# UMC 203 Assignment 01

Sahil Chaudhary(22244)

28 February, 2024

1. **Naive Bayes Classifier**
Recall the Bayes classifier; the critical component of a Bayes classifier is computing the posterior. However, this computation scales with the dimension of the data. This can be seen in, $P(y,x) = P(y, x_1, x_2, \dots, x_d) = P(y)P(x_1|y)P(x_2|x_1, y) \dots P(x_d|x_1 x_2 \dots x_d, y)$ where x=$(x_1, \dots, x_d)$ is a d-dimensional vector and y denotes the class label. To get around this issue, we make a simplifying assumption

   (a) Assume each coordinate of the vector is mutually independent, i.e.,
   $P(x_i|x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_d, y) = P(x_i|y)$ Therefore, $P(x,y) = P(y)\prod_{i=1}^{d} P(x_i|y)$

   (b) Assume that class-conditioned distributions are normally distributed, i.e.,
   for $j \in \{0,1\}, (X|Y=j) \sim N(\mu_j, \Sigma_j)$ where $X \in R^d$

In class, you have studied the Bayes classifier under the 0-1 loss function, which assumes that misclassifi cation of any kind is punished equally. However, in the real world, some mistakes are costlier than others. Think of an intruder detection algorithm that fails to catch someone breaking in versus making an annoying beep whenever the dog snores too loud. We capture this by setting l(0,1) and l(1,0) to different values. We will call this new loss function the modified loss function. Please use the function q1_get_loss described below to obtain these values. You are now given a dataset with 5-dimensional features and asked to learn a classifier that minimizes the modified misclassification loss. You decide to make assumptions (A1) and (A2) and build the Bayes classifier for the modified loss. Please use the function q1_get_train_set and q1_get_test_set described below to obtain these datasets.

   (a) The parameters for your model are $\mu_0, \mu_1, \Sigma_0, \Sigma_1$. Using assumption (A1), what can you say about $\Sigma_0, \Sigma_1$?
   **Solution:**

$$\mu_0 = E[X|Y=0] \tag{1}$$

$$\begin{aligned}\mu_0 &= E[X|Y=0]\\ &= \begin{bmatrix} E[X_1|Y=0] \\ E[X_2|Y=0] \\ E[X_3|Y=0] \\ E[X_4|Y=0] \\ E[X_5|Y=0] \end{bmatrix}\end{aligned}$$

$$\mu_1 = E[X|Y=1] \tag{2}$$

$$\begin{aligned}\mu_1 &= E[X|Y=1]\\ &= \begin{bmatrix} E[X_1|Y=1] \\ E[X_2|Y=1] \\ E[X_3|Y=1] \\ E[X_4|Y=1] \\ E[X_5|Y=1] \end{bmatrix}\end{aligned}$$

   Using assumption (A1), each co-ordinate of the vector is mutually independent.

$$\Sigma_0 = Cov[X|Y=0] \tag{3}$$

$$Cov[X|Y=0] = \begin{bmatrix} Cov[X_1|Y=0, X_1|Y=0] & Cov[X_1|Y=0, X_2|Y=0] & \dots & Cov[X_1|Y=0, X_5|Y=0] \\ Cov[X_2|Y=0, X_1|Y=0] & Cov[X_2|Y=0, X_2|Y=0] & \dots & Cov[X_2|Y=0, X_5|Y=0] \\ \vdots & \vdots & \ddots & \vdots \\ Cov[X_5|Y=0, X_1|Y=0] & Cov[X_5|Y=0, X_2|Y=0] & \dots & Cov[X_5|Y=0, X_5|Y=0] \end{bmatrix}$$

Since, each co-ordinate of the vector is mutually independent, the covariance matrix will be a diagonal matrix.

$$Cov[X|Y=0] = \begin{bmatrix} Var[X_1|Y=0] & 0 & \dots & 0 \\ 0 & Var[X_2|Y=0] & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Var[X_5|Y=0] \end{bmatrix}$$

Similarly, $\Sigma_1 = Cov[X|Y=1]$ will also be a diagonal matrix.

$$\Sigma_1 = Cov[X|Y=1] \tag{4}$$

$$Cov[X|Y=1] = \begin{bmatrix} Var[X_1|Y=1] & 0 & \dots & 0 \\ 0 & Var[X_2|Y=1] & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & Var[X_5|Y=1] \end{bmatrix}$$

(b) State the Bayes classifier under the modified loss function.

**Solution:**

We have the following loss function:

$$l(0,1) = 6 \tag{5}$$
$$l(1,0) = 5 \tag{6}$$
$$l(0,0) = 0 \tag{7}$$
$$l(1,1) = 0 \tag{8}$$

Choose class h(x)=1 if $E_{Y|X}[l(1,Y)] < E_{Y|X}[l(0,Y)]$
Choose class h(x)=0 if $E_{Y|X}[l(1,Y)] > E_{Y|X}[l(0,Y)]$
Condition when h(x)=1:

$E_{Y|X}[l(1,Y)] = P(Y=0|X)l(1,0) + P(Y=1|X)l(1,1) < E_{Y|X}[l(0,Y)] = P(Y=0|X)l(0,0) + P(Y=1|X)l(0,1)$

$$P(Y=0|X)l(1,0) + P(Y=1|X)l(1,1) < P(Y=0|X)l(0,0) + P(Y=1|X)l(0,1)$$
$$5P(Y=0|X) + 0P(Y=1|X) < 0P(Y=0|X) + 6P(Y=1|X)$$
$$5P(Y=0|X) < 6P(Y=1|X)$$
$$P(Y=0|X) < \frac{6}{5}P(Y=1|X)$$
$$\frac{P(Y=0|X)}{P(Y=1|X)} < \frac{6}{5}$$

Then, the Bayes classifier under the modified loss function is:

$$h(x) = \begin{cases} 1 & \text{if } \frac{P(Y=0|X)}{P(Y=1|X)} < \frac{6}{5} \\ 0 & \text{otherwise} \end{cases}$$

(c) Obtain sample estimates of the class conditional means and variances for your model with n =2,10,20,50,100,500,1000 samples. Write down the estimates for all the parameters in a table for different values of n.

**Solution:**

We can estimate the class conditional means and variances assuming (A1) and (A2) using the following equations:

$$ln(P(X|\mu, \Sigma)) = -\frac{ND}{2} ln(2\pi) - \frac{N}{2} ln(|\Sigma|) - \frac{1}{2} \sum_{i=1}^{N} (X_i - \mu)^T \Sigma^{-1} (X_i - \mu) \tag{9}$$

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^{N} X_i \tag{10}$$

$$\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (X_i - \hat{\mu})(X_i - \hat{\mu})^T \tag{11}$$

Using the same idea, following are the estimates for the parameters for different values of n:

| n | $\hat{\mu_0}$ | $\hat{\Sigma}_0$ |
|---|---|---|
| 2 | $\begin{bmatrix} -1.10 \\ -0.01 \\ -0.27 \\ 1.04 \\ -0.7 \end{bmatrix}$ | $\begin{bmatrix} 1.94 & 2.03 & -2.79 & -0.97 & -1.64 \\ 2.03 & 2.12 & -2.92 & -1.01 & -1.71 \\ -2.79 & -2.92 & 4.02 & 1.39 & 2.36 \\ -0.97 & -1.01 & 1.39 & 0.48 & 0.81 \\ -1.64 & -1.72 & 2.36 & 0.82 & 1.39 \end{bmatrix}$ |
| 10 | $\begin{bmatrix} -1.52023 \\ -1.698765 \\ -0.674521 \\ -1.28683 \\ -1.892129 \end{bmatrix}$ | $\begin{bmatrix} 2.20409065 & -0.70897033 & -1.52012916 & 0.32983363 & -0.85392442 \\ -0.70897033 & 2.17018988 & 0.61634396 & -0.56701789 & -0.61139648 \\ -1.52012916 & 0.61634396 & 1.8952868 & 0.1477997 & -0.40486102 \\ 0.32983363 & -0.56701789 & 0.1477997 & 2.51164119 & 1.14612134 \\ -0.85392442 & -0.61139648 & -0.40486102 & 1.14612134 & 4.31921586 \end{bmatrix}$ |
| 20 | $\begin{bmatrix} -0.9440725 \\ -0.943722 \\ -1.49528 \\ -1.044779 \\ -1.093172 \end{bmatrix}$ | $\begin{bmatrix} 1.85338079 & 0.18993976 & -1.54024669 & -0.59454894 & -0.20181833 \\ 0.18993976 & 2.88328464 & 0.4994077 & -0.84828654 & -0.26184565 \\ -1.54024669 & 0.4994077 & 4.07590723 & 1.09917592 & -0.72155111 \\ -0.59454894 & -0.84828654 & 1.09917592 & 4.34059797 & -2.46268364 \\ -0.20181833 & -0.26184565 & -0.72155111 & -2.46268364 & 5.09839533 \end{bmatrix}$ |
| 50 | $\begin{bmatrix} -1.3262168 \\ -0.5947532 \\ -0.453961 \\ -1.0742048 \\ -0.911841 \end{bmatrix}$ | $\begin{bmatrix} 1.98250285 & -0.38848063 & -0.09779611 & -0.50523961 & 0.24792637 \\ -0.38848063 & 2.1219595 & 0.08121971 & -0.30544928 & -0.17729515 \\ -0.09779611 & 0.08121971 & 3.23124224 & -0.1139626 & 0.82947096 \\ -0.50523961 & -0.30544928 & -0.1139626 & 4.25568047 & -0.36085824 \\ 0.24792637 & -0.17729515 & 0.82947096 & -0.36085824 & 4.33149526 \end{bmatrix}$ |
| 100 | $\begin{bmatrix} -0.9290768 \\ -1.2022883 \\ -1.0824965 \\ -1.2040802 \\ -1.3218391 \end{bmatrix}$ | $\begin{bmatrix} 2.87564438 & -0.318950261 & -0.326499492 & 0.423347808 & 0.0006312 \\ -0.31895 & 2.503230 & 0.21478 & -0.588 & -0.08189 \\ -0.32649949 & 0.2147 & 2.1495632 & -0.599123 & 0.00503566 \\ 0.423347 & -0.588629 & -0.5991 & 5.029462 & -0.834208 \\ 0.0006312594 & -0.08189087 & 0.005035664 & -0.834208 & 5.72707 \end{bmatrix}$ |
| 500 | $\begin{bmatrix} -0.90848252 \\ -1.02960764 \\ -0.95140424 \\ -1.0302202 \\ -0.93154514 \end{bmatrix}$ | $\begin{bmatrix} 2.32556305 & 0.06871617 & 0.07725202 & 0.02398197 & 0.1200431 \\ 0.06871617 & 2.34887973 & 0.1687388 & -0.26967689 & 0.18576101 \\ 0.07725202 & 0.1687388 & 2.2837962 & -0.18022004 & -0.05147337 \\ 0.02398197 & -0.26967689 & -0.18022004 & 4.52396422 & 0.33020893 \\ 0.1200431 & 0.18576101 & -0.05147337 & 0.33020893 & 4.34237267 \end{bmatrix}$ |
| 1000 | $\begin{bmatrix} -1.06094952 \\ -1.06322811 \\ -1.03086935 \\ -1.00683137 \\ -1.07125371 \end{bmatrix}$ | $\begin{bmatrix} 2.394096 & 0.108637 & 0.01035053 & 0.0101109 & 0.0264421 \\ 0.1086377 & 2.4655199 & -0.03647597 & -0.00441317 & 0.1039463 \\ 0.01035053 & -0.0364759 & 2.2809901 & 0.1379287 & 0.02528989 \\ 0.01011094 & -0.004.4131760 & 0.13792876 & 4.513258 & 0.08250462 \\ 0.02644210 & 0.1039463 & 0.025289898 & 0.08250462 & 4.452842 \end{bmatrix}$ |

| n | $\hat{\mu_1}$ | $\hat{\Sigma_1}$ |
|---|---|---|
| 2 | $\begin{bmatrix} 1.92 \\ 0.80 \\ -2.01 \\ 1.81 \\ 0.41 \end{bmatrix}$ | $\begin{bmatrix} 0.19 & -0.33 & -0.16 & -0.34 & 0.28 \\ -0.34 & 0.59 & 0.28 & 0.61 & -0.50 \\ -0.16 & 0.28 & 0.13660416 & 0.29363242 & -0.24252413 \\ -0.34565366 & 0.61104302 & 0.29363242 & 0.63116669 & -0.52130876 \\ 0.28549079 & -0.50468772 & -0.24252413 & -0.52130876 & 0.43057219 \end{bmatrix}$ |
| 10 | $\begin{bmatrix} 1.427877 \\ 0.225468 \\ 0.046948 \\ 1.335006 \\ 0.905356 \end{bmatrix}$ | $\begin{bmatrix} 1.57396717 & 0.90107623 & -0.08201351 & -0.1545357 & -0.69822252 \\ 0.90107623 & 2.51965115 & 0.10175686 & 1.43473445 & -0.56592598 \\ -0.08201351 & 0.10175686 & 5.58127669 & 2.37964385 & 0.81023582 \\ -0.1545357 & 1.43473445 & 2.37964385 & 3.27554611 & 0.67199657 \\ -0.69822252 & -0.56592598 & 0.81023582 & 0.67199657 & 1.26651214 \end{bmatrix}$ |
| 20 | $\begin{bmatrix} 0.9538235 \\ 0.2633515 \\ 0.6862135 \\ 1.024088 \\ 0.5066135 \end{bmatrix}$ | $\begin{bmatrix} 2.11375563 & 0.07809414 & -1.10411738 & 0.23504073 & -0.58704819 \\ 0.07809414 & 2.34635567 & 0.38326802 & 0.42207962 & -0.60376699 \\ -1.10411738 & 0.38326802 & 4.14694213 & 1.06496262 & -0.05221493 \\ 0.23504073 & 0.42207962 & 1.06496262 & 2.99026365 & 1.03303517 \\ -0.58704819 & -0.60376699 & -0.05221493 & 1.03303517 & 2.95185958 \end{bmatrix}$ |
| 50 | $\begin{bmatrix} 1.1573002 \\ 0.693935 \\ 0.6801964 \\ 1.0461014 \\ 0.8602632 \end{bmatrix}$ | $\begin{bmatrix} 2.13625588 & -0.04985265 & -0.94700026 & -0.10852518 & 0.03461513 \\ -0.04985265 & 2.58246167 & 0.44051035 & -0.29606067 & -0.40862959 \\ -0.94700026 & 0.44051035 & 3.49445255 & 0.78474267 & -0.12797897 \\ -0.10852518 & -0.29606067 & 0.78474267 & 3.52653203 & -0.68052625 \\ 0.03461513 & -0.40862959 & -0.12797897 & -0.68052625 & 4.26852977 \end{bmatrix}$ |
| 100 | $\begin{bmatrix} 0.9155417 \\ 1.0495909 \\ 1.1131177 \\ 0.9859483 \\ 0.9742111 \end{bmatrix}$ | $\begin{bmatrix} 2.11782654 & -0.30514948 & -0.62706059 & -0.29233987 & 0.11372288 \\ -0.30514948 & 2.4787017 & 0.41483605 & -0.32214878 & -0.25243613 \\ -0.62706059 & 0.41483605 & 3.55026825 & 0.30934848 & 0.40007647 \\ -0.29233987 & -0.32214878 & 0.30934848 & 3.89472464 & -0.52754656 \\ 0.11372288 & -0.25243613 & 0.40007647 & -0.52754656 & 4.31299664 \end{bmatrix}$ |
| 500 | $\begin{bmatrix} 0.88786128 \\ 0.95054512 \\ 1.00328014 \\ 0.96933674 \\ 0.8624077 \end{bmatrix}$ | $\begin{bmatrix} 2.596741 & -0.1914665 & -0.2804898 & 0.0952633 & 0.2032001 \\ -0.19146656 & 2.3824209 & 0.1631414 & 0.004478118 & -0.25560298 \\ -0.28048981 & 0.1631414 & 2.5084217 & -0.023833737 & 0.15085060 \\ 0.0952633 & 0.004478118 & -0.0238337 & 4.5644775 & -0.0587699 \\ 0.2032001 & -0.2556029 & 0.150850 & -0.0587699 & 4.67762324 \end{bmatrix}$ |
| 1000 | $\begin{bmatrix} 0.98968938 \\ 0.96046874 \\ 1.02593795 \\ 0.96955827 \\ 0.96543128 \end{bmatrix}$ | $\begin{bmatrix} 2.47152136 & -0.0603647 & -0.0993117 & 0.05964521 & 0.17211231 \\ -0.0603647 & 2.36574881 & 0.16616495 & -0.13259719 & -0.03389862 \\ -0.0993117 & 0.16616495 & 2.39662234 & -0.10202187 & 0.0520229 \\ 0.05964521 & -0.13259719 & -0.10202187 & 4.54422095 & 0.13574233 \\ 0.17211231 & -0.03389862 & 0.0520229 & 0.13574233 & 4.52061181 \end{bmatrix}$ |

(d) Compute and report the misclassification loss under the modified loss with the test set provided by the oracle for each Bayes classifier obtained with the different parameter estimates. Write down these losses in a table.

**Solution:**

To use the above estimates, let us simplify the Bayes classifier under the modified loss function.

$$h(x) = \begin{cases} 1 & \text{if } \frac{P(Y=0|X)}{P(Y=1|X)} < \frac{6}{5} \\ 0 & \text{otherwise} \end{cases}$$

We know,

$$P(Y=0|X) = \frac{P(X|Y=0)P(Y=0)}{P(X)} \tag{12}$$

and

$$P(Y=1|X) = \frac{P(X|Y=1)P(Y=1)}{P(X)} \tag{13}$$

Then,

$$\frac{P(Y=0|X)}{P(Y=1|X)} = \frac{P(X|Y=0)P(Y=0)}{P(X|Y=1)P(Y=1)} \tag{14}$$

From the output generated for test sets, it is clear that the number of outputs for class 0 is equally likely to be class 1. Therefore, $P(Y=0) = P(Y=1) = 0.5$.

$$\frac{P(Y=0|X)}{P(Y=1|X)} = \frac{P(X|Y=0)}{P(X|Y=1)}$$

Note that,

$$P(X|Y=0) = \frac{1}{(2\pi)^{\frac{5}{2}}|\Sigma_0|^{\frac{1}{2}}} e^{-\frac{1}{2}(X-\mu_0)\Sigma_0^{-1}(X-\mu_0)^T}$$

$$\frac{P(X|Y=0)}{P(X|Y=1)} = \frac{|\Sigma_1|^{\frac{1}{2}}}{|\Sigma_0|^{\frac{1}{2}}} e^{\frac{1}{2}\{\{(X-\mu_1)\Sigma_1^{-1}(X-\mu_1)^T\}-\{(X-\mu_0)\Sigma_0^{-1}(X-\mu_0)^T\}\}}$$

Thus, define a new classifier with these specification. Assuming non-diagonal entries of the classifier are zero,

| S.N | n | Number of misclassification | Misclassification loss | Percentage of misclassification |
|---|---|---|---|---|
| 1 | 2 | 43 | 250 | 21.5% |
| 2 | 10 | 19 | 99 | 9.5% |
| 3 | 20 | 18 | 95 | 9% |
| 4 | 50 | 16 | 85 | 8% |
| 5 | 100 | 14 | 76 | 7% |
| 6 | 500 | 16 | 87 | 8% |
| 7 | 1000 | 17 | 93 | 8.5% |

You are now provided 2000 samples each from 2 classes of the CIFAR10 dataset and asked to learn a classifier that minimizes the modified misclassification loss. Again, you build a classifier under assumptions (A1) and (A2). Note: Each image from CIFAR10 is represented as a 3072-dimensional vector. Please use the function q1_get_cifar10_train_test described below to obtain this dataset.

(e) Build a Bayes classifier under the modified loss function after computing class conditional estimates of the parameters and report the test accuracy on the test set provided.
**Solution:**
Since, the dimension of the vector is very large, there is a good chance that we might get OverflowError: math range error. Therefore, let's modify the classifier:

$$h(x) = \begin{cases} 1 & \text{if } \frac{P(Y=0|X)}{P(Y=1|X)} < \frac{6}{5} \\ 0 & \text{otherwise} \end{cases}$$
$$= \begin{cases} 1 & \text{if } log(\frac{P(X|Y=0)}{P(X|Y=1)}) < log(\frac{6}{5}) \\ 0 & \text{otherwise} \end{cases}$$

$$log(\frac{P(X|Y=0)}{P(X|Y=1)}) = log(\frac{|\Sigma_1|^{\frac{1}{2}}}{|\Sigma_0|^{\frac{1}{2}}} e^{\frac{1}{2}\{\{(X-\mu_1)\Sigma_1^{-1}(X-\mu_1)^T\}-\{(X-\mu_0)\Sigma_0^{-1}(X-\mu_0)^T\}\}})$$
$$= log(\frac{|\Sigma_1|^{\frac{1}{2}}}{|\Sigma_0|^{\frac{1}{2}}}) + \frac{1}{2}\{\{(X-\mu_1)\Sigma_1^{-1}(X-\mu_1)^T\} - \{(X-\mu_0)\Sigma_0^{-1}(X-\mu_0)^T\}\}$$
$$= \frac{1}{2}log(\frac{|\Sigma_1|}{|\Sigma_0|}) + \frac{1}{2}\{\{(X-\mu_1)\Sigma_1^{-1}(X-\mu_1)^T\} - \{(X-\mu_0)\Sigma_0^{-1}(X-\mu_0)^T\}\}$$

The number of misclassification and misclassification loss under same schemes as before is 861 and 4757 respectively. The Percentage of misclassification is,

$$\frac{861}{4000} \times 100 = 21.525\%$$

and the test accuracy is, 78.475%

2. **Perceptron Algorithm**
Recall that the Perceptron algorithm is guaranteed to converge when the data is linearly separable. You will need to implement the perception algorithm. Use the Oracle q2_perceive(srn) to obtain data D.
Report the following:

(a) The value of w obtained using the perceptron algorithm
**Solution:**
The value of w obtained using the perceptron algorithm is

$$\begin{bmatrix} 34.73074 & 34.87084 & 35.18411 & 20.89604 & 20.79986 \end{bmatrix}$$

(b) Number of errors made by the algorithm.
**Solution:**
The number of errors made by the algorithm is 1573.

(c) Report the Margin given by your final classifier.
**Solution:**
The margin given by the final classifier is 0.0001951635771629387.

(d) Radius of the data set.

$$R = \max_{i \in (1,|D|)} ||x_i||$$

**Solution:**
The radius of the data set is 0.40944837305819154.

Use the Oracle q2_mnist(srn) to obtain linearly separable samples from the mnist data $D_{mnist}$. Run the perceptron algorithm on this dataset. Report the following:

(e) The value of w obtained using the perceptron algorithm. You must report this vector in a comma-separated file named AIML_2024_A1_LastFiveDigitsOfSRNumber_q2_w.csv

(f) Number of errors made by the algorithm.
**Solution:**
The number of errors made by the algorithm is 279

(g) Report the Margin given by your final classifier.
**Solution:**
The margin given by the final classifier is 2.5894659108547295

(h) Radius of the data set.
**Solution:**
The radius of the data set is 1195.2727722156144.

3. One way to view a linear classification model is dimensionality reduction. As seen in class, the Fisher Linear discriminant projects the data into lower dimensions, ensuring a large separation between the projected class means while giving a small variance within each class, thereby minimizing the class overlap. The Fisher Linear Discriminant is given by: $w^T x = b$ where w is the Projection vector, and b is the threshold for classification.

(a) **Fisher Discriminant on IRIS Dataset**
IRIS dataset is one of the earliest datasets available on UCI Machine Learning repository. Also known as Fisher's Iris data set is a multivariate data set used and made famous by the British statistician and biologist Ronald Fisher in his 1936 paper "The use of Multiple Measurements in Taxonomic Problems" as an example of linear discriminant analysis. It is used in the literature on classification methods and is widely used in statistics and machine learning. The data set contains three classes of 50 instances each, where each class refers to a type of iris plant.
Your task is to create a Fisher linear classifier on the iris dataset. You will be given the train set of the iris dataset containing two classes, namely 'setosa' and 'vertsicolor'.

i. Find the normalized projection vector w as per the idea proposed in Fisher discriminant to project the data to one dimension and the threshold b to classify the two classes such that the error on train data is minimal. Report the vector w and the threshold b in the PDF file.
**Solution:**
We want to find w such that the projected data is linearly separable. i.e.

$$\hat{w} = argmax_{w \in R^d} \frac{|w^T(\mu_1 - \mu_2)|^2}{w^T \Sigma_1 w + w^T \Sigma_2 w}$$
$$= argmax_{w \in R^d} \frac{w(\mu_1 - \mu_2)(\mu_1 - \mu_2)^T w}{w^T(\Sigma_1 + \Sigma_2) w}$$

Let $S_b = (\mu_1 - \mu_2)(\mu_1 - \mu_2)^T$ and $S_w = \Sigma_1 + \Sigma_2$. Then, the above equation can be written as:

$$\hat{w} = argmax_{w \in R^d} \frac{w S_b w^T}{w S_w w^T}$$

The solution to the above equation is the eigenvector corresponding to the largest eigenvalue of $S_b w = \lambda S_w w$.
We know,

$$\hat{w} = S_w^{-1}(\mu_1 - \mu_2)$$

Then, the normalized projection vector w is:

$$\begin{bmatrix} 0.11332533 \\ 0.60270643 \\ -0.32326528 \\ -0.72069542 \end{bmatrix}$$

and the threshold b is 0.9964615551352254.

ii. Plot the histogram of the projected data with different colors for each class. Also, plot the classifier boundary obtained using the above threshold on the same plot. Add this plot in the PDF File.
**Solution:**
The histogram of the projected data with different colors for each class is as follows:

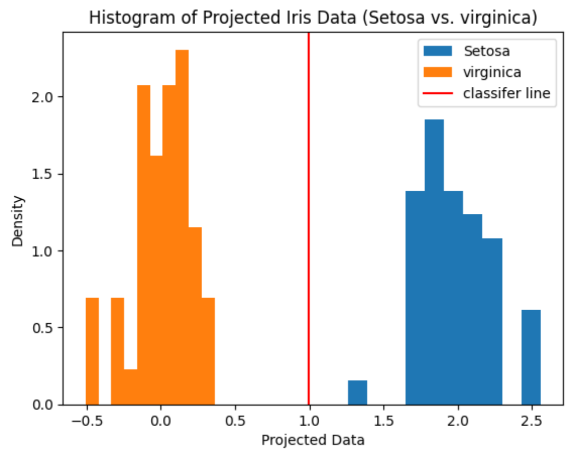


Figure 1: Histogram of the projected data

(b) **Fisher Discriminant on Dataset given by Oracle** The Oracle file provided to you, AIML_A1.py, contains the following functions related to question 3.
q3_get_data(srn): returns training dataset. It returns a list containing all the data points. Each entry of the list is a tuple of two elements; the first element is the 2-dimensional data $[x_1, x_2]$, and the second element is the label (1 or 0).
q3_get_test_data(srn): returns test dataset. It returns a list containing all the data points. Each entry of the list is a tuple of two elements; the first element is the 2-dimensional data $[x_1, x_2]$, and the second element is the label (1 or 0).
Answer the following questions.

i. Find the normalized projection vector w using fisher discriminant. plot the histogram of projected data. Give different colors for both classes. Is the projected data is linearly separable?
**Solution:**
The normalized projection vector w using fisher discriminant is:

$$\begin{bmatrix} 0.66436384 \\ 0.74740932 \end{bmatrix}^T$$

and the threshold b is 2.7832574279571255e-17 The histogram of the projected data with different colors for each class is as follows: The projected data is not linearly separable.

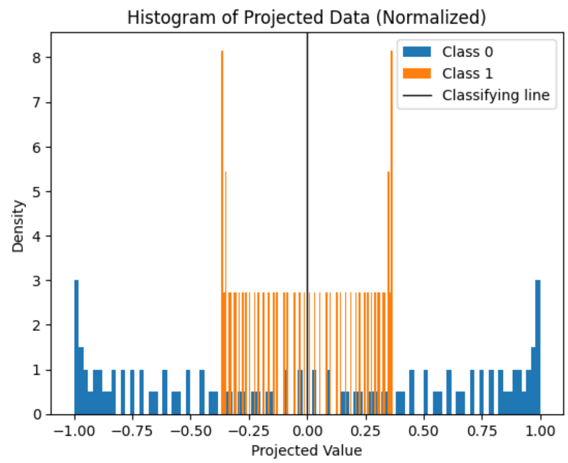


Figure 2: Histogram of the projected data

ii. Add two more dimensions to each data point to get a 4-dimensional dataset. call it as $\phi(x)$
example: if $x = [x_1, x_2]$ then $\phi(x) = [x_1, x_2, x_1^2, x_2^2]$ Now use this new dataset $\phi(x)$ with the same labels to find the normalized projection vector w. Is this new data linearly separable? If the data is linearly separable, Find the threshold b such that the error on the training dataset is minimal. Report the vector w and the threshold b in the PDF file.
**Solution:**
The new data is linearly separable as seen in the picture below: The normalized

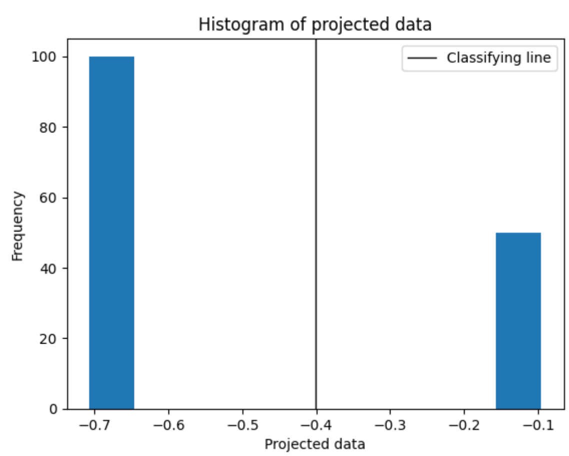


Figure 3: Histogram of the projected data

projection vector w using fisher discriminant is:

$$\begin{bmatrix} -1.99245292e-18 \\ -6.52075501e-18 \\ -7.07106781e-01 \\ -7.07106781e-01 \end{bmatrix}^T$$

and the threshold b is -0.4014389843833197.

iii. Plot original data and the classifier boundary $w^T\phi(x) = b$. If any entry of w is less than $10^{-6}$, you can assume it to be zero. Add this plot to the PDF file. [Hint : Ideally, the first two entries $w_0$ and $w_1$ should be close to zero, hence we can ignore them.]

**Solution:**

From the last define the classifier boundary $w^T\phi(x) = b$ as:

$$\begin{bmatrix} 0 & 0 & -0.707106781 & -0.707106781 \end{bmatrix}$$

With this new w, the histogram should look like following:

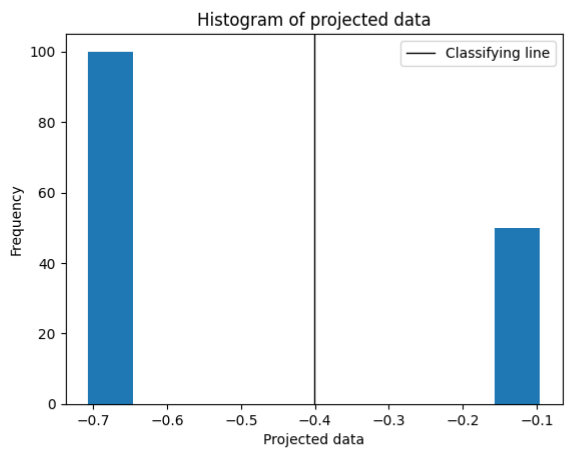


Figure 4: Histogram of the projected data

and the threshold is -0.40143898 Similarly, the classifier boundary $w^T\phi(x) = b$ can be rewritten as follows:

$$\begin{bmatrix} 0 & 0 & -0.707106781 & -0.707106781 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_1^2 \\ x_2^2 \end{bmatrix} = -0.40143898$$

$$0 + 0 + (-0.707106781)x_1^2 + (-0.707106781)x_2^2 = 0.40143898$$

$$x_1^2 + x_2^2 = \frac{-0.40143898}{-0.707106781}$$

$$x_1^2 + x_2^2 = 0.567$$

and the plot of the classifier boundary is as follows:

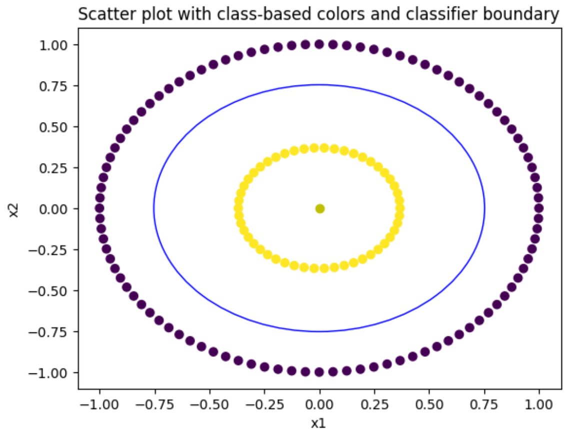


Figure 5: Histogram of the projected data

iv. Load the test dataset using the Oracle function. Find test accuracy with thresholds = [b -0.3,b-0.2,b-0.1,b,b+0.1,b+0.2,b+0.3] where b is threshold found above. Report these values as a table in the PDF file.
**Solution:**
The test accuracy with thresholds = [b -0.3,b-0.2,b-0.1,b,b+0.1,b+0.2,b+0.3] is as follows:

| Threshold | Mistakes Count | Test Accuracy |
|---|---|---|
| b-0.3 | 20 | 80% |
| b-0.2 | 15 | 85% |
| b-0.1 | 7 | 93% |
| b | 7 | 93% |
| b+0.1 | 6 | 94% |
| b+0.2 | 13 | 87% |
| b+0.3 | 30 | 70% |

(c) **Logistic regression:**
Answer the following questions:

i. Perform K-Fold Cross Validation with K = 6 and run logistic regression on the dataset using scikit-learn.
**Solution:**
The K-Fold Cross Validation with K = 6 and run logistic regression on the dataset using scikit-learn is as follows:
K-Fold Cross-Validation Scores: [0.75, 0.81, 0.72, 0.78, 0.76, 0.85]
Mean Accuracy: 0.7783333333333333

ii. Evaluate the K models using a confusion matrix; for more details, check here and calculate the following metrics:

A. Recall: Proportion of true positive predictions among all actual positive instances

B. Precision: Proportion of true positive predictions among all positive predictions

C. Accuracy: Proportion of correctly classified instances among the total instances

D. F1-score: Harmonic mean of precision and recall, providing a balanced measure of model perfor mance

**Solution:**
The confusion matrix is as follows:

$$\begin{bmatrix} \begin{bmatrix} 54 & 9 \\ 16 & 21 \end{bmatrix} \\ \\ \begin{bmatrix} 59 & 3 \\ 16 & 22 \end{bmatrix} \\ \\ \begin{bmatrix} 48 & 13 \\ 15 & 24 \end{bmatrix} \\ \\ \begin{bmatrix} 57 & 5 \\ 17 & 21 \end{bmatrix} \\ \\ \begin{bmatrix} 60 & 10 \\ 14 & 16 \end{bmatrix} \\ \\ \begin{bmatrix} 69 & 5 \\ 10 & 16 \end{bmatrix} \end{bmatrix}$$

The metrics are as follows:

A. Recall: [0.567, 0.579, 0.615, 0.553, 0.533, 0.615]

B. Precision: [0.7, 0.88, 0.648, 0.807, 0.615, 0.761]

C. Accuracy: [0.75, 0.81, 0.72, 0.78, 0.76, 0.85]

D. F1-score:[0.627, 0.698, 0.631, 0.65625, 0.571, 0.680]

iii. Plot barplot of all Metrics for all K Models (Make every metric in [0,1] ).
**Solution:**
The barplot of all Metrics for all K Models is as follows:

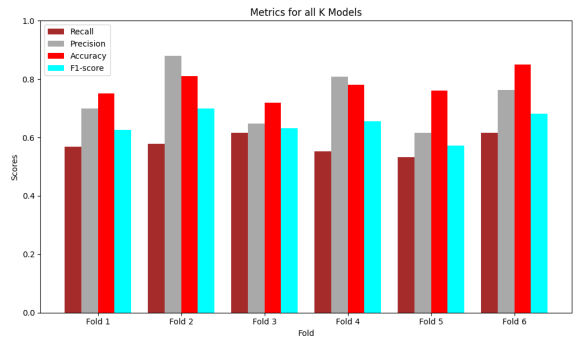


Figure 6: Barplot of all Metrics for all K Models

iv. Apply the trained model to the test data provided on the day of submission(25th Feb 2024) and submit the test results as AIML_2024_A1_LastFiveDigitsOfSRNumber_q4.csv
**Solution:**
The results are saved in the file as instructed.