Mid Term Project Report

March 2025

RESEARCH PAPER:

‘The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients’

I-Cheng Yeh a,*, Che-hui Lien b

a Department of Information Management, Chung-Hua University, Hsin Chu 30067, Taiwan, ROC
bDepartment of Management, Thompson Rivers University, Kamloops, BC, Canada

DATASET : ‘ default_credit_score’

SHAPE : (3000, 25)

Group 4

Marcelo, Dana, Aspyn

Project repo: [DS_301_Mid_term]

Project Summary

Our goal is to analyze and replicate existing research on the **"Default of Credit Card Clients"** dataset. By understanding the methodologies and results presented in prior studies, we aim to interpret their findings and validate their conclusions through hands-on implementation.

To achieve this, we will reproduce the original results and then explore ways to enhance the model's performance using the knowledge we have gained over the past week on Machine Learning Classification Models. This involves testing different algorithms, optimizing hyperparameters, and applying feature selection techniques to improve predictive accuracy.

Ultimately, our objective is to not only understand the dataset and the research behind it but also to experiment with alternative approaches that may yield better results.

Motivation Behind the Project

We chose this project to explore credit risk analysis and apply ML classification models to predict credit card defaults. The dataset is well-documented, making it ideal for benchmarking and improvement.

- **It aligns with our learning objectives**, allowing us to apply different ML classification techniques and assess their effectiveness.
- **It provides a challenging yet structured problem**, helping us gain hands-on experience in data preprocessing, feature selection, and model optimization.
- **It allows us to compare our findings with previous research**, giving us the opportunity to understand the reasoning behind their approach and explore ways to enhance it.

Research Paper Details

The research paper "The Comparisons of Data Mining Techniques for the Predictive Accuracy of Probability of Default of Credit Card Clients" by I-Cheng Yeh & Che-hui Lien (2009) examines different machine learning models for predicting credit card default probability.

Objective of the Paper

The study analyzes six data mining techniques to determine which provides the best predictive accuracy for credit default in Taiwan. The key question is:

Can machine learning models accurately estimate the probability of default, not just classify clients as risky or non-risky?

Dataset & Features

The dataset includes 25,000 credit cardholders from a Taiwanese bank. The features include:

- Demographic data (age, gender, education, marital status)
- Credit history (credit limit, past payment records)
- Billing and payment amounts (six months of historical data)
- Default Payment (Yes/No) as the target variable

Machine Learning Models Compared

1. K-Nearest Neighbors (KNN)
2. Logistic Regression (LR)
3. Discriminant Analysis (DA)
4. Naïve Bayes (NB)
5. Artificial Neural Networks (ANNs)
6. Classification Trees (CTs)

Key Findings

★ KNN had the highest accuracy in training, but ANNs performed best in validation.
★ ANNs were the only model that could accurately estimate the real probability of default using a novel method called Sorting Smoothing Method (SSM).
★ Decision Trees and Naïve Bayes also performed well, while Logistic Regression and Discriminant Analysis were weaker in this dataset.

Conclusion

The study suggests that ***Artificial Neural Networks (ANNs) are the best model for estimating credit default probability***, outperforming traditional statistical methods. It highlights that probability estimation is more valuable than binary classification in financial risk management

---

How ANNs Work & Why They Were Better

Artificial Neural Networks (ANNs) mimic the human brain, processing data through **layers of neurons** that learn patterns via **backpropagation**.

Why ANNs Outperformed Binary Classification?

1. **Captured Nonlinear Relationships** – Unlike simpler models, ANNs identified complex interactions in credit risk factors.
2. **Estimated Probability, Not Just Labels** – Instead of a simple **"default/no default"**, ANNs predicted a **probability**, making risk assessment more precise.
3. **Higher Accuracy in Validation**

Dataset details

shape (30000, 25)

Feature Name	Description	Values
default_payment_next_month	Whether the customer will default next month	**1:** Default (Yes) / **0:** No Default (No)
LIMIT_BAL	Credit limit amount (NT dollars)	Numeric
Gender	1: Male, 2: Female	**1, 2**
EDUCATION	Education Level	**1:** Graduate / **2:** University / **3:** High School / **4:** Others
MARRIAGE	Marital Status	**1:** Married / **2:** Single / **3:** Others
AGE	Age in years	Numeric
PAY_ST9 - PAY_ST4	Repayment status (Apr - Sep 2005)	**1:** Paid on time / **2:** Not used / **1~9:** Delay (months)
BILL_AMT9 - BILL_AMT4	Bill amount (NT dollars) for each month (Apr - Sep 2005)	Numeric
PAY_AMT9 - PAY_AMT4	Amount paid (NT dollars) for each month (Apr - Sep 2005)	Numeric

Sample from Dataset

	ID	LIMIT_BAL	SEX	EDUCATION	MARRIAGE	AGE	PAY_0	PAY_2	PAY_3	PAY_4	...
0	1	20000	2	2	1	24	2	2	-1	-1	...
1	2	120000	2	2	2	26	-1	2	0	0	...
2	3	90000	2	2	2	34	0	0	0	0	...

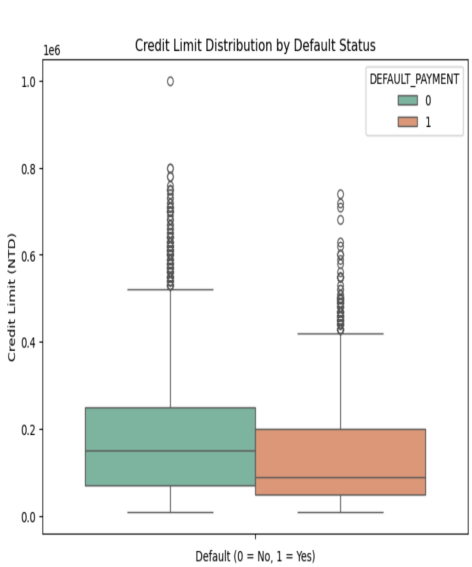
Sample from Dataset - Y Feature/ Target

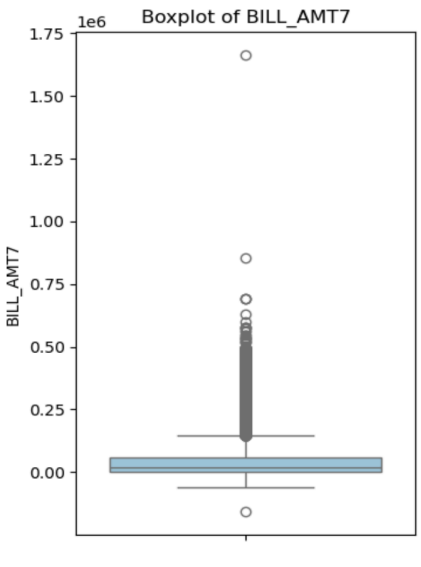
PAY_AMT1	PAY_AMT2	PAY_AMT3	PAY_AMT4	PAY_AMT5	PAY_AMT6	default payment next month
0	689	0	0	0	0	1
0	1000	1000	1000	0	2000	1
1518	1500	1000	1000	1000	5000	0
2000	2019	1200	1100	1069	1000	0
2000	36681	10000	9000	689	679	0

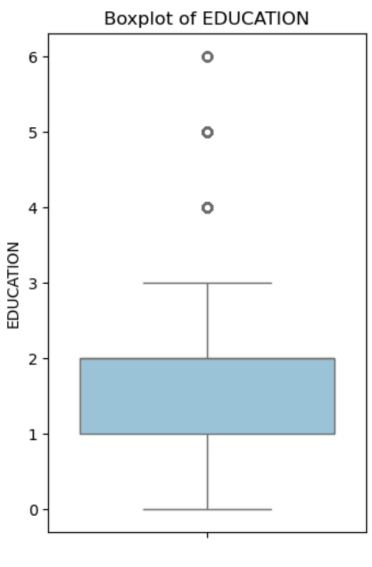
Data preprocessing and feature engineering

Outlier Detection & Handling

There was a value in the Credit Limit Amount that seemed like an outlier (≥ 1,000,000), so we filtered it to the top 5%.







A negative value of bill amount means the customer overpaid, but since our target is **default payment next month**, we don't need to consider overpayments. Therefore, we replaced all negative values with **0**. Additionally, we handled the outliers in the **Bill Amount** and **Pay Amount** columns using the **Interquartile Range (IQR) method**. This helped identify and remove extreme values that could negatively impact the analysis. We replaced values **5 and 6** in the "EDUCATION" column with **4** since their exact meaning is unknown, categorizing them as "Others."

For Feature Scaling, we did StandardScaler to standardize the variables and ensure they are on the same scale.

Steps reproduced from the paper

In this study, we implemented **Logistic Regression (LR), K-Nearest Neighbors (KNN), and Classification Trees** as described in the paper. Each model was trained and evaluated to analyze its effectiveness in handling imbalanced data and predicting default cases.

1. Logistic Regression (LR)

- Implemented **Logistic Regression as a baseline model** for binary classification.
- Applied **class weighting (class_weight={0:1, 1:2})** to address class imbalance.
- Performance evaluation results: **Accuracy = 0.80**

2. K-Nearest Neighbors (KNN)

- Implemented KNN for classification, **optimizing k** using hyperparameter, finding the best '*K value'*
- **Standardized features** before training to improve distance calculations.
- **Tried feature selection using Random Forest feature importance**, but results remained the same.
- **Applied SMOTE** (Synthetic Minority Over-sampling Technique) to handle class imbalance, but performance worsened, likely due to KNN's sensitivity to synthetic data.
- Performance evaluation results: **Accuracy = 0.81**

3. Classification Trees

- Implemented **Decision Tree-based classification models** following the methodology in the paper.

- **Hyperparameter tuning** was performed to control model complexity, with the following key settings:
  - **max_depth=4**: Limited tree depth to prevent overfitting.
  - **min_samples_split=200**: Allowed node splitting only when at least 200 samples were present, ensuring model stability by preventing excessive fragmentation.
  - **min_samples_leaf=50**: Required at least 50 samples per leaf node, simplifying the model and improving generalization.
  - **random_state=42**: Ensured reproducibility of results.
- Performance evaluation results: **Accuracy = 0.82**

While the classification tree achieved high overall accuracy, **it struggled to detect default cases effectively**.

This study faithfully reproduced the models presented in the paper and provided insights into how each model handles imbalanced data and predicts default cases.

Model Performance Evaluation

The performance of multiple classification models was assessed based on Accuracy, Precision, Recall, and F1-score, with a particular focus on the minority class (default cases). The results are summarized in Table 1.

Table 1. Performance Comparison of Classification Model

Model	**Accuracy**	Precision (Class1)	Recall (Class1)	F1-score (Class1)	***Error Rate Training***	***Area Ratio Training***
Logistic Regression	**0.80**	0.56	0.43	0.48	***0.19***	***0.41***
KNN (k=14)	**0.82**	0.65	0.30	0.41	***0.18***	***0.68***
Classification Trees	**0.82**	0.69	0.33	0.45	***0.17***	***0.49***
SVM	**0.81**	0.67	0.32	0.43	***0.18***	***0.40***
XGBoost Classifier	**0.80**	0.55	0.44	0.49	***0.15***	**-**

Table from Research Paper

Model	*Accuracy*	***Error Rate***	***Area Ratio (Training)***	***Area Ratio (Validation)***
Logistic Regression	*0.80*	***0.20***	***0.41***	***0.44***
KNN	*0.82*	***0.18***	***0.68***	***0.45***
Classification Trees	*0.82*	***0.17***	***0.49***	***0.53***
Discriminant Analysis	*0.81*	***0.18***	***0.40***	***0.43***
Naive Bayes	*0.79*	***0.21***	***0.47***	***0.53***
Artificial Neural Networks	*0.81*	***0.19***	***0.55***	***0.54***

CONCEPTS:

- **Accuracy Score -** The percentage of correctly classified cases.
- **Error Rate -** The percentage of misclassified cases (1 - Accuracy)
- **Area Ratio** - Measures how well the model ranks defaulters before non-defaulters (higher is better).

KEY INSIGHTS:

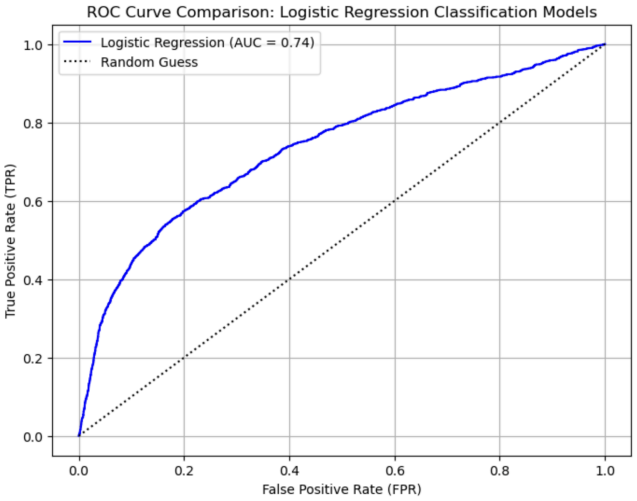
★ **KNN ranked defaulters best in training (0.68 area ratio)** but didn't generalize as well

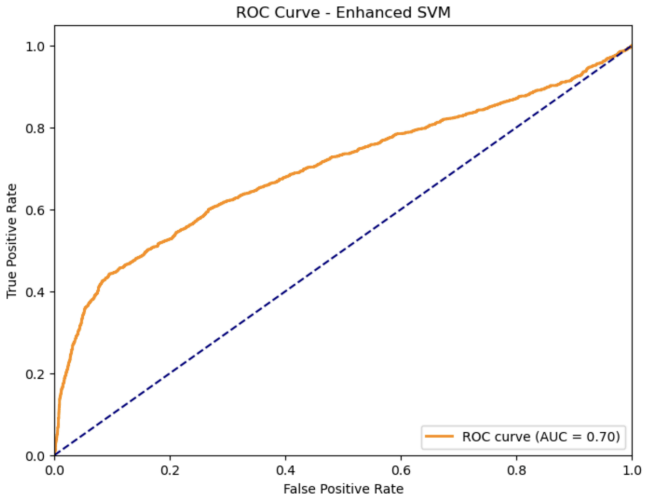
★ **ANN had the best validation (0.54 area ratio)**, meaning it worked better on new data**.**

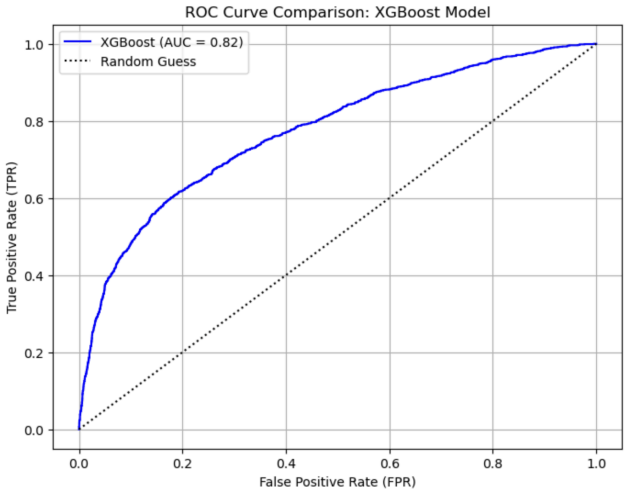
Results and Discussion

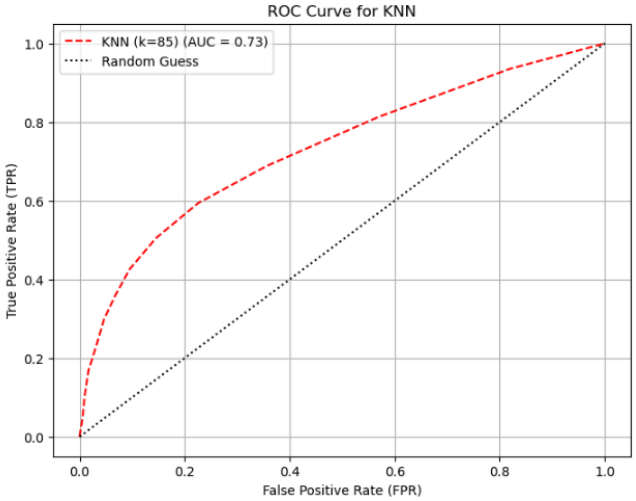
- **Logistic Regression and XGBoost demonstrated similar overall accuracy (0.80),** but XGBoost exhibited slightly higher recall for the minority class (0.44 vs. 0.43), indicating a marginal improvement in detecting default cases.
- **SVM achieved the highest accuracy (0.81),** but its recall (0.32) suggests that it failed to effectively capture the minority class.
- **KNN (k=14) recorded the lowest accuracy (0.81) but the highest recall (0.30),** indicating a tendency to favor the minority class at the expense of overall accuracy.
- **Classification Trees achieved the highest accuracy (0.82) but had a recall of only 0.33,** showing that it struggled with detecting default cases.

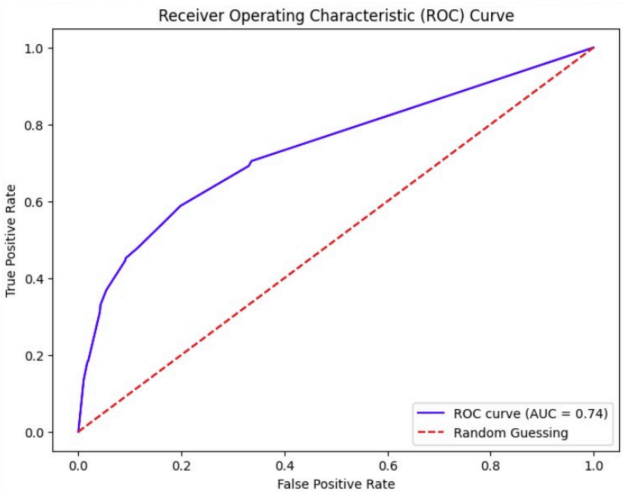
- Overall, **XGBoost provided a balanced trade-off between accuracy and recall**, making it a more suitable choice for addressing class imbalance in this dataset.











Contributions

Exploring Additional Machine Learning Models for Credit Default Prediction

In this project, our contributions centered around expanding the scope of predictive modeling by testing alternative machine learning algorithms beyond those explored in the original paper.

- New Modules: Designed and integrated additional components to enhance the preprocessing pipeline and streamline feature engineering.
- XGBoost Implementation: Applied the XGBoost algorithm to leverage its gradient boosting capabilities, improving predictive accuracy and handling imbalanced data more effectively.
- SVM & Hyperparameter Tuning: Implemented Support Vector Machines (SVM) as a comparative model and conducted hyperparameter tuning to optimize kernel functions, regularization parameters, and other key settings.

Significant improvements

Classification Tree Module Enhancement

Automated Hyperparameter Tuning & Cross-Validation
Used GridSearchCV with stratified k-fold cross-validation to automatically select optimal decision tree parameters, ensuring robust and reliable performance without manual guesswork.

Optimized SSM Parameter Tuning
Systematically fine-tuned the SSM half-window size to better estimate real default probabilities, enhancing the match between predicted and actual outcomes.

Enhanced Calibration

- **Our Classification Trees Results:**
  - Intercept: 0.0201
  - Slope: 0.9027
  - R^2: 0.8783
- **Paper's CT Results:**
  - Intercept: 0.0276
  - Slope: 1.111
  - R^2: 0.278

In contrast, our results show a calibration line much closer to the ideal y=xy=xy=x and a significantly higher R^2, indicating that our predictions are far better aligned with the actual default rates.

Challenges

One of the challenges was **handling class imbalance** while optimizing **XGBoost performance (AUC score)**. Initially, **poor minority class prediction (AUC < 0.79)** was observed due to skewed data distribution. To address this, **SMOTE was applied but led to overfitting**, reducing generalization. As an alternative, **XGBoost's scale_pos_weight was used**, improving performance without data augmentation. Additionally, **hyperparameter tuning (adjusting max_depth, learning rate, and Early Stopping)** helped prevent overfitting. Results showed that **scale_pos_weight was more effective than SMOTE**, and **AUC & F1-score were better evaluation metrics than Accuracy**. The final model achieved **AUC 0.82**, demonstrating improved minority class prediction.

Another challenge was overfitting in the **Classification Tree model**. Initially, an unpruned tree captured too much noise from the training data, leading to poor generalization on the validation set. To address this, pruning techniques were applied by limiting the tree depth (**max_depth=4**) and setting minimum sample thresholds for splits (**min_samples_split=200**) and leaves (**min_samples_leaf=50**). Additionally, a robust scaling step was incorporated to mitigate the impact of outliers. Evaluation using a lift chart and area ratio analysis confirmed that these adjustments improved the model's generalization, reducing the validation error rate while maintaining predictive performance.

Conclusion and Future Scope

This study successfully replicated and analyzed various machine learning models for predicting credit card defaults. In the paper's result ANN had the best overall predictions with a good balance between Error validation, Accuracy and calibration. During our implementation XGBoost emerged as a strong contender, handling class imbalance effectively and providing a balanced trade-off between accuracy and recall.

Areas for Improvement:

2. **Feature Engineering**: Enhancing preprocessing with techniques like Recursive Feature Elimination (RFE) and Principal Component Analysis (PCA) may optimize model performance.
3. **Ensemble Methods**: Combining multiple models, such as XGBoost with ANN or SVM, could enhance predictive accuracy..
4. **Real-World Deployment**: Developing an API for real-time credit risk assessment would enhance practical implementation.

REFERENCES

Research Source: Default of Credit Card Clients

Research Paper: The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients