CSC321: Project4

Due on Tuesday, April 4, 2017

Xiaodi Lu

April 5, 2017

Part 1

1.
We know the functions after we put them into Lagrangian for π function is:

$\mathcal{L}_\pi = \sum_{i=1}^{N}\sum_{k=1}^{K} r_k^{(i)} log\pi_k + \lambda(1 - \sum_{k=1}^{K}\pi_k) + \sum_{k=1}^{K}(a_k - 1)log\pi_k$

Setting the derivative function of this to 0, would give us the maximum:

$\frac{\partial \mathcal{L}}{\partial \pi_k} = \frac{\sum_{i=1}^{N} r_k^{(i)}}{\pi_k} - \lambda + \frac{a_k-1}{\pi_k} = 0$

thus:

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

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

$\sum_{k=1}^{K}\pi_k = \sum_{k=1}^{K}\frac{a_k-1+\sum_{i=1}^{N} r_k^{(i)}}{\lambda}$

$= \frac{\sum_{k=1}^{K}(a_k-1+\sum_{i=1}^{N} r_k^{(i)})}{\lambda} = 1$

$\lambda = \sum_{k=1}^{K}(a_k - 1 + \sum_{i=1}^{N} r_k^{(i)})$

$\frac{a_k-1+\sum_{i=1}^{N} r_k^{(i)}}{\pi_k} = \sum_{k'=1}^{K}(a_k' - 1 + \sum_{i=1}^{N} r_{k'}^{(i)})$

$\pi_k = \frac{a_k-1+\sum_{i=1}^{N} r_k^{(i)}}{\sum_{k'=1}^{K}(a_{k'}-1+\sum_{i=1}^{N} r_{k'}^{(i)})}$

Which means that the π update would be:

$\pi_k \leftarrow \frac{a_k-1+\sum_{i=1}^{N} r_k^{(i)}}{\sum_{k'=1}^{K}(a_{k'}-1+\sum_{i=1}^{N} r_{k'}^{(i)})}$

Then similar for θ:

$\mathcal{L}_\theta = \sum_{i=1}^{N} r_k^{(i)} log\theta_{k,j}^{x_j} log(1-\theta_{k,j})^{1-x_j} + log\theta_k^{a-1} log(1-\theta_{k,j})^{b-1}$

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

Calculating its partial derivative regarding θ we get:

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

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

$(a+b-2+\sum_{i=1}^{N} r_k^{(i)})\theta_{k,j} = a - 1 + \sum_{i=1}^{N} r_k^{(i)} x_j^i x_j^i$

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

Therefore, we have the update rules for θ:

$\theta_{k,j} \leftarrow \frac{a-1+\sum_{i=1}^{N} r_k^{(i)} x_j^i}{a+b-2+\sum_{i=1}^{N} r_k^{(i)}}$

2.

The output of running mixture.print_part1_values():

pi[0] 0.085

pi[1] 0.13

theta[0, 239] 0.642710622711

theta[3, 298] 0.465736124958

Part 2

1.

$$Pr(z=k|x^i) = \frac{p(z=k)p(m^i * x^i|z=k)}{\sum_{k'=1}^{K} p(z=k')p(m^i * x^i|z=k')} = \frac{\pi_k \prod_{j=1}^{D} p(m_j^i * x_j^i|z=k)}{\sum_{k'=1}^{K} \pi_k \prod_{j'=1}^{D} p(m_j^i * x_{j'}^i|z=k')} = \frac{\pi_k \prod_{j=1}^{D} \theta_{k,j}^{m_j^i * x_j^i} (1-\theta_{k,j})^{m_j^i * (1-x_j^i)}}{\sum_{k'=1}^{K} \pi_{k'} \prod_{j'=1}^{D} \theta_{k',j'}^{m_{j'}^i * x_{j'}^i} (1-\theta_{k',j'})^{m_{j'}^i * (1-x_{j'}^i)}}$$

2.

Output:

R[0, 2] 0.174889514921
R[1, 0] 0.688537676109
P[0, 183] 0.651615199813
P[2, 628] 0.474080172491

Part 3

1. If $a = b = 1$, then $a - 1 = 0$ and $a + b - 2 = 0$, therefore the following update rule $\theta_{k,j} \leftarrow \frac{a-1+\sum_{i=1}^{N} r_k^{(i)} x_j^i}{a+b-2+\sum_{i=1}^{N} r_k^{(i)}}$ would be: $\theta_{k,j} \leftarrow \frac{\sum_{i=1}^{N} r_k^{(i)} x_j^i}{\sum_{i=1}^{N} r_k^{(i)}}$. In the case of all of X's term is 0 or 1, θ would equal to 0 or 1, but it is also a probability, therefore cause data sparsity, which is a bad model design.

2. Since part2's model has 100 class and part1's model has 10 class, so the probability of getting into a class is higher but less specific, therefore, model 2 is more accurate, thus, each elements has higher confidence, so it performs better.

3. No. Since we are only given top half of each image to predict the digit, and the number 1 has no other digits that has a top half shape similar to it, but the number 8 might get predicted to the number 9 since their top halves are very similar to each other. Thus the probability of this image being 8 or 9 are very close, but it is smaller than log probability of 1.