TP 2: Expectation-Maximisation algorithm – Importance sampling

Exercise 1: Discrete distributions

Let $K \in \mathbb{N}^*$ and $\mathcal{X} = \{v_1, \dots, v_K\}$ be a set of K distinct real numbers. Let $(\pi_k)_{k\in[\![1,K]\!]}$ be a sequence of real numbers such that:

$$\forall k \in [\![1, K]\!], \quad \pi_k > 0 \qquad \text{and} \qquad \sum_{k=1}^{K} \pi_k = 1.$$

1. Explain a method to generate a random variable Y that follows the discrete distribution on $\mathcal{X}$ given by $(\pi_k)_{k\in[\![1,K]\!]}$:

$$\forall k \in [\![1, K]\!], \quad \mathbb{P}(Y = v_k) = \pi_k.$$

2. Write in Python the corresponding algorithm

3. Generate a sequence $(Y_i)_{i\in[\![1,N]\!]}$ of N *i.i.d.* random variables having the same distribution as Y for a large value of N (e.g., $N = 10000$). Compare the empirical distribution (histogram) to the theoretical distribution of Y. (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 follows 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 encodes the class of X_i. More formally, we consider a mixture of m Gaussians: 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 θ, of the model and write down the likelihood of θ 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 iterations of the algorithm.

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

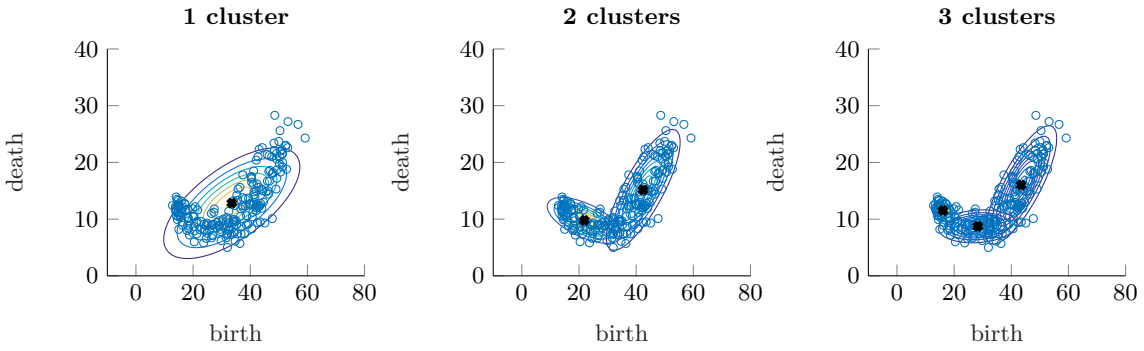

Figure 1: Importance of the number of clusters – Crude Birth/Death Rate.

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* of 2023 – See `https://population.un.org/wpp/downloads` for instance, choose Standard projection, and download "the 1950-2100, medium (gz)". Column usefull are CBR and CDR for Crude Birth/Death Rate.– and plot the associated scatter graph. What do you think about using a Gaussian mixture model ?

6. Interpret the different clusters for m = 3.

7. Estimate the parameters θ 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

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

Objective 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 holds 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

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

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

Importance distribution The performance of Importance Sampling depends on the choice of *importance density* (or *importance function*). The "best" importance density q^* is chosen so as to minimize the variance of the related Monte-Carlo estimate:

$$q^* = \operatorname*{argmin}_q \mathrm{Var}_q\left[\frac{p(X)}{q(X)}g(X)\right], \quad X \sim q(\cdot)\,. \tag{$\star$}$$

It can be shown (see for instance [RK16]) that the optimal density minimizing objective ($\star$) is given by

$$q^*(x) = \frac{g(x)p(x)}{\int g(y)p(y)\,\mathrm{d}y}\,,$$

however this expression requires the explicit use of $\int g(y)p(y)\,\mathrm{d}y$, which is the unknown quantity of interest which we are trying to find.

In order to circumvent this issue, we instead choose q among a parametric family of densities $\mathcal{Q}$ and try to find the distribution that best matches with q^*. Given a density q on $\mathbb{R}^d$, the approximation is measured in terms of the Kullback-Leibler divergence $K(q^* \,\|\, q)$ given by

$$K(\nu_1 \,\|\, \nu_2) = \int \log\left(\frac{\nu_1(x)}{\nu_2(x)}\right)\nu_1(x)\,\mathrm{d}x\,.$$

Therefore, the new problem to be solved to perform efficient Importance Sampling writes as follows:

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

The parametric family $\mathcal{Q}$ of distributions on $\mathbb{R}^d$ should be chosen large enough to allow for a close match with q^* and be such that the optimization problem $(\star\star)$ is feasible.

3.A – Poor Importance Sampling

Before studying the above optimization problem $(\star\star)$, we will illustrate the importance of choosing carefully the distribution q and explore the effects of selecting a poor distribution to cover p.

In this section, proceeding as in [Cev08], 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 density p (defined below) that is similar to a χ 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

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

Note that neither p nor q are proper distributions here without normalization.

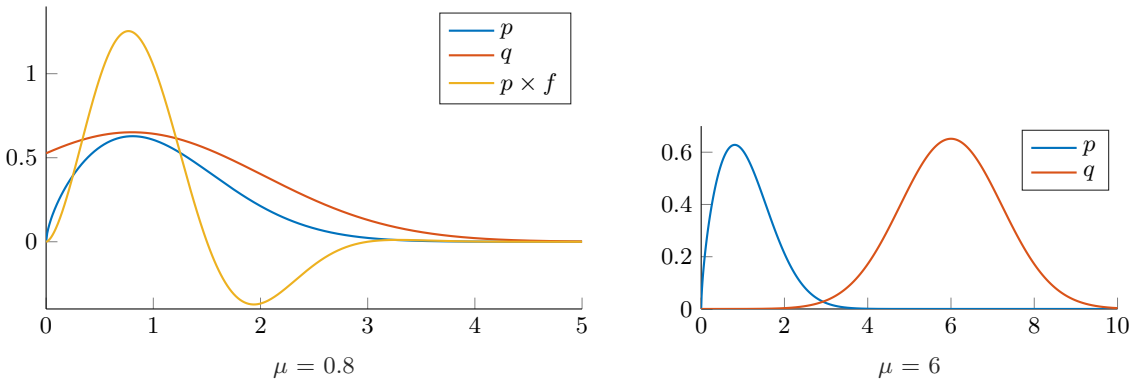

Figure 2: Distributions p and q for two choices of mean μ.

1. Implement a simple importance sampling procedure for the previous functions.
Be careful when sampling from q supported on $\mathbb{R}$ to discard any samples $x < 0$ when p is supported only for $x \geqslant 0$.

2. Compute the mean and the variance of the importance sampling estimate of $\mathbb{E}_p[f(X)]$. You can use several sample sizes, for instance $N = 10$, 100, 10^3 and 10^4.

3. Shift the mean of q, $\mu = 6$, so that the centers of mass for each distribution are far apart and repeat the experiment. Compare the importance weights for both values of μ.

3.B – Adaptative Importance Sampling

In this section, we show how Importance Sampling can be used to solve problem ($\star\star$) in a more general setting, where we wish to find the distribution q^* best approximating a distribution ν:

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

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 (μ_i, Σ_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 ($\star\star'$) can be rewritten:

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

The solution to ($\star\star\star$) cannot always be obtained in closed-form due to the density p which makes the exact computation impossible. The *Population Monte Carlo* algorithm described at page 6 is a method which aims at approximating this solution q_{θ^*}.

4. Explain how the EM algorithm can be used to maximize the empirical criterion in step (iii) of the algorithm on page 6. Derive the parameters update.

Remark In practice, the Population Monte Carlo algorithm allows solving problem ($\star\star$). Importance Sampling is thus used in two different ways in the overall process: first for Population Monte Carlo, in order to find the best distribution $q^* \in \mathcal{Q}$ approximating $p(x)g(x)/c$; then to compute the expectation of interest $\mathbb{E}_p[g(X)]$ using q^* as importance distribution.

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

The target density $\nu(x)$ 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 μ and covariance Σ, we have, up to a normalizing constant:

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

If we choose $d = 5$, $\sigma_1^2 = 1$ and $b = 0.4$, ν results in a banana-shaped density in the first two dimensions.

5. Using the Adaptive Importance Sampling, write an algorithm which allows drawing samples from the density ν. You may display the results for the banana-shaped density in the first two coordinates.

Population Monte Carlo Algorithm

The 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 $(X_1^{(0)}, \ldots, X_n^{(0)})$ be *i.i.d.* random variables generated from $q^{(0)}$. The exact criterion in $(\star\star\star)$ is approximated using normalized importance weights:

$$\sum_{i=1}^{n} \tilde{\omega}_i^{(0)} \log \left(\sum_{j=1}^{M} \alpha_j \varphi(X_i^{(0)}; \theta_j) \right) .$$

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

$$\sum_{i=1}^{n} \tilde{\omega}_i^{(0)} \log \left(\sum_{j=1}^{M} \alpha_j \varphi(X_i^{(0)}; \theta_j) \right)$$

with respect to α, μ and Σ. These new parameters define an importance density $q^{(1)}$.

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

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.

[RK16] Reuven Y Rubinstein and Dirk P Kroese. *Simulation and the Monte Carlo method*, volume 10. John Wiley & Sons, 2016.