Machine Learning HW2

109550159 李驊恩

Coding Part

1. the mean vectors mi (i=1, 2) of each 2 classes on training data

```
mean vector of class 1: [ 0.99253136 -0.99115481] mean vector of class 2: [-0.9888012   1.00522778]
Within-class scatter matrix SW: [[ 4337.38546493 -1795.55656547]
 [-1795.55656547  2834.75834886]]
Between-class scatter matrix SB: [[ 3.92567873 -3.95549783]
 [-3.95549783  3.98554344]]
Fisher’s linear discriminant: [-0.37003809  0.92901658]
Accuracy of test-set 0.8912
```

2. the within-class scatter matrix SW on training data

```
mean vector of class 1: [ 0.99253136 -0.99115481] mean vector of class 2: [-0.9888012   1.00522778]
Within-class scatter matrix SW: [[ 4337.38546493 -1795.55656547]
 [-1795.55656547  2834.75834886]]
Between-class scatter matrix SB: [[ 3.92567873 -3.95549783]
 [-3.95549783  3.98554344]]
Fisher’s linear discriminant: [-0.37003809  0.92901658]
Accuracy of test-set 0.8912
```

3. the between-class scatter matrix SB on training data

```
mean vector of class 1: [ 0.99253136 -0.99115481] mean vector of class 2: [-0.9888012   1.00522778]
Within-class scatter matrix SW: [[ 4337.38546493 -1795.55656547]
 [-1795.55656547  2834.75834886]]
Between-class scatter matrix SB: [[ 3.92567873 -3.95549783]
 [-3.95549783  3.98554344]]
Fisher’s linear discriminant: [-0.37003809  0.92901658]
Accuracy of test-set 0.8912
```

4. the Fisher’s linear discriminant w on training data

```
mean vector of class 1: [ 0.99253136 -0.99115481] mean vector of class 2: [-0.9888012   1.00522778]
Within-class scatter matrix SW: [[ 4337.38546493 -1795.55656547]
 [-1795.55656547  2834.75834886]]
Between-class scatter matrix SB: [[ 3.92567873 -3.95549783]
 [-3.95549783  3.98554344]]
Fisher’s linear discriminant: [-0.37003809  0.92901658]
Accuracy of test-set 0.8912
```

5. accuracy score on testing data with K values from 1 to 5

K = 5, Accuracy = 0.8912 `Accuracy of test-set 0.8912`

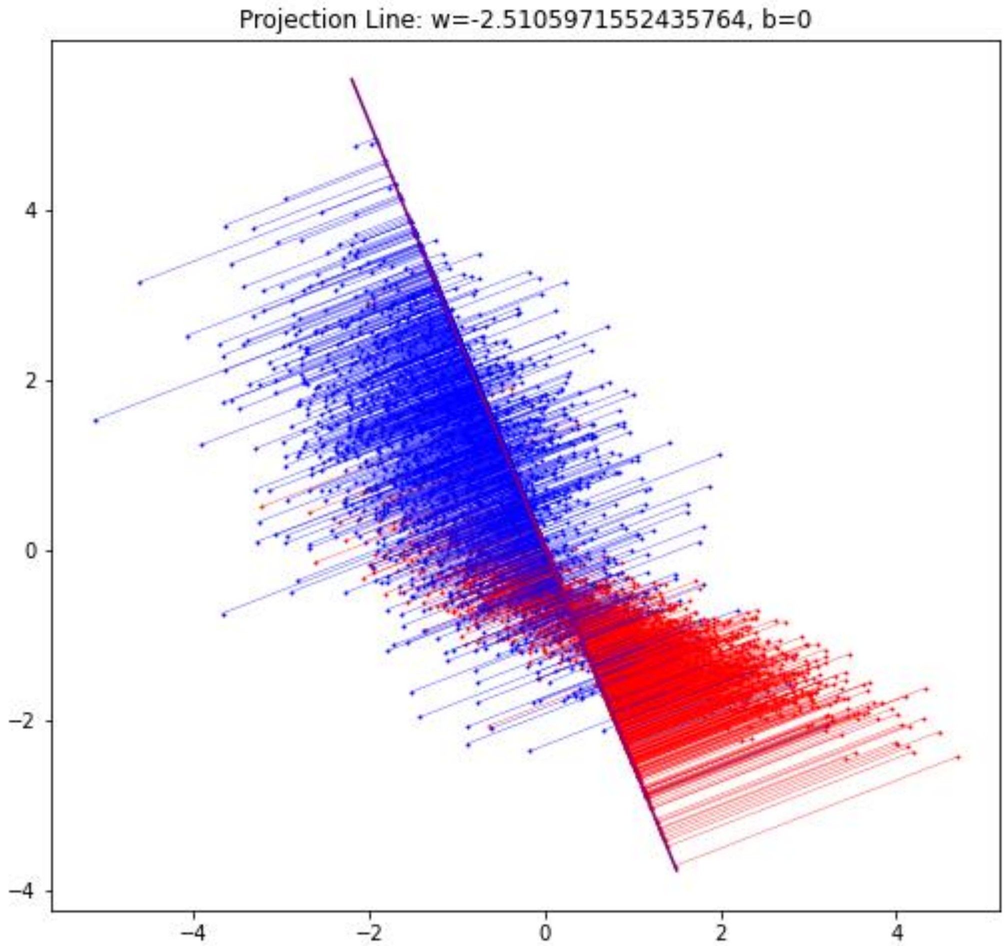
K = 4, Accuracy = 0.8824 `Accuracy of test-set 0.8824`

K = 3, Accuracy = 0.8792 `Accuracy of test-set 0.8792`

K = 2, Accuracy = 0.8704 `Accuracy of test-set 0.8704`

K = 1, Accuracy = 0.8488 `Accuracy of test-set 0.8488`

6.



Question Part

1. What's the difference between the Principle Component Analysis and Fisher’s Linear Discriminant?

Principle Component Analysis is an unsupervised technique to reduce dimension. In PCA, we have to find a projection axis to maximize the data separation after projecting. We don’t have to know what class the data belong to in PCA. In Fisher’s Linear Discriminant, we want the separation of the data to be maximized after projection similarly. What’s difference is that we want the separation of the data ” between each classes “ as large as possible. Thus, FLD is a supervised learning technique since we consider separation between different classes of data, which is different from PCA.

2. Please explain in detail how to extend the 2-class FLD into multi-class FLD (the number of classes is greater than two).

2.

To extend the 2-class FLD into multiclass FLD, we first

assume that the dimension of input space is D, $D > K > 2$

Then linear weight vectors $y_k = W_k^T x$ where $k = 1, \dots D'$, $D' \geq 1$

$\Rightarrow y = W^T x$ where weight vectors $\{W_k\}$ are columns of W

The within-class covariance matrix then becomes:

$$S_W = \sum_{k=1}^{K} S_k \quad \text{where } S_k = \sum_{n \in C_k} (x_n - m_k)(x_n - m_k)^T, \quad m_k = \frac{1}{N_k} \sum_{n \in C_k} x_n$$

N_k is the number of pattern in class C_k

The between-class covariance matrix then becomes:

$$S_B = \sum_{k=1}^{K} N_k (m_k - m)(m_k - m)^T, \quad m = \frac{1}{N} \sum_{n=1}^{N} x_n$$

However, we can't directly extend the objective that $J(w) = \frac{W^T S_B W}{W^T S_W W}$.

because W and W^T are no longer a single vector. They are matrices that gather the weights vector together and projects each data to a D'-dimensional space.

A better choice for objective is $J(w) = Tr\{(W S_W W^T)^{-1} (W S_B W^T)\}$

As for D', the dimension of the projected space by FLD is at most $K-1$ because the rank of the between-class covariance matrix S_B is at most $K-1$.

3.

3.

By $s_k^2 = \sum_{n \in C_k} (y_n - m_k)^2$, $y = W^T x$, $m_k = W^T m_k$,

$$s_1^2 = \sum_{n \in C_1} (y_n - m_1)^2 = \sum_{n \in C_1} (W^T x_n - W^T m_1)^2 = \sum_{n \in C_1} [W^T(x_n - m_1)]^T [W^T(x_n - m_1)]$$

$$= \sum_{n \in C_1} [(x_n - m_1)^T W]^T [(x_n - m_1)^T W] = \sum_{n \in C_1} W^T (x_n - m_1)(x_n - m_1)^T W$$

$$= W^T \sum_{n \in C_1} (x_n - m_1)(x_n - m_1)^T W$$

And thus $s_2^2 = W^T \sum_{n \in C_2} (x_n - m_2)(x_n - m_2)^T W$

Since $S_W = \sum_{n \in C_1} (x_n - m_1)(x_n - m_1)^T + \sum_{n \in C_2} (x_n - m_2)(x_n - m_2)^T$,

$$s_1^2 + s_2^2 = W^T S_W W$$

By $m_2 - m_1 = W^T(m_2 - m_1)$,

$$(m_2 - m_1)^2 = [W^T(m_2 - m_1)]^2 = W^T (m_2 - m_1)(m_2 - m_1)^T W$$

Since $S_B = (m_2 - m_1)(m_2 - m_1)^T$, $(m_2 - m_1)^2 = W^T S_B W$

By $s_1^2 + s_2^2 = W^T S_W W$ and $(m_2 - m_1)^2 = W^T S_B W$,

$$J(w) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2} = \frac{W^T S_B W}{W^T S_W W} \#$$

4.

4.

By chain rule, $\frac{\partial E}{\partial a_k} = \frac{\partial y_k}{\partial a_k} \cdot \frac{\partial E}{\partial y_k}$

$$\frac{\partial y_k}{\partial a_k} = \frac{\partial}{\partial a_k}\sigma(a_k) = \frac{\partial}{\partial a_k}\left(\frac{1}{1+e^{-a_k}}\right) = \frac{1}{(1+e^{-a_k})^2} \cdot e^{-a_k} = \frac{1}{1+e^{-a_k}}\left(1 - \frac{1}{1+e^{-a_k}}\right)$$

$$= \sigma(a_k)[1-\sigma(a_k)] = y_k(1-y_k)$$

$$\frac{\partial E}{\partial y_k} = \frac{\partial}{\partial y_k}\Big[-(t_1 \ln y_1 + (1-t_1)\ln(1-y_1)) - (t_2 \ln y_2 + (1-t_2)\ln(1-y_2)) - \cdots$$

$$-(t_k \ln y_k + (1-t_k)\ln(1-y_k)) \cdots - (t_N \ln y_N + (1-t_N)\ln(1-y_N))\Big]$$

$$= 0 + 0 + \cdots + \frac{\partial}{\partial y_k}\Big[-(t_k \ln y_k + (1-t_k)\ln(1-y_k))\Big] + \cdots + 0$$

$$= -\left(\frac{t_k}{y_k} + \frac{1-t_k}{1-y_k}\cdot(-1)\right) = -\frac{t_k}{y_k} + \frac{1-t_k}{1-y_k}$$

$$\Rightarrow \frac{\partial E}{\partial a_k} = \frac{\partial y_k}{\partial a_k} \cdot \frac{\partial E}{\partial y_k} = (y_k(1-y_k)) \cdot \left(-\frac{t_k}{y_k} + \frac{1-t_k}{1-y_k}\right) = -(1-y_k)t_k + y_k(1-t_k)$$

$$= -t_k + y_k t_k + y_k - y_k t_k = y_k - t_k$$

#

5.

5.

$y_k(x, w) = p(t_k = 1 | x)$

$p(t | w_1, \cdots w_K) = \prod_{k=1}^{K} y_k^{t_k}$

$\Rightarrow$ for data set of N points: $p(T | w_1, \cdots w_K) = \prod_{n=1}^{N} \prod_{k=1}^{K} y_{nk}^{t_{nk}}$

To derive the error function, take the negative logarithm:

$$-\ln[p(T | w_1, \ldots w_K)] = -\ln\left(\prod_{n=1}^{N} \prod_{k=1}^{K} y_{nk}^{t_{nk}}\right)$$

$$= -\ln\left(\prod_{n=1}^{N} y_{n1}^{t_{n1}} y_{n2}^{t_{n2}} \cdots y_{nK}^{t_{nK}}\right)$$

$$= -\ln(y_{11}^{t_{11}} \cdots y_{1K}^{t_{1K}})(y_{21}^{t_{21}} \cdots y_{2K}^{t_{2K}}) \cdots (y_{N1}^{t_{N1}} \cdots y_{NK}^{t_{NK}})$$

$$= -(t_{11} \ln y_{11} + \cdots t_{1K} \ln y_{1K} + \cdots t_{N1} \ln y_{N1} + \cdots t_{NK} \ln y_{NK})$$

$$= -\sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} \ln y_{nk}$$

$$= -\sum_{n=1}^{N} \sum_{k=1}^{K} t_{nk} \ln y_k(x_n, w) = E(w)$$

#