Introduction to Probabilistic Graphical Models
Homework 2

Umut Şimşekli
Télécom ParisTech, Université Paris-Saclay, Paris, France
`umut.simsekli@telecom-paristech.fr`

Instructions: (please read carefully)

1. This homework can be done in groups of **maximum 3** people. I personally encourage group work, try to form groups.
2. Prepare your report in English by using LaTeXor an ipython (jupyter) notebook. Do not submit scanned papers.
3. Put all your files (code and/or report) in a zip file: *surname_name_hw2.zip* and upload it to moodle before the deadline (check moodle for the deadline). Late submissions will not be accepted.

MAP estimation on NMF

Consider the following probabilistic non-negative matrix factorization (NMF) model: (for $f = 1, \dots, F$, $n = 1, \dots, N$, $k = 1, \dots, K$)

$$\begin{aligned} w_{fk} &\sim \mathcal{G}(w_{fk}; \alpha_w, \beta_w) \\ h_{kn} &\sim \mathcal{G}(h_{kn}; \alpha_h, \beta_h) \\ v_{fn}|w_{f,:}, h_{:,n} &\sim \mathcal{PO}(v_{fn}; \sum_{k=1}^{K} w_{fk} h_{kn}) \end{aligned}$$

where $\mathcal{G}$ and $\mathcal{PO}$ denote the gamma and the Poisson distributions, respectively. Here, $w_{f,:}$ denotes the collection $\{w_{fk}\}_{k=1}^{K}$. We define $h_{:,n}$ similarly.

Question 1

Derive an Expectation-Maximization algorithm for finding the maximum a-posteriori estimate (MAP), defined as follows:

$$(W^\star, H^\star) = \arg\max_{W,H} \log p(W, H|V), \tag{1}$$

where V, W, and H are the matrices with the form: $V = [v_{fn}]_{f,n}$, $W = [w_{fk}]_{f,k}$, $H = [h_{kn}]_{k,n}$. Define auxiliary latent random variables (i.e. data augmentation) if necessary. You need to end up with some multiplicative update rules similar to the ones that we have seen during the lectures. Show all your work.

Hint: Let us consider the simplified probabilistic model

$$\begin{aligned} \theta &\sim p(\theta) \\ x|\theta &\sim p(x|\theta) \end{aligned}$$

where θ is the latent random variable and x is the observation. We are interested in the MAP estimate:

$$\begin{aligned}\theta^\star &= \arg\max_\theta \log p(\theta|x) \\ &= \arg\max_\theta \log \frac{p(x|\theta)p(\theta)}{p(x)} \\ &= \arg\max_\theta \big(\log p(x|\theta) + \log p(\theta)\big),\end{aligned}$$

or equivalently:

$$\theta^\star = \arg\max_\theta \log p(x, \theta)$$

The EM algorithm for this problem can be defined as follows: (by following the same derivations that we did in the class)

$$\begin{aligned}&\text{E-Step:} \qquad \mathcal{L}_t(\theta) = \mathbb{E}[\log p(x, z, \theta)]_{p(z|x,\theta_t)} \\ &\text{M-Step:} \qquad \theta_{t+1} = \arg\max_\theta \mathcal{L}_t(\theta),\end{aligned}$$

where t denotes the iteration number. Notice that the main difference between the maximum likelihood case (i.e. the case that we covered in the lectures) is the E-Step (obviously the result of the M-Step will also be different due to the E-Step).

Question 2

In this part, our aim will be to analyze a dataset of face images. We will use the AT&T Database of Faces [1]. This dataset contains face images from 40 distinct subjects, where there are 10 images for each subject. In total there are 400 images in the dataset, where the size of each image is 92 x 112 pixels, with 256 gray levels per pixel. A preview image of the dataset is given in Figure 1.

In this experiment, we will 'vectorize' all the images in the dataset and concatenate these vectors in order to represent the whole dataset as a matrix. Finally, we obtain an observed matrix V of dimensions $F = 92 \times 112$ and $N = 400$. We provide the matrix V obtained from the dataset in the file "attfaces.mat"[1]. You are also provided a Matlab script 'visualize_data.m' for loading and visualizing the dataset[2].

The goal of the experiment is to obtain a 'parts-based-representation' of faces by learning an NMF model on this dataset. Our hope is that, when we estimate the matrices W and H by using V, the columns of W will correspond to some images that only contain a particular part of a face (remember that each column of W is a vector of size $F = 92 \times 112$. Therefore you can reshape this vector to obtain an image (matrix) of size 92×112.).

1. Implement the EM algorithm that you developed in Question 1.
2. Run the algorithm on the face dataset. Set $K = 25$, $\alpha_w = \alpha_h = 1$. Try different values for β_w and β_h. Visualize the columns of estimated W matrices (you can use the script 'visualize_data.m' for visualizing the columns of W after estimating it). What do you observe when you change the parameters?
3. Run the algorithm with $K = 25$, $\beta_w = \beta_h = 1$. Try different values for α_w and α_h. Visualize the columns of estimated W matrices. What do you observe when you change the parameters?

I am expecting you to obtain figures similar to the Figure 5 (top) of [2].

[1] The dataset can also be downloaded from `http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html`.

[2] You are free to code in python or Matlab.

References

[1] Ferdinando S Samaria and Andy C Harter, "Parameterisation of a stochastic model for human face identification," in *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on.* IEEE, 1994, pp. 138–142.

[2] Ali Taylan Cemgil, "Bayesian inference for nonnegative matrix factorisation models," *Computational Intelligence and Neuroscience*, vol. 2009, 2009.

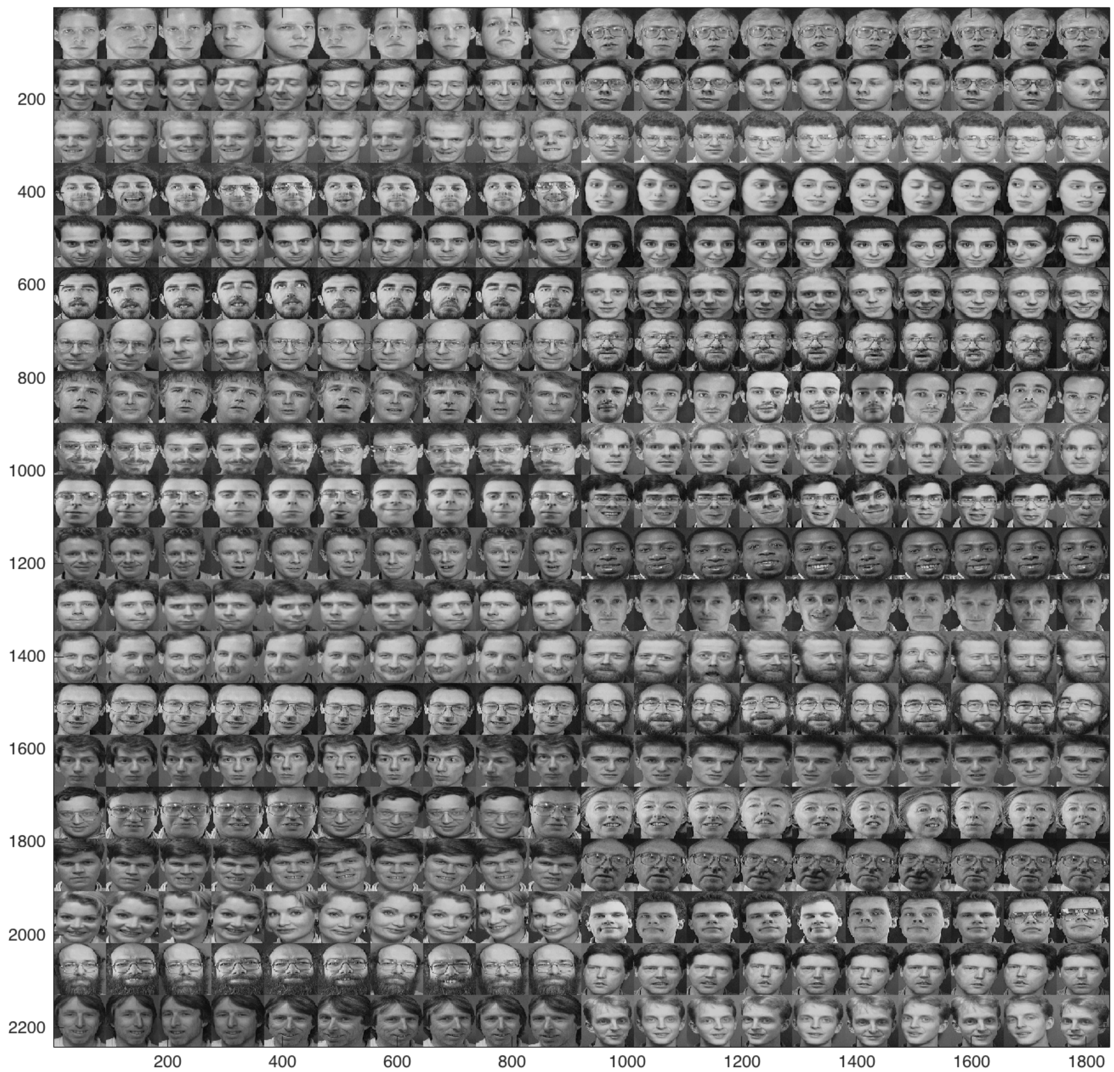

Figure 1: Preview of the dataset.