CSC411H1 FALL 2018

HW6

LINO LASTELLA 1001237654

Part 1

1.

$$\frac{\partial}{\partial \pi_k}\left[\sum_{i=1}^{N}\sum_{k=1}^{K} r_k^{(i)}\left[\log P_h(z^{(i)}=k) + \log p(x^{(i)} \mid z^{(i)}=k)\right] + \right.$$

$$\left. + \log p(\pi) + \log p(\theta)\right] =$$

$$= \frac{\partial}{\partial \pi_k}\left[\sum_{i=1}^{N}\sum_{k=1}^{K} r_k^{(i)}\left[\log \pi_k + \log\left(\prod_{j=1}^{D} \theta_{k,j}^{x_j^{(i)}} (1-\theta_{k,j})^{1-x_j^{(i)}}\right)\right] + \right.$$

$$\left. + \log\left(\prod_{k=1}^{K} \pi_k^{\alpha_k - 1}\right) + \log\left(\prod_{k=1}^{K}\prod_{j=1}^{D} \theta_{k,j}^{\beta-1}(1-\theta_{k,j})^{\beta-1}\right)\right]$$

$$= \sum_{i=1}^{N} r_k^{(i)} \cdot \frac{1}{\pi_k} + \frac{\partial}{\partial \pi_k}\left[\sum_{k=1}^{K}(\alpha_k - 1)\log \pi_k\right]$$

$$= \sum_{i=1}^{N} r_k^{(i)} \cdot \frac{1}{\pi_k} + \frac{\alpha_k - 1}{\pi_k} \qquad \text{(Partial w.r.t. } \pi_k\text{)}$$

Similarly,

$$\frac{\partial}{\partial \theta_{k,j}} \left[\sum_{i=1}^{N} \sum_{k=1}^{K} r_k^{(i)} \left[\log P_k(z^{(i)}=k) + \log p(x^{(i)} \mid z^{(i)}=k) \right] + \right.$$

$$\left. + \log p(\pi) + \log p(\theta) \right] = \{\text{Same first step as before}\}$$

$$= \frac{\partial}{\partial \theta_{k,j}} \left[\sum_{i=1}^{N} \sum_{k=1}^{K} r_k^{(i)} \sum_{j=1}^{D} \left[x_j^{(i)} \log \theta_{k,j} + (1-x_j^{(i)}) \log(1-\theta_{k,j}) \right] + \right.$$

$$\left. + \sum_{k=1}^{K} \sum_{j=1}^{D} \left[(a-1) \log \theta_{k,j} + (b-1) \log(1-\theta_{k,j}) \right] \right] =$$

$$= \sum_{i=1}^{N} r_k^{(i)} \left[\frac{x_j^{(i)}}{\theta_{k,j}} + \frac{x_j^{(i)} - 1}{1-\theta_{k,j}} \right] + \frac{a-1}{\theta_{k,j}} + \frac{1-b}{1-\theta_{k,j}}$$

(Partial w.r.t. $\theta_{k,j}$)

In the second case we can just set the partial equal to zero and solve for $\theta_{k,j}$.

$$0 = \frac{\sum_{i=1}^{N} r_k^{(i)} x_j^{(i)}}{\theta_{k,j}} + \frac{\sum_{i=1}^{N} (x_j^{(i)} - 1) r_k^{(i)}}{1-\theta_{k,j}} + \frac{a-1}{\theta_{k,j}} + \frac{1-b}{1-\theta_{k,j}}$$

$$\rightarrow (1-\theta_{k,j}) \left[\sum_{i=1}^{N} r_k^{(i)} x_j^{(i)} + a - 1 \right] = \theta_{k,j} \left(-1 + b + \sum_{i=1}^{N} r_k^{(i)} (1 - x_j^{(i)}) \right)$$

$$\rightarrow 0 = \theta_{k,j} \left[\left(-1 + b + \sum_{i=1}^{N} r_k^{(i)} (1 - x_j^{(i)}) \right) + \sum_{i=1}^{N} r_k^{(i)} x_j^{(i)} + a - 1 \right]$$

$$- \sum_{i=1}^{N} r_k^{(i)} x_j^{(i)} + 1 - a$$

$$\rightarrow \theta_{k,j} = \frac{\sum_{i=1}^{N} r_k^{(i)} x_j^{(i)} - 1 + a}{\left[\sum_{i=1}^{N} r_k^{(i)} x_j^{(i)} + \sum_{i=1}^{N} r_k^{(i)}\right] + b - 1 + \sum_{i=1}^{N} r_k^{(i)} x_j^{(i)} + a - 1}$$

$$= \frac{a - 1 + \sum_{i=1}^{N} r_k^{(i)} x_j^{(i)}}{a + b - 2 + \sum_{i=1}^{N} r_k^{(i)}} .$$

On the other hand, for π_k we need to apply Lagrange multiplier theorem:

$$\frac{\sum_{i=1}^{N} r_k^{(i)}}{\pi_k} + \frac{a_k - 1}{\pi_k} + \frac{\partial}{\partial \pi_k}\left[\lambda \sum_{k=1}^{K}(\pi_k - 1)\right] = 0$$

$$\rightarrow \frac{\sum_{i=1}^{N} r_k^{(i)} + \alpha - 1}{\pi_k} = -\lambda \qquad \left[\text{Since all } a_k\text{'s are equal, call them } \alpha\right].$$

$$\rightarrow \pi_k = \frac{\sum_{i=1}^{N} r_k^{(i)} + \alpha - 1}{-\lambda} .$$

Since $\sum_{k'=1}^{K} \pi_{k'} = 1$, $\sum_{k'=1}^{K} \sum_{i=1}^{N} r_{k'}^{(i)} + \alpha - 1 = \lambda$.

therefore, $\pi_k = \frac{\sum_{i=1}^{N} r_k^{(i)} + \alpha - 1}{\sum_{k'=1}^{K}\left[\sum_{i=1}^{N} r_{k'}^{(i)} + \alpha - 1\right]}$.

Part 2

1.

$$P_n(z=k \mid x_{obs}) = \frac{P(x_{obs} \mid z=k)\, P(z=k)}{P(x_{obs})} \quad \text{[Bayes' rule]}$$

$$= \frac{P(x_{obs} \mid z=k)\, P(z=k)}{\sum_{k'=1}^{K} P(x_{obs} \mid z=k)\, P(z=k)} \quad \text{[Law of total pr.]}$$

$$= \frac{\pi_k \cdot \prod_{j=1}^{D} \theta_{k,j}^{x_j^{(i)}} (1-\theta_{k,j})^{1-x_j^{(i)}}}{\sum_{k'=1}^{K} \left[\prod_{j=1}^{D} \theta_{k',j}^{x_j^{(i)}} (1-\theta_{k',j})^{1-x_j^{(i)}} \cdot \pi_{k'} \right]}$$

Note: $x_j^{(i)}$ should be $x_{obs\,j}^{(i)}$ everywhere.

Part 3.

1.

If instead of a Beta prior we used a Uniform prior, the MAP learning algorithm would have the problem that it might assign zero prob. to images in the test set because that image might not have appeared consistently in the training set. For instance, if a pixel is always zero in the training set, but 1 in the test set, then the algorithm would not be able to identify the proper cluster to maximize in the M-step.

2.

I think model Part 2 performs better because the partial observations give a closer approximation of the clusters for each step

3.

No, the model is just outlining that predicting a 1 is far easier than predicting an 8. If we sample from its distribution we won't get more 1's than 8's because they are all equally likely to be sampled.