Introduction to ML (CS771), Autumn 2018
Indian Institute of Technology Kanpur
Homework Assignment Number 3

Student Name: Shahsi Kant Gupta
Roll Number: 160645
Date: November 2, 2018

QUESTION 1

Expanding the objective function we get:

$$f(\alpha + \delta e_n) = (\alpha + \delta e_n)^T 1 - \frac{1}{2}(\alpha + \delta e_n)^T G(\alpha + \delta e_n) \tag{1}$$

$$f(\alpha + \delta e_n) = \alpha^T 1 + \delta e_n^T 1 - \frac{1}{2}(\alpha^T G\alpha + 2\delta e_n^T G\alpha + \delta^2 e_n^T e_n) \tag{2}$$

$$\frac{\partial f(\alpha + \delta e_n)}{\partial \delta} = e_n^T 1 - \frac{1}{2}(2e_n^T G\alpha + 2\delta e_n^T e_n) \tag{3}$$

Above equation should be equal to zero to maximize f w.r.t δ

$$\Rightarrow e_n^T 1 - e_n^T G\alpha - \delta e_n^T e_n) = 0 \tag{4}$$

$$\Rightarrow 1 - \sum \alpha_i G_{in} - \delta = 0 \tag{5}$$

$$\Rightarrow \delta_* = 1 - \sum \alpha_i G_{in} \tag{6}$$

Now, this will not be the final answer as we need to project the δ_* as such that α should be in the range of $0 \leq \alpha_n \leq C$. Applying the conditions we get a project δ_* to be:

$$\delta_* = min(C - \alpha_n, max(-\alpha_n, 1 - \sum \alpha_i G_{in})) \tag{7}$$

Pseudo Code/ Algorithm:

```
initialise alpha
while converges:
  for i in range(N):
    delta = max(-alpha[n], 1 - dot(G, alpha))
    delta = min(C - alpha[n], delta)
    alpha[n] = alpha[n] + delta
```

Student Name: Shahsi Kant Gupta
Roll Number: 160645
Date: November 2, 2018

QUESTION 2

Clearly, we can write:

$$\sum_{n,m} ||x_n - x_m||^2 = \sum_{n,m} \{f_n = f_m\} ||x_n - x_m||^2 + \sum_{n,m} \{f_n \neq f_m\} ||x_n - x_m||^2 \tag{8}$$

And since the data is fixed the LHS of the above equation will be constant. Therefore minimising the first term in RHS will maximise the second term of RHS as their sum need to be constant. The second term is nothing but the sum of squared distances between pairs of points in different clusters. Hence proved!

Introduction to ML (CS771), Autumn 2018
Indian Institute of Technology Kanpur
Homework Assignment Number 3

Student Name: Shahsi Kant Gupta
Roll Number: 160645
Date: November 2, 2018

QUESTION 3

Part (1)
Using $m = missed$ and $o = observed$.

$$p(x_n^m|x_n^o, \mu, \Sigma) = p(x_n^m|\mu_n^{m|o}, \Sigma_n^{m|o}) \tag{9}$$

Using the results from the book:

$$\mu_n^{m|o} = \mu_n^m + \Sigma_n^{mo}(\Sigma_n^{oo})^{-1}(x_n^o - \mu_n^o) \tag{10}$$

$$\Sigma_n^{m|o} = \Sigma_n^{mm} - \Sigma_n^{mo}(\Sigma_n^{oo})^{-1}\Sigma_n^{om} \tag{11}$$

Part (2)
Taking $x_n = (x_n^o; x_n^m) \implies E[x_n] = (x_n^o; E[x_n^m])$. Therefore, our expected CLL can be easily written as:

$$Q(\theta, \theta^{t-1}) = \sum_{n=1}^{N} log(N(E[x_n]|\mu, \Sigma)) \tag{12}$$

Part (3)
Solving the above MLE will yield:

$$\mu = \frac{1}{N}\sum_{n=1}^{N} E[x_n] \tag{13}$$

$$\Sigma = \frac{1}{N}\sum_{n=1}^{N} E[x_n x_n^T] - \mu\mu^T \tag{14}$$

EM Algorithm:
Initialise u, sigma
E Step: Compute $\mu_n^{m|o}$ and $\Sigma_n^{m|o}$ for each n.

$$\mu_n^{m|o} = \mu_n^m + \Sigma_n^{mo}(\Sigma_n^{oo})^{-1}(x_n^o - \mu_n^o) \tag{15}$$

$$\Sigma_n^{m|o} = \Sigma_n^{mm} - \Sigma_n^{mo}(\Sigma_n^{oo})^{-1}\Sigma_n^{om} \tag{16}$$

M Step: re-estimate u, sigma via MLE

$$\mu = \frac{1}{N}\sum_{n=1}^{N} E[x_n] \tag{17}$$

$$\Sigma = \frac{1}{N}\sum_{n=1}^{N} E[x_n x_n^T] - \mu\mu^T \tag{18}$$

If doesnt converge go back to E Step!

Introduction to ML (CS771), Autumn 2018
Indian Institute of Technology Kanpur
Homework Assignment Number 3

Student Name: Shahsi Kant Gupta
Roll Number: 160645
Date: November 2, 2018

QUESTION 4

This problem can be simply solved by taking it to be similar to clustering problem, where we have added advantage of estimating the parameters by using actuall datas during the M step of EM Algorithm.

Taking $y^{'}_{n+m}$ to be the new class data, we can write $y^{'}_{n+m} = (y_n; z_m) \implies y^{'}_{(n+m)k} = (y_{nk}; z_{mk})$. Let the $n + m = i; 0 \leq i \leq N + M$. We can write $E[y^{'}_{ik}] = (y_{nk}; E[z_{mk}])$. Therefore, our new CLL can be written as:

$$Q(\theta, \theta^{old}) = \sum_{i=1}^{N+M} \sum_{k=1}^{K} E[y^{'}_{ik}][log(\pi_k) + log(N(x_n|\mu_k, \Sigma_k)) \tag{19}$$

EM Algorithm:
(1) Initialise: $\{\pi_k, \mu_k, \Sigma_k\}_{k=1}^{K}$
(2) E Step: Compute expectation of z_m for all m, k

$$E[z_{mk}] = \frac{\pi_k N(x_{(N+m)}|\mu_k, \Sigma_k)}{\sum_{l=1}^{K} \pi_l N(x_{(N+m)}|\mu_l, \Sigma_l)} \tag{20}$$

(3) M Step: re-estimate parameters via MLE

$$N_k = \sum_{i=1}^{N+M} E[y^{'}_{ik}] \tag{21}$$

$$\mu_k = \frac{1}{N_k} \sum_{i=1}^{N+M} E[y^{'}_{ik}]x_i \tag{22}$$

$$\Sigma_k = \frac{1}{N_k} \sum_{i=1}^{N+M} E[y^{'}_{ik}](x_i - \mu_k)(x_i - \mu_k)^T \tag{23}$$

$$\pi_k = \frac{N_k}{N + M} \tag{24}$$

(4) Go to step 2 if not yet converged!

Introduction to ML (CS771), Autumn 2018
Indian Institute of Technology Kanpur
Homework Assignment Number 3

Student Name: Shahsi Kant Gupta
Roll Number: 160645
Date: November 2, 2018

QUESTION 5

Part (1)
A standard linear model will only work for those solutions where we have to regress a linear curve whereas this model can be a combination of K different linear curves basically what the model is doing is that at first it is clustering the data on k different linear curves and then the prediction are made for y. This will also help in the reduction of outliers in a linear curve as the outliers may get separate out due to clustering.

Part (2 and 3)
Here, our latent variable model becomes:

$$p(z_n = k|y_n, \theta) = \frac{p(z_n = k)p(y_n|z_n = k, \theta)}{\sum_{l=1}^{K} p(z_n = l)p(y_n|z_n = l, \theta)} \tag{25}$$

$$p(y_n, z_n|\theta) = p(y_n|z_n, \theta)p(z_n|\theta) \tag{26}$$

where:

$$p(z_n = k) = \pi_k \tag{27}$$

$$p(y_n|z_n, \theta) = N(w_{z_n}^T x_n, \beta^{-1}) \tag{28}$$

ALT-OPT Algorithm
Step 1 find the best z_n:

$$z_n = argmax_{z_n} \frac{\pi_k N(w_{z_n}^T x_n, \beta^{-1})}{\sum_{l=1}^{K} \pi_l N(w_l^T x_n, \beta^{-1})} \tag{29}$$

$$\implies z_n = argmax_{z_n} \frac{\pi_k exp(\frac{-\beta}{2}(y_n - w_{z_n}^T x_n)^2)}{\sum_{l=1}^{K} \pi_l exp(\frac{-\beta}{2}(y_n - w_l^T x_n)^2)} \tag{30}$$

Step 2 re-estimate the parameters:

$$N_k = \sum_{n=1}^{N} z_{nk} \tag{31}$$

$$w_k = (X_k^T X_k)^{-1} X_k^T y_k \tag{32}$$

$$\pi_k = N_k/N \tag{33}$$

Here X_k are N_kxD matrix containing training sets which is clustered in class k. And y_n are N_kx1 vectors containing training sets labels which is clustered in class k.

If $\pi_k = 1/K$ then:

$$z_n = argmax_{z_n} \frac{exp(\frac{-\beta}{2}(y_n - w_{z_n}^T x_n)^2)}{\sum_{l=1}^{K} exp(\frac{-\beta}{2}(y_n - w_l^T x_n)^2)} \tag{34}$$

This update is equivalent to multi output logistic regression.

EM Algorithm
E Step find $p(z_k)$ for each k:

$$p(z_n = k|y_n, \theta) = \frac{\pi_k N(w_k^T x_n, \beta^{-1})}{\sum_{l=1}^{K} \pi_l N(w_l^T x_n, \beta^{-1})} \tag{35}$$

$$\implies p(z_n = k|y_n, \theta) = \frac{\pi_k exp(\frac{-\beta}{2}(y_n - w_k^T x_n)^2)}{\sum_{l=1}^{K} \pi_l exp(\frac{-\beta}{2}(y_n - w_l^T x_n)^2)} \tag{36}$$

M Step re-estimate the parameters:

$$N_k = \sum_{n=1}^{N} E[z_{nk}] \tag{37}$$

$$w_k = (X_k^T X_k)^{-1} X_k^T y_k \tag{38}$$

$$\pi_k = N_k/N \tag{39}$$

Here X_k are N_kxD matrix containing training sets which is clustered in class k. And y_n are N_kx1 vectors containing training sets labels which is clustered in class k.

As $\beta \to \infty$:

$$p(z_n = k|y_n, \theta) \to \pi_k \tag{40}$$

Therefore, $E[z_{nk}] = z_{nk}$ which means EM reduces to ALT-OPT

Introduction to ML (CS771), Autumn 2018
Indian Institute of Technology Kanpur
Homework Assignment Number 3

Student Name: Shahsi Kant Gupta
Roll Number: 160645
Date: November 2, 2018

QUESTION
6

Problem 1:
Part (1)
As we increase the regularisation hyperparameter error increases! The possible reason is that the train set and test sets both seems to be taken from a same sin curve without much outliers so less the regularisation better curve toward train data, indirectly better curve toward test data.

Part (2)
Lesser the value of L more the error in prediction, the reason being that less number feature point taken. The value of L=50 is good enough as increasing L to 100 changes the rmse by just 0.003!

Problem 2:
Part (1)
After plotting the datas it can be easily seen that the clusters are radially distributed around origin. Therefore for the hand crafted part feature transformed was used to be the distance from origin.

Part (2)
As the given data can be easily clustered if we use a feature transform which based on distance of the point from origin. One landmark sometime clusters the data well when the choosen landmark is closer to the origin whereas when it is farther to the origin it doesnt cluster well because the datas are distributed radially around the origin!

Note: landmark points are shown in red colours with star mark as the blue color points where not visible whent the lanmark belongs to the blue clusters