Introduction to probabilistic graphical model
Mini Project

Done by BEJAOUI Ahmed

18 November 2018

1 MAP estimation on NMF:

Consider the following probabilistic non-negative matrix factorization (NMF) model: ($\forall f = \{1,..,F\}, n = \{1,..,N\}, k = \{1,..,K\}$) Let $V = (v_{fn})_{f,n}$, $W = (w_{fk})_{f,k}$, $H = (h_{kn})_{k,n}$ with $w_{f,:} = \{w_{fk}\}_{k=1}^{K}$ and $h_{:,n} = \{h_{kn}\}_{k=1}^{K}$

$$w_{fk} \sim \mathcal{G}(w_{fk}, \alpha_w, \beta_w)$$

$$h_{kn} \sim \mathcal{G}(h_{kn}, \alpha_h, \beta_h)$$

$$v_{fn}|w_{f,:}, h_{:,n} \sim \mathcal{PO}(v_{fn}, \hat{V}_{f,n}) \; where \; \hat{v}_{f,n} = \sum_{k=1}^{K} w_{f,k} h_{k,n}$$

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

2 Question 1 :

1. Draw the directed graphical model :
   Answer:

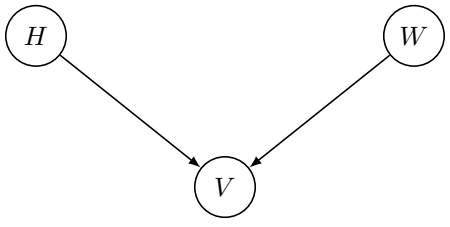

Figure 1: Directed graphical model using matrix representation

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

$$(W^*, H^*) = \underset{W,H}{\operatorname{argmax}} \log p(W, H|V)$$

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 (in that case draw the new graphical model). You need to end up with some 'multiplicative update rules'.
Show all your work.

Answer:

To make the calculations more easier and to make the problem more comprehensible, let's introduce the latent variable $\theta = (W, H)$, Then

$$v_{fn}|\theta \sim \mathcal{PO}(v_{fn}, \hat{v}_{f,n}) \; where \; \hat{v}_{f,n} = \sum_{k=1}^{K} w_{f,k} h_{k,n}$$

We can also re-write then the maximimum a-posteriori estimate (MAP) as follows:

$$(\theta^*) = \underset{\theta}{\operatorname{argmax}} \log p(\theta|V)$$

Finding the parameters θ^* is usually hard to obtain so we'll follow a few step in order to solve this problem.

- **First step :** We will introduce a hidden random variable $S = (S_{fnk})_{1 \leq f \leq F, 1 \leq n \leq N, 1 \leq k \leq K}$:

  $\forall f = \{1, .., F\}, n = \{1, .., N\}, k = \{1, .., K\}$ $S_{f,n,k}$ is defined as follows:

  $$\begin{cases} S_{f,n,k}|\theta \sim \mathcal{PO}(s_{f,n,k}, w_{f,k} h_{k,n}) \\ \\ v_{f,n}|S_{f,n} \sim \delta(v_{f,n} - \sum_{k=1}^{K} s_{f,n,k}) \; where S_{f,n} = (s_{f,n,1}, .., s_{f,n,K}) \end{cases}$$

  we obtained the following graph:

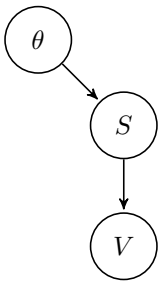

Figure 2: the final directed graphical model after introducing the hidden variable S and the latent variable θ

Now, we have to make sure that the $P(S|V,\theta)$ *is tractable*
According to bayes's formula, We have:

$$\begin{aligned} P(S|V,\theta) &= \prod_{f=1}^{F}\prod_{n=1}^{N} P(S_{f,n}|v_{f,n},\theta) \\ &= \prod_{f=1}^{F}\prod_{n=1}^{N} \frac{P(V_{f,n}|S_{f,n},\theta).P(S_{f,n}|\theta)}{P(v_{f,n}|\theta)} \end{aligned}$$

we have

$$\begin{cases} P(S_{f,n}|\theta) = \prod_{k=1}^{K} P(s_{f,n,k}|\theta) \\ P(v_{f,n}|\theta) = \frac{(\hat{v}_{f,n})^{v_{fn}}}{v_{fn}!}\exp\left(-\hat{v}_{fn}\right) \\ P(v_{f,n}|S_{f,n},\theta) = 1 \ (case\ where\ v_{f,n} = \sum_{k=1}^{K} s_{f,n,k}) \end{cases}$$

then :

$$P(S_{f,n}|v_{fn},\theta) = \frac{v_{fn}!}{\prod_{k=1}^{K} s_{fnk}!}\prod_{k=1}^{K}(\frac{w_{f,k}h_{k,n}}{\hat{v}_{fn}})^{s_{fnk}}$$

and consequently : $S_{fn}|v_{f,n},\theta \sim \mathcal{M}(v_{fn},\pi_1,\pi_2,..,\pi_k)$ *where* $\pi_k = \frac{w_{f,k}h_{k,n}}{\hat{v}_{fn}} \forall k \in \{1..K\}$

$$\begin{aligned} P(S|V,\theta) &= \prod_{f=1}^{F}\prod_{n=1}^{N} P(S_{f,n}|v_{f,n},\theta) \\ &= \prod_{f=1}^{F}\prod_{n=1}^{N}(\frac{v_{fn}!}{\prod_{k=1}^{K} s_{fnk}!}\prod_{k=1}^{K}(\frac{w_{f,k}h_{k,n}}{\hat{v}_{fn}})^{s_{fnk}}) \end{aligned}$$

We find then that $(S|V,\theta)$ is the product of mutinomial law.

- **Second step: E-step**
  Now we'll have to compute the expression of

$$\mathcal{L}_t(\theta) = \mathbf{E}(\log(P(S,V,\theta))_{P(S|\theta^t)}$$

We have the following result

$$\begin{cases} P(S,V,\theta) = P(\theta).P(S|\theta).P(V|S,\theta) = P(\theta).P(S|\theta) \\ \log P(\theta) = \log P(W) + \log P(H) \end{cases}$$

Or $w_{fk} \sim \mathcal{G}(W_{f,k}, \alpha_w, \beta_w)$ then

$$\begin{aligned} \log P(W) &= \sum_{f=1}^{F}\sum_{k=1}^{K} log(P(w_{f,k})) \\ &= \sum_{f=1}^{F}\sum_{k=1}^{K}(\alpha_w - 1)\log(w_{f,k}) + \alpha_w \log(\beta_w) - \beta_w w_{f,k} \\ &\sim \sum_{f=1}^{F}\sum_{k=1}^{K}(\alpha_w - 1)\log(w_{f,k}) - \beta_w w_{f,k} \end{aligned}$$

By analogy

$$\log(P(H)) \sim \sum_{n=1}^{N}\sum_{k=1}^{K}(\alpha_h - 1)\log(h_{k,n}) - \beta_h h_{k,n}$$

and finally

$$\begin{aligned} \log(P(S|\theta)) &= \log(\prod_{f=1}^{F}\prod_{n=1}^{N}\prod_{k=1}^{K} P(S_{f,n,k}|\theta)) \\ &= \sum_{f=1}^{F}\sum_{n=1}^{N}\sum_{k=1}^{K}\log(P(s_{f,n,k}|\theta)) \\ &= \sum_{f=1}^{F}\sum_{n=1}^{N}\sum_{k=1}^{K} s_{f,n,k}\log(w_{f,k}h_{k,n}) - \log(s_{f,n,k}!) - w_{f,k}h_{k,n} \\ &\sim \sum_{f=1}^{F}\sum_{n=1}^{N}\sum_{k=1}^{K} s_{f,n,k}\log(w_{f,k}h_{k,n}) - w_{f,k}h_{k,n} \end{aligned}$$

We must also recall that $S_{fn}|v_{f,n}, \theta \sim \mathcal{M}(v_{fn}, \pi_1, \pi_2, .., \pi_k)$ then

$$E(s_{f,n,k})_{P(S,V|\theta^t)} = v_{f,n}\pi_k = v_{f,n}\frac{w_{f,k}^t h_{k,n}^t}{\hat{v}_{f,n}^t}$$

Now we are able to compute $\mathcal{L}_t(\theta)$

$$\begin{aligned}
\mathcal{L}_t(\theta) &= \mathbf{E}(\log(P(S,V,\theta))_{P(S,V|\theta^t)} \\
&= \mathbf{E}(\log(P(S|\theta))_{P(S,V|\theta^t)} + log(P(W)) + \log(P(H)) \\
&\sim \sum_{f=1}^{F}\sum_{n=1}^{N}\sum_{k=1}^{K}(\mathbf{E}(s_{f,n,k})_{P(S,V|\theta^t)}\log(w_{f,k}h_{k,n}) - w_{f,k}h_{k,n}) + \log(P(W)) + \log(P(H)) \\
&\sim \sum_{f=1}^{F}\sum_{n=1}^{N}\sum_{k=1}^{K}(v_{f,n}\frac{w^t_{f,k}h^t_{k,n}}{\hat{v}^t_{f,n}}\log(w_{f,k}h_{k,n}) - w_{f,k}h_{k,n}) + \log(P(W)) + \log(P(H)) \\
&\sim \sum_{f=1}^{F}\sum_{n=1}^{N}\sum_{k=1}^{K}(v_{f,n}\frac{w^t_{f,k}h^t_{k,n}}{\hat{v}^t_{f,n}}\log(w_{f,k}h_{k,n}) - w_{f,k}h_{k,n}) + \sum_{f=1}^{F}\sum_{k=1}^{K}(\alpha_w - 1)\log(w_{f,k}) \\
&- \beta_w w_{f,k} + \sum_{n=1}^{N}\sum_{k=1}^{K}(\alpha_h - 1)\log(h_{k,n}) - \beta_h h_{k,n}
\end{aligned}$$

- **Final step: The M-step**
  We'll determine now the expression of $\theta^{t+1} = \{W^{t+1}, H^{t+1}\}$ using the M-step at each iteration. We are going to solve 2 optimization problem

$$\begin{cases} \textbf{Pb1: } W^{t+1} = \text{argmax}_W\, \mathcal{L}_t(W, H^t)\ (i.e\ we\ fix\ H^t\ ) \\ \textbf{Pb2: } H^{t+1} = \text{argmax}_H\, \mathcal{L}_t(W^t, H)\ (i.e\ we\ fix\ W^t\ ) \end{cases}$$

to solve **Pb1:** we will do the derivation with respect to each single variable $w_{f,k}$, we find:

$$\begin{aligned}
\frac{\partial}{\partial w_{fk}}\mathcal{L}_t(W,H^t)_{|w_{fk}=w^{t+1}_{fk}} = 0 \iff & \sum_{n=1}^{N}(v_{f,n}\frac{w^t_{f,k}h^t_{k,n}}{\hat{v}^t_{f,n}}\frac{1}{w^{t+1}_{f,k}}) - h_{k,n}) + \\
& \frac{(\alpha_w - 1)}{w^{t+1}_{f,k}} - \beta_w = 0 \\
\iff & w^{t+1}_{fk} = \frac{(\alpha_w - 1) + \sum_n (v_{fn}\frac{w^t_{fk}h^t_{kn}}{\hat{v}^t_{fn}})}{\beta_w + \sum_k h^t_{kn}} \forall f,k
\end{aligned}$$

We have $W^{t+1} = (W^{t+1}_{f,k})_{1\le f\le F\},1\le k\le K}$ Then

$$W^{t+1} = \frac{(\alpha_w - 1) * 1_{F,K} + W^t \odot ((V \oslash \hat{V})(H^t)^T}{\beta_w * \mathbf{1_{F,K}} + \mathbf{1_{F,N}}(H^t)^T}$$

where $\odot$ and $\oslash$ are symbols that denote respectively the element-wise

product and the element-wise division and :

$$\mathbf{1_{L,M}} = \begin{bmatrix} 1 & 1 & 1 & . & . & . & 1. & 1. & 1 \\ 1 & 1 & 1 & . & . & . & 1. & 1. & 1 \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ 1 & 1 & 1 & . & . & . & 1. & 1. & 1 \end{bmatrix} \in R^{L,M} \forall L, M$$

We solve the **Pb2:** with the same method. By analogy , we obtain

$$h_{kn}^{t+1} = \frac{(\alpha_h - 1) + \sum_f (v_{fn} \frac{w_{fk}^t h_{kn}^t}{\hat{v}_{fn}^t})}{\beta_h + \sum_f w_{fk}^t}$$

We have $H^{t+1} = (H_{k,n}^{t+1})_{1 \leq K \leq K\}, 1 \leq n \leq N}$ *and then*

$$H^{t+1} = \frac{(\alpha_h - 1) * \mathbf{1_{K,N}} + H^t \odot ((W^t)^T (V \oslash \hat{V})}{\beta_h * \mathbf{1_{K,N}} + (W^t)^T}$$

3 Question 2:

1. Part 1: Implement the EM algorithm that you developed in Question 1

   Answer: see the notebook

2. Run the algorithm on the face dataset. Set K = 25, $\alpha_w = 1$ $\alpha_w = 1$. Try different values for β_w β_h. Visualize
   answer: see the notebook

   - For small values of β_w β_H, we are able to see faces that are a bit noisy (not clear). However, we are unable to isolate the facial features (eyes, hair, nose, etc.). This is due to the fact that the matrix W and H are dense. And therefore we can say that the extracted features are not exploitable.
   - For large values of β_w β_H, the matrix W and H will be sparse and we are able then to detect some facial features. For instance, when we choose $\beta_w = 100$ and $\beta_H = 100$, the images are darker than the previous examples and we can see that we can detect in some images particular features like the eyebrow, the mouth, the eyes,etc..

   - We can see from the graph below that when we increase the value of β, the gamma distribution tends to have a thiny peak close to zero that's why the majority of the pixel will be dark and then the obtained matrix w will be sparse as majority of its values(pixels) are close to zero

3. Run the algorithm on the face dataset. Set K = 25, $\beta_w = 1$ $\beta_h = 1$. Try different values for α_w α_h. Visualize

   answer: see the notebook
   - For $\alpha_w = 0.01$, the images are not exploitable, they are very noisy and not clear.
   - we can also remark from the figures that the sparsity of the matrix W depends mainly on the value of α_h. In fact, when we increase the value of α_h, the matrix we will become sparser and therefore we obtain a darker image)
   - As we increase the value of α_w and α_h the images become clearer and we are able to detect some facial features. The best result is achieved for $\alpha_w = 10$ and $\alpha_h = 100$

4. Now try changing the number of components K. What do you observe?

   answer: see the notebook
   - The choice of K results in a compromise between Data fitting and Model complexity.
   In fact, when K is big, it will lead to a better approximation (i.e $V \sim \hat{v} = W.H$). But in return, the complexity of our model will increase (i.e. the matrix W and H will be larger and harder to estimate) and it might also induce a problem of interpretability when visualizing the facial features of the face represented by the column of W. And when k is too low, our model will be very simple and it will lead to a bad approximation of W. Moreover, we will get a lot of facial features combined in each image.

   - In our case, the best parameter is obtained for $k = 20$. In fact, we can detect a particular part of facial features in every image