Work Origination Certification

By submitting this document, I, the author of this deliverable, certify that

1. I have reviewed and understood the Academic Honesty section of the current version of FITs Student Handbook available at http://www.fit.edu/studenthandbook/, which discusses academic dishonesty (plagiarism, cheating, miscellaneous misconduct, etc.)

2. The content of this report reflects my personal work and, in cases it is not, the source(s) of the relevant material has/have been appropriately acknowledged after it has been first approved by the courses instructor.

3. In preparing and compiling all this report material, I have not collaborated with anyone and I have not received any type of help from anyone but from the courses instructor.

Shreenidhi Sudhakar

Full Name (please PRINT)

11/03/2016

Signature Date

Task 2

a) - $r(\theta)$ is referred to as penalty term as increase / decrease in it's value, directly affects how the classifier classifies i/p data samples. $r(\theta)$ in general helps to reduce over-fitting by helping the model achieve a good trade off b/w bias and variance.

- The term getting penalised here is the log likelihood function that is used to classify data

- Increase in P_k, decreases model classification ability and vice-versa.

b) We know that

$$l_r(\theta) = \sum_{n=1}^{N} \sum_{i=1}^{C} y_{ni} \log \pi_i + \sum_{n=1}^{N} \sum_{i=1}^{C} y_{ni} \log[g(x_n)]$$

$$+ -\tfrac{1}{2} \sum_{i=1}^{C} P_k N_k \, \text{trace}\{C_k^{-1} R_k\} \rightarrow ①$$

$$\frac{\partial l_r(\theta)}{\partial \hat{\pi}_k} = 0 \Rightarrow \hat{\pi}_k^* = \frac{\sum_{y=1}^{N} y_{nk}}{\sum_{y=1}^{N} \sum_{i=1}^{C} y_{ni}} = \frac{N_k}{N} \quad ; k = 1, 2 \ldots C$$

$$\frac{\partial l_r(\theta)}{\partial \hat{\mu}_k} = 0 \Rightarrow \hat{\mu}_k^* = \frac{1}{N_k} \cdot \sum_{l(x_n) = k} x_n$$

The class priors Π_k^* & class means μ_k^* do not change since regularisation term does not depend on Π_k & μ_k. Hence, class priors Π_k^* & class means μ_k^* for $k=1,2\ldots C$ that maximise l_r coincide with maximum likelihood estimates utilised in QDA.

c) Differentiating ① with $\hat{C}_k$, we get

$$\frac{\partial l_r(\hat{\theta})}{\partial \hat{C}_k} = \frac{\partial l(\hat{\theta})}{\partial \hat{C}_k} - \frac{1}{2} N_k P_k C_k^{-1} R_k C_k^{-1}$$

$$= (-)\sum_{n=1}^{N} y_{nk} \left(\frac{1}{2}\left[C_k^{-1} - C_k^{-1}(x-\mu)(x-\mu)^T C_k^{-1}\right]\right)$$

$$+ \sum_{n=1}^{N} y_{nk} \cdot \frac{1}{2} \cdot P_k C_k^{-1} R_k C_k^{-1} = 0$$

$$\Rightarrow \sum_{n=1}^{N} y_{nk} \left(-\frac{1}{2} C_k^{-1} - \frac{C_k^{-1}}{2}\left((x-\mu)(x-\mu)^T + P_k R_k\right) C_k^{-1} \right) = 0$$

$$\Rightarrow C_k^* = \hat{C}_k + P_k R_k \quad , k=1,2\ldots C$$

W.k.t

d) $$C_k^* = \hat{C}_k + P_k R_k$$

$$\Rightarrow V^T\left(\hat{C}_k + P_k R_k\right)V$$ here, V = dummy matrix

$$\Rightarrow V^T \hat{C}_k V + P_k V^T R_k V$$

here,

$v^T \hat{C}_k v \geq 0$, only when $\hat{C}_k$ is semi +ve definite

$\rho_k \geq 0$, positive

$v^T R_k v > 0 \Rightarrow v^T R_k v$ is +ve defnite

Thus, when we add $\rho_k R_k$; we ensure that $\hat{C}_k^*$ is always positive defnite

e) - We use RQDA instead of QDA , when data sample size is less than # of dimensions i.e, features of data set. Here, we get a singular covariance matrix.

- In order to choose ρ_k , we can use cross validation techniques like LOOC /K-fold.

Task 4

a) We know that,

$$P(C_i/x) = \frac{P(x|C_i)\ P(C_i)}{\sum_{k=1}^{c} P(x|C_k)\ P(C_k)}$$

$$= \frac{g(\eta_i)\ h(x)\ exp(\eta_i^T x) * P(C_i)}{\sum_{k=1}^{c} g(\eta_k)\ h(x)\ exp(\eta_k^T x) * P(C_i)}$$

$$P(C_i/x) = \frac{\cancel{P(C_i)} \exp\left[\log\left(P(\omega_i)\, g(\eta_i)\right)\right] \exp(\eta_i^T x)}{\sum_{k=1}^{c} \exp\left[\log\left[\left(P(\omega_k)\, g(\eta_i)\right)\right]\right] \exp(\eta_k^T x)} \quad \left[\because \exp\log(x) = x\right]$$

$$= \frac{\exp\left\{\log\left[P(C_i)\, g(\eta_i)\right] + \eta_i^T x\right\}}{\sum_{k=1}^{c} \exp\left\{\log\left[P(C_k)\, g(\eta_i)\right] + \eta_k^T x\right\}}$$

$$= \frac{\exp\left\{1 + \dfrac{\eta_i^T x}{\log\{P(C_i)\, g(\eta_i)\}}\right\}}{\sum_{k=1}^{c} \exp\left\{1 + \dfrac{\eta_k^T x}{\log\{P(\omega_k)\, g(\eta_k)\}}\right\}}$$

Let, $\dfrac{\eta_k^T x}{\log\{P(\omega_k)\, g(\eta_k)\}} = \omega_k^T$; $\dfrac{\eta_i^T x}{\log\{P(\omega_i)\, g(\eta_i)\}} = \omega_i^T$

$$\Rightarrow P(C_i/x) = \frac{e^1 \cdot e^{\omega_i^T x}}{e^1 \cdot \sum_{k=1}^{c} e^{\omega_k^T x}} = \frac{e^{\omega_i^T x}}{\sum_{k=1}^{c} e^{\omega_k^T x}}$$

$\downarrow$

PDM arising from MNR model (given)

b) Cross Entropy function is given as:

$$E_r = E + r(\omega) = -\sum_{n=1}^{N} \sum_{k=1}^{C} t_{kn} \log y_{kn} + \rho \sum_{k=1}^{C} \|\omega_k\|^2$$

$$\Rightarrow \frac{\partial E_r}{\partial \omega_i} = \sum_{n=1}^{N} (y_{in} - t_{in}) x_n + 2 \rho \omega_i$$

$$\Rightarrow \omega_i^{k+1} = \omega_i^k - \lambda \left(\sum_{n=1}^{N} (y_{in} - t_{in}) x_n + 2 \rho \omega_i^k \right)$$

$\hookrightarrow$ GD update equation

here,

λ = learning rate

ρ = regularisation parameter

- Adding regulariser term to weight update equation, ensures that the new weight obtained is not very much different from the old weight values.
- Thus, it prevents over-fitting. Resulting decision boundary will have a smooth curve.

Task 1

Run Task1.m to observe Graphs

Model	Classifier	Avg LOOC Error	Test Set Error
General Case	LDA	0.02	0.06
Naive Bayes	LDA	0.00	0.05
Isotropic	LDA	0.00	0.04
General Case	QDA	0.02	0.04
Naive Bayes	QDA	0.00	0.04
Isotropic	QDA	0.00	0.04

Table: LOOC and Test Set Error for Various Classifier Models

b) From the table, we realize that Isotropic LDA Classifier is the champion Model. LOOC is used to find the average validation set error. It helps us to ensure that during training phase, we use all combinations of input samples and thus we end up realizing the best classifier model.

c) From the table, we realize that QDA model is better than LDA model by observing the test set errors. Results are as expected as decision boundaries for QDA are non-linear in shape when compared to LDA model. This is the reason why we get a lower test error for QDA models.

d) Decision boundaries for QDA are non-linear in shape when compared to LDA model. This is the reason why we get a lower test error for QDA models as they are able to classify data better than LDA that generate linear decision boundaries. The curve or non-linear boundaries is obtained for QDA as we add the prior weighted covariance matrices of all class together to form the log likelihood function.

Task 3

Run Task3.m to observe Graphs

a)

Average LOOC Error	Test Set Error
0.02	0.08

b) Decision boundary obtained classifies the data with above accuracy. When compared to LDA and QDA models, the decision boundary of MNR is much more precise i.e., it's not well spread as in case of LDA and QDA. This because they are not constrained by decision boundaries that is based on prior weighted covariance matrices. As a result, MNR have more freedom to generate precise decision boundaries.

Note:

Convergence of MNR model is seen post 100 iterations as mean / gradient found is not divided by total # of Data Samples i.e. it's not normalized.