STA4042 24-25T1 Homework 4:
Expectation-Maximisation algorithm – Importance sampling

The answer should be provided in a single Jupyter notebook whose template is already provided to you (you can use markdown cells to write answers in Latex with sign "$$").

Problem 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:

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

We have seen in course a simple method to generate a random variable X having the discrete distribution on X given by $(p_i)_{i\in[1,n]}$:

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

from a uniform distribution Unif($[0, 1]$).

Q1 Write the corresponding algorithm.

Q2 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`.)

Problem 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 us 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:

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

and

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

Unless otherwise stated, we suppose that m is fixed.

Q1 Show that the probability of X knowing Z writes:

$$p_X(x|\theta) = \sum_{j=1}^m \alpha_j \phi_j(x_i)$$

where θ contains the parameters of the model $(\alpha_j, \mu_j, \Sigma_j)_{j \in [m]}$.

Q2 Sample a set of observations according to a Gaussian mixture law, with the parameters of your choice. Use the hierarchical model and the first exercise.

We won't prove fully the validity of the EM algorithm, but just give some heuristic on its formulation. The EM algorithm seeks to find the maximum likelihood estimate of the marginal likelihood:

$$\begin{aligned}
\log\left(p(X_1 = x_1, \dots, X_n = x_n)\right) &= \sum_{i=1}^n \log(p(X = x_i|\theta)) \\
&= \sum_{i=1}^n \sum_{j=1}^m p(Z = j|X = x_i, \theta) \log\left(p(X = x_i|\theta)\right) \\
&= \sum_{i=1}^n \sum_{j=1}^m p(Z = j|X = x_i, \theta) \log\left(\frac{p(X = x_i, Z = j|\theta)}{p(Z = j|X = x_i, \theta)}\right)
\end{aligned}$$

It then relies on two steps:

- **Expectation step (E step)**: For all $j \in [p]$, replace "$p(Z = j|X = x_i, \theta)$" with its expectation:

$$\tau_{i,j} \equiv \mathbb{E}[p(Z = j|X = x_i, \theta)] = \frac{\alpha_j^{(t)} \phi_j^{(t)}(x_i)}{\sum_{j=1}^m \alpha_j^{(t)} \phi_j^{(t)}(x_i)}$$

given $X = x_i$ and the current estimates of the parameters $\theta^{(t)} = (\alpha^{(t)}, \mu^{(t)}, \Sigma^{(t)})$.

- **Maximization step (M step)**: Find the parameters $\theta^{(t+1)}$that maximize this quantity:

$$\theta^{(t+1)} = \arg\max_{\theta} \sum_{i=1}^{n} \sum_{j=1}^{m} \tau_{i,j} \log\left(\frac{p(X = x_i, Z = j|\theta)}{\tau_{i,j}}\right)$$

Q3 Explain the procedure in our particular example and compute the Argmax for each step of the EM.

Q4 Implement the EM algorithm (with, say a 100 iterations) 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. We advise you to use scipy.special.logsumexp as much as possible for stability reasons.

Q5 Are the estimated parameters far from the original ones?

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:

$$\hat{m} = \arg\min_{m \geq 1} \left\{ -\log L(x_1, \ldots, x_n; \theta) + \frac{\mathrm{df}(m) \log(n)}{2} \right\},$$

where df is the number of degrees of freedom of the mixture model with m clusters. Here it can be computed followingly:

- α_j: p parameters with the constraint $\sum_{j=1}^{p} \alpha_j = 1$, hence $p - 1$ degrees of freedom.
- μ_j: $p \times d$ independent parameters, hence $p \times d$ degrees of freedom.
- Σ_j: $p \times d^2$ parameters with the constraint $\Sigma^T = \Sigma$, hence $p \times \frac{d(d+1)}{2}$ degrees of freedom.

The total number of degrees of freedom is therefore $df = p \cdot \frac{(d+1)(d+2)}{2} - 1$.

Q6 **Application:** Download the data "pop_df" and plot the associated scatter graph. 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`.)

Problem 3: Importance sampling

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

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

Objective

Classical Monte Carlo integration requires to generate i.i.d. random variables $(X_1, \ldots, X_n)$ 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:

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

which holds for any density q such that $\mathrm{Supp}(g \times p) \subseteq \mathrm{Supp}(q)$. The density q is called *importance density*. If $(Y_1, \ldots, Y_n)$ is a sample from q, $\mathbb{E}_p[g(X)]$ can therefore be approximated by

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

The (ω_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

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

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

A – Poor Importance Sampling

Before studying the above optimization problem (1), 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

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

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 $p(x) < kq(x)$ for all $x \in \mathbb{R}^+$ where $k \in \mathbb{R}^+$. Let consider

$$\begin{aligned} p(x) &= x^{(1.65)-1}e^{-\frac{x^2}{2}}\mathbf{1}_{\mathbb{R}^+}(x) \quad \text{and} \\ q(x) &= \frac{2}{\sqrt{2\pi(1.5)}}e^{-\frac{(0.8-x)^2}{2(1.5)}}. \end{aligned}$$

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

Q1 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 $x \geq 0$.

Q2 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.

Q3 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.

B – Adaptive Importance Sampling

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:

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

We have shown that the optimal density minimizing objective $(\star)$ is given by

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

however this expression requires the explicit use of $\int g(y)p(y)\,dy$, which is exactly 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 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 given for any density $\nu_1, \nu_2 : \mathbb{R} \to \mathbb{R}$ by:

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

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

$$\hat{q}^* = \arg\min_{\nu \in Q} K(q^* \| \nu). \tag{1}$$

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

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

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 ν. Because the family Q is a parametric family of distributions, the optimization problem (1) can be rewritten:

$$\begin{aligned}\text{Find} \quad \theta^* &= \arg \min_{\theta=(\alpha_i,\mu_i,\Sigma_i)_{1\leq i\leq d}} K(q^*||\nu) \\ &= \arg \min_{\theta=(\alpha_i,\mu_i,\Sigma_i)_{1\leq i\leq d}} \int \log(q^*(x))q^*(x)dx - \int \log(\nu(x))q(x)dx \\ &= \arg \max_{\theta=(\alpha_i,\mu_i,\Sigma_i)_{1\leq i\leq d}} \int \log\left(\sum_{i=1}^{M} \alpha_i \varphi(x;\mu_i,\Sigma_i)\right) q^*(x)\,dx. \end{aligned} \tag{2}$$

The solution to (2) cannot always be obtained in closed-form due to the density q^* which makes the exact computation impossible.

Q4 (optional) Explain briefly how the EM algorithm can be used to maximize the empirical criterion in step (iii) of the Population Monte Carlo Algorithm described below. Derive the parameters update.

Remark: In practice, the Population Monte Carlo algorithm allows solving problem (1). 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 $\hat{q}^* \in Q$ approximating $p(x)g(x)/c$; then to compute the expectation of interest $\mathbb{E}_{X\sim p}[g(X)] = \mathbb{E}_{Y\sim \hat{q}^*}[g(Y)p(Y)/\hat{q}^*(Y)]$, using $\hat{q}^*$ as the importance distribution.

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 $\nu^{(0)}$ as follows:

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

(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 $\nu^{(0)}$. The exact criterion in (2) is approximated with the expression:

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

where $(\omega_i^{(0)})_{i\in[n]}$ are the normalized weights estimated as:

$$\omega_i^{(0)} = \frac{p(X_i^{(0)})g(X_i^{(0)})/\nu^{(0)}(X_i^{(0)})}{\sum_{l=1}^{n} p(X_l^{(0)})g(X_l^{(0)})/\nu^{(0)}(X_l^{(0)})}$$

(since we do not know the normalizing constant of $q^* \propto pg$, we need to empiricallly normalize it).

(iii) New parameters $(\alpha^{(1)}, \mu^{(1)}, \Sigma^{(1)})$ are obtained by maximizing with an EM procedure:

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

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

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

References

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

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