Assignment -2 Solutions

Prasenjeet Biswal **CS 7641**

GTID - 903260510 **Prof. Le Song**

1.

(a)

Given $p(x) = \sum_{k=1:K}(\pi_k N(x|u_k,\sum_k))$ ...(1)

$p(z) = \prod_{k=1:K}(\pi_k^{z_k})$

$p(x|z) = \prod_{k=1:K}(N(x|u_k, \sum_k))^{z}_k$

$p(x) = \sum_{z \varepsilon Z} p(z)p(x|z)$ where $Z = \{z^{(1)}, z^{(2)}, ... z^{(k)}\}$. .....(2)

To show that $\sum_{z \varepsilon Z} p(z)p(x|z) = \sum_{k=1:K}(\pi_k N(x|u_k,\sum_k))$

RHS

The probability of a data point from kth gaussian component = π_k can be written as $\prod_{k=1:K}(\pi_k^{z_k})$ because z_k is a latent variable and thus it only has values 0 if the point does not belong to k^{th} component else it will be 1. This is only for the k^{th} component. Summing over all components,

$\sum_{k=1:K} \pi_k = \sum_{z \varepsilon Z} \prod_{k=1:K}(\pi_k^{z_k})$

Similarly, $\sum_{k=1:K} N(x|u_k,\sum_k)$ can be written as $\sum_{z \varepsilon Z} \prod_{k=1:K}(N(x|u_k, \sum_k))^{z}_k$.

So eq(1) can written as $\sum_{z \varepsilon Z} \prod_{k=1:K}(\pi_k^{z_k}) \prod_{k=1:K}(N(x|u_k, \sum_k))^{z}_k$.

$= \sum_{z \varepsilon Z} \prod_{k=1:K}(\pi_k^{z_k})(N(x|u_k, \sum_k))^{z}_k$.

$= \sum_{z \varepsilon Z} p(z)p(x|z)$

=RHS

Thus Proved.

(b) Using Bayes Rule, $p(z_k^n|x_n) = (p(x_n|z_k^n)p(z_k^n)) / p(x_n)$

$p(x_n) = \sum \pi_k N(x|u_{k,} \sum_k)$

For data point x_n belonging to component $z_{k,}$

$p(z_k^n) = \pi_k$

and $p(x_n | z_k^n) = N(x_n | u_{k,} \sum_k)$

So, $p(z_k^n|x_n) = (\pi_k N(x_n | u_{k,} \sum_k)) / (\sum \pi_k N(x|u_{k,} \sum_k))$

(c)

We refer to the $p(z_k^n|x_n)$ as $\gamma_{z_{nk}}$.

The log likelihood is

$L = \sum_{n=1:N}\sum_{k=1:K} \gamma_{z_{nk}} (\log(\pi_k) - (x_i - u_k)^T\sum_k^{-1}(x_i - u_k) + \log|\sum|) + C$

Since we have a constraint, $\sum_{k=1:K} \pi_k = 1$, we will have to use lagrangian operator,

$L' = \sum_{n=1:N}\sum_{k=1:K} \gamma_{z_{nk}} (\log(\pi_k) - (x_i - u_k)^T\sum_k^{-1}(x_i - u_k) + \log|\sum|) + C + \lambda(1 - \sum_{k=1:K} \pi_k)$

Differentiating L' wrt π_k setting it equal to 0, $\sum_{n=1:N} \gamma_{z_{nk}} / \pi_k - \lambda = 0$

$$=> \pi_k \lambda = \sum_{n=1:N} \gamma_{z_{nk}}$$

Summing $\sum_{k=1:K}$ on both sides, we get

$1 = (1/\lambda)\sum_{n=1:N} 1$ ....because $\sum_{k=1:K} \gamma_{z_{nk}} = 1$.

so $\lambda = N$

replacing λ, we get

$\boldsymbol{\pi_k = N_k / N}$ **where** $\mathbf{N_k} = \sum_{n=1:N}\boldsymbol{\gamma_{z_{nk}}}$

To estimate u_k, we differentiate L wrt u_k and set it equal to 0,we get

$0 = \sum_{n=1:N} \gamma_{z_{nk}}(x_n - u_k) \sum_k^{-1}$

$\sum_{n=1:N} \gamma_{z_{nk}} x_n \sum_k = \sum_{n=1:N} \gamma_{z_{nk}} u_k \sum_k^{-1}$

Multiplying both sides by $\sum_k$, we get

$\sum_{n=1:N} \gamma_{z_{nk}} x_n = \sum_{n=1:N} \gamma_{z_{nk}} u_k$

so $\mathbf{u_k = (\sum_{n=1:N} \gamma_{z_{nk}} x_n) / N_k}$

Differentiating L wrt $\sum_k$, and equating to 0, we get

$0 = \sum_{n=1:N} \gamma_{z_{nk}}((1/\sum_k) + (x_i - u_k)^T\sum_k^{-2}(x_i - u_k))$

$\sum_{n=1:N} \gamma_{z_{nk}}\sum_k = \sum_{n=1:N} \gamma_{z_{nk}}(x_i - u_k)^T(x_i - u_k)$

$\mathbf{\sum_k = (\sum_{n=1:N} \gamma_{z_{nk}}(x_i - u_k)^T(x_i - u_k)) / N_k}$

(d)

Covariance matrix is given by ϵI where ϵ is variance parameter and I is the identity matrix.

So , $p(x| u_k, \sum_k) = (1/2\ \pi\epsilon) \exp((-1/2\epsilon)||x-u_k||^2)$, we get

Considering the EM algorithm for a mixture of K gaussians and ϵ to be fixed

$\boldsymbol{\gamma_{z_{nk}} = (\pi_k \exp(-||x_n - u_k||^2 / 2\epsilon)) / (\sum_j \pi_j \exp(-||x_n - u_j||^2 / 2\epsilon))}$

Since ϵ -> 0, we see that in the denominator, the term corresponding to smallest $||x_n - u_j||^2$ will go to zero most slowly. Hence, $\gamma_{z_{nk}}$ for point x_n all go to 0 except for term j, for which it goes to 1. Thus for this limit, each point is assigned a definite cluster.

Thus maximizing the expected complete log likelihood is the same as minimizing J for the K means algorithm

2.

(a) Given the probability of a point x_i lying over region i i.e. $P(x_i)= h_i$.

To maximize, we will multiply over all regions.

so $P(x_1)*P(x_2)..... = h_{j(1)*}h_{j(2).....}$ - the likelihood of all points x_i falling into region h_j.

so $\prod_i P(x_i) = \prod_i h_{j(i)}$

Taking log on both sides, we get

$\log(\prod_i P(x_i)) = \log(\prod_i h_{j(i)})$

so , **the log likelihood $\sum_i \log P(x_i) = \sum_i \log(h_{j(i)})$**

(b) Since there is a constraint of $\sum_i h_i \Delta_i = 1$

we use the LaGrange operator.

$\sum_i \log P(x_i) = \sum_i \log(h_j) - \lambda(\sum_i h_i \Delta_i - 1)$

Derivating wrt h_j and equating to 0, we get

$(\sum_{1:N} 1)/ h_j + \lambda \Delta_j = 0$ .... (1)

As we see we are summing from 1 to n, but only n_j fall in region j so ,

$\sum_{i=1:N} 1 = n_j$

Changing the equation (1), we get

$(n_j / h_j) + \lambda \Delta_j = 0$

$n_j = - \lambda \Delta_j h_j$

To get λ, we sum over all regions j,

$\sum_j n_j = - \lambda \sum_j \Delta_j h_j$

$\lambda = -N$

Putting the value of λ in equation 1 we get

maximum likelihood estimator $\mathbf{h_j = (n_j / N\Delta_j)}$

(c)

Non-parametric density estimation usually does not have parameters. - False, Non parametric density estimators have parameters that depends on the size of data. The non-parametric density estimators have hyper-parameters that has to be changed to fit the model.

The Epanechnikov kernel is the optimal kernel function for all data. - True, the Epanechnikov kernel minimizes the mean integrated square error, and other kernels are measured relative to Epanechnikov kernel. So Epanechnikov kernel is the optimal kernel function for all data.

Histogram is an efficient way to estimate density for high-dimensional data. - False, because when the dimensions of data increase to d, the memory required increases to n^d, which makes the histogram quite inefficient.

Parametric density estimation assumes the shape of probability density. - True, the parametric density estimation makes assumptions about the shape of the data, and thus finds the parameters involved. For e.g. the Gaussian distribution assumes a uni-modal distribution.

3(a)

To Prove $- H(X,Y) \leq H(X) + H(Y)$

LHS -

$-\sum_{x\varepsilon X}\sum_{y\varepsilon Y} p(x,y) \log(p(x)p(y|x))$

$= -\sum_{x\varepsilon X}\sum_{y\varepsilon Y} p(x,y) \log(p(x)) - -\sum_{x\varepsilon X}\sum_{y\varepsilon Y} p(x,y) \log(p(y|x))$

$= H(x) + H(Y|X)$

In RHS, we have H(X) + H(Y),

We need to show that $H(Y|X) \leq H(Y)$,

By the property, $I(X,Y) \geq 0$ where I(X,Y) is information gain

I(X,Y) can written as H(Y) - H(Y\|X) (See part b of this question).

so $H(Y) - H(Y|X) \geq 0$

so $H(Y) \geq H(Y|X)$

so **$H(X,Y) \leq H(X) + H(Y)$. Thus Proved.**

(b) To Prove $I(X,Y) = H(Y) + H(X) - H(X,Y)$

$I(X,Y) = \sum_{x\varepsilon X}\sum_{y\varepsilon Y} p(x,y) \log(p(x,y)/p(x)p(y))$

$= \sum_{x\varepsilon X}\sum_{y\varepsilon Y} p(x,y) \log(p(x|y)/p(x))$

$= \sum_{x \epsilon X} \sum_{y \epsilon Y} p(x,y) \log(p(y|x)/p(y))$

$= - \sum_{x \epsilon X} \sum_{y \epsilon Y} p(x,y) \log(p(y)) + \sum_{x \epsilon X} \sum_{y \epsilon Y} p(x,y) \log(p(y|x))$

so I(X,Y) = H(Y) - H(Y|X) .....(1)

From previous part, H(X,Y) = H(X) + H(Y|X)

replacing H(Y|X) in equation (1), we get,

I(X,Y) = H(Y) +H(X) - H(X,Y)

(c) Find under what conditions does H(Z) = H(X) + H(Y).

Since Z is a function (X,Y), Z can be represented as f(X,Y).

We know that H(f(X)) ≤ H(X) --- property of entropy.

Thus H(Z) ≤ H(X,Y). .... From above

and H(X,Y) ≤ H(X) + H(Y) .... from part (a)

The equality will only be satisfied if and only if H(X,Y) = H(X) + H(Y) which is only true **when X and Y are independent.** (when X and Y are independent, H(Y|X) = H(Y)).

4. I ran mycluster.m for 200 iterations and the statistics for accuracy and running time are as follows -

mean(accuracy) - 77.4412

standard deviation(accuracy) - 8.0292

Median (accuracy) - 79.1042

Mean running time - 1.3632736 s

Median Running Time - 1.2034 s

The converging condition for our algorithm is when u_{jc} and π_c do not change or the number of iterations reaches 1000. I saw only two observations when the running time went to around 6s when the algorithm stopped after 1000 iterations.