Yohandi 120040025

DDA3020 Assignment 1

1.1 Suppose:

X is a $m \times m$ vector

w is a $m \times 1$ vector

y is a $m \times 1$ vector

$$\Rightarrow X^T w = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mm} \end{bmatrix}^T \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}$$

$$= \begin{bmatrix} x_{11} w_1 + x_{21} w_2 + \dots + x_{m1} w_m \\ x_{12} w_1 + x_{22} w_2 + \dots + x_{m2} w_m \\ \vdots \\ x_{1m} w_1 + x_{2m} w_2 + \dots + x_{mm} w_m \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{m} x_{i1} w_i \\ \sum_{i=1}^{m} x_{i2} w_i \\ \vdots \\ \sum_{i=1}^{m} x_{im} w_i \end{bmatrix}$$

$$\frac{d(X^T w)}{dw} = \begin{bmatrix} \frac{\partial(\sum_{i=1}^{m} x_{i1} w_i)}{\partial w_1} & \cdots & \frac{\partial(\sum_{i=1}^{m} x_{im} w_i)}{\partial w_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial(\sum_{i=1}^{m} x_{i1} w_i)}{\partial w_m} & \cdots & \frac{\partial(\sum_{i=1}^{m} x_{im} w_i)}{\partial w_m} \end{bmatrix}$$

$$= \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & & x_{2m} \\ \vdots & & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mm} \end{bmatrix}$$

$$= X$$

$$\Rightarrow y^T X w = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}^T \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mm} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{m} y_i x_{i1} & \cdots & \sum_{i=1}^{m} y_i x_{im} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}$$

$$= w_1 \sum_{i=1}^{m} y_i x_{i1} + w_2 \sum_{i=1}^{m} y_i x_{i2} + \dots + w_m \sum_{i=1}^{m} y_i x_{im}$$

$$= \sum_{j=1}^{m} w_j \sum_{i=1}^{m} y_i x_{ij}$$

$$\frac{d(y^T X w)}{dw} = \begin{bmatrix} \frac{\partial(\sum_{j=1}^{m} w_j \sum_{i=1}^{m} y_i x_{ij})}{\partial w_1} \\ \vdots \\ \frac{\partial(\sum_{j=1}^{m} w_j \sum_{i=1}^{m} y_i x_{ij})}{\partial w_m} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{m} x_{i1} y_i \\ \vdots \\ \sum_{i=1}^{m} x_{im} y_i \end{bmatrix}$$

$$= \begin{bmatrix} x_{11} & x_{21} & \cdots & x_{m1} \\ x_{12} & x_{22} & \cdots & x_{m2} \\ \vdots & & \ddots & \vdots \\ x_{1m} & x_{2m} & \cdots & x_{mm} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}$$

$$= X^T y$$

$$\Rightarrow w^T X w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}^T \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mm} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{m} w_i x_{i1} & \cdots & \sum_{i=1}^{m} w_i x_{im} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}$$

$$= w_1 \sum_{i=1}^{m} w_i x_{i1} + w_2 \sum_{i=1}^{m} w_i x_{i2} + \dots + w_m \sum_{i=1}^{m} w_i x_{im}$$

$$= \sum_{j=1}^{m} w_j \sum_{i=1}^{m} w_i x_{ij}$$

$$= \sum_{i=1}^{m} \sum_{j=1}^{m} (w_i w_j) x_{ij}$$

$$\frac{d(w^T X w)}{dw} = \begin{bmatrix} \dfrac{\partial(\sum_{i=1}^{m}\sum_{j=1}^{m}(w_i w_j)\, x_{ij})}{\partial w_1} \\ \vdots \\ \dfrac{\partial(\sum_{i=1}^{m}\sum_{j=1}^{m}(w_i w_j)\, x_{ij}}{\partial w_m} \end{bmatrix}$$

$$= \begin{bmatrix} 2w_1x_{11} + w_2x_{12} + w_2x_{21} + \dots + w_m x_{m1} \\ w_1x_{12} + w_1x_{21} + 2w_2x_{22} + \dots + w_m x_{m2} \\ \vdots \\ w_1x_{1m} + w_1x_{m1} + w_2x_{2m} + \dots + 2w_m x_{mm} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_{i=1}^{m}(x_{i1}+x_{1i})\, w_i \\ \vdots \\ \sum_{i=1}^{m}(x_{im}+x_{mi})\, w_i \end{bmatrix}$$

$$= \begin{bmatrix} (X_{11}+X_{11}) & (X_{12}+X_{21}) & \cdots & (X_{1m}+X_{m1}) \\ (X_{21}+X_{12}) & (X_{22}+X_{22}) & & (X_{2m}+X_{m2}) \\ \vdots & & \ddots & \vdots \\ (X_{m1}+X_{1m}) & (X_{m2}+X_{2m}) & \cdots & (X_{mm}+X_{mm}) \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_m \end{bmatrix}$$

$$= (X+X^T)\, w$$

1.2(1) We have a target function $\sum_{i=1}^{N}\alpha_i\|y_i-Wx_i-b\|^2$. Assume $y_i\in\mathbb{R}^{k\times 1}$ and $x_i\in\mathbb{R}^{d\times 1}$

Rewrite as matrix:

$$\sum_{i=1}^{N}\alpha_i(y_i-Wx_i-b)(y_i-Wx_i-b)^T$$

$$=\sum_{j=1}^{N}\alpha_i(y_i-\bar{w}\bar{x_i})(y_i-\bar{w}\bar{x_i})^T,$$

where $\bar{w}=[b;W]$ and $\bar{x_i}=(1,x_i^T)^T$

$$=\sum_{j=1}^{N}\alpha_i(y_i y_i^T-2\bar{w}\bar{x_i}y_i^T+\bar{w}\bar{x_i}\bar{x_i}^T\bar{w}^T)$$

$$=\sum_{j=1}^{N}\alpha_i y_i y_i^T-2\bar{w}\sum_{i=1}^{N}\alpha_i\bar{x_i}y_i^T+\bar{w}\left(\sum_{i=1}^{N}\alpha_i\bar{x_i}\bar{x_i}^T\right)\bar{w}^T$$

where $X=[\bar{x_1},\bar{x_2},\dots,\bar{x_N}]$

suppose $Y=[y_1,y_2,\dots,y_N]$ and $\alpha=\begin{bmatrix}\alpha_1&0&\cdots&0\\0&\alpha_2&\cdots&0\\\vdots&\vdots&\ddots&\vdots\\0&0&\cdots&\alpha_N\end{bmatrix}$,

then the function

$$=Y\alpha Y^T-2\bar{w}X\alpha Y^T+\bar{w}X\alpha X^T\bar{w}^T$$

denote the function as $f(\bar{w})$, then differentiate

$$\frac{\partial f(\bar{w})}{\partial\bar{w}^T}=-2X\alpha Y^T+(X\alpha^T X^T)\bar{w}^T+(X\alpha X^T)\bar{w}^T$$

since $\alpha^T=\alpha$

$$\Rightarrow -2X\alpha Y^T+2(X\alpha X^T)\bar{w}^T=0$$

$$\Rightarrow \bar{w}^T=(X\alpha X^T)^{-1}(X\alpha Y^T)$$

$$\Rightarrow \bar{w}=Y\alpha X^T(X\alpha X^T)^{-1}$$

$$\Rightarrow [b|W]=Y\alpha X^T(X\alpha X^T)^{-1}$$

(2) $\bar{w}^T$ can be updated by gradient descent algorithm,

$$\bar{w}^T\leftarrow\bar{w}^T-\beta(-2X\alpha Y^T+2(X\alpha X^T)\bar{w}^T)$$

where β is step-size

1.3 (1) $f(x)=x^4$, $0\le\theta\le1$

$$f(\theta p+(1-\theta)q)=(\theta p+(1-\theta)q)^4$$
$$=((\theta p+(1-\theta)q)^2)^2$$
$$\le(\theta p^2+(1-\theta)q^2)^2$$
$$\le \theta p^4+(1-\theta)q^4$$
$$=\theta f(p)+(1-\theta)f(q)$$

(proved as convex)

(2) $f(x)=|x|$, $0\le\theta\le1$

$$f(\theta p+(1-\theta)q)=|\theta p+(1-\theta)q|$$
$$\le|\theta p|+|(1-\theta)q|$$
$$=\theta|p|+(1-\theta)|q|$$
$$=\theta f(p)+(1-\theta)f(q)$$

(proved as convex)

(3) $f(x)=\|Ax-b\|^2$

$$=(Ax-b)^T(Ax-b)$$
$$\nabla f(x) : 2A^T(Ax-b)$$
$$\nabla^2 f(x) : 2A^TA$$

for Any A, $\nabla^2 f(x)\succeq 0$ for all $x\in \mathrm{dom} f$

(proved as convex)

1.4 Suppose $L(\mu,\sigma)$ denotes the likelihood function,

we have: $L(\mu,\sigma)=\prod_{i=1}^{n} f(x_i;\mu,\sigma)$

$$=\prod_{i=1}^{n}\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2}$$

$$\Rightarrow \ln(L(\mu,\sigma))=\sum_{i=1}^{n}\left(\ln\left(\frac{1}{\sigma\sqrt{2\pi}}\right)+\left(-\frac{1}{2}\left(\frac{x_i-\mu}{\sigma}\right)^2\right)\right)$$
$$=-n\ln(\sigma\sqrt{2\pi})-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i-\mu)^2$$

to maximize $\ln(L(\mu,\sigma))$, we derivate it firstly to μ :

$$\frac{\partial(\ln(L(\mu,\sigma)))}{\partial\mu}=-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(-2(x_i-\mu))$$
$$=\frac{1}{\sigma^2}\sum_{i=1}^{n}(x_i-\mu)=0$$
$$\Rightarrow \mu_{MLE}=\frac{\sum_{i=1}^{n}x_i}{n}$$

next, derivate to σ and put μ as μ_{MLE}

$$\frac{\partial(\ln(L(\mu,\sigma)))}{\partial\sigma}=-\frac{n}{\sigma}+\frac{1}{\sigma^3}\sum_{i=1}^{n}(x_i-\mu)^2=0$$
$$\Rightarrow \sigma^2_{MLE}=\frac{\sum_{i=1}^{n}(x_i-\mu_{MLE})^2}{n}=\frac{1}{n}\sum_{i=1}^{n}\left(x_i-\frac{\sum_{i=1}^{n}x_i}{n}\right)^2$$

Programming Question

Yohandi [SID: 120040025]

Regression

According to the question, there is a file named `Regression.csv`, which contains a dataset for regression. There are about 7750 samples with 25 features that are going to be used as the next-day temperature-bound prediction. This data is for the purpose of bias correction of next-day maximum and minimum air temperatures forecast of the LDAPS model operated by the Korea Meteorological Administration over Seoul, South Korea. This data consists of summer data from 2013 to 2017. The input data is largely composed of the LDAPS model's next-day forecast data, in-situ maximum and minimum temperatures of present-day, and geographic auxiliary variables. There are two outputs (i.e. next-day maximum and minimum air temperatures) in this data.

Hyperparameter Settings

In this task, we are firstly asked to remove both the station and date attributes in the collected data. This simply can be done with a `drop()` function provided in the `pandas` library.

Data Loading

The program uses `read_csv()` function, which is provided in the `pandas` library, to read the information in the file. Notice that the loading of the data is using relative path instead of absolute path. Hence, when executing the code of the model, one should place the CSV file under the same directory of the model python file.

In the provided data, there occurs to be some missing values (NaN) which preclude the learned information. Usually, to solve this, there are two common methods. The first common method is to impute the data with relevant information such as mean, median, or mode. The second common method is to drop the data that lacks of information or irrelevant. In this program, the second method is used.

After read in, the dataset is split into two parts. The first part is the first 21 columns (after excluded the first two columns), which acts as the input feature values. Denote this part as input data X. The second part is the last 2 columns, which serve as the actual label value. Denote this part as Y. As required, 80% of the data will be used as training data and the rest of the data will be used as testing data.

Training

According to Lecture 5 - Linear Regression, the closed-form solution for the learning model is $W = (X^T X)^{-1} X^T Y$. The model is obtained according to the value of X and Y that are already loaded previously.

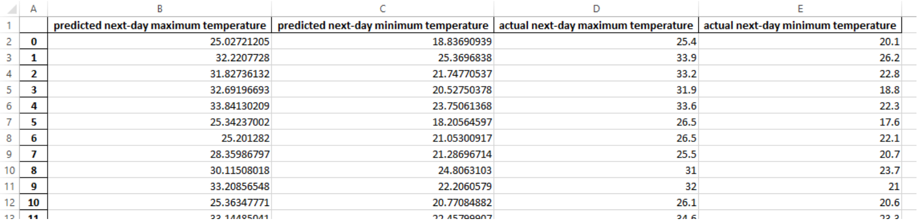
Testing

With the previous learning model, the prediction data is obtained with $f_{w,b}(X_{new}) = (X_{new}W)$.

Analysis

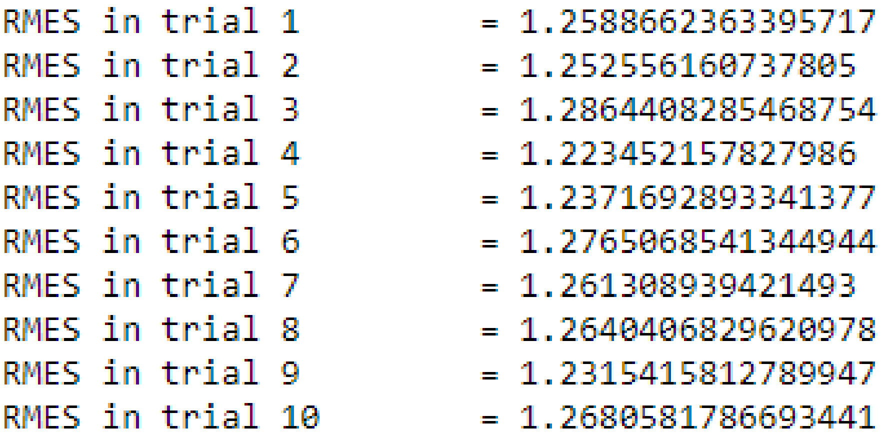
In 10 trials, each result is printed to an excel with the format name `regression_result_trial_{number}.xlsx`. In the submission, the files are in the `results` folder.

`regression_result_trial_1.xlsx` shows as following:



	A	B	C	D	E
1		predicted next-day maximum temperature	predicted next-day minimum temperature	actual next-day maximum temperature	actual next-day minimum temperature
2	0	25.02721205	18.83690939	25.4	20.1
3	1	32.2207728	25.3696838	33.9	26.2
4	2	31.82736132	21.74770537	33.2	22.8
5	3	32.69196693	20.52750378	31.9	18.8
6	4	33.84130209	23.75061368	33.6	22.3
7	5	25.34237002	18.20564597	26.5	17.6
8	6	25.201282	21.05300917	26.5	22.1
9	7	28.35986797	21.28696714	25.5	20.7
10	8	30.11508018	24.8063103	31	23.7
11	9	33.20856548	22.2060579	32	21
12	10	25.36347771	20.77084882	26.1	20.6
13	11	33.14485041	22.45799907	34.6	23.3

Those results are asked to use RMES (Root Mean Error Square) to measure the



```
RMES in trial 1          = 1.2588662363395717
RMES in trial 2          = 1.252556160737805
RMES in trial 3          = 1.2864408285468754
RMES in trial 4          = 1.223452157827986
RMES in trial 5          = 1.2371692893341377
RMES in trial 6          = 1.2765068541344944
RMES in trial 7          = 1.261308939421493
RMES in trial 8          = 1.2640406829620978
RMES in trial 9          = 1.2315415812789947
RMES in trial 10         = 1.2680581786693441
```

errors.
The measurements show that the used method serves an outstanding result in regressing the data. It also provides almost a similar value to the actual data.

Classification Iris

According to the question, there is a file named `Classification iris.xlsx`, which contains the iris dataset for classification. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are NOT linearly separable from each other. There are fiver attributes: 1. sepal length in cm; 2. sepal width in cm; 3. petal length in cm; 4. petal width in cm; 5. class: – Iris Setosa – Iris Versicolour – Iris Virginica.

Data Loading

The program uses `read_excel()` function, which is provided in the `pandas` library, to read the information in the file. Notice that the loading of the data is using relative path instead of absolute path. Hence, when executing the code of the model, one should place the CSV file under the same directory of the model python file.

After read in, the dataset is split into two parts. The first part is the first 4 columns, which acts as the input feature values. Denote this part as input data X. The second part is the last column, which serve as the actual label value. Denote this part as y. As required, 80% of the data will be used as training data and the rest of the data will be used as testing data.

Training

According to Lecture 5 - Linear Regression, there are two methods to classify this data. First, repetitive binary classifications. Second, multi-category classification. As there is a class that has linearly separable property, the first method is used.

In the lecture, closed-form solution for the learning model is shown to be $W = (X^T X)^{-1} X^T y$. The model is obtained according to the value of X and y that are already loaded previously. Notice that, $y_i \in \{-1, +1\}$ for $i = 1, ...,$ (size of y).

The program starts by finding a particular class that has the linearly separable property and prints the classification result of that particular class. The rest are then re-classified.

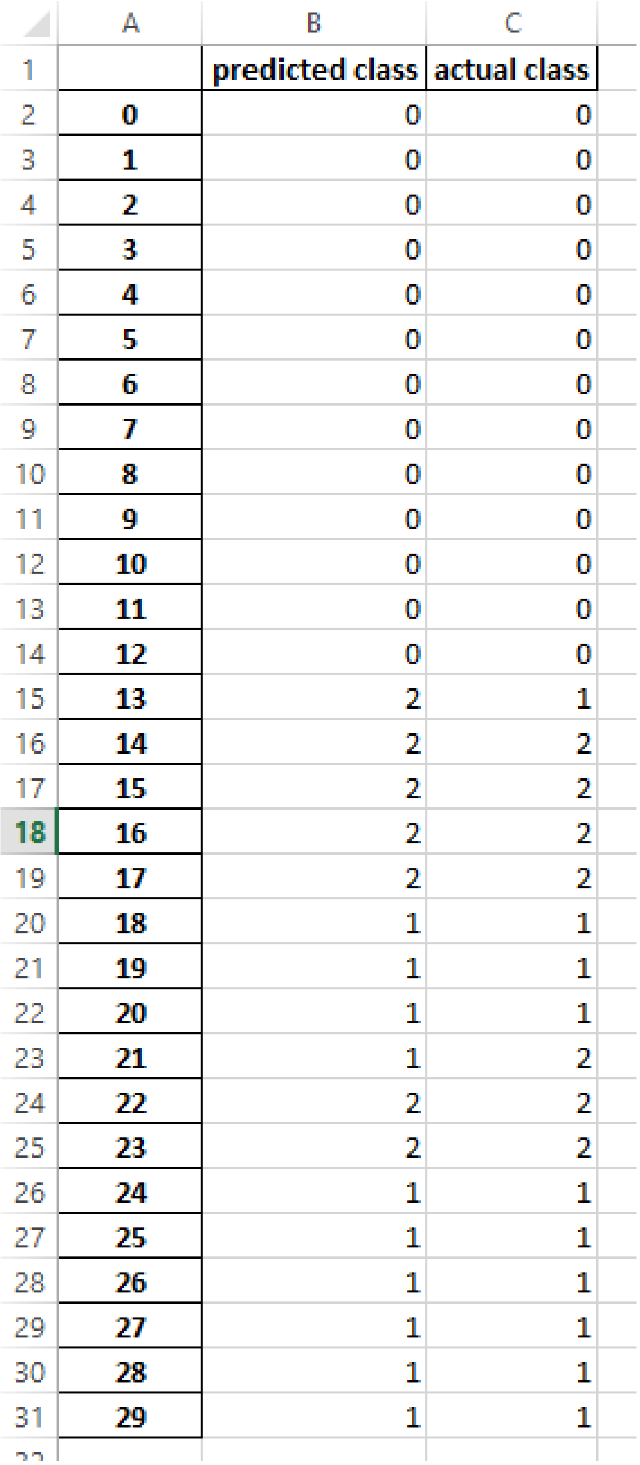
Testing

With the previous learning model, the prediction data is obtained with $f_{w,b}(x_{new}) = sgn(x_{new}^T W)$.

Analysis

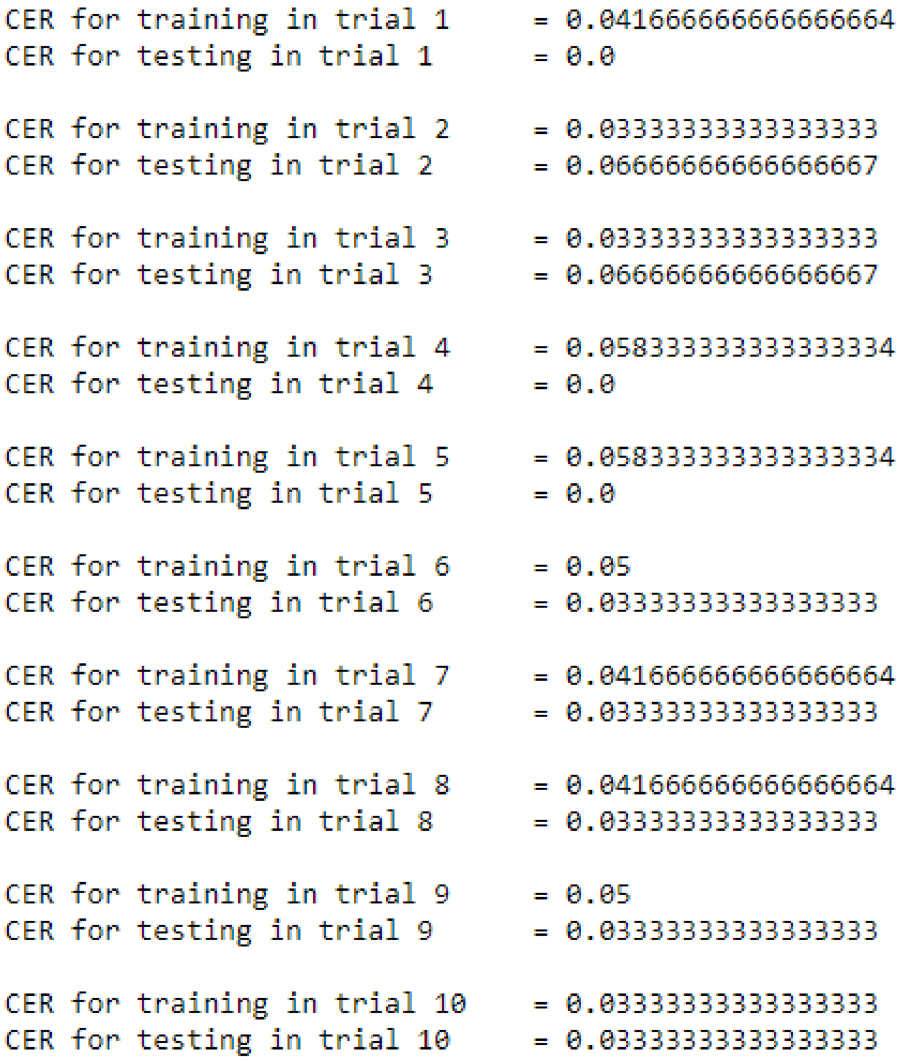
In 10 trials, there are two results printed to excels with the format name `classification_iris_testing_result_trial_{number}.xlsx` for the testing result and `classification_iris_training_result_trial_{number}.xlsx` for the training result. In the submission, the files are in the results folder.

`classification_iris_testing_result_trial_1.xlsx` shows as following:



	A	B	C
1		predicted class	actual class
2	**0**	0	0
3	**1**	0	0
4	**2**	0	0
5	**3**	0	0
6	**4**	0	0
7	**5**	0	0
8	**6**	0	0
9	**7**	0	0
10	**8**	0	0
11	**9**	0	0
12	**10**	0	0
13	**11**	0	0
14	**12**	0	0
15	**13**	2	1
16	**14**	2	2
17	**15**	2	2
18	**16**	2	2
19	**17**	2	2
20	**18**	1	1
21	**19**	1	1
22	**20**	1	1
23	**21**	1	2
24	**22**	2	2
25	**23**	2	2
26	**24**	1	1
27	**25**	1	1
28	**26**	1	1
29	**27**	1	1
30	**28**	1	1
31	**29**	1	1

Those results are asked to use classification error rate, which is the number of miss-classified samples divided by number of all samples.



```
CER for training in trial 1     = 0.041666666666666664
CER for testing in trial 1      = 0.0

CER for training in trial 2     = 0.03333333333333333
CER for testing in trial 2      = 0.06666666666666667

CER for training in trial 3     = 0.03333333333333333
CER for testing in trial 3      = 0.06666666666666667

CER for training in trial 4     = 0.058333333333333334
CER for testing in trial 4      = 0.0

CER for training in trial 5     = 0.058333333333333334
CER for testing in trial 5      = 0.0

CER for training in trial 6     = 0.05
CER for testing in trial 6      = 0.03333333333333333

CER for training in trial 7     = 0.041666666666666664
CER for testing in trial 7      = 0.03333333333333333

CER for training in trial 8     = 0.041666666666666664
CER for testing in trial 8      = 0.03333333333333333

CER for training in trial 9     = 0.05
CER for testing in trial 9      = 0.03333333333333333

CER for training in trial 10    = 0.03333333333333333
CER for testing in trial 10     = 0.03333333333333333
```

The measurements show that the used method serves an outstanding result in regressing the data. It also provides relatively a low rate of error in classifying the data.