Master M2 MVA 2019/2020 - Graphical models

Homework 2

These exercises are due on or before December 31st 2019 and should be submitted on Moodle. They can be done in groups of two students. The write-up can be in English or in French. Please submit your answers as a pdf file that you will name MVA DM1 your name.pdf if you worked alone or MVA DM1 name1 name2.pdf with both of your names if you work as a group of two. Indicate your name(s) as well in the documents. Please submit your code as a separate zipped folder and name it MVA DM1 .zip if you worked alone or MVA DM1 name1 name2.zip with both of your names if you worked as a group of two. Note that your files should weight no more than 16Mb.

Classification: K-means, and the EM algorithm

Consider a mixture model, with K components, where datapoints X_i, $i = 1, \ldots, n$ have a probability p_k to be in component k: $P(Z_i = k) = p_k$, and, conditional on $Z_i = k$, $X_i \sim N(\mu_k, D_k)$, a multivariate Gaussian distribution with mean μ_k, and **diagonal** (not full) covariance matrix D_k.

1. Derive the expressions of the parameters (p_k, μ_k, D_k) at each iteration of the corresponding EM algorithm. (Explain your derivations.)
2. What may be the advantage of such a model, compared to the more standard Gaussian mixture model, where covariance matrices are full?
3. Implement the algorithm, and compare the results with:
   - K-means
   - EM for a standard Gaussian mixture (full covariance)

   for the IRIS data set, and $K = 2, 3, 4$. Note that the data set is available in scikit-learn and that these two methods are also implemented in that module. (EM for a Gaussian mixture with diagonal covariance matrices is also implemented, but please do your own implementation!)

   Please provide meaningful plots to help the comparison, i.e. scatter plots for every pair of dimensions, with clusters represented with different colors, and even better with a sur-imposed ellipsis. (What should the ellipses represent?)

4. In which situations K-means is going to be significantly outperformed by the two EM algorithms discussed above? (Think about the shape of the clusters for instance.) Construct a synthetic dataset to illustrate this point (show that K-means fails to capture some of the clusters found by EM).

Graphs, algorithms and Ising

To avoid over or underflow, it is often recommended to store and compute small quantities (such as probabilities, densities, etc.) on the *log scale*. To multiply two quantities stored on the log-scale, we just add their logs. To add two such quantities, we use the log-sum trick: if $a \geq b$, compute $\log(e^a + e^b)$ as: $a + \log(1 + e^{b-a})$; otherwise swap a and b.

1. Implement the sum-product algorithm for an undirected chain, using this trick, in order to compute all the forward and backward messages. (Explain how you represent the input of the algorithm, i.e. the functions ψ_i and $\psi_{i,i+1}$).

The Ising model assumes n binary variables $X_1, \ldots, X_n$, which are jointly distributed as follows:

$$p(x_1, \ldots, x_n) = \frac{1}{Z(\alpha, \beta)} \exp\{\alpha \sum_i x_i + \beta \sum_{i \sim j} \mathbb{I}_{x_i = x_j}\} \tag{1}$$

where the relation $i \sim j$ means that i and j are "neighbours". Specifically, each variable is associated to a point in a 2D grid, of size $h \times w$ (height times width), and two variables are neighbours if they are at distance one on that grid (i.e. immediately to the left, right, up or down).

2. For $w = 10$, $h = 100$, $\alpha = 0$, use your implementation from point 1 to compute *exactly* $Z(\alpha, \beta)$ as a function of β, and plot it. (Hint: recall the idea behind the junction tree algorithm).

3. Implement loopy belief propagation to obtain a faster approximation of $Z(\alpha, \beta)$. (Explain.) For which values of β the approximation error gets larger?