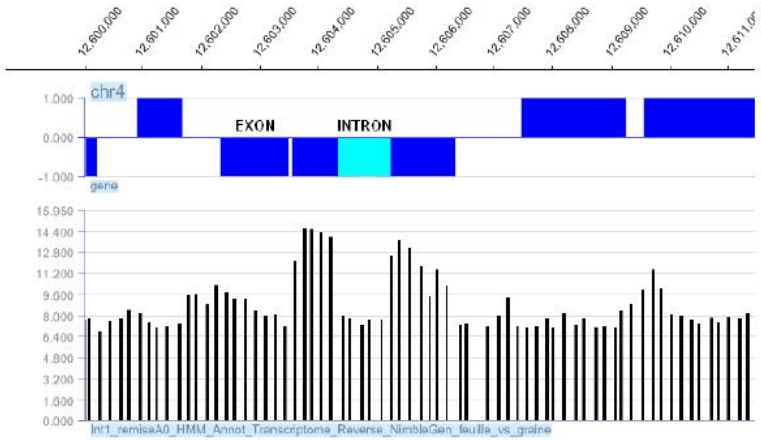
- π is the proportion of enriched probes
- $\phi_{kr}(\cdot|x)$ stands for the probability density function of a Gaussian distribution with mean $a_{kr} + b_{kr}x$ and variance σ^2.

Generalization for analyzing several replicates



Accounting for the genomic localisation

- Probes are (almost) equally spaced along the genome
- Probes tend to be clustered



Natural framework: Hidden Markov model (HMM)

- $$(IP_{ir}|Input_{ir}, Z_i = d) \sim \prod_r f(\cdot; \gamma_{dr})$$
- But the latent variables are (Markov-)dependent: $\{Z_i\} \sim MC(\Pi)$

$$\pi_{k\ell} = \Pr\{Z_i = k|Z_{i-1} = \ell\}$$

Evaluation of the model improvement

Based on simulated data :

- two scenarios: (i) well-separated non-enriched and enriched probes (slope parameters 0.6 and 0.99) (ii) overlapping populations of non-enriched and enriched probes (slope parameters 0.5 and 0.65).
- Two biological replicates for each scenario : $\sigma_1^2 = 0.7$ and $\sigma_1^2 = 0.75$
- The transition matrix is set to (0.97 ; 0.03 ; 0.01; 0.9)

Evaluation results with a ROC curve

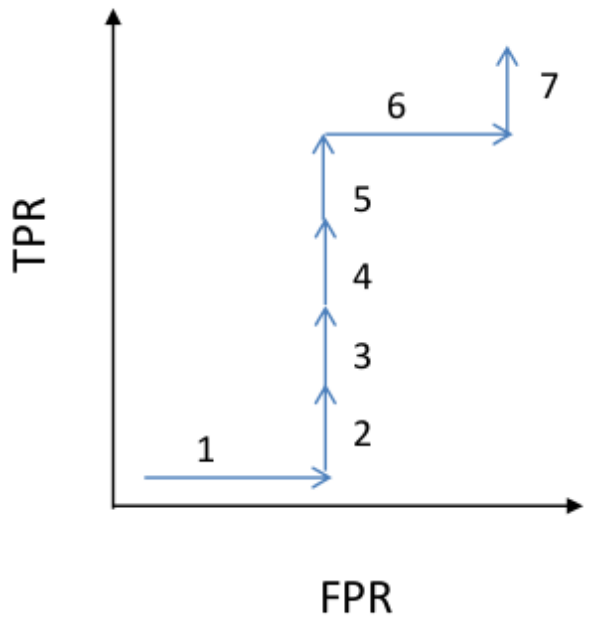
Definition of a ROC curve

Drawing a ROC curve:
1- sort genes by increasing raw p-value
2- knowing the truth (DE or NDE) for each gene, go down the sorted list counting the proportion of all the DE genes encountered so far (TPR) and the proportion of all the NDE genes encountered so far in the list (FPR)

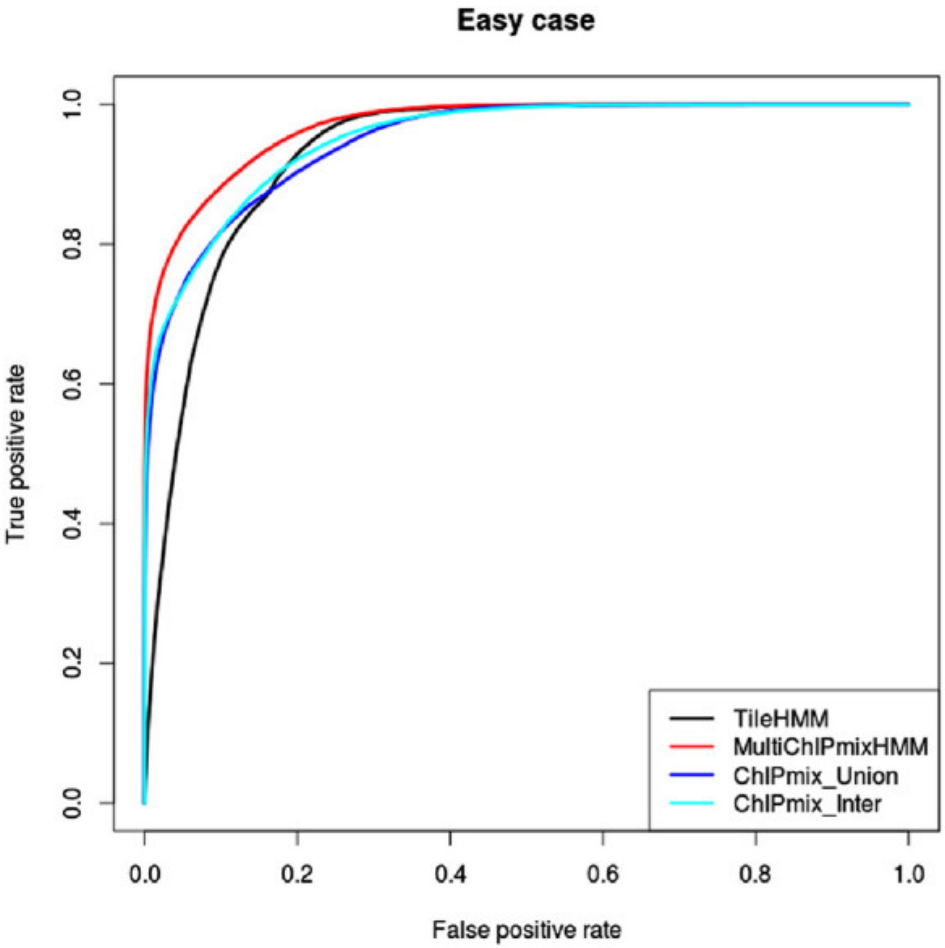
Example:

7 genes: 5 DE and 2 NDE

rank	gene	p-value	truth	TPR	FPR
1	G1	p1	NDE	0/5	1/2
2	G2	p2 (>p1)	DE	1/5	1/2
3	G3	p3(>p2)	DE	2/5	1/2
4	G4	p4(>p3)	DE	3/5	1/2
5	G5	p5(>p4)	DE	4/5	1/2
6	G6	p6(>p5)	NDE	4/5	2/2
7	G7	p7(>p6)	DE	5/5	2/2



Results presented with a ROC curve



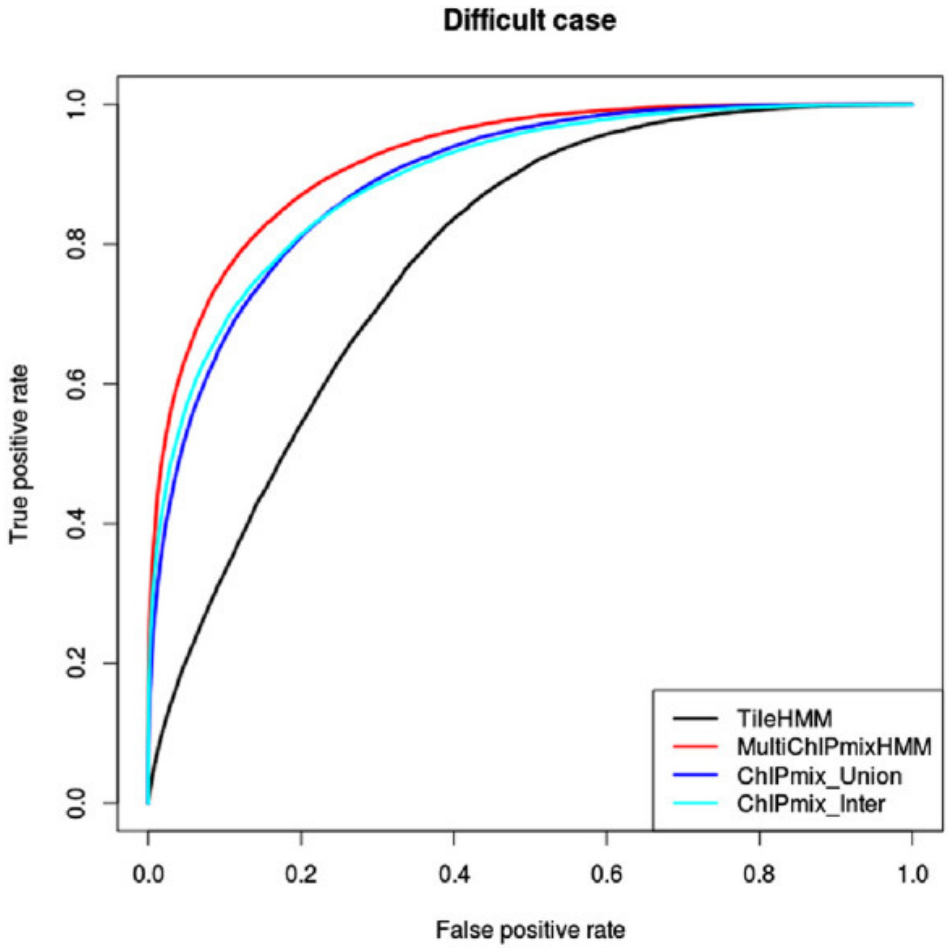


Table of contents

Conclusions

- Here, HMM is viewed as a mixture genarlization
- HMM takes a known spatial structure into account

  How to model an HMM ?
- Specify the latent variable Z and its distribution
- Specify the conditional distribution of the observations

  How to estimate the parameters ?
- Parameters are estimated with an EM-like algorithm
- E-step is replaced by a Forward-Backward algorithm

  How to select the number of components ?
- BIC-like or ICL-like can be derived

Table of contents

Final conclusions

To date, mixtures and HMM are not very popular.
However they are very useful for

- co-expression study

  GEM2net project: http://urgv.evry.inra.fr/GEM2NET/

  MAP kinases study (Frey-dit-frei et al., Genome Biology, 2014)

  of RNAseq data: HTSCluster (Rau et al., Bioinformatics, 2015)

- Tiling array analysis

  the first epigenomic map of Arabidopsis thaliana (Roudier, EMBO journal, 2011)

  Exosome study (Lange et al, PloS Genetics, 2014)

- Network characterization

  Group the nodes of a graph to find communities (Daudin et al., Statistics and computing, 2008)