# ANALYZING SUPERVISED LEARNING METHODS FOR CREDIT CARD FRAUD DETECTION

**Jennifer Gao**
D. W. Daniel High School
MehtA+Tutoring

**Karen Situ**
University Hill Secondary School
MehtA+Tutoring

**Grace Tian**
Yorktown High School
MehtA+Tutoring

July 24, 2020

**ABSTRACT**

Credit card fraud has been a growing issue both in the United States and worldwide. Unfortunately, the countless losses caused by fraudulent transactions are usually paid for by the sales companies and credit card companies. This project analyzes supervised machine learning models for identifying fraudulent transactions based on the transaction information. In doing so, we have worked with three models: decision tree, K-nearest neighbors, and random forest, analyzing both the individual models and all possible combinations of models. A final fraud score is calculated as a probability of fraud based on all models, which achieved a recall of 82% with a precision of 84%. Our final model is a probability and not an absolute decision of fraud, which can allow credit card companies to factor in personal concerns when deciding their course of action in response to possible future frauds.

*Keywords:* Fraud Detection, Supervised Learning, Decision Tree, K Nearest Neighbors, Random Forest, SMOTE

## 1 Introduction

Every year, billions of dollars are lost worldwide due to credit card fraud. In the U.S. alone, 9.47 billion dollars were lost in 2018, and this amount is projected to increase in the coming years [1]. By analyzing the patterns of current credit card fraud data, future fraudulent credit card transactions could be predicted in advance and stopped.

Supervised learning methods perform better than unsupervised methods for credit card fraud detection [2]. Past studies have used supervised learning with decision trees (DT) [3], K-nearest neighbors (KNN) [4], and random forests (RF) [5] as individual models and have also compared their effectiveness [6]. However, there has not been evaluation of a combined DT-KNN-RF model. By producing various combinations of the three models, we were able to obtain higher precision and recall rates than any of the three models alone yielded.

## 2 Methodology

### 2.1 Dataset

The "Credit Card Fraud Detection" dataset from Kaggle consists of 492 fraudulent and 284807 nonfraudulent transactions, collected within a span of around 2 days [7]. Note that this dataset is heavily imbalanced, with less than 0.2% of the transactions being fraudulent. The 30 features include time in seconds after the first transaction in the dataset, transaction amount, and 28 other anonymized features representing sensitive data after principal component analysis (Figure 1).

**Preprocessing** In order to deal with the heavily imbalanced data classes, we use synthetic minority over-sampling technique (SMOTE), which generates synthetic data in the minority class to create a class balance for unbiased training [8]. We first split the imbalanced data using a 67:33 train:test split. This allows us to achieve the final desired 80:20 train:test split after generating more fraudulent data to train on using SMOTE (Figure 2).

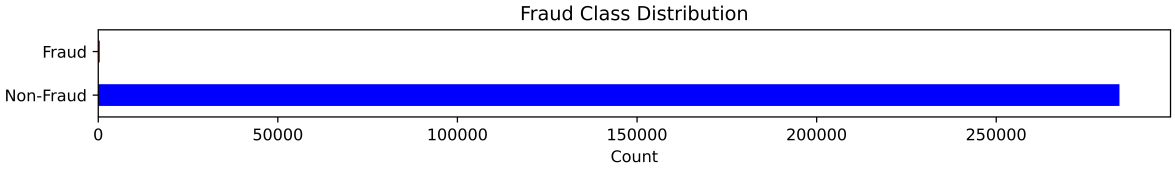


Figure 1: Imbalance in fraud class distribution.

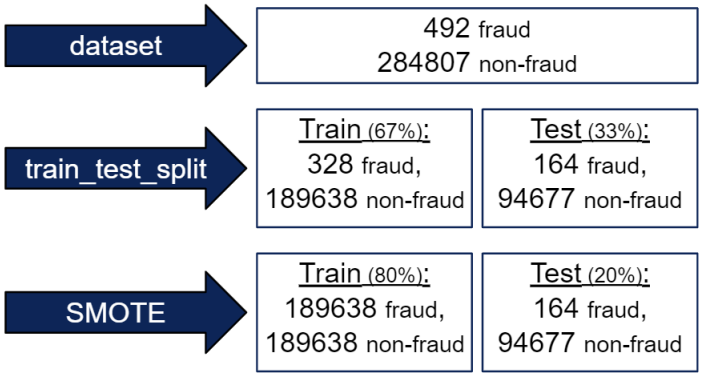


Figure 2: Use of SMOTE to balance testing data.

### 2.2 Models

We developed three separate supervised classification models: decision tree, k-nearest neighbors, and random forest. We then combine the three models in all possible combinations, and use both the individual and combined models to calculate a final fraud score for probability of fraud.

#### Decision Tree

We built a decision tree classifier with a 600:1 class weighting to reflect to approximate ratio of nonfraud:fraud. After conducting experiments on the max_depth parameter, we found that a maximum tree depth of 6 gave the highest recall without compromising precision. Although some higher max_depth yielded greater recall values, these increases were minor compared to the large decreases in precision, as the model begins to overfit after a maximum tree depth of 6 (Table 1).

Table 1: Evaluating decision tree at different values of max_depth.

| max_depth | Precision (%) | Recall (%) |
|---|---|---|
| 1 | 76.8 | 64.6 |
| 2 | 82.3 | 70.7 |
| 3 | 83.7 | 72.0 |
| 4 | 88.3 | 78.0 |
| 5 | 89.7 | 79.3 |
| **6** | **91.0** | **80.5** |
| 7 | 87.2 | 79.3 |
| 8 | 85.2 | 80.5 |
| 9 | 84.4 | 82.3 |
| 10 | 81.7 | 81.7 |

### K-Nearest Neighbors Classifier

We built a K-nearest neighbors classifier using a distance weight to put a larger emphasis on closer points. After conducting experiments varying the number of neighbors, we found that using four nearest neighbors optimized recall rate, so we used four nearest neighbors to classify our test data.

### Random Forest

A random forest algorithm has its advantage of not overfitting, which comes useful especially when training on imbalanced data. Random forests work by taking the most frequent results given by a set of decision trees. After testing, we have decided to implement the random forest classifier using 800 trees.

### Combined Models

We created three new models by combining our original K-nearest neighbors, random forest, and decision forest models pairwise. In each of these new combination models, a transaction is classified as fraud if both of the models combined labeled the transaction as fraud and classified as nonfraud otherwise. In addition, we created a fourth combination model by combining all three original models (K-nearest neighbors, decision tree, and random forest). In this model, a transaction is classified as fraud if and only if it is classified as fraud in all three of the original models.

### Final Fraud Score

Each of the three individual models and four combined models predicts either fraud or nonfraud for each datapoint. We converted this binary classification to a probability that each datapoint is fraud by imputing predictions of $0$ (nonfraud) and $1$ (fraud) with $\Pr(\text{fraud} \mid \text{pred nonfraud})$ and $\Pr(\text{fraud} \mid \text{pred fraud})$, respectively, for each of the seven models.

## 3 Results

### 3.1 Decision Tree

Our final decision tree had a recall of 80.5%, which means it successfully detected 80.5% of true frauds. It also had an accuracy of 99.95% which is due to the imbalanced test set. The decision tree's precision was 91.0%, so 91.0% of predicted frauds were actually frauds (Figure 3).

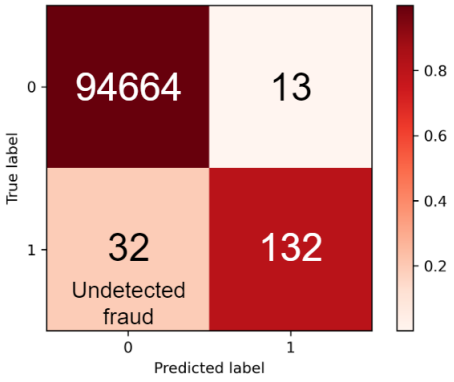


Figure 3: Confusion matrix for decision tree.

Due to the high interpretability of decision tree models, we analyzed the feature importance in the decision tree to determine which features were most influential in the classification process. From our analysis, we see that feature pc17 is the most important, which is confirmed since feature pc17 decides the first node in the decision tree (Figure 4). Although the pc17 feature is a result of anonymizing original sensitive features under principal component analysis, future research can look into uncovering the original sensitive features which are correlated to pc17. This could lead to more efficient methods of credit card fraud detection which are based on observable patterns in fraudulent transactions.

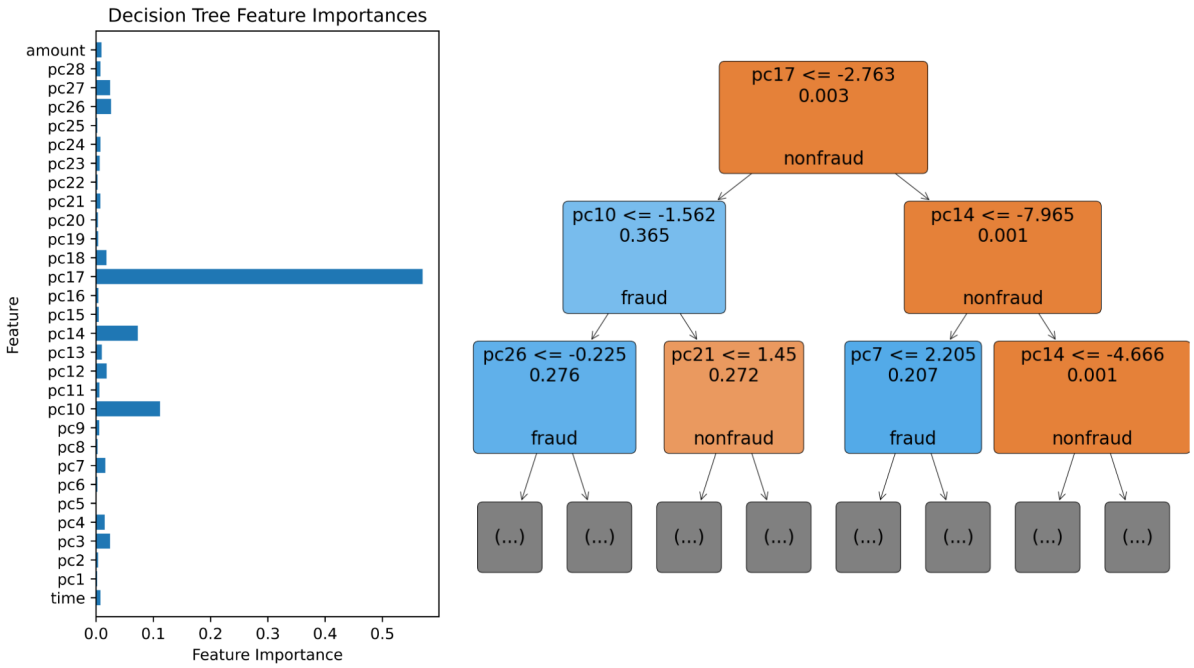


Figure 4: A graph of feature importances in the decision tree (left). A visualization of the decision tree at a depth of 2, where orange represents nonfraud and blue represents fraud (right).

### 3.2 K-Nearest Neighbors Classifier

Our KNN model had a recall rate of 56.1%,meaning it successfully detected 84.1% of true frauds (Figure 5). It also had a 95.74% accuracy rate. While this model has a lower recall and precision rate than the other two models, when combined with either, it yields a higher precision rate than any of the models alone.

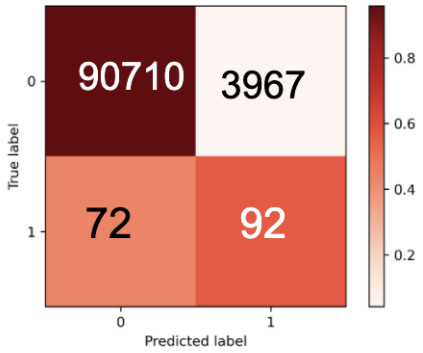


Figure 5: Confusion matrix for KNN classifier.

### 3.3 Random Forest

Our random forest model had a recall rate of 81.7% and a precision of 87.6%. Out of the three original models, the random forest the achieved the highest recall rate, meaning that it was the most successful at detecting fraud.

### 3.4 Combined Models

Our four combined models all achieved higher precision rates than any of the three individual models, with a precision rate of 95.6% for the DT-RF model, and above 98.8% for the other three. The combined models had lower recall rates than the individual models, with a recall of 80.0% for DT-RF, and around 52% for the other three. This increase in precision and decrease in recall can be attributed to using an AND logic gate on frauds for our combined models.The DT-RF model had a higher recall and lower precision that the other combined models since the individual DT and RF models yielded similar performances compared to the KNN.

### 3.5 Final Fraud Score

Our primary goal was to maximize recall, while not compromising precision. In order to combat the lowered recall rates of the combined model, we created a method of computing a final fraud score for probability of fraud using all seven models. When evaluated at a cutoff of 80% probability of fraud, our final fraud score had a precision of 83.9% and a recall of 82.3|%. The recall for our final fraud score model is the highest of all our models (Table 2).

Table 2: Final fraud detection model evaluations

| Model | Precision (%) | Recall (%) |
|---|---|---|
| DT | 91.0 | 80.5 |
| KNN | 2.27 | 56.1 |
| RF | 87.6 | 81.7 |
| DT-KNN | **98.9** | 52.4 |
| DT-RF | 95.6 | 79.9 |
| KNN-RF | **98.9** | 52.4 |
| DT-KNN-RF | **98.8** | 51.8 |
| Score* | 83.9 | **82.3** |

*Final fraud score evaluated at 80% cutoff.

## 4 Conclusion and Future Work

We first developed three supervised learning models for credit card fraud detection, where DT and RF both had a precision rate around 90% and a recall rate around 80%, and KNN performed relatively poorly on its own. Combining these models gave us four new models, all with precision rate above 95%, but recall rate under 80%. Our novel final fraud score model resulted in a precision of 84%, and a recall of 82%, which was higher than any of the individual or combined models.

Since our final score model determines a probability of fraud rather than a discrete prediction, this can allow credit card companies to take more personalized approaches when utilizing our model to make informed decisions against future credit card frauds. For example, if our model predicts a 3% probability of fraud for a transaction, some companies decide preventative action, whereas other companies may wish to delegate more of their resources against cases with higher probability of fraud.

Further factor adjustments could be made to our original K-nearest neighbors, decision tree, and random forest models. Instead of trying to optimize the recall for each individual model, parameters could be chosen to optimize the recall of the combined models. In addition, further models could be developed by combining different individual models, such as support vector machine and gradient boosting. Different methods of combination could also be tested.

## 5 Division of Labor

We divided the work as follows:

- Preprocessing and SMOTE: Grace Tian
- Decision Tree: Grace Tian

- K-Nearest Neighbors: Jennifer Gao
- Random Forest: Karen Situ
- Combined Models and Fraud Score: Grace Tian
- Paper and Poster: Jennifer Gao, Karen Situ, Grace Tian
- Website: Karen Situ

## 6 Acknowledgements

We would like to acknowledge our TA mentor Andrea Jaba for guiding us throughout this project, our instructor Haripriya Mehta for teaching us, and TAs Bhagirath Mehta and Marwa AlAlawi for their continued support through MehtA+ Tutoring.

## References

[1] The Nilson Report. Payment card fraud losses reach $27.85 billion, Nov 2019.

[2] Xuetong Niu, Li Wang, and Xulei Yang. A comparison study of credit card fraud detection: Supervised versus unsupervised. *arXiv preprint arXiv:1904.10604*, 2019.

[3] Suraj Patil, Varsha Nemade, and Piyush Kumar Soni. Predictive modelling for credit card fraud detection using data analytics. *Procedia computer science*, 132:385–395, 2018.

[4] N Malini and M Pushpa. Analysis on credit card fraud identification techniques based on knn and outlier detection. In *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, pages 255–258. IEEE, 2017.

[5] M Suresh Kumar, V Soundarya, S Kavitha, ES Keerthika, and E Aswini. Credit card fraud detection using random forest algorithm. In *2019 3rd International Conference on Computing and Communications Technologies (ICCCT)*, pages 149–153. IEEE, 2019.

[6] Masoumeh Zareapoor, KR Seeja, and M Afshar Alam. Analysis on credit card fraud detection techniques: based on certain design criteria. *International journal of computer applications*, 52(3), 2012.

[7] Machine Learning Group ULB. Credit card fraud detection, Mar 2018.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002.