# TP 2: Expectation-Maximisation algorithm – Importance sampling

## Exercise 1: Discrete distributions

Let $n \in \mathbb{N}^*$ and $X = \{x_1, \dots, x_n\}$ a set of $n$ distinct real numbers. Let $(p_i)_{i \in [\![1,n]\!]}$ a sequence of real numbers such that :

$$\forall i \in [\![1, n]\!], \quad p_i > 0 \qquad \text{and} \qquad \sum_{i=1}^{n} p_i = 1.$$

**1.** Explain how to generate a random variable $X$ having the discrete distribution on $X$ given by $(p_i)_{i \in [\![1,n]\!]}$ :

$$\forall i \in [\![1, n]\!], \quad \mathbb{P}(X = x_i) = p_i.$$

**2.** Write (in Python, Julia, Matlab, Octave…) the corresponding algorithm.

**3.** Generate a sequence $(X_i)_{i \in [\![1,N]\!]}$ of *i.i.d.* random variables having the same distribution as $X$ for large values of $N$. Compare the empirical distribution to the theoretical distribution of $X$. (In Python, you can use the function `numpy.histogram`).

## Exercise 2: Gaussian mixture model and the EM algorithm

A Gaussian mixture model (GMM) is useful for modelling data that comes from one of several groups: the groups might be different from each other, but data points within the same group can be well-modelled by a Gaussian distribution. The main issue is to estimate the parameters of the mixture, *i.e* to find the most likely ones. Moreover, we aim to determine if our sample follow a Gaussian mixture distribution or not.

Let consider a $n$-sample. For each individual, we observe a random variable $X_i$ and assume there is an unobserved variable $Z_i$ for each person which encode the class of $X_i$. More formally, we consider a mixture of $m$ Gaussian : let $(\alpha_1, \dots, \alpha_m) \in \mathbb{R}_+^m$ such that $\sum_{i=1}^m \alpha_i = 1$ and the following hierarchical model :

$$\forall i \in [\![1, n]\!], \quad \forall j \in [\![1, m]\!], \qquad \mathbb{P}_\theta(Z_i = j) = \alpha_j$$

and

$$\forall i \in [\![1, n]\!], \quad \forall j \in [\![1, m]\!] \qquad X_i \mid \theta, \{Z_i = j\} \ \sim \ \mathcal{N}(\mu_j, \Sigma_j)\,.$$

Unless otherwise stated, we suppose that $m$ is fixed.

**1.** Identify the parameters, denoted $\theta$, of the model and write down the likelihood of $\theta$ given the outcomes $(x_i)_{i \in [\![1,n]\!]}$ of the i.i.d $n$-sample $(X_i)_{i \in [\![1,n]\!]}$, *i.e* the p.d.f

$$\mathcal{L}(x_1, \dots, x_n; \theta) = \prod_{i=1}^{n} f_\theta(x_i)\,.$$

**2.** Sample a set of observation according to a Gaussian mixture law, with the parameters of your choice. Use the hierarchical model and the first exercise.

**3.** Implement the EM algorithm in order to estimate the parameters of this model from your observations and plot the log-likelihood over the number of iteration of the algorithm.

**4.** Are the estimated parameters far from the original ones ?

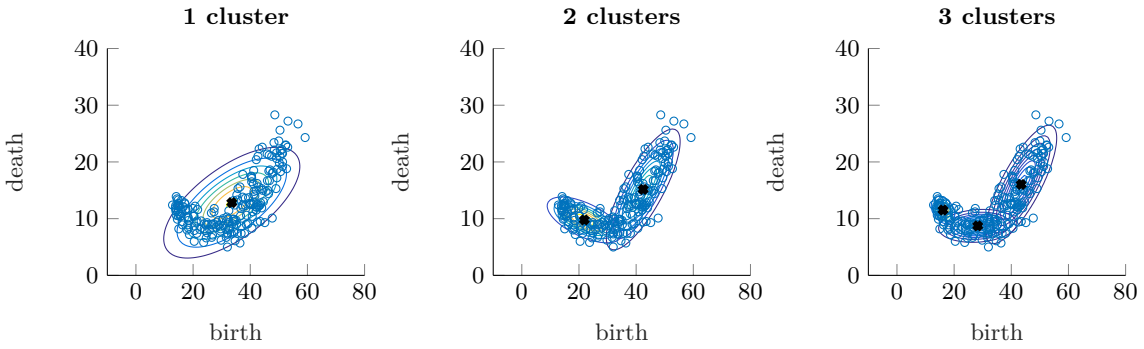



Figure 1: Importance of the number of clusters – Crude Birth/Death Rate.

In practice, determining the right number of clusters is an important issue. A good criterion is to minimize the BIC – Bayesian Information Criterion. See for example [Gir15] for more information on the BIC.

$$\widehat{m} \;=\; \underset{m\geqslant 1}{\operatorname{argmin}} \left\{ -\log \mathcal{L}(x_1,\dots,x_n;\theta) + \frac{\mathrm{df}\,(m)\log(n)}{2} \right\}$$

where df is the number of degrees of freedom of the mixture model with $m$ clusters.

**5.** **Application :** Download the data *Crude Birth/Death Rate* – See `esa.un.org/unpd/wpp/` for instance – and plot the associated scatter graph. What do you think about using a Gaussian mixture model ?

**6.** Estimate the parameters $\theta$ for different values of $m$, try to interpret them and compute the BIC. Plot the corresponding p.d.f over the scatter plot. (In Python, you can use `plt.contour`).

## Exercise 3: Importance sampling

Let $p$ be a density on $\mathbb{R}^d$, $d \in \mathbb{N}^*$. *Importance Sampling* aims at evaluating

$$\mathbb{E}_p\left[g(X)\right] = \int g(x)p(x)\,dx\,.$$

Classical Monte Carlo integration requires to generate *i.i.d.* random variables $(X_1, \dots, X_n)$ from $p$ in order to approximate $\mathbb{E}_p[g(X)]$ by $\frac{1}{n}\sum_{i=1}^n g(X_i)$. Sampling from other distributions than the original distribution $p$ *can* improve the variance of the estimator and reduce the number of samples needed.

Importance sampling is based on the following fundamental equality

$$\mathbb{E}_p[g(X)] \;=\; \int g(x)p(x)\,\mathrm{d}x \;=\; \int g(x)\frac{p(x)}{q(x)}q(x)\,\mathrm{d}x \;=\; \mathbb{E}_q\left[g(X)\frac{p(X)}{q(X)}\right]$$

which hold for any density $q$ such that $\mathrm{Supp}(g \times p) \subset \mathrm{Supp}(q)$. The density $q$ is called *importance density*. If $(X_1, \dots, X_n)$ is a sample from $q$, $\mathbb{E}_p[g(X)]$ can therefore be approximated by

$$\frac{1}{n}\sum_{i=1}^n \frac{p(X_i)}{q(X_i)}g(X_i) = \frac{1}{n}\sum_{i=1}^n \omega_i g(X_i) \qquad \text{with} \qquad \omega_i = \frac{p(X_i)}{q(X_i)}\,.$$

The $(\omega_i)_i$ are called *importance weights*. In Bayesian inference, the density $p$ might be known only up to a normalizing constant. In this case, $\mathbb{E}_p[g(X)]$ can be approximated by

$$\sum_{i=1}^n \tilde{\omega}_i g(X_i) \qquad \text{where} \qquad \tilde{\omega}_i = \frac{\omega_i}{\sum_{j=1}^n \omega_j}\,.$$

The $(\tilde{\omega}_i)_i$ are called *normalized importance weights* and do not depend on the normalizing constant of $p$.

### 3.A – Poor Importance Sampling

The performance of Importance Sampling depends on the choice of *importance density* (or *importance function*). The "best" importance density $q^*$ is chosen among a parametric family of densities $\mathcal{Q}$. Given a density $q$ on $\mathbb{R}^d$, the approximation is measured in terms of the Kullback-Leibler divergence $K(p \,\|\, q)$ given by

$$K(p \,\|\, q) = \int \log\left(\frac{p(x)}{q(x)}\right) p(x)\,\mathrm{d}x$$

therefore

$$q^* = \operatorname*{argmin}_{q \in \mathcal{Q}} K(p \,||\, q)\,. \qquad (\star)$$

The parametric family $\mathcal{Q}$ of distributions on $\mathbb{R}^d$ should be chosen large enough to allow for a close match with $p$ and be such that the optimization problem $(\star)$ is feasible. Before studying the above optimisation problem, we will illustrate the importance of choosing carefully the distribution $q$ and explore the effects of selecting a poor distribution to cover $p$.

We proceed as in [Cev08].

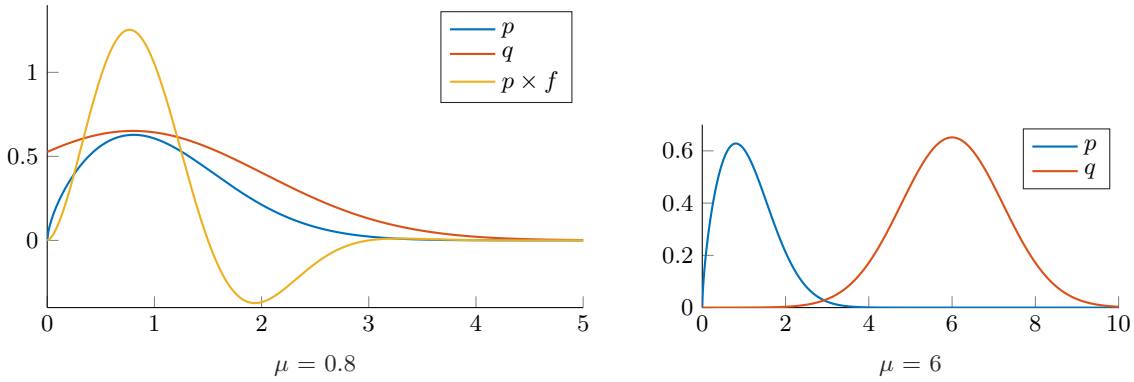
In this section, we will implement importance sampling in order to calculate the expectation of a function $f$ defined by

$$f(x) = 2\sin\left(\frac{\pi}{1.5}x\right)\,\mathbb{1}_{R^+}(x)$$

where $x$ is distributed according to a distribution similar to a $\chi$ distribution. We will use a scaled normal distribution $\mathcal{N}(0.8, 1.5)$ as our sampling distribution where the parameters are chosen so that $p(x) < k\,q(x)$ for all $x \in \mathbb{R}+$ where $k \in \mathbb{R}^+$. Let consider

$$p(x) \;=\; x^{(1.65)-1}\,\mathrm{e}^{-\frac{x^2}{2}}\;\mathbb{1}_{\mathbb{R}+}(x) \qquad \text{and} \qquad q(x) \;=\; \frac{2}{\sqrt{2\pi(1.5)}}\,\mathrm{e}^{-\frac{((0.8)-x)^2}{2(1.5)}}\;.$$

Mainly, neither $p$ nor $q$ are proper distributions here without normalization.




1. Implement a simple importance scheme for the previous functions.
   Be careful when sampling from $q$ supported on $\mathbb{R}$ to discard any samples $x < 0$ while $p$ is supported only for $x \geqslant 0$.
2. Compare the estimate and the importance weight for several sample size, $N = 10,\ 100,\ 10^3,\ 10^4$ for instance.
3. Shift the mean of $q$, $\mu = 6$, so that the centers of mass for each distribution are far apart and repeat the experiment.

### 3.B – Adaptative Importance Sampling

In the following, we choose $\mathcal{Q}$ to be the family of mixtures of $M$ Gaussian distributions on $\mathbb{R}^d$. An element of $q \in \mathcal{Q}$ is of the form

$$q(x) = \sum_{i=1}^{M} \alpha_i \varphi(x; \mu_i, \Sigma_i)$$

where, for all $i$, $\alpha_i > 0$, $\sum_{i=1}^{M} \alpha_i = 1$ and $(\mu_i, \Sigma_i)$ are mean and covariance parameters which parametrize the $i$-th Gaussian component of $q$. Because the family $\mathcal{Q}$ is a parametric family of distributions, the

optimization problem $(\star)$ can be rewritten :

$$\text{Find} \quad \theta^* = \underset{\theta=(\alpha_i,\mu_i,\Sigma_i)_{1\leq i\leq d}}{\operatorname{argmax}} \int \log\left(\sum_{i=1}^{M} \alpha_i \varphi(x;\mu_i,\Sigma_i)\right) p(x)\, \mathrm{d}x. \qquad (\star\star)$$

The solution to $(\star\star)$ cannot always be obtained in closed-form due to the density $p$ which makes the exact computation impossible. The *Population Monte Carlo* is an algorithm which aims at approximating this solution $q_{\theta^*}$.

**4.** Explain how the EM algorithm can be used to maximize the empirical criterion in step (iii). Derive the parameters update.

**Population Monte Carlo** :
The Population Monte Carlo algorithm iterates between the following steps :

(i) Choose mixture parameters $(\alpha^{(0)}, \mu^{(0)}, \Sigma^{(0)})$. This choice of parameters defines an importance density $q^{(0)}$ as follows :

$$\forall x \in \mathbb{R}^d, \quad q^{(0)}(x) = \sum_{i=1}^{M} \alpha_i^{(0)}\, \varphi\left(x\,;\, \mu_i^{(0)}, \Sigma_i^{(0)}\right).$$

(ii) This importance density is used to compute an Importance Sampling estimate of the quantity of interest. Let $(X_1, \ldots, X_n)$ be *i.i.d.* random variables generated from $q^{(0)}$. The exact criterion in $(\star)$ is approximated using normalized importance weights :

$$\sum_{i=1}^{n} \tilde{\omega}_i \log\left(\sum_{j=1}^{M} \alpha_j \varphi(X_i;\theta_j)\right).$$

(iii) New parameters $(\alpha^{(1)}, \mu^{(1)}, \Sigma^{(1)})$ are obtained by maximizing

$$\sum_{i=1}^{n} \tilde{\omega}_i \log\left(\sum_{j=1}^{M} \alpha_j \varphi(X_i;\theta_j)\right)$$

with respect to $\alpha$, $\mu$ and $\Sigma$. The new parameters define a density $q^{(1)}$.

(iv) We start again with steps from (i) to (iii) until convergence.

### 3.C – Application to a "banana"-shaped density

The target density is based on a Gaussian distribution in $\mathbb{R}^d$ with mean 0 and covariance matrix $\Sigma = \mathrm{diag}(\sigma_1^2, 1, \ldots, 1)$. This density defined on $\mathbb{R}^d$ is twisted by changing the second coordinate $x_2$ to

$x_2 + b(x_1^2 - \sigma_1^2)$. If $\Phi(\cdot; \mu, \Sigma)$ denotes the density function of the $d$-dimensional Gaussian with mean $\mu$ and covariance $\Sigma$, we have :

$$\forall x = (x_1, \dots, x_d) \in \mathbb{R}^d, \qquad p(x) = \Phi\left(x_1,\, x_2 + b(x_1^2 - \sigma_1^2),\, x_3,\, \dots,\, x_d\right).$$

If we choose $d = 10$, $\sigma_1^2 = 100$ and $b = 0.03$, $p$ results in a banana-shaped density in the first two dimensions.

**5.** Using the Adaptive Importance Sampling, write an algorithm which allows to exploring the density $p$. You may display the results for the banana-shaped density in the first two coordinates.

## References

[Bie09] Christophe Biernacki. Pourquoi les modèles de mélange pour la classification ? *La revue Modulad*, 40, 2009.

[Cev08] Volkan Cevher. Importance sampling. Lecture note, Rice University, 2008.

[Gir15] Christophe Giraud. *Introduction to High-Dimensional Statistics*. Chapman and Hall, CRC, 2015.