NUS
National University
of Singapore

DBA3713
Assignment 1 Report

Group 2

Name	Matriculation Number
Chern Tat Sheng	A0183739N
Chew Zhe Ming, Ryan	A0183631H
Lim Shu Min, Cambrie	A0204096J
Tan Jia Yi, Megan	A0188664L

Introduction

In this assignment, we aim to predict which customers will result in good loans depending on various features. This allows the bank to avoid lending to customers who may default on their loans, thereby reducing revenue loss. To do this, we will perform a logistic regression on the dataset “lendingclub_full_data_set_no_id.csv”, evaluate the results, and then conduct regularization to improve the model.

1. Data Extraction and Cleaning

The dataset consists of a sample of customers and the characteristics of their loans, such as their status of Home Ownership, the Interest Rate, Purpose of Loan, and more.

Before a logistic regression can be performed, we will first clean the data. We identify the dependent variable as “loan_status” and convert it into binary, where a good loan is represented by 1 and a bad loan is represented by 0. We also convert all other relevant categorical variables ('verification_status', 'home_ownership', 'purpose', 'sub_grade', 'term', 'addr_state') into binary and inspect the resulting dataset for missing values. The identified missing values are replaced with the median of their respective columns. The resulting dataset has 116 independent variables.

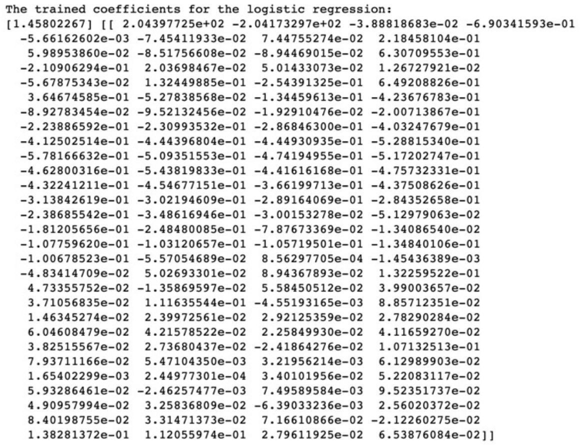
The cleaned dataset is then shuffled randomly and then split into 3 sets for training, validation, and testing. We also standardized the independent variables by scaling the data based on the mean and standard deviations of the training set and proceed with the logistic regression.

2. Predicting with Logistic Regression and Evaluation

Model Training: Basic LR with selected features, no regularisation

Using the transformed 116 features, we first fitted the Logistic Regression to the training data using the newton conjugate gradient algorithm with no regularisation and obtained the following results.

Model Interpretation



```
The trained coefficients for the logistic regression:
[1.45802267] [[ 2.04397725e+02 -2.04173297e+02 -3.88818683e-02 -6.90341593e-01
  -5.66162602e-03 -7.45411933e-02  7.44755274e-02  2.18458104e-01
   5.98953860e-02 -8.51756608e-02 -8.94469015e-02  6.30709553e-01
  -2.10906294e-01  2.03698467e-02  5.01433073e-02  1.26727921e-02
  -5.67875343e-02  1.32449885e-01 -2.54391325e-01  6.49208826e-01
   3.64674585e-01 -5.27838568e-02 -1.34459613e-01 -4.23676783e-01
  -8.92783454e-02 -9.52132456e-02 -1.92910476e-02 -2.00713867e-01
  -2.23886592e-01 -2.30993532e-01 -2.86846300e-01 -4.03247679e-01
  -4.12502514e-01 -4.44396804e-01 -4.44930935e-01 -5.28815340e-01
  -5.78166632e-01 -5.09351553e-01 -4.74194955e-01 -5.17202747e-01
  -4.62800316e-01 -5.43819833e-01 -4.41616168e-01 -4.75732331e-01
  -4.32241211e-01 -4.54677151e-01 -3.66199713e-01 -4.37508626e-01
  -3.13842619e-01 -3.02194609e-01 -2.89164069e-01 -2.84352658e-01
  -2.38685542e-01 -3.48616946e-01 -3.00153278e-02 -5.12979063e-02
  -1.81205656e-01 -2.48480085e-01 -7.87673369e-02 -1.34086540e-02
  -1.07759620e-01 -1.03120657e-01 -1.05719501e-01 -1.34840106e-01
  -1.00678523e-01 -5.57054689e-02  8.56297705e-04 -1.45436389e-03
  -4.83414709e-02  5.02693301e-02  8.94367893e-02  1.32259522e-01
   4.73355752e-02 -1.35869597e-02  5.58450512e-02  3.99003657e-02
   3.71056835e-02  1.11635544e-01 -4.55193165e-03  8.85712351e-02
   1.46345274e-02  2.39972561e-02  2.92125359e-02  2.78290284e-02
   6.04608479e-02  4.21578522e-02  2.25849930e-02  4.11659270e-02
   3.82515567e-02  2.73680437e-02 -2.41864276e-02  1.07132513e-01
   7.93711166e-02  5.47104350e-03  3.21956214e-03  6.12989903e-02
   1.65402299e-03  2.44977301e-04  3.40101956e-02  5.22083117e-02
   5.93286461e-02 -2.46257477e-03  7.49589584e-03  9.52351737e-02
   4.90957994e-02  3.25836809e-02 -6.39033236e-03  2.56020372e-02
   8.40198755e-02  3.31471373e-02  7.16610866e-02 -2.12260275e-02
   1.38281372e-01  1.12055974e-01  2.79611925e-02  6.53876084e-02]]
```

Fig. 1: Regression Coefficients (Basic LR)

All variables used in the model are statistically significant. However, it can be noted that the large number of significant features (116) makes it difficult to interpret the model as a whole (Fig. 1).

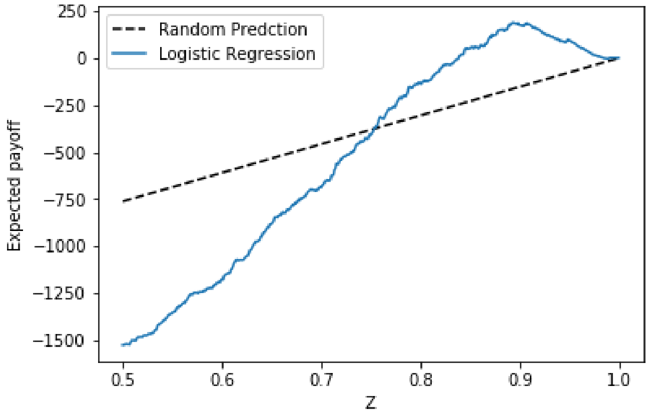
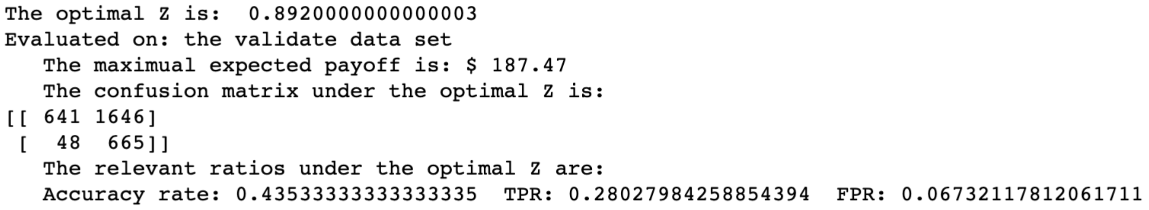


Fig. 2: Expected Payoff Curve of LR model



```
The optimal Z is:  0.8920000000000003
Evaluated on: the validate data set
   The maximual expected payoff is: $ 187.47
   The confusion matrix under the optimal Z is:
[[ 641 1646]
 [  48  665]]
   The relevant ratios under the optimal Z are:
   Accuracy rate: 0.43533333333333335  TPR: 0.28027984258854394  FPR: 0.06732117812061711
```

Based on the results, we found out that the optimal threshold, labelled “optimal z” is **0.892** (To 3 s.f.). This corresponds with the results shown in the maximum point of the expected payoff curve (Fig. 2) given that a threshold of 0.892 on the graph returns a expected payoff of $187.47. Until the maximum point, the curve shows that as the threshold increases, the expected payoff increases.

Model Performance

Computed from the confusion matrix, the number of positives in the dataset is TP + FN = 2,287 and the number of negatives is TN + FP = 713. This shows that the dataset is heavily imbalanced. The TPR of 28.03% (To 2.d.p) indicates low sensitivity and 1 - FPR of 99.33% (To 2.d.p) indicates high specificity of the model. The model is able to predict with an accuracy rate of **43.53%** (To 2.d.p) on the validation set. It is to be noted that accuracy is not a reliable metric for datasets with class imbalance.

- MLE Loss function for training set: 0.473
- MLE Loss function for validation set: 0.503
- MLE Loss function for test set: 0.497

We later compute the loss function value for the validation set which is **0.503** (To 3.sf). Both the relatively low loss and accuracy meant that our model makes small errors in most of the data. This is not ideal as what we expect are high accuracy and low loss which meant few small errors. Hence, the basic LR model with no regularisation offers relatively poor performance.

The AUC measure of the model on the test data is **0.686**. Since 0.5<AUC<1, this shows that there is a high chance that the classifier will be able to distinguish the positive class values from the negative class values. This is because the classifier is able to detect a greater number of True Positives and True Negatives than False Negatives and False Positives. In our next section, we will explore regularization in improving our models predictions.

3. **Improving Prediction with regularization**

Non-zero coefficients with respect to Lambda

Regularization is a technique that fits the function appropriately on the given training set and avoid overfitting.

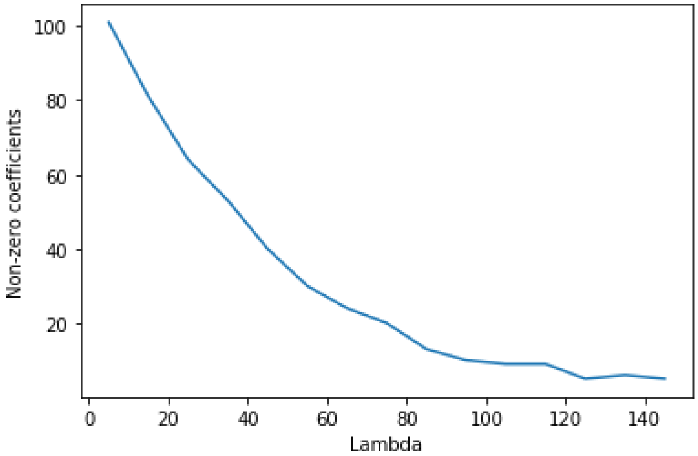


Fig. 3: Plot of non-zero coefficients against Lambda

From the plot (Fig. 3) of non-zero coefficients against the a given range of Lambda for our L1 regularized model, as Lambda increases the strength of regularization increases and an increasing number of coefficients that are close to zero shrink to 0, resulting in a fall in the number of non-zero coefficients. As Lambda approaches 120, we observe that over 90% of the initial non-zero coefficients are removed, and thereafter, the non-zero coefficients begins to shrink at a decreasing rate. These observations imply that the initial model has many coefficients that are close to zero, and that only less than 10% of the initial coefficients are significant. Thus, L1 regularization is ideal in feature selection for this model, as it contains a small number of significant parameters that can be highlighted more clearly by increasing the Lambda.

MLE Loss function with respect to Lambda

The aim of regularization is to improve the prediction power of the model in the validation and test sets, by reducing the MLE loss, through the reduction of overfitting in the training set. To evaluate this improvement in our model after regularization, we created a plot for MLE loss function values on the Train, Test and Validation sets before (Dotted lines) and after regularization (Curves).

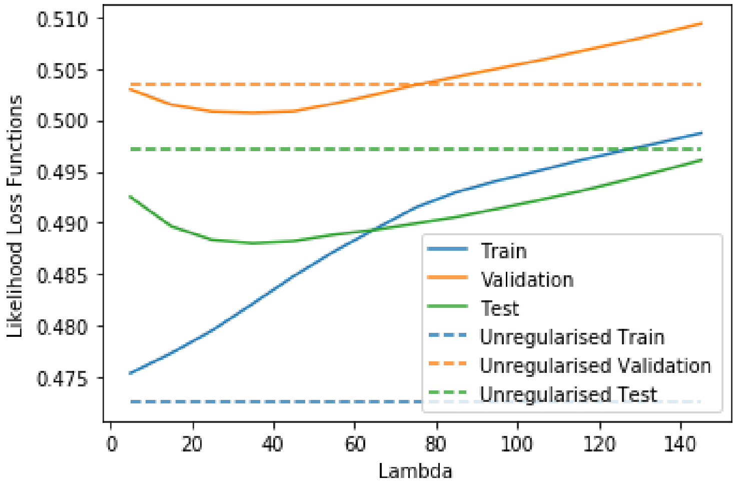


Fig. 4: Plot of likelihood loss functions against Lambda

From the plot (Fig. 4), unregularized training loss is low and likely due to overfitting in the original model. However, predictive power in the validation and test sets are weak with the high MLE loss at close to 0.50 for both test and validation sets. After regularization, while the training set has a higher MLE loss due to the reduction of overfitting, the MLE loss for validation and test sets are reduced for **some** values of Lambda, most significantly when Lambda is around 35 to 40. The curves demonstrate the ideal hyperparameter (Lambda) to obtain the lowest validation and test loss, and show that at the optimal Lambda, our model can have a better predictive power in validation and test sets.

AUC Measures with respect to Lambda

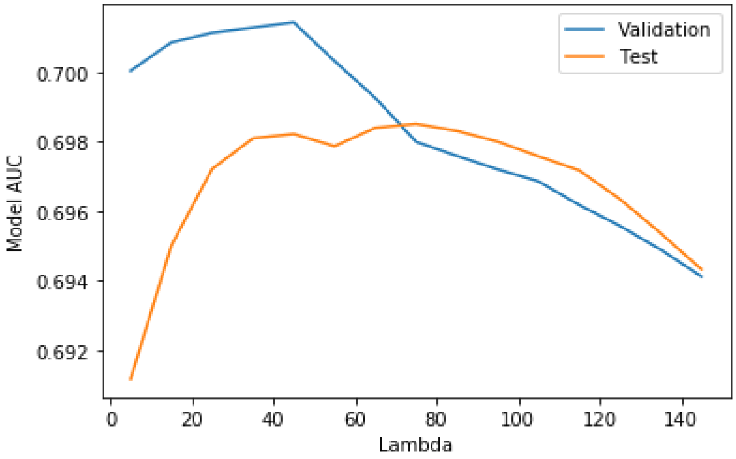


Fig. 5: Plot of Model AUC against Lambda

The AUC measure is the main measure of how accurately a model can make predictions. By introducing a plot (Fig. 5) of the AUC measures in a given range of Lambda, may use the maximal points to obtain the hyperparameter that gives the most ideal performance in test and validation sets.

Our group found the hyperparameters leading to the highest AUC values to be:

- Validation: λ = 45 (AUC = 0.701)
- Test: λ = 75 (AUC = 0.699)

Findings using suggested optimal lambda

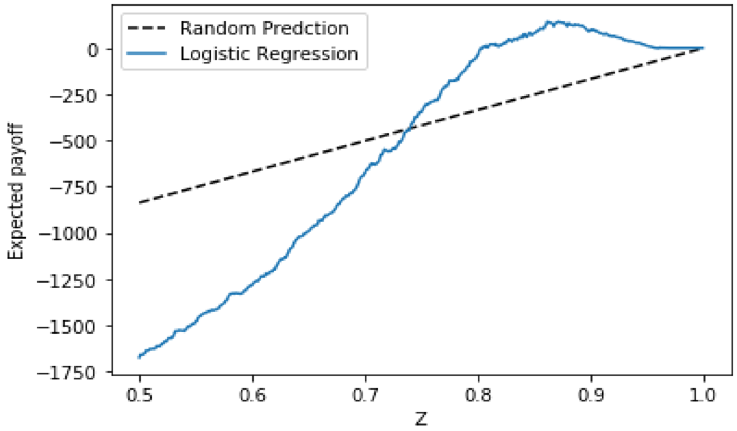


Fig. 6: Expected Payoff Curve of Regularized model (Validation set and Test set)

After using the suggested optimal lambda of 40, our group obtained the following results on the validation set:

- z_optimal = **0.871**
- Confusion Matrix

TP = 528	FN = 1759
FP = 42	TN = 671

- EP of regularized model = **$142.68**

Observation and evaluation: Our group initially found it odd that EP of regularized model is lower than EP of unregularized model ($187.47). However, this can be explained by the fact that the optimal lambda was not obtained through the tuning of the lambda (Fig. 5). More importantly, while the regularized model which has a lower MLE loss in the validation and test sets, a model with good prediction power is not aware of the business decision-aware objective, to maximise the expected profits, thus the lower EP results. In this case, a false positive prediction will lead to a complete loss of capital, thus the model can be improved by focusing on reducing the fall-out rate to maximise EP.

On the test set, our group managed to yield the final metrics using the optimal lambda and the newly found z_optimal_l1 of **0.871**.

- AUC score of regularized classifier = **0.698**
- Confusion Matrix

TP = 490	FN = 1544
FP = 38	TN = 556

- EP of regularized model = **$153.52**

Observation and evaluation: In line with our expectations, as seen in the MLE loss function curve, the test set has a lower MLE loss than the validation set. As such, the EP of the regularised model is higher than that of the validation set. However, we must keep in mind the decision aware objective in order to yield better results.

4. Using model to set Interest Rates

Using the optimal decision criteria z to set interest rates: Given the optimal model of the training data to issue a loan for a given customer if the predicted probability of paying it off is z, this means we want to optimize the interest rate feature so the model predicts as close to z as possible for any given customer. Too low an interest rate and the profit decreases, too high and the default risk increases. Hence, we can simply try out a range of interest rates for any given customer's set of features and select the rate that gives us the model output closest to z.