NYCU Introduction to Machine Learning, Homework 2

110550071, 田松翰

Part. 1, Coding (50%):

In this coding assignment, you are requested to implement Logistic Regression and Fisher's Linear Discriminant by using only Numpy. After that, train your model on the provided dataset and evaluate the performance on the testing data.

(15%) Logistic Regression

Requirements:

- Use Gradient Descent to update your model
- Use CE (Cross-Entropy) as your loss function.

Criteria:

1. (0%) Show the hyperparameters (learning rate and iteration) that you used.

```
LR = LogisticRegression(learning_rate=0.0005, iteration=150000)
```

2. (5%) Show the weights and intercept of your model.

```
Weights: [-0.05660093 -1.75183585  1.04929478 -0.2387437   0.03556569 -0.65359616],
```

```
Intercept: -0.1587813655607662
```

3. (10%) Show the accuracy score of your model on the testing set. The accuracy score should be greater than 0.75.

```
Accuracy: 0.7540983606557377
```

(35%) Fisher's Linear Discriminant (FLD)

Requirements:

- Implement FLD to reduce the dimension of the data from 2-dimensional to 1-dimensional.

Criteria:

4. (0%) Show the mean vectors mi (i=0, 1) of each class of the training set.

```
Class Mean 0: [ 56.75925926 137.7962963 ], Class Mean 1: [ 52.63432836 158.97761194]
```

5. (5%) Show the within-class scatter matrix SW of the training set.

```
With-in class scatter matrix:
[[ 19184.82283029 -16006.39331122]
 [-16006.39331122 106946.45135434]]
```

(next page)

6. (5%) Show the between-class scatter matrix SB of the training set.

```
Between class scatter matrix:
[[ 17.01505494 -87.37146342]
 [-87.37146342 448.64813241]]
```

7. (5%) Show the Fisher's linear discriminant w of the training set.

```
w:
[-0.19485739  0.98083158]
```

8. (10%) Obtain predictions for the testing set by measuring the distance between the projected value of the testing data and the projected means of the training data for the two classes. Show the accuracy score on the testing set. The accuracy score should be greater than 0.65.

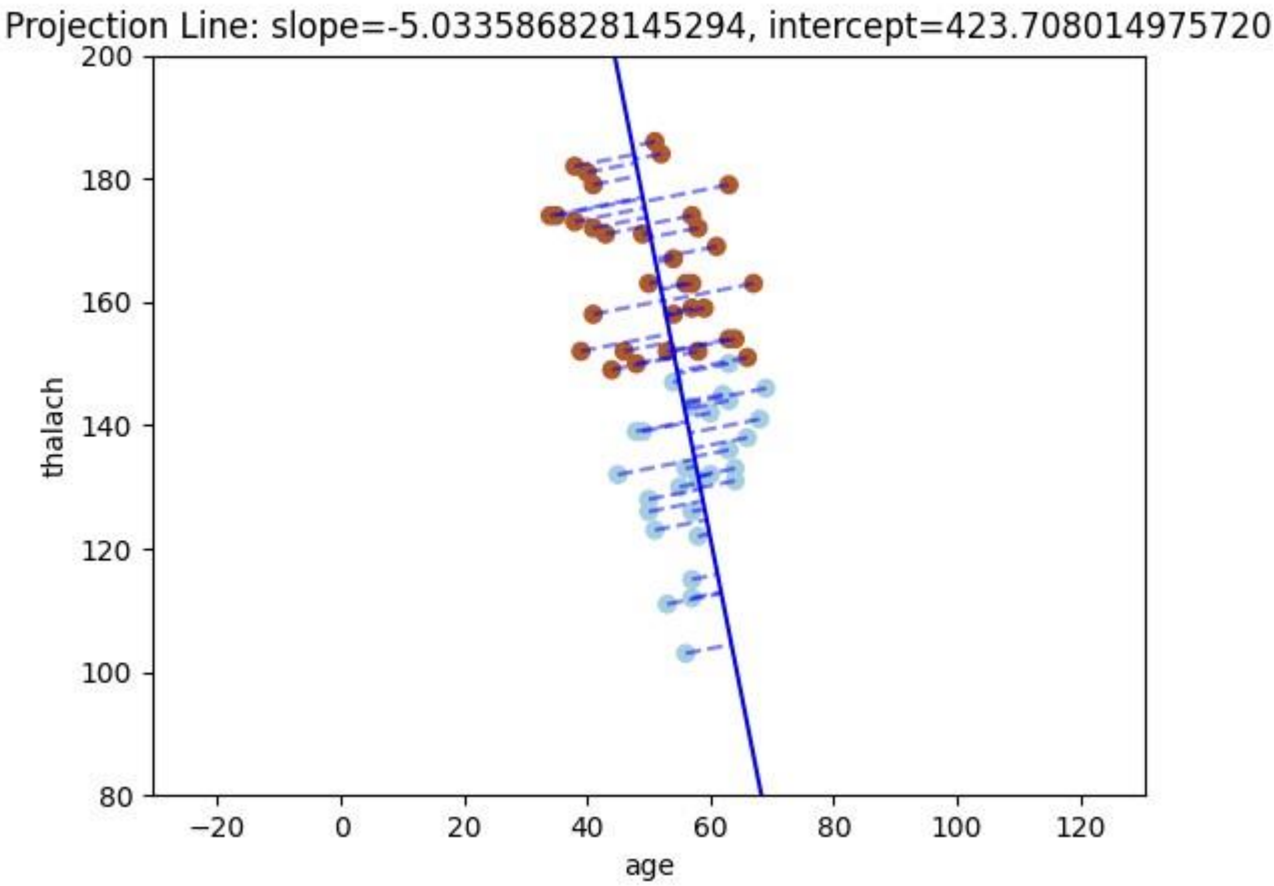
```
Accuracy of FLD: 0.6721311475409836
```

9. (10%) Plot the projection line (x-axis: age, y-axis: thalach).

   1) Plot the projection line trained on the training set and show the slope and intercept on the title (you can choose any value of intercept for better visualization).

   2) Obtain the prediction of the testing set, plot and colorize them based on the prediction.

   3) Project all testing data points on your projection line. Your result should look like the below image.



(next page)

Part. 2, Questions (50%):

1. (5%) What's the difference between the sigmoid function and the softmax function? In what scenarios will the two functions be used? Please at least provide one difference for the first question and answer the second question respectively.
   Sigmoid function maps the input values to the result as the range of [0, 1]. It's commonly used in binary classification problems to make a decision between two classes. It can be used at determining an email is spam or not.
   Softmax function is used for multiclass classification problems. It is suitable for problems where an input can belong to one of several classes. It can be used at classifying the handwriting numbers.

2. (10%) In this homework, we use the cross-entropy function as the loss function for Logistic Regression. Why can't we use Mean Square Error (MSE) instead? Please explain in detail.
   Usually, we use MSE for regression problems and CE for classification problems.
   CE penalized heavier than MSE on predictions that are confidently wrong.
   CE provides more smoother gradients during optimization compared to MSE.

3. (15%) In a multi-class classification problem, assume you have already trained a classifier using a logistic regression model, which the outputs are P1, P2, ... Pc, how do you evaluate the overall performance of this classifier with respect to its ability to predict the correct class?
   1. (5%) What are the metrics that are commonly used to evaluate the performance of the classifier? Please at least list three of them.
      Observe the value with below metrics:
      Confusion matrix: a table shows that the true positives, true negatives, false positives, false negatives.
      Accuracy: (true positives + true negatives) / total observations
      Precision: true positives / (true positives + false positives)
   2. (5%) Based on the previous question, how do you determine the predicted class of each sample?
      Choose the highest probability among P1, P2, ..., Pc for each sample.
   3. (5%) In a class imbalance dataset (say 90% of class-1, 9% of class-2, and 1% of class-3), is there any problem with using the metrics you mentioned above and how to evaluate the model prediction performance in a fair manner?
      For metrics like Accuracy, if the classifier is not "well" and always predicts "class-1", it will get a high accuracy but it's a bad classifier. On the other hand, using Precision can solve this problem, it can concern the performance of each class respectively.

(next page)

4. (20%) Calculate the results of the partial derivatives for the following equations. (The first one is binary cross-entropy loss, and the second one is mean square error loss followed by a sigmoid function. σ is the sigmoid function.)

4.1 (10%)

$$\frac{\partial}{\partial x}\left(-t * \ln\left(\sigma(x)\right) - (1-t) * \ln\left(1-\sigma(x)\right)\right)$$

Sol:

σ(x) = 1 / (1+exp(-x))

Original =

$$\frac{\partial}{\partial x}(tln(1+e^{-x}) - ln(1-\frac{1}{1+e^{-x}})(1-t))$$

$$= -\frac{te^{-x}}{1+e^{-x}} - \frac{1-t}{e^{-x}+1}$$

$$= \frac{-e^{-x}t+1-t}{1+e^{-x}}$$

4.2 (10%)

$$\frac{\partial}{\partial x}(\,(\,t-\sigma(x)\,)^2\,)$$

Sol:

Original =

$$2(t-\frac{1}{1+e^{-x}})\frac{\partial}{\partial x}(t-\frac{1}{1+e^{-x}})$$

$$= 2(t-\frac{1}{1+e^{-x}})(-\frac{e^{-x}}{(e^{-x}+1)^2})$$

$$= \frac{2e^{-x}(t(1+e^{-x})-1)}{(1+e^{-x})^3}$$