NYCU Introduction to Machine Learning, Homework 2

110550126, 曾家祐

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.

```
145     LR = LogisticRegression(learning_rate=0.00045, iteration=25000)
```

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

```
Weights: [-0.05405787 -0.63060222  0.86880291 -0.03076196  0.02723661 -0.50335275], Intercept: -0.05715263991951218
```

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 m_i (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 S_W of the training set.

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

6. (5%) Show the between-class scatter matrix S_B 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:
[ 0.28737344 -0.95781862]
```

8. (10%) Obtain predictions for the testing set by measuring the distance between the project ed value of the testing data and the projected means of the training data for the two classe s. Show the accuracy score on the testing set. The accuracy score should be greater than 0. 65.

```
Accuracy of FLD: 0.6557377049180327
```

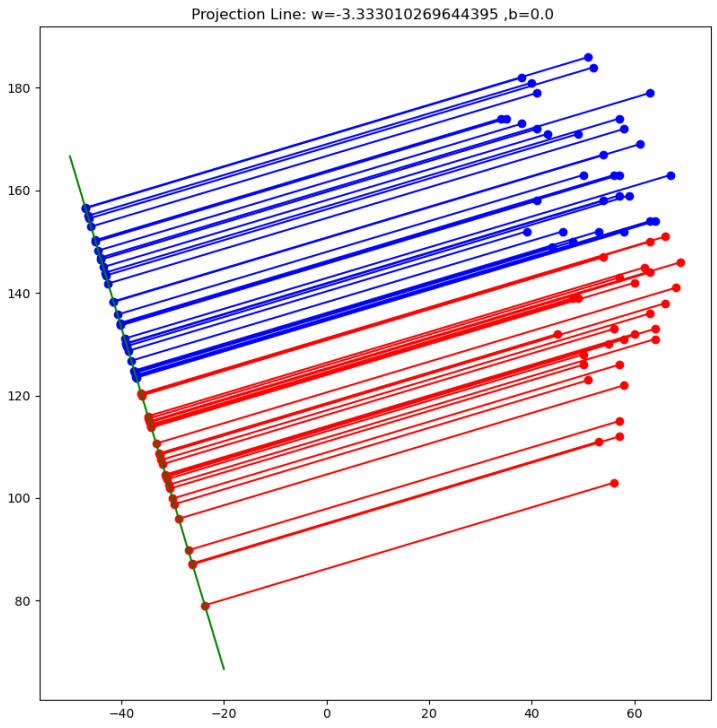
(next page)

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 th e 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 b elow image.



Part. 2, Questions (50%):

1. (5%) What’ s the difference between the sigmoid function and the softmax function? In w hat scenarios will the two functions be used? Please at least provide one difference for the first question and answer the second question respectively.

1. sigmoid function is a logistic function. It will convert any number into [0, 1]

$Sigmoid(x) = \frac{1}{1+e^{-x}}$ 。

，input is a number, ouput is a number too.

softmax function also known as Normalized Exponential Function, it will takes vectors of real numbers as inputs, and normalizes them into a probability distribution proportio nal to the exponentials of the input numbers. every number in vector will between 0 1 and sum will be 1.

$Softmax(x) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}$ 。

2. Sigmoid fuction is suitable for use in binary classification problems. If the output is grea ter than 0.5, it is considered class1, otherwise it is class0.

The softmax function is suitable for use in multi-class classification problems. We will take argmax from the output array as the class we predict.

2. (10%) In this homework, we use the cross-entropy function as the loss function for Logisti c Regression. Why can’ t we use Mean Square Error (MSE) instead? Please explain in det ail.

There are some reason.

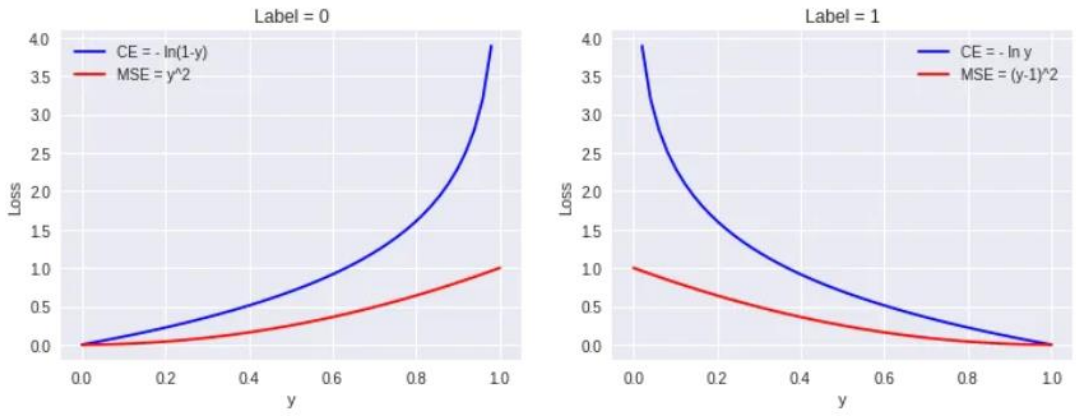
1. Nature of the problem :

Logistic regression is primarily used for binary classification problems, where the goal is to predict the probability of an instance belonging to a particular class (usually 0 or 1). The output of logistic regression is a probability value between 0 and 1.

Mean Square Error (MSE) is commonly used for regression problems, where the g oal is to predict a continuous output. It measures the average squared difference between p redicted and actual values.

2. Sensitivity to Outliers:

Although both MSE and CE can reach a minimum value of 0 if classified correctl y, there is a big difference between the two in the gradient part. The following figure (taki ng binary as an example) is the gradient of two loss functions. Because the model will go t hrough the sigmoid function and output the value between 0 and 1, we can see that the gra dient of CE is significantly larger than MSE. At the end of training, MSE will be difficult t o continue to update the weights because the gradient is too low, while CE is relatively It c an continue to update the weights because it has a larger gradient, so the performance will be better than MSE.



(next page)

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 cla ss?

3.1. (5%) What are the metrics that are commonly used to evaluate the performance of the classifier? Please at least list three of them.

1. **Accuracy:** *Number of Correct Predictions / Total Number of Predictions*
2. **Precision:** Ratio of true positives to the sum of true positives and false positive s.
3. **Recall:** Ratio of true positives to the sum of true positives and false negatives.
4. **F1-Score**: the harmonic mean of precision and recall
5. **Confusion Matrix** : showing the number of true positives, true negatives, false p ositives, and false negatives for each class.

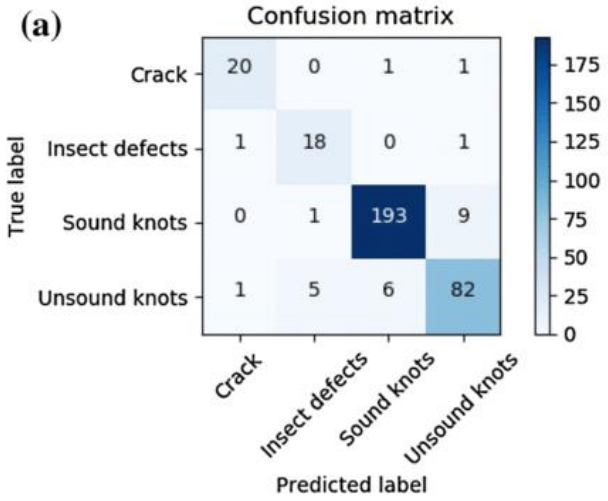
3.2. (5%) Based on the previous question, how do you determine the predicted class of each sample?

Since the model outputs is P1 P2 P3 ⋯ PC So we can use the argmax function to get the index of the maximum Px so we can determine the predicted class is x.

3.3. (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?

there will be some problem when using accuracy, the whole model performance will be dominated by the accuracy if class-1 and the accuracy of class-2 and class-3 will be ignored.
It is better to use confusion metrix to evaluate the model prediction performance. Using confusion matrix we can see the whole prediction result. we can see the accuracy of class-1, class-2 and class-3 . We can also see the wrong prediction we predict to which class.

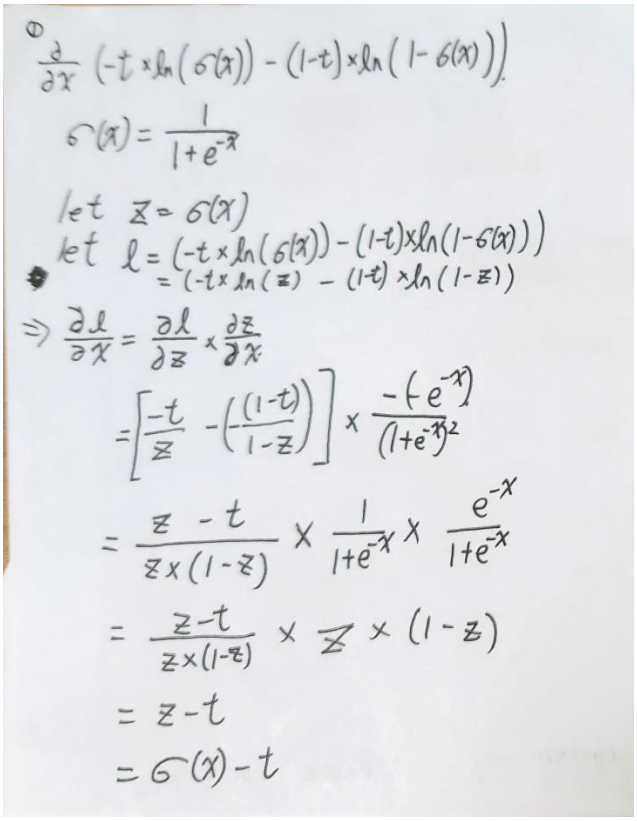
ex:



4. (20%) Calculate the results of the 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%)

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



4.2. (10%)

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

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

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

let $z = \sigma(x)$

let $l = (t-\sigma(x))^2$

$= (t-z)^2$

$\Rightarrow \frac{\partial l}{\partial x} = \frac{\partial l}{\partial z} \times \frac{\partial z}{\partial x}$

$= -2(t-z) \times z \times (1-z)$

$= -2(t-\sigma(x)) \times \sigma(x) \times (1-\sigma(x))$.