UNIVERSITI TUNKU ABDUL RAHMAN

Assignment 2

Course Code: UECS2153

Course Name: Artificial Intelligence

Lecturer: Dr. Ng Oon-Ee

Academic Session: 2019/05

Title: Supervised Learning

Student ID	Student Name	Major
16UEB03890	Chin Kai Xiang	Software Engineering (SE)

Contents

Introduction

The data chosen in this project is called “Default of Credit Card Clients Dataset”. This dataset contains information on default payments, demographic factors, credit data, history of payment, and bill statements of credit card clients in Taiwan from April 2005 to September 2005. This is the link to the dataset: https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients. The aim of this project is to construct a predictive model using various machine learning algorithms.

Data Cleaning

1.

	PAY_5	PAY_6	BILL_AMT1	BILL_AMT2
0	2	21034	27321
2	0	48657	45079

Figure 1.1 shows that some of the features have very different scales

When features are highly varying in range, the features with high magnitudes will have a bigger effect on the distance calculations as compared to features with low magnitudes. This will affect the learning performance of the classifier, and classification accuracy and precision. Therefore, the RobustScaler is used to scale the features to a common range. RobustScaler is chosen because it uses statistics that are robust to outliers so the model training process will not be affected by outliers yielding a more accurate and precise result.

2.

	male	grad_school	university	high_school	married	single
0	0	1	0	0	1
1	0	1	0	0	1

Figure 1.2 shows the one-hot encoding of three categorical attributes

The SEX, EDUCATION and MARRIAGE are categorical data with their values represented by numerical values such as 1, 2, 3 and so on. This might confuse the machine learning algorithms into thinking that these attributes have some kind of order or hierarchy and take into account the magnitudes of the values in the training process. As a result, the classification result might not be correct and therefore, these three attributes are one hot encoded to avoid this issue.

3.

```
Plotting for column PAY_0
```

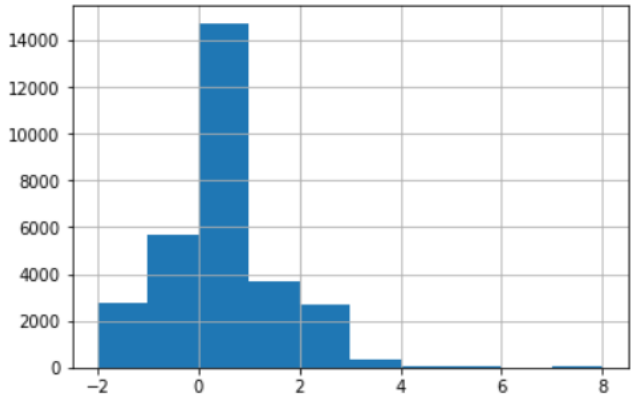

Figure 1.3 shows the distribution of column PAY_0

Figure 1.3 shows that the PAY_0 column contains undocumented categories which are -2 and 0. The same goes to the other PAY_n columns. According to the documentation, the PAY_n variables are just variables represent the number of months (payment delay) and -1 is used to indicate "pay duly". Therefore, we can infer that PAY_n features represent payment delayed for 0 months i.e. “pay duly” if it is less than or equal to 0. To solve this issue, all values of PAY_n features which are less than 0 are modified to belong to category 0.

Hyperparameter Tuning

All the hyperparameter tuning are using GridSearchCV. GridSearchCV will perform an exhaustive search over specified parameter values for an estimator. Besides, the grid-search process over the parameter grid is cross-validated automatically to optimize the parameters of the estimator. Note that all the grid-search processes in this project are performed with 3-fold cross-validation as the GridSearchCV instances are all initialized with parameter (cv=3). Thus, the mean test score i.e. mean accuracy of each classification is a test result of 3-fold cross-validation.

1. Logistic Regression

For Logistic Regression classifier, the hyperparameters that are being tuned are C (inverse of regularization strength) and penalty (norm used in penalization).

```
Best mean test score and best parameters:
0.817875 {'C': 2.7825594022071245, 'penalty': 'l1'}

List of Mean test scores and respective parameters:
0.817625 {'C': 1.0, 'penalty': 'l1'}
0.8176666666666667 {'C': 1.0, 'penalty': 'l2'}
0.817875 {'C': 2.7825594022071245, 'penalty': 'l1'}
0.81775 {'C': 2.7825594022071245, 'penalty': 'l2'}
```

Figure 1.4 shows the result of the grid-search process for Logistic Regression classifier

After performing the grid-search, the best parameters obtained are '2.7825594022071245' for hyperparameter 'C' and 'l1' for hyperparameter 'penalty'. This best parameters pair yielded the highest mean accuracy of 0.817875.

2. Decision Tree

For Decision Tree classifier, the hyperparameters that are being tuned are criterion (function to measure the quality of a split), max_depth (maximum depth of the tree) and min_samples_split (minimum number of samples required to split an internal node).

```
Best mean test score and best parameters:
0.8197083333333334 {'criterion': 'gini', 'max_depth': 5, 'min_samples_split': 2}

List of Mean test scores and respective parameters:
0.7729583333333333 {'criterion': 'gini', 'max_depth': None, 'min_samples_split': 2}
0.774375 {'criterion': 'gini', 'max_depth': None, 'min_samples_split': 4}
0.8197083333333334 {'criterion': 'gini', 'max_depth': 5, 'min_samples_split': 2}
0.8196666666666667 {'criterion': 'gini', 'max_depth': 5, 'min_samples_split': 4}
```

Figure 1.5 shows the result of the grid-search process for Decision Tree classifier

After performing the grid-search, the best parameters obtained are ‘gini’ for hyperparameter ‘criterion’, ‘5’ for hyperparameter ‘max_depth’ and ‘2’ for hyperparameter ‘min_samples_split’. This best parameters pair yielded the highest mean accuracy of ~0.819708.

3. Neural Network (Feedforward)

For Neural Network classifier, the hyperparameters that are being tuned are batch_size (number of samples per gradient update), epochs (number of epochs to train the model) and optimizer (optimizer instance).

```
Best mean test score and best parameters:
0.8190416660308838 {'batch_size': 24, 'epochs': 50}

List of Mean test scores and respective parameters:
0.8087916670838992 {'batch_size': 24, 'epochs': 30}
0.8190416660308838 {'batch_size': 24, 'epochs': 50}
0.8153333333333334 {'batch_size': 32, 'epochs': 30}
0.8187916666666667 {'batch_size': 32, 'epochs': 50}

Best mean test score and best parameters:
0.8190833332538605 {'optimizer': 'Adam'}

List of Mean test scores and respective parameters:
0.7844999993840853 {'optimizer': 'SGD'}
0.787708332846562 {'optimizer': 'Adagrad'}
0.8190833332538605 {'optimizer': 'Adam'}
```

Figure 1.6 shows the result of the grid-search process for Neural Network classifier

Note that the hyperparameters batch_size and epochs are tuned separately from hyperparameter optimizer due to some processing power limitations. After performing the grid-search for ‘batch_size’ and ‘epochs’ hyperparameters, the best parameters’ values obtained are ‘24’ and ’50’ for ‘batch_size’ and ‘epochs’ respectively. This best parameters pair yielded the highest mean accuracy of ~0.808792. Meanwhile, the best parameter value for hyperparameter ‘optimizer’ obtained from the grid-search process is ‘Adam’ and the highest mean accuracy achieved is ~0.819083.

After these best parameters pairs are determined, all the models are trained using these best parameters pair.

Comparison of Performance of Various Predictive Models

1. Basic Metrics Analyses

Logistic Regression

PREDICTION TRUE	pay	default	Total
pay	4480	193	4673
default	892	435	1327
Total	5372	628	6000

Decision Tree

PREDICTION TRUE	pay	default	Total
pay	4459	214	4673
default	864	463	1327
Total	5323	677	6000

Feedforward Neural Network

PREDICTION TRUE	pay	default	Total
pay	4442	231	4673
default	848	479	1327
Total	5290	710	6000

Table 1.1 shows the confusion matrixes of models

	LogisticReg	DecisionTree	NeuralNet
accuracy	81.9167	82.0333	82.0167
precision	69.2675	68.39	67.4648
recall	32.7807	34.8907	36.0965
f1-score	44.5013	46.2076	47.0299
AUC	76.2671	75.9425	77.6403

Figure 1.7 shows the evaluation metrics in numerical form.

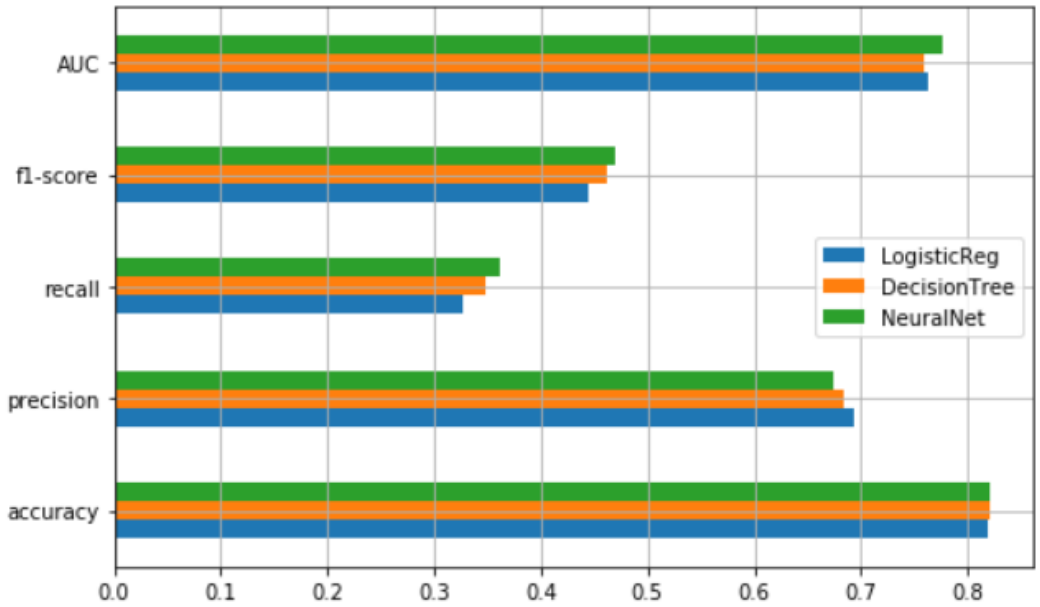

Figure 1.8 shows the evaluation metrics in graphical form.

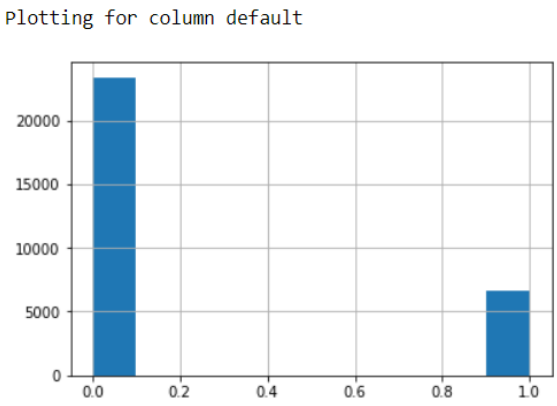

Figure 1.9 shows the uneven distribution of the labels of the target column 'default'.

In the context of this application, accuracy represents the overall frequency that the model predicts the defaulters and non-defaulters correctly. In terms of accuracy, Decision Tree classifier has the highest accuracy (82.0333) as shown in Figure 1.7 and 1.8.

Meanwhile, precision represents the frequency that the model predicts defaults correctly. The Logistic Regression classifier achieved the highest hyperparameters (69.2675) as shown in Figure 1.7 and 1.8.

On the other hand, recall represents the proportion of actual defaulters that the model will predict correctly. Neural Network classifier achieved the highest recall (36.0965) as shown in Figure 1.7 and 1.8.

Moreover, f1-score represents a balance between Precision and Recall and thus, is used as a comparison indicator between precision and recall values. Neural Network classifier has the highest f1-score (47.0299) as shown in Figure 1.7 and 1.8.

Although there are many metrics shown above but not all the metrics are important for evaluating the models. In the context of this application, false negatives i.e. Type II error (A person who will default predicted as payer) are worse than false positives i.e. Type I error (A person who will pay predicted as defaulter) because the banks will lose more money if they lend more money to people who will not pay them. Therefore, the recall metric is more important in the current context as it takes into account the false-negative rate and generally, the recall value should be higher. Furthermore, f1-score is a better measure when there is an uneven class distribution which is the exact same situation in this dataset (as demonstrated in Figure 1.9). This is because f1-score keeps a balance between precision and recall to prevent misleading results when they are interpreted separately.

As a result, recall and f1-score are the most appropriate metrics to be used to evaluate the model performance in the current context. From the above comparisons, we can conclude that Neural Network is the best model to predict the credit card default since it has the highest recall and f1-score.

2. Further Supporting Analyses

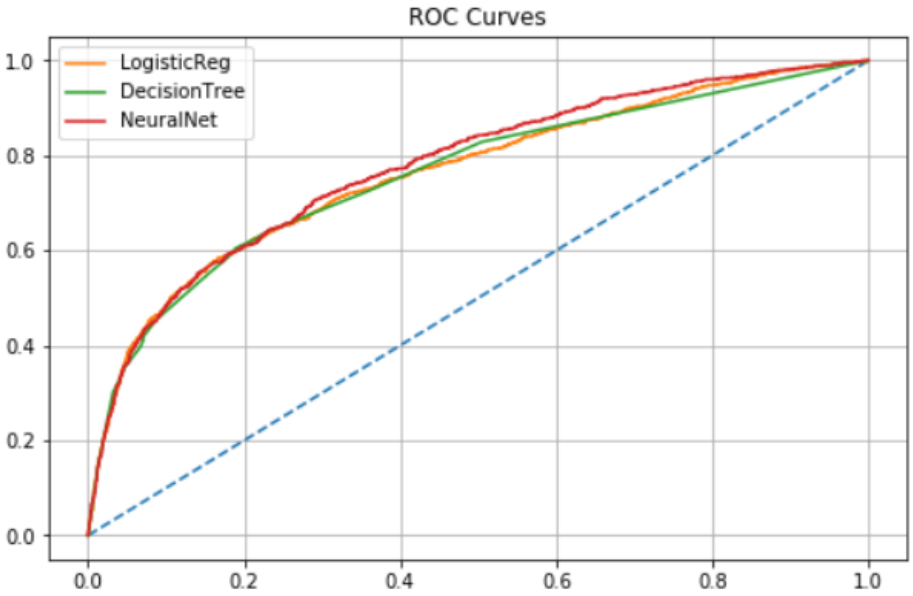

Figure 1.10 shows the comparison of ROC Curves of various models.

ROC curve is a probability curve which compares the True Positive Rate (TPR) and False Positive Rate (FPR). The area under the ROC curve (AUC) measures the capability of the model to distinguish between classes. Thus, the higher the AUC, the better the model is at distinguishing between defaulters and non-defaulters. In terms of AUC, Neural Network has the highest AUC of 77.6403 (as shown in Figure 1.7 and 1.8) as compared to two other models. Although the ROC curve is more suitable to use when there is an even distribution of class instances but it can still serve as a good indicator to the performance of the model at predicting and differentiating distinct classes.

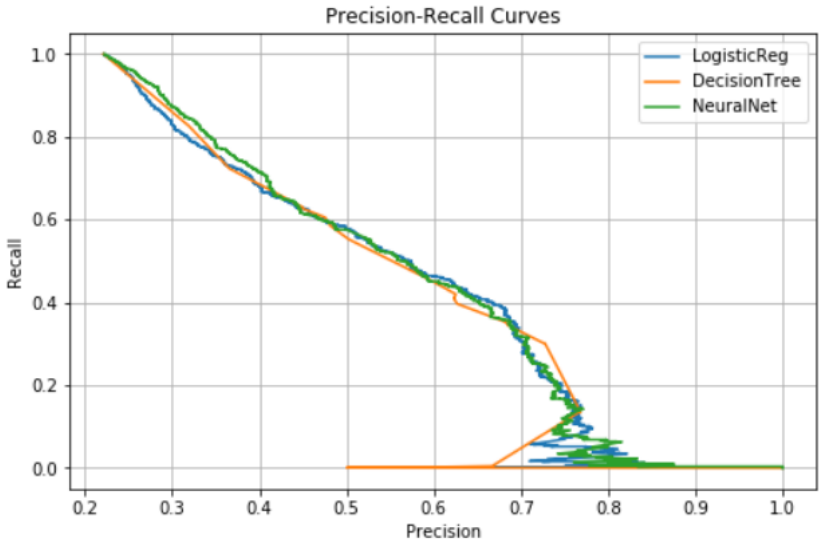

Figure 1.11 shows the comparison of Precision-Recall Curves of various models.

Precision-Recall curves summarize the trade-off between the true positive rate and the positive predictive value for a predictive model using different probability thresholds. Similar to f1-score, it is suitable when there is an uneven distribution of class instances. From the curve, we can infer that the Neural Network has the highest recall and the precisions of different models are quite close to each other.

The Verdict

The neural network classifier works well in this dataset most probably due to the reason that this dataset is concerning finance data which is highly non-linear and random. A neural network is effective at finding the non-linear relationships between data and using them to predict or classify new data. Furthermore, a neural network consisting of multiple layers of neurons is able to model almost any complex and non-linear problems which are the exact same situation in the real-world problems.

From the interpretation of all the evaluation metrics above as well as the ROC curve and Precision-Recall curve, we can conclude that Neural Network classifier performs the best for this credit card default dataset. The best model was chosen based on mainly the minimum value of Type 2 error i.e. higher recall value and high f1-score.

By having this predictive model, the credit card issuers can better understand the behaviour of its current and potential future clients. Besides, this model can also provide them with insightful information to plan their future strategies regarding loan products targeting. Lastly, this model allows the issuers to make informed decisions about whether or not to approve a credit card application as well as the credit limit granted.