COMP 4211 - Machine Learning Programming Assignment 1 Report

Cheng Chi Fung
cfchengac@connect.ust.hk

Abstract

In this assignment, we used linear regression, logistic regression and single layer neural network to perform the regression and classification on three binary classification and regression data sets.

1 Linear Regression

1.1 Experienment Model Settings

For linear regression, we used the built-in models **LinearRegression** in skitlearn to perform the linear regression on three regression datasets .This model fits a linear model with coefficients $w = (w_1, ..., w_p)$ to minimize the residual sum of squares between the observed responses in the dataset, and the responses predicted by the linear approximation. This model has **no hyperparameter** to set.

Table 1: Experiments Datasets Settings

Dataset	Training set	Test set
Fifa	13191	4397
Finance	2754	918
Orbits	9642	3215

1.2 Experiment Results

The following are the experiment results which using the above model setting on fifa, finanace and orbits regression datasets.

Table 2: Experiments Results of Linear Regression on Three Regression Datasets

Coefficient of Determination (R^2 score)		
Dataset	Training set	Test set
Fifa	0.51944	0.51223
Finance	0.42637	0.28766
Orbits	0.61671	0.61016

From the experiment results, we found out that excepts the **finance** datasets, the R^2 score of the training set and test set for the other twos are almost the same. And, for the finance datasets, the R^2 score of the test set (0.42637) are a much higher than the training set (0.28766). It means that this model **overfit** the training data of the finance datasets.

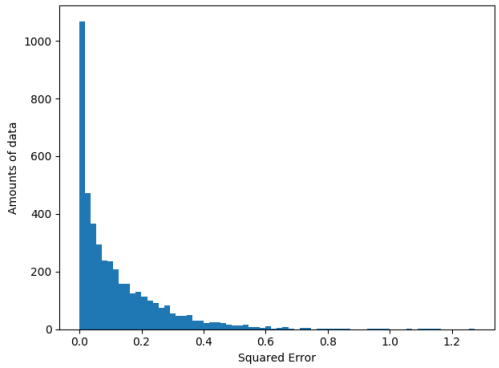

Figure 1: Squared Error Histogram on test set of orbits data set

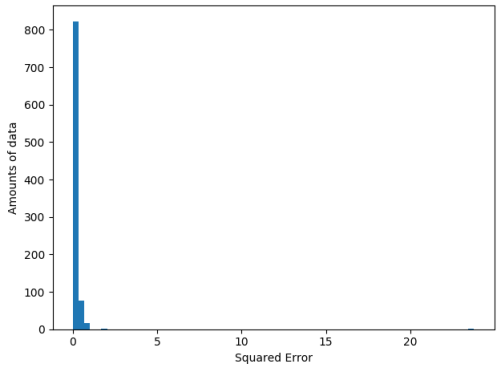

Figure 2: Squared Error Histogram on test set of finance data set

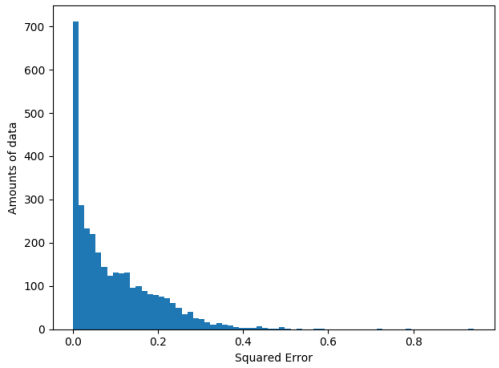

Figure 3: Squared Error Histogram on test set of orbit data set

From the square loss histogram of test sets, we found out that the mean squared error of the most of the data points are around zero. It means that this model perform quite well for most of the data points. However, for the finance datasets, there are some data points that have the sqaured error far away from zero (23.77581) which means that there might be some **outliers** that far away from the curve.

2 Logistic Regression

2.1 Experienmental Model Settings

To perform the logistic regression with gradient-descent algorithm by minimizing the cross-entropy loss on the three classification datasets, we used the built-in model **SGDClassifier** in skitlearn. This model implements regularized linear models with **stochastic gradient descent** (SGD) learning.

To get better training results, we first performed the **hyperparameters tunning** by using **GridSearchCV** in skitlearn to obtain best hyperparameters parameters on the training set of each datasets. The following are the settings for parameters tuning.

Table 3: Model Settings for Hyperparameters Tuning (Tuned hyperparameters *)

Name	Parameter Setttings
max_iter	5000
learning_rate	optimal
loss	log
tol	0.000000001
alpha*	$0.1, 0.01, ..., 1^{-9}, 1^{-10}$
cv	5

Table 4: Best Hyperparameters after Hyperparameters Tuning

Dataset	alpha
fifa	1^{-5}
finance	0.001
orbits	1^{-6}

2.2 Experiment Results

The following are the experiment results which using the same setting as the hyperparameter tuning except the **alpha** using the best results after hyperparameters tuning on fifa, finanace and orbits classification datasets.

Table 5: Experiments Results of Logistic Regression on Three Classification Datasets

Classification Accuracy		
Dataset	Training set	Test set
Fifa	0.82533	0.81669
Finance	0.80355	0.79084
Orbits	0.98610	0.98755

AUC Score		
Dataset	Training set	Test set
Fifa	0.93828	0.93449
Finance	0.90664	0.89185
Orbits	0.99995	0.99992

From the experiment results, we found out that the model perform **quite well** on three datasets which getting $> 79\%$ on three datasets. However, its classification performace still **worse than single layer neural network** which will be disscussed in the next section. Furthermore, the better the AUC Score, the better the performace of the classifier. And from experiment results, we got a very high AUC score which mean that the model can classify the dataset very well.

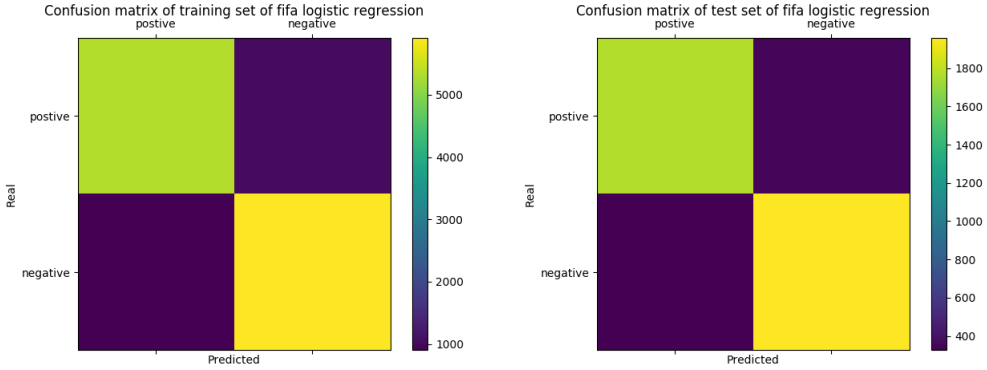

Figure 4: Confusion matrix of fifa data set

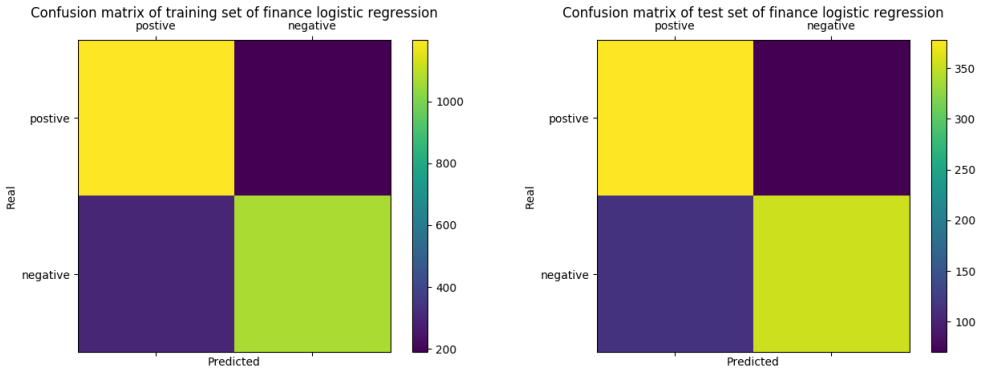

Figure 5: Confusion matrix of finance data set

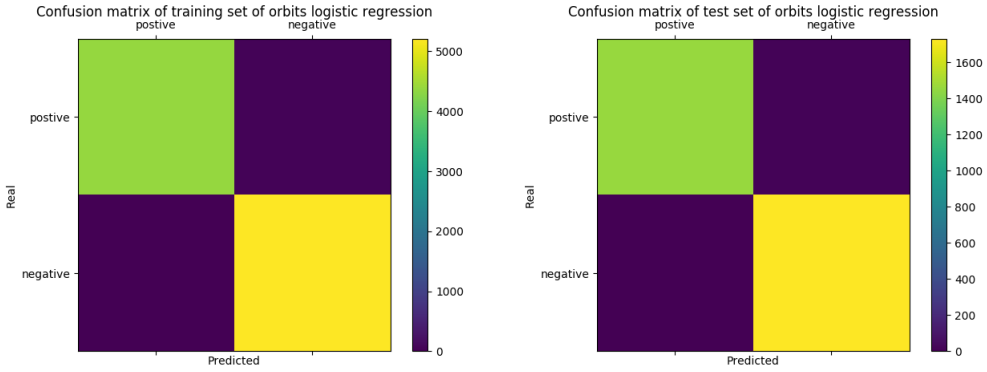

Figure 6: Confusion matrix of orbits data set

From the visulization of confusion matrix on three datasets, we found out that the model perform very well on both fifa and orbits dataset since there are deep purple color on the top right and the bottom left in the confusion matrix which means that it has **high true positive and true negative** rate during prediction. However, for the finance datasets, the bottom left purple color are not as deep as the other two which means that the prediction of this model on the finance datasets have a little bit higher **false postive** rate.

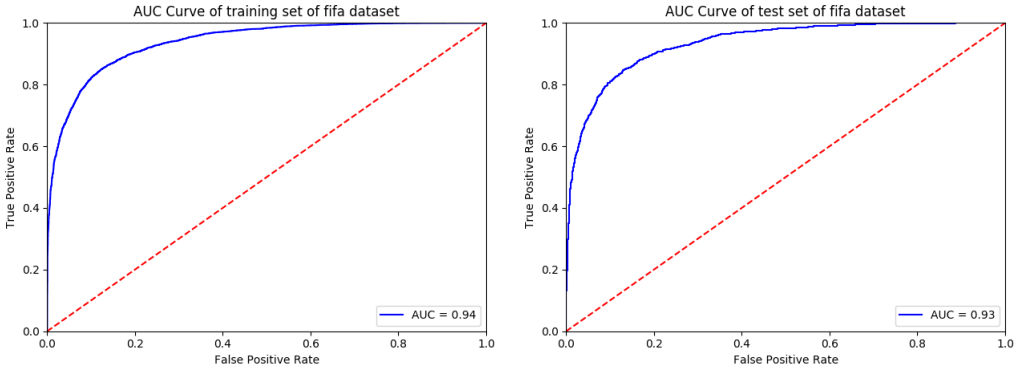

Figure 7: ROC curve of fifa data set

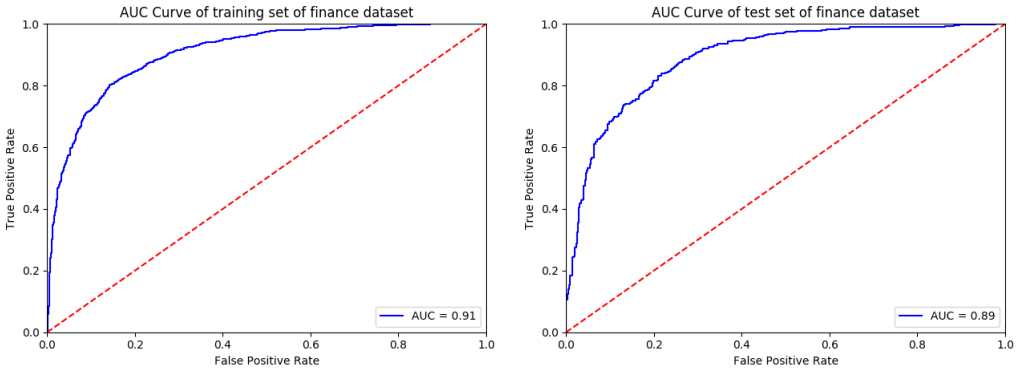

Figure 8: ROC curve of finance data set

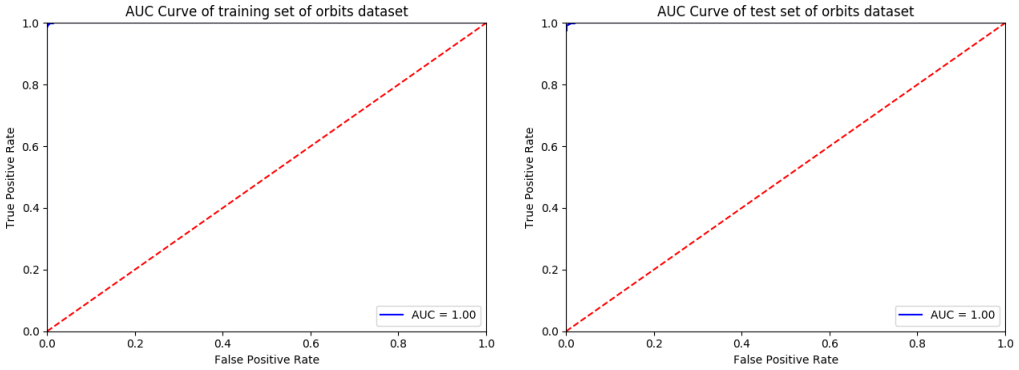

Figure 9: ROC curve of orbits data set

From the visulization of ROC Curve, since the curve are smooth on three datasets. Therefore, this model **does not overfit** the dataset.

3 Single Layer Neural Network

3.1 Experienmental Model Settings

To perform the classification with single layer neural network on the three classification datasets, we used the built-in model **MLPClassifier** in skitlearn. This model implements a multi-layer perceptron classifier and optimizes with the log-loss function using stochastic gradient descent.

To determine the number of **hidden units** should be used in the hidden layer, we evaludate the The generalization performance of the model with i hidden units with **cross validation** (CV) using the utilities **cross_val_score** in skitlearn. The following are the model settings for cross validation and the results of that.

Table 6: Model Settings for Cross Validation

Name	Parameter Setttings
max_iter	5000
learning_rate	optimal
solver	sgd
tol	0.000000001
alpha	determined by using grid search for each i hidden unit
cv	5

Table 7: Performance of the Model with i Hidden Units using Cross Validation

i hidden units	Score (Average Classification Accuracy)
	Fifa, Finance, Orbits
1	0.65346, 0.67176, 0.99398
2	0.86142, 0.68424, 0.90243
3	0.86028, 0.80501, 0.99429
4	0.86119, 0.80647, 0.99367
5	0.86089, 0.80464, 0.99367
6	0.85869, 0.80900, 0.99336
7	0.85975, 0.80247, 0.99398
8	0.86020, 0.80501, 0.99387
9	0.86134, 0.80429, 0.99419
10	0.86088, 0.80320, 0.99263
Number of hidden units with highest score	$2(0.86142)$, $6(0.80900)$, $3(0.99429)$

Furthermore, to **avoid overfitting**, we performed the parameters tuning on the **number of iterations** after getting the best number of hidden units with cross validation. The model settings for parameters tuning used the same model settings as cross validation and the number hidden units using what we got from cross validation. We tuned the number of iteraion by returning the maximum iterations before loss of the test set start to decrease.

Table 8: Model Settings for Parameter Tuning(Tuned hyperparameters *)

Name	Parameter Setttings
max_iter*	$1000, 1500, 2000, ..., 8000, 8500$
learning_rate	optimal
solver	sgd
tol	0.000000001
alpha	determined by using grid search before
hidden units	determined by using CV before

Table 9: Best Hyperparameters after Hyperparameters Tuning

Dataset	max_iter
fifa	1000
finance	4500
orbits	8000

After hyperparameters tuning on number of iterations, we obtain the best **max_iteration** before overfitting to the training set. Finally, we got the following model settings for our final training.

Table 10: Model Settings of Using Single Layer Nerual Network

Name	Parameter Setttings
	Fifa, Finance, Orbits
max_iter	1000, 1000, 1500
learning_rate	optimal
solver	sgd
tol	0.000000001
alpha	0.0001, 1^{-9}, 0.01
number of hidden units	2, 6, 3

3.2 Experienment Results

The following are the experiment results which using the above model setting on fifa, finanace and orbits classification datasets.

Table 11: Experiments Results of Logistic Regression on Three Classification Datasets

Classification Accuracy		
Dataset	Training set	Test set
Fifa	0.93828	0.93449
Finance	0.82933	0.80718
Orbits	0.99533	0.99502

Loss		
Dataset	Training set	Test set
Fifa	0.31560	0.32098
Finance	0.38444	0.39380
Orbits	0.07369	0.05241

AUC Score		
Dataset	Training set	Test set
Fifa	0.94828	0.93449
Finance	0.91933	0.89718
Orbits	0.99997	0.99995

From the experiment results, we found out that the single layer neural network **perform better than logistic regression** which has higher classification accuracy on three datasets. And the loss of the training set and test set are similiar which means that the model are **not overfit** to the training data set. And for the AUC score, single layer neural network perform slightly better than logistic.

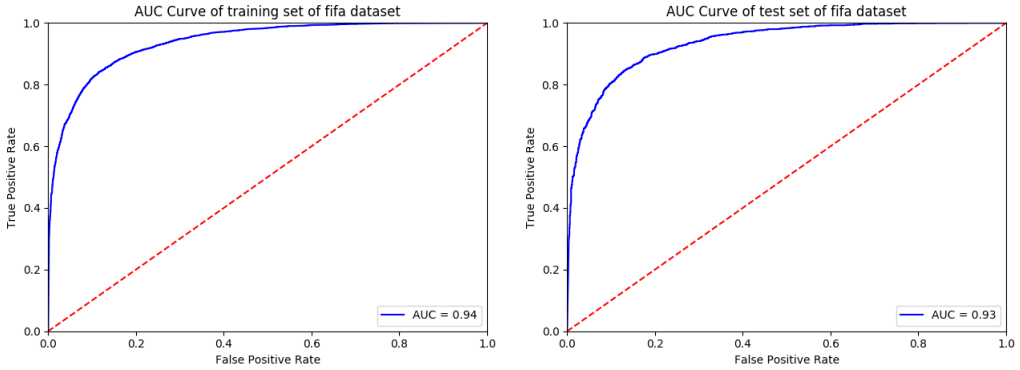

Figure 10: ROC curve of fifa data set

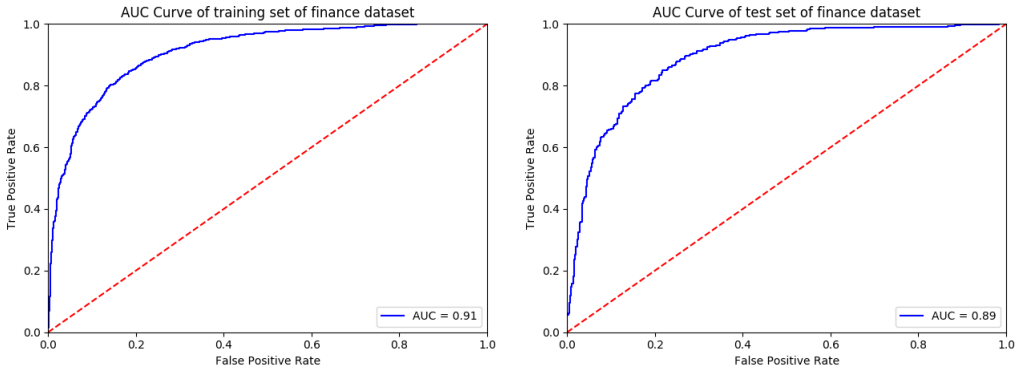

Figure 11: ROC curve of finance data set

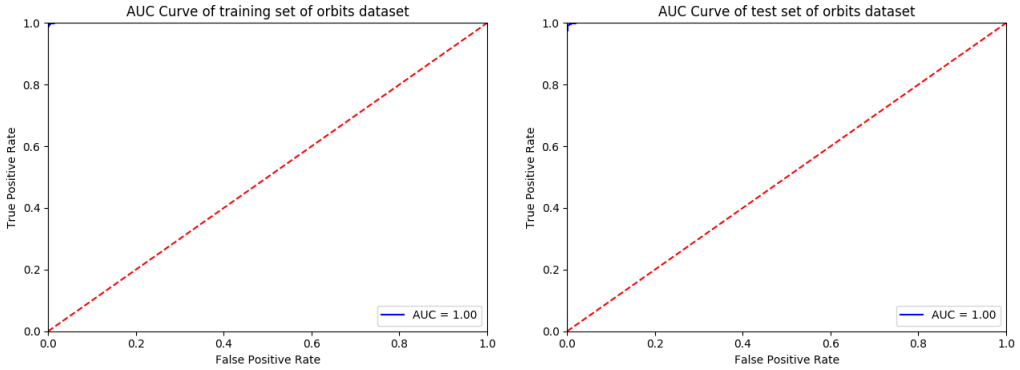

Figure 12: ROC curve of orbits data set

From the visulization of ROC Curve, since the curve are smooth on three datasets. Therefore, this model **does not overfit** the dataset.

The following are the visualization of the peroformace during training. (Y axis = loss / X axis = iterations)

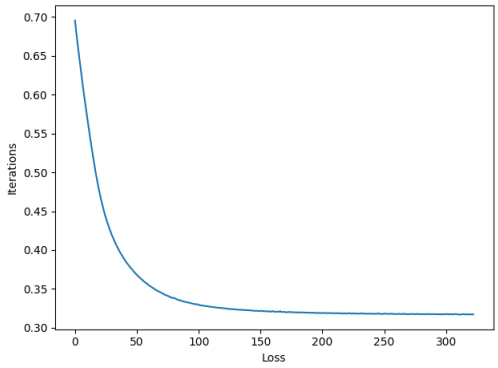

Figure 13: Loss curve of train set of fifa data set

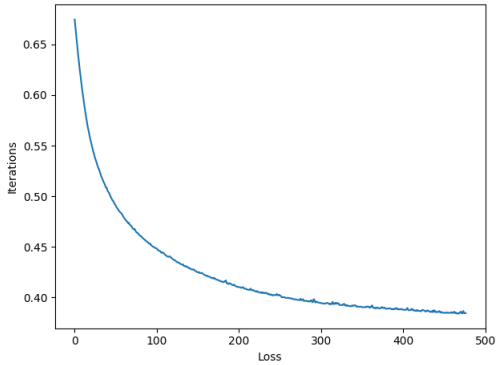

Figure 14: Loss curve of train set of finance data set

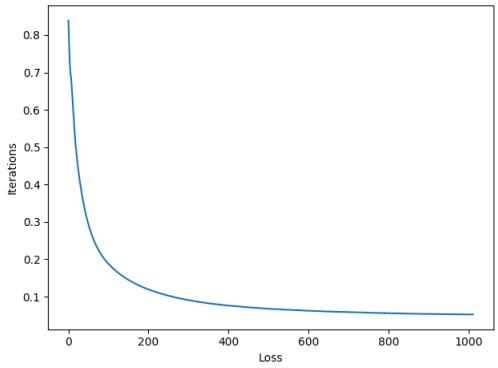

Figure 15: Loss curve of train set of orbits data set

4 Computing Environment and Time Used

4.1 Computing Environment

Table 12: Computing Environment

	Environment Settings
CPU	AMD Ryzen™ 7 2700X
GPU	GTX 1080Ti
RAM	16GB
STORAGE	1TB M2 NVME SSD
OS	Ubuntu 16.04
IDE	Pycharm
Python Version	Python 3.5.2
Libraries Used	skitlearn, numpy, matplotlib.pyplot

4.2 Time Used

Table 13: Total Running Time For Each Tasks

	Wall Time (seconds)
Linear Regression	0.470023
Logistic Regression	10.86583
Single Layer Neural Network	8376.50412