Research on Default Payments of Credit Card

Using Machine Learning Methods

Wanxin Chen

chen.wanx@husky.neu.edu

Abstract

Excessive issuance of credit cards for unqualified applicants may result in some cardholders, regardless of their repayment ability, excessive use of credit card consumption and accumulation of a large number of credit problems. This crisis is not only a cardholder but also a huge economic risk for banks. This research aims to predict customers' credit risk and to reduce the damage and uncertainty. The secondary aim of this project is to find out if traditional Machine Learning methods perform better than Artificial Neural Network classifiers as well as Deep Learning networks. After training Unsupervised Learning model (*K*-means clustering), Supervised Learning model (Logistic Regression & Random Forest) and Neural Network model, the results obtained suggest that the performance of Logistic Regression is the worst among the models (with the accuracy 80.80%), but Neural Network model (with the accuracy 81.57%) performs similar with Random Forest model (with the accuracy 81.27%). The findings related to this project could provide vital information for the detection of default payments using machine learning in the future and financial research.

Keywords

Data Mining, *K*-means Clustering, Logistic Regression, Random Forest, Artificial Neural Network

1. Introduction

In the era of information explosion, companies generate and collect large amounts of data every day. Discovering useful knowledge from the database and turning it into actionable results is a major challenge for companies.

Data mining is the process of exploring and analyzing large amounts of data in an automated or semi-automated manner to discover meaningful patterns and rules. Currently, data mining is an indispensable tool in decision support systems and plays a key role in market segmentation, customer service, fraud detection, credit and behavioral scoring, and benchmarking.

This project takes payment data in October, 2005, from an important bank (a cash and credit card issuer) in Taiwan and the targets were credit card holders of the bank. Among the total 30,000 observations, 6636 observations (22.12%) are the cardholders with default payment. This research employed a binary variable – default payment (Yes = 1, No = 0), as the response variable.

The primary aim of this project is to identify the most efficient machine learning model to predict the default of credit card clients and fine tune the model to obtain the highest accuracy.

2. Methods

2.1 K-means clustering

K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable *K*. The algorithm works iteratively to assign each data point to one of *K* groups based on the features that are provided. Data points are clustered based on feature similarity. The results of the *K*-means clustering algorithm are:

1. The centroids of the *K* clusters, which can be used to label new data
2. Labels for the training data (each data point is assigned to a single cluster)

Rather than defining groups before looking at the data, clustering allows you to find and analyze the groups that have formed organically.

2.2 Logistic regression

Logistic regression, also known as logistic regression analysis, is a predictive analysis model, which is often used in data mining, automatic disease diagnosis, economic forecasting and other fields. It is a basic machine learning technique that uses linear weighted combinations of features and predicts the probability of different classes when the dependent variable is dichotomous (binary).

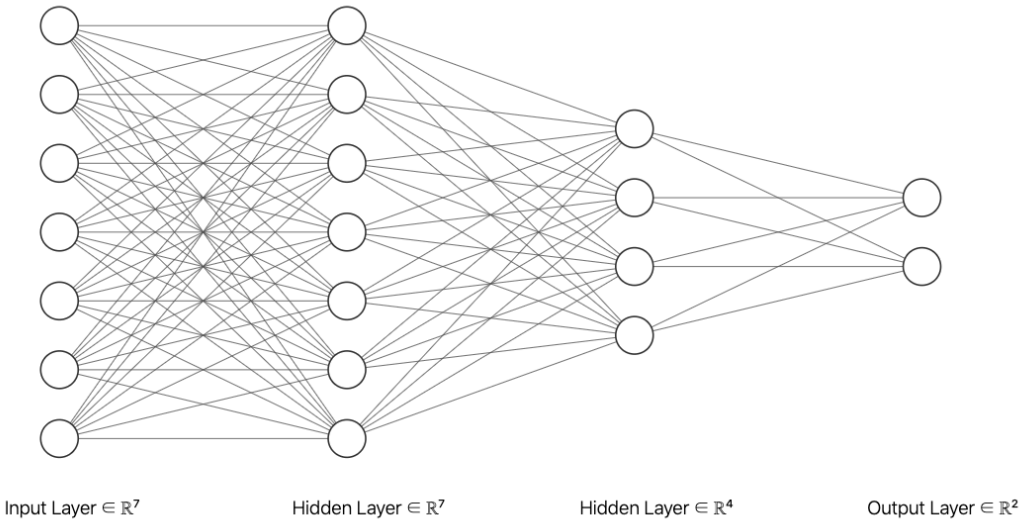
2.3 Random Forest

Random Forest is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one

of the most used algorithms, because its simplicity and the fact that it can be used for both classification and regression tasks. To say it in simple words: Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

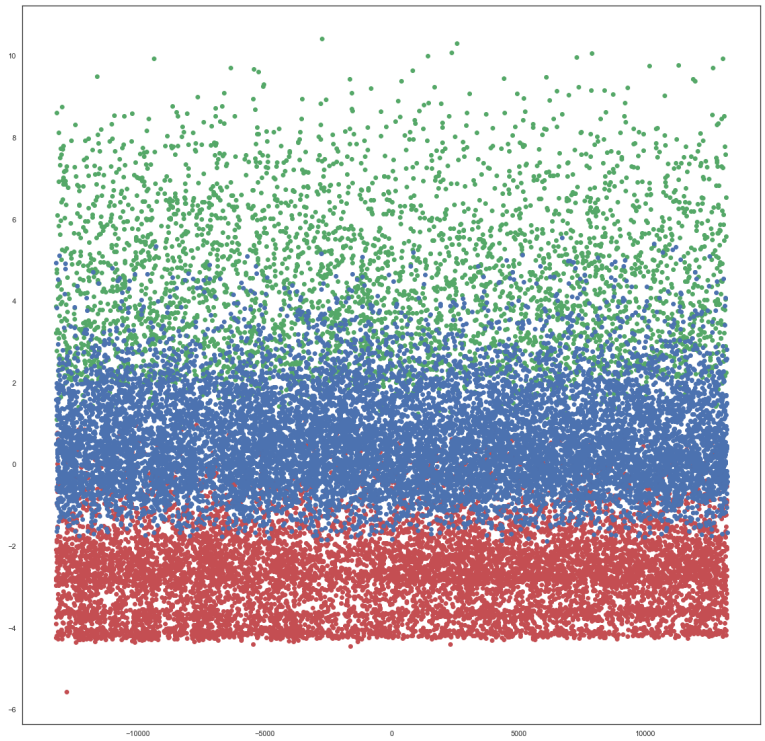
2.4 Artificial Neural Network

Artificial Neural Network (ANN), referred to as Neural Network (NN). In the fields of machine learning and cognitive science, it is a mathematical model or a computational model that mimics the structure and function of biological neural networks (The central nervous system of animals, especially the brain) as well as estimates or approximates a function. The neural network is calculated by a large number of artificial neuronal connections. In most cases, artificial neural networks can change the internal structure based on external information, which means that it is an adaptive system. Modern neural network is a kind of nonlinear statistical data modeling and usually optimized by a learning method based on mathematical statistics type, so it is also a practical application of mathematical statistical methods. The standard mathematical method we can get a lot of local structure space that can be expressed by function. On the other hand, in the field of artificial perception of artificial intelligence, we can use the application of mathematical statistics to make the decision of artificial perception. It is said that through statistical methods, artificial neural networks can have simple decision-making ability and simple judgment ability similar to humans. Therefore, this method is more advantageous than formal logical reasoning calculation.



3. Results

3.1 K-means clustering:



3.2 Logistic Regression:

```
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
# Use sklearn's LogisticRegression function
clf = LogisticRegression()
X2 = dataset[['SEX', 'EDUCATION','MARRIAGE',
       'PAY_0','PAY_2','PAY_3','PAY_4']]
y2 = dataset[['default.payment.next.month']]
X2 = np.array(X2)
y2 = np.array(y2)
trainX,testX, trainy, testy = train_test_split(X2,y2,test_size=0.2, random_state=0)
clf.fit(trainX, trainy)
```

```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
          intercept_scaling=1, max_iter=100, multi_class='warn',
          n_jobs=None, penalty='l2', random_state=None, solver='warn',
          tol=0.0001, verbose=0, warm_start=False)
```

```
print ('The training accuracy of Logistic Regression is' ,clf.score(trainX, trainy))
```

```
The training accuracy of Logistic Regression is 0.8085891311545192
```

```
print ('The test accuracy of Logistic Regression is' ,clf.score(testX,testy))
```

```
The test accuracy of Logistic Regression is 0.8043889519485433
```

```
# Cross validation
from sklearn.model_selection import cross_val_score
acc = cross_val_score(clf, X2, y2.ravel( ), cv=10, scoring='accuracy').mean()
print('The test accuracy after cross validation is',acc)
```

```
The test accuracy after cross validation is 0.8080160383838308
```

3.3 Random Forest:

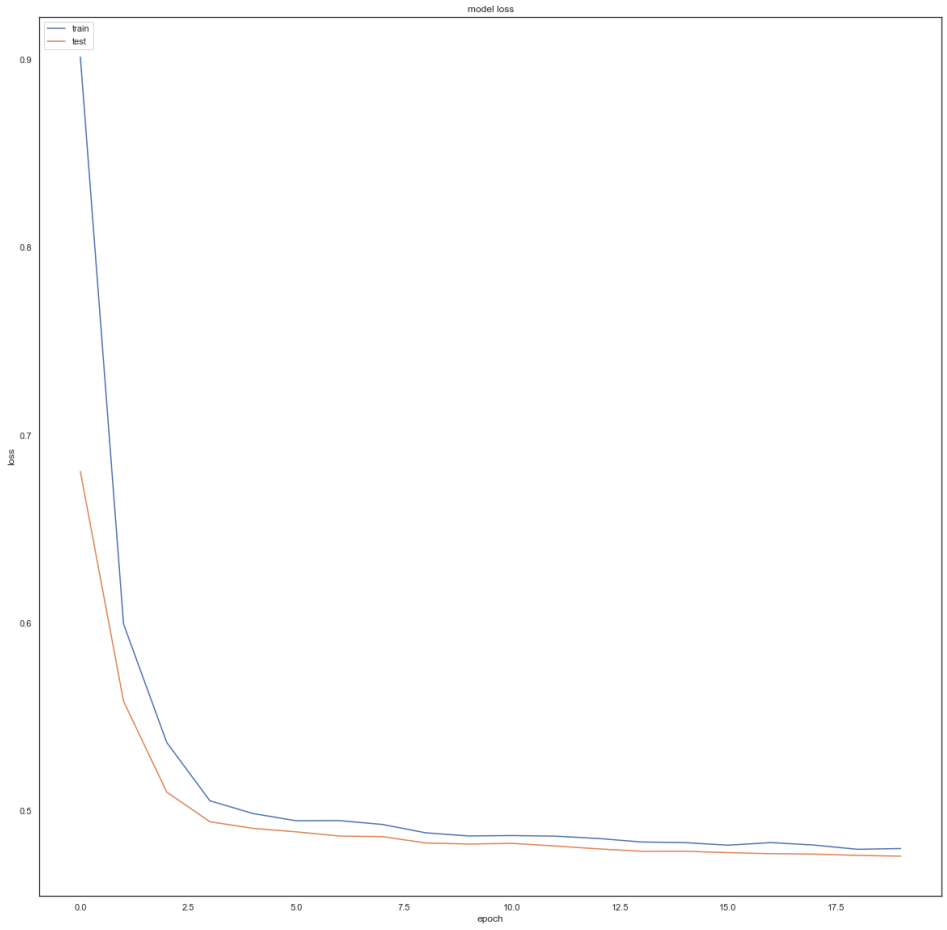
```
# What hyper-parameter values work best
print(dtrfModel.best_params_)
print(dtrfModel.best_estimator_)

print('The best accuracy is',dtrfModel.best_score_)
print('The test accuracy is', dtrfModel.score(testX, testy))
```

```
{'criterion': 'gini', 'max_depth': 6, 'min_samples_leaf': 50, 'n_estimators': 10}
RandomForestClassifier(bootstrap=True, class_weight=None, criterion='gini',
            max_depth=6, max_features='auto', max_leaf_nodes=None,
            min_impurity_decrease=0.0, min_impurity_split=None,
            min_samples_leaf=50, min_samples_split=2,
            min_weight_fraction_leaf=0.0, n_estimators=10, n_jobs=None,
            oob_score=False, random_state=None, verbose=0,
            warm_start=False)
The best accuracy is 0.8163458355011115
The test accuracy is 0.8127128263337117
```

3.4 Artificial Neural Network:

```
score, acc = model.evaluate(test_x_a, test_y_a,
                            batch_size=100)
print('Test score:', score)
print('Test accuracy:', acc)
```

```
5286/5286 [==============================] - 0s 6us/step
Test score: 0.468473495473656
Test accuracy: 0.8157396821239614
```



4. Discussion

An important finding in machine learning is that no single algorithm works best across all possible scenarios. Thus, no algorithm strictly dominates in all applications; the performance of machine learning algorithms varies wildly depending, for example, on the application and the dimensionality of the dataset. Accordingly, a good practice is to compare the performance of different learning algorithms to find the best one for the particular problem.

In this case, from our experiments, we can conclude that traditional machine learning algorithms such as Logistic Regression and Random Forest aid in less efficient classification of data compared to Deep Learning Neural Networks. There are several possibilities for this outcome which will be discussed in this section.

However, often there is not enough time and/or money to test and optimize every algorithm in order to its quality in a specific context. On the other hand, particular weaknesses of an approach can lead to avoid a specific algorithm in a specific context. In these cases, a decision about an algorithm has to be made before starting the project.

Random Forests require much less input preparation. They can handle binary features, categorical features as well as numerical features and there is no need for feature normalization. Random Forests are quick to train and to optimize according to their hyperparameters. Thus, the computational cost and time of training a Random Forest are comparatively low. Furthermore, a Random Forest can be trained with a relative small amount of data. Neural Networks usually need more data to achieve the same level of accuracy. On the other hand, Random Forests often have little performance gain when a certain amount of data is reached, while Neural Networks usually benefit from large amounts of data and continuously improve the accuracy.

The intention of this research was to show that Neural Networks, despite their current high visibility in the media, not always need to be the first choice in

selecting a machine learning methodology. Random Forests not only achieve (at least) similarly good performance results in practical application in many areas, they also have some advantages compared to Neural Networks in specific cases. This includes their robustness as well as benefits in cost and time. They are particularly advantageous in terms of interpretability. If we were faced with the choice of taking a model with 81.27% accuracy that we understand or a model with 81.57% accuracy that we don't currently understand, we would probably choose the first one for many applications, for example, if the model is supposed to be responsible for investigating patients and suggesting medical treatment. These may be the reasons for the increasing popularity of Random Forests in practice.

References

B. Baesens, R. Setiono, C. Mues, J. Vanthienen. (2003). Using neural network rule extraction and decision tables for credit-risk evaluation. Management Science, 49 (3) (2003), pp. 312-329

Yeh, I. C., & Lien, C. H. (2009). The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. Expert Systems with Applications, 36(2), 2473-2480.

Ahmad, M.W.; Mourshed, M.; Rezgui, Y. (2017). Trees vs Neurons: Comparison between Random Forest and ANN for high-resolution prediction of building energy consumption. In: Energy and Buildings, volume 147, pp. 77–89.

Code with Documentation

https://github.com/WanxinChen/INFO6105_FinalProject

License