Introduction to Probabilistic Graphical Models and Deep Generative Models

P. Latouche

PR UCA
Ecole Polytechnique / ENS Paris Saclay
pierre.latouche@math.cnrs.fr
`https://lmbp.uca.fr/~latouche/`
`https://lmbp.uca.fr/~latouche/mva/IntroductiontoProbabilisticGraphicalModelsMVA.html`

ENS Paris Saclay, Master MVA

Part I

Lecture 2: K-means
EM
Gaussian mixtures

Outline Part 1

Clustering

Gaussian mixture models

EM algorithm

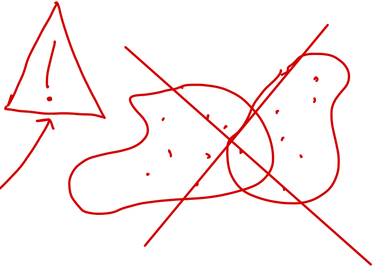
Model selection

Introduction to clustering

- we are provided with a data set $(x_i)_i = (x_1, \ldots, x_n)$ (sample of size n) with $x_i \in \mathbb{R}^d$
- goal : retrieve groups $=$ clusters $=$ classes of individuals where :
  - 2 individuals within a group must be as similar as possible
  - 2 individuals of different groups must be as different as possible
- unsupervised learning setting : no target variable
- we aim at uncovering (learning) what is hidden in the data set

Types of clusters :

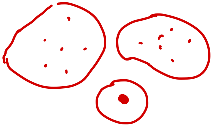
- disjoint
- hierarchical
- ~~overlapping~~



Def

A partition of a data set $(x_i)_i$ into K clusters $\mathcal{P} = (C_k)_k = (C_1, \ldots, C_K)$ verifies :

1. $\cup_{k=1}^{K} C_k = (x_i)_i$
2. $C_k \cap C_l = \emptyset, \forall k \neq l$



So each observation is clustered into a unique cluster

Def

The nth Bell number B_n counts the number of different ways to partition a set that has exactly n elements

Theorem

The Bell numbers satisfy the following recurrence relation :

$$B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k$$

ex : $B_{18} = 682\,076\,806\,159$

Dobinski's formula

$$B_n = \frac{1}{e} \sum_{k=0}^{\infty} \frac{k^n}{k!}$$

- so exact clustering is a combinatorial problem (NP-hard)
- two most famous heuristics : kmeans and hierarchical clustering
- the statistical point of view : mixture models and expectation maximisation

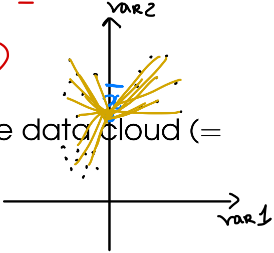
Inertia

Def

The total inertia of a data cloud of points in $\mathbb{R}^d$ (sample of observations) is :

ex: $d = 2$

$$S = \frac{1}{n}\sum_{i=1}^{n} ||x_i - \bar{x}||^2, \geq 0$$



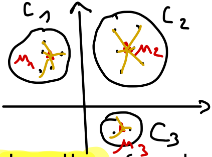
where $\bar{x} = (1/n)\sum_{i=1}^{n} x_i$ is the barycentre of the data cloud (= empirical mean of the sample)

Remark

The total inertia can be written :

$$S = \sum_{j=1}^{d} \hat{\sigma}_j^2,$$

where $\hat{\sigma}_j^2 = (1/n)\sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2$ is the empirical (biased) variance of variable j

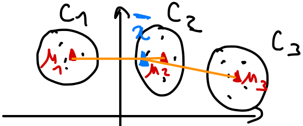


Def

The intra class inertia of a data cloud of points in $\mathbb{R}^d$ (sample of observations), for a partition $\mathcal{P} = (C_k)_k$ with K clusters, is :

$$W = \frac{1}{n} \sum_{k=1}^{K} \sum_{x_i \in C_k} ||x_i - \mu_k||^2, \geq 0$$

where $\mu_k = (1/n_k) \sum_{x_i \in C_k} x_i$ is the empirical mean of the observations in cluster C_k and n_k is the number of observations in C_k



Def

The inter class inertia of a data cloud of points in $\mathbb{R}^d$ (sample of observations), for a partition $\mathcal{P} = (C_k)_k$ with K clusters, is :

$$B = \frac{1}{n}\sum_{k=1}^{K} n_k ||\mu_k - \bar{x}||^2, \geqslant 0$$

Huygens theorem

$$S = W + B$$

Remarks

- S does not depend on the partition $\mathcal{P}$ contrary to W and B
- so, when W decreases, B increases, and vice versa

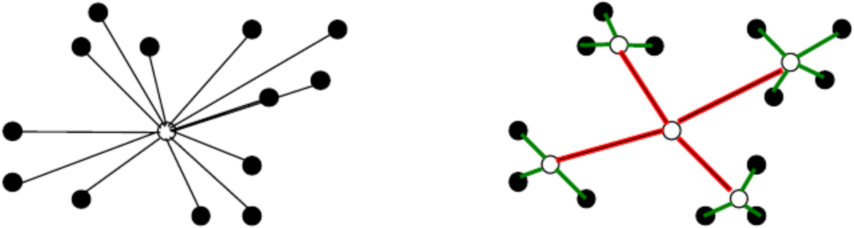
A clustering task

Find $\mathcal{P}$ which minimises W (maximises B) with $K < n$

Remark

When $K = n$ (each observation is in its own cluster), $W = 0 \rightarrow$ useless in practice

Inertia and Huygens theorem



- again : combinatorial problem (NP hard) (if $K < n$)
- heuristics

kmeans

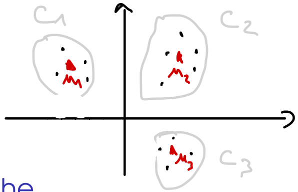
The kmeans algorithm focuses directly on the optimisation of W, K being fixed

- init : initialise all the (bary)centres μ_k (at random in $\mathbb{R}^d$ or on random observations)

1. each observation is clustered in the cluster with the closest centre
2. recompute the centres
3. if the μ_k have moved (no eps convergence) back to 1.

$d = 2$

$K = 3$



Pros and cons

kmeans :

- + fast. Complexity : $\mathcal{O}(nK)$
- +- dependent on the initialisation
- + easy to parallelise
- - fixed K

Outline Part 1

Clustering

Gaussian mixture models

EM algorithm

Model selection

Gaussian mixture models

$X_i \in \mathbb{R}^d$

iid

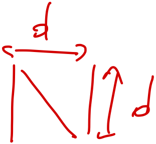
Let us first consider a random sample $(X_1, \ldots, X_n)$ where $X_i \sim \mathcal{N}(\mu, \Sigma)$ (assumed mulvariate Gaussian).

Property

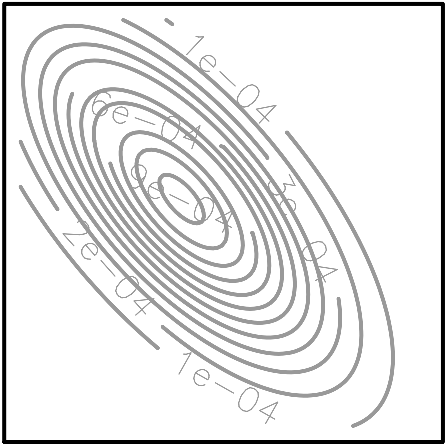
The MLE for μ and Σ are :

- $\hat{\mu} = \bar{x}$ (empirical mean)
- $\hat{\Sigma} = (1/n)\sum_{i=1}^{n}(x_i - \hat{\mu})(x_i - \hat{\mu})^\intercal$ (empirical variance-covariance matrix)

$\mu \in \mathbb{R}^d$, $\Sigma =$



Multivariate Gaussian density ($d = 2$)



Mixture of densities

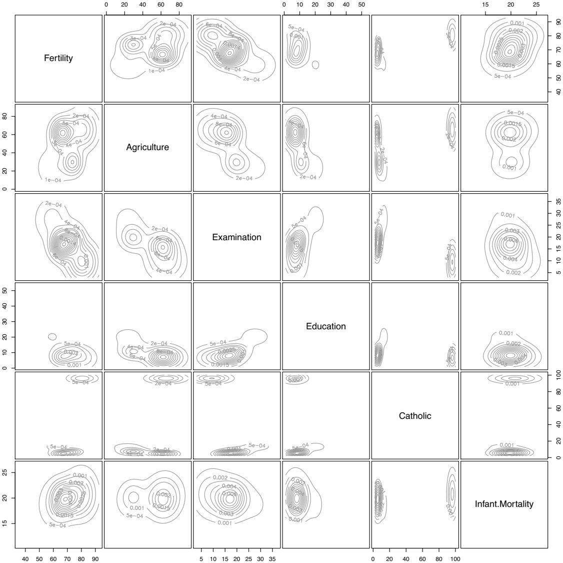


Figure: Analysis of the *swiss* data set with Mclust

GMM

* $f(x) \geq 0, \forall x$

* $\int f(x)\,dx = 1$

Def

A Gaussian mixture model with K components is defined through the density :

density function in x

$$= \frac{1}{\sqrt{(2\pi)^d |\Sigma_k|}} \exp\left\{-\frac{1}{2}(x-\mu_k)^T \Sigma_k^{-1} (x-\mu_k)\right\}$$

$$p(x|\underbrace{\pi,\theta}_{}) = \sum_{k=1}^{K} \pi_k \boxed{\mathcal{N}(x;\mu_k,\Sigma_k),}$$

where $\theta = (\mu_k, \Sigma_k)_k$ and $\pi = (\pi_1, \dots, \pi_K)^\intercal$ the vector of mixing weights lies in the standard K-simplex :

- $\pi_k \in]0,1[, \forall k \in \{1,\dots,K\}$
- $\sum_{k=1}^{K} \pi_k = 1$

$\mathcal{N}(x;\mu_k,\Sigma_k)$ denotes here the multivariate Gaussian density with parameters μ_k and Σ_k evaluated at $x \in \mathbb{R}^d$

Starting point

- as usual, we are provided with a random sample $(X_1, \ldots, X_n)$ but now $X_i \overset{\text{iid}}{\sim} \sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \Sigma_k)$
- we aim at estimating π and the component parameters θ

Property

The log-likelihood of a Gaussian mixture model is given by :

$$
\begin{aligned}
L_{(x_1,\ldots,x_n)}(\pi, \theta) &= \sum_{i=1}^{n} \log p(x_i|\pi, \theta) \\
&= \sum_{i=1}^{n} \log \left(\sum_{k=1}^{K} \pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k) \right)
\end{aligned}
$$

$$\underset{\text{random sample}}{(X_1, \ldots, X_n)} \longrightarrow \underset{\text{Realisation}}{(x_1, \ldots, x_n)}$$

Property

- the optimisation task is not trivial
- no analytical expression for the estimators of π and θ
- can rely on numerical algorithms for optimisation (conjugate gradient descent for instance) but . . .

Another point of view

$X_i \overset{iid}{\sim} \sum_{k=1}^{K} \pi_k \mathcal{N}(\mu_k, \Sigma_k), \forall i$

The Gaussian mixture model can be rewritten by introducing auxiliary variables :

1. $Z_i \sim \mathcal{M}(1, \pi), \forall i \in \{1, \dots, n\}$ iid

multinomial law

2. $X_i | Z_{ik} = 1 \sim \mathcal{N}(\mu_k, \Sigma_k)$

So

- $Z_i \in \{0, 1\}^K$ such that $\sum_{k=1}^{K} Z_{ik} = 1$
- $Z_{ik} = 1$ encodes the fact that observation i is from component k
- by definition of the multinomial law : $\mathbb{P}(Z_{ik} = 1) = \pi_k$
- the observations are now sampled conditionally on their components

Outline Part 1

Clustering

Gaussian mixture models

EM algorithm

Model selection

Def

Considering the (complete) random sample of couples $((X_1, Z_1) \dots, (X_n, Z_n))$, the complete data log-likelihood is given by :

$$L_{(x_i,z_i)_i}(\pi, \theta) = \log p((x_i, z_i)_i | \pi, \theta)$$

$$= \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} \log \left(\pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)\right)$$

$$= \sum_{k=1}^{K} \sum_{x_i \in C_k} \log \left(\pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)\right)$$

Property

The estimators of π and θ maximising the complete data log-likelihood are :

- $\hat{\pi}_k = (1/n) \sum_{i=1}^{n} z_{ik}$
- $\hat{\mu}_k = (1/n_k) \sum_{i=1}^{n} z_{ik} x_i \;=\; \frac{1}{n_k} \sum_{x_i \in C_k} x_i$
- $\hat{\Sigma}_k = (1/n_k) \sum_{i=1}^{n} z_{ik} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^{\intercal} \;=\; \frac{1}{n_k} \sum_{k=1}^{K} (x_i - \mu_k)(x_i - \mu_k)^{T}$

However :

- the z_i are unknown in practice
- this is the clustering information we are looking for
- how estimating the parameters without knowing the clusters ?
- $\rightarrow$ the expectation maximisation (EM) algorithm (DLR77)

Remark

Link with the kmeans algorithm : $\pi_k = 1/K$ and $\Sigma_k = I_d, \forall k \in \{1, \dots, K\}$

The EM algorithm relies on two fondamental properties :

Property

Given the observations (and the parameters), all the Z_i are independent :

$$p((z_i)_i|(x_i)_i, \pi, \theta) = \prod_{i=1}^{n} p(z_i|x_i, \pi, \theta)$$

Recall that Z_i is discrete so $p(z_i|x_i, \pi, \theta)$ translates into $\mathbb{P}(Z_i = z_i|x_i, \pi, \theta)$

Property

The probabilities $p(z_i|x_i, \pi, \theta)$ have analytical forms :

$$p(z_i|x_i, \pi, \theta) = \mathcal{M}(z_i; 1, \tau_i)$$

where $\underset{\uparrow}{\tau_i} = (\tau_{i1}, \dots, \tau_{iK})^\intercal$

τ_{ik} : probability that observation i is from cluster k

Property

τ_{ik} is given by :

$$\tau_{ik} = \frac{\pi_k \mathcal{N}(x_i; \mu_k, \Sigma_k)}{\sum_{l=1}^{K} \pi_l \mathcal{N}(x_i; \mu_l, \Sigma_l)}$$

- τ_{ik} is the probability for observation i to be in cluster k, given x_i and the (current) value of the parameters

Remark

The Z_i being unknown, they are treated as random vectors in the complete data log-likelihood :

$$L_{(x_i, Z_i)_i}(\pi, \theta)$$

Remark

The Z_i being unknown, the expectation of the complete data log-likelihood is computed

Property

The expectation of the complete data log-likelihood is given by :

$$\mathbb{E}_{(Z_i)_i}[L_{(x_i,Z_i)_i}(\pi,\theta)] = \sum_{i=1}^{n}\sum_{k=1}^{K}\tau_{ik}\log\left(\pi_k\mathcal{N}(x_i;\mu_k,\Sigma_k)\right),$$

where the expectation is taken with respect to the random variables $Z_i \sim \mathcal{M}(1;\tau_i)$

Property

The estimators of π and θ maximising the expected complete data log-likelihood are :

- $\hat{\pi}_k = (1/n)\sum_{i=1}^{n} \tau_{ik}$
- $\hat{\mu}_k = (1/n_k)\sum_{i=1}^{n} \tau_{ik} x_i$
- $\hat{\Sigma}_k = (1/n_k)\sum_{i=1}^{n} \tau_{ik}(x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^\intercal$

EM algorithm for Gaussian mixture models

- init : initialise the clusters (with kmeans for instance)

M compute $\hat{\pi}$ and $\hat{\theta}$ with respect to the τ_i

E compute the τ_i with respect to π and θ

- if the log-likelihood has changed (or the parameters) (no eps convergence) back to M.

Remarks

The parameters can also be initialised (instead of the clusters) through a sampling. In that case, the algorithm starts with the E step

Property

- the EM iteration does increase the log-likelihood $L_{(x_i)_i}(\pi, \theta)$
- in general, no guarantee to converge to the global maximum

Outline Part 1

Clustering

Gaussian mixture models

EM algorithm

Model selection

Model selection

In order to estimate the number K of components from the data, the EM algorithm is run for various values of K and the one maximising a criterion is chosen :

- M_K denoting the total number of (free) parameters in the model with K components
- Bayesian information criterion : $\mathrm{BIC}(K) = L_{(x_i)_i}(\hat{\pi}, \hat{\theta}) - (M_K/2)\log n$
- Akaike's information criterion : $\mathrm{AIC}(K) = L_{(x_i)_i}(\hat{\pi}, \hat{\theta}) - M_K$

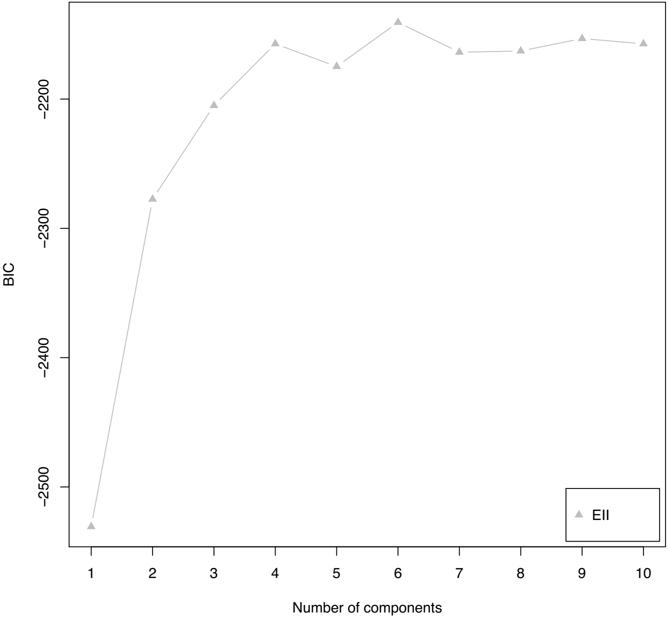


Figure: Analysis of the *swiss* data set with Mclust

References I

A.P. Dempster, N.M. Laird, and D.B. Rubin, *Maximum likelihood for incomplete data via the em algorithm*, Journal of the Royal Statistical Society **B39** (1977), 1–38.