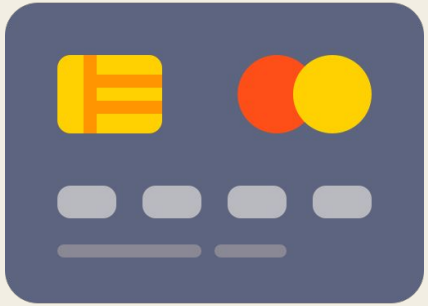


Credit Card Default Prediction

Group 2:

Theng Foo Yin | Lim Gabriel | Ng Yew Kong

Chai Si Ting | Ho Faye

Content Overview

Introduction and Problem Statement

With **globalization and concerns over Covid-19** being transmitted through contaminated surfaces, contactless payments have been made the new norm. Credit cards usage has also accelerated significantly through near field communications (NFC) cards, phone apps and wearables, providing both convenience and easy accessibility for usage.

This change in living behaviour, in turn, put card issuers like banks at risk of **credit card delinquencies**, especially during poor or uncertain economic environment.

To reduce the susceptibility of banks with a hefty write-down on outstanding balances left unpaid, **AI modeling** can provide solutions to **better predict potential credit card defaults**, and help identifying key factors leading to a default.

Objectives

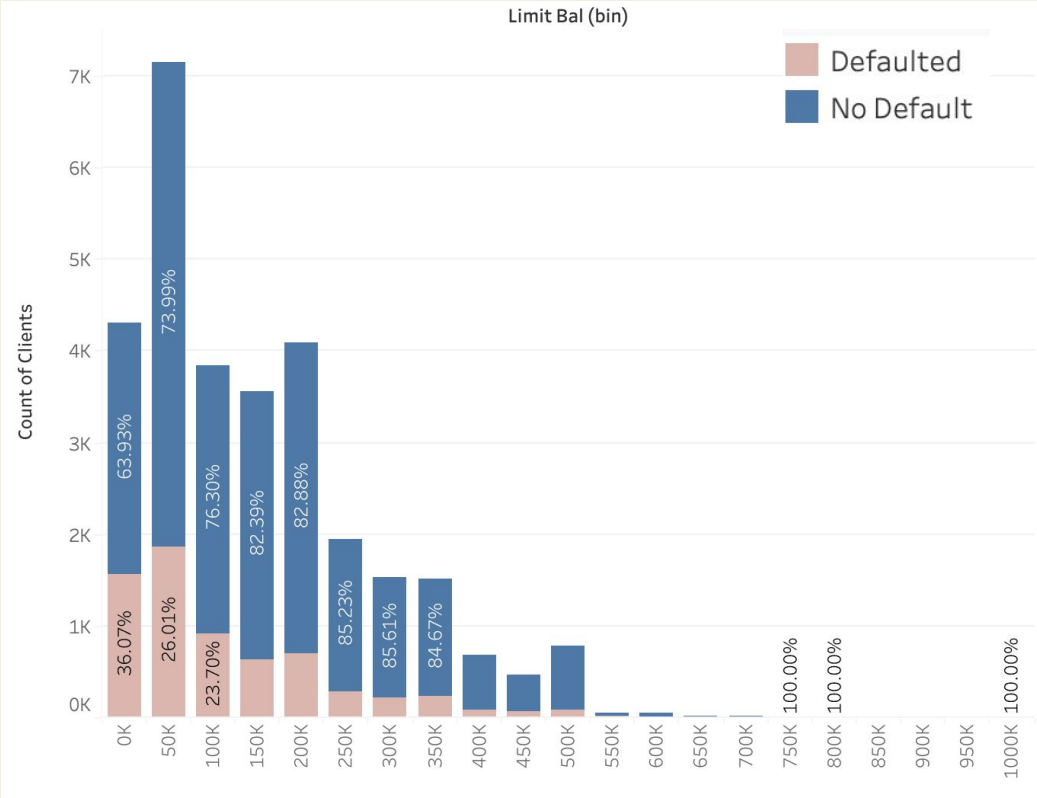
- In order to **reduce the risk** of banks' exposure in large **credit card default** incidents, utilize various **data classification techniques** with a large data set of customer records to screen for potential credit card defaulters.
- Identify common **key traits, features and conditions** through the pool of customers' dataset for early detection and trigger for preventive measures.
- To avoid experiencing customer defaults with a snow-balling effect, derive accurate **predictions and recommendations** of test results to flag for default accounts in advance.

EDA - Data Visualisation

Exploratory data analysis (EDA) - approach of analyzing data using statistical graphics and data visualization methods.

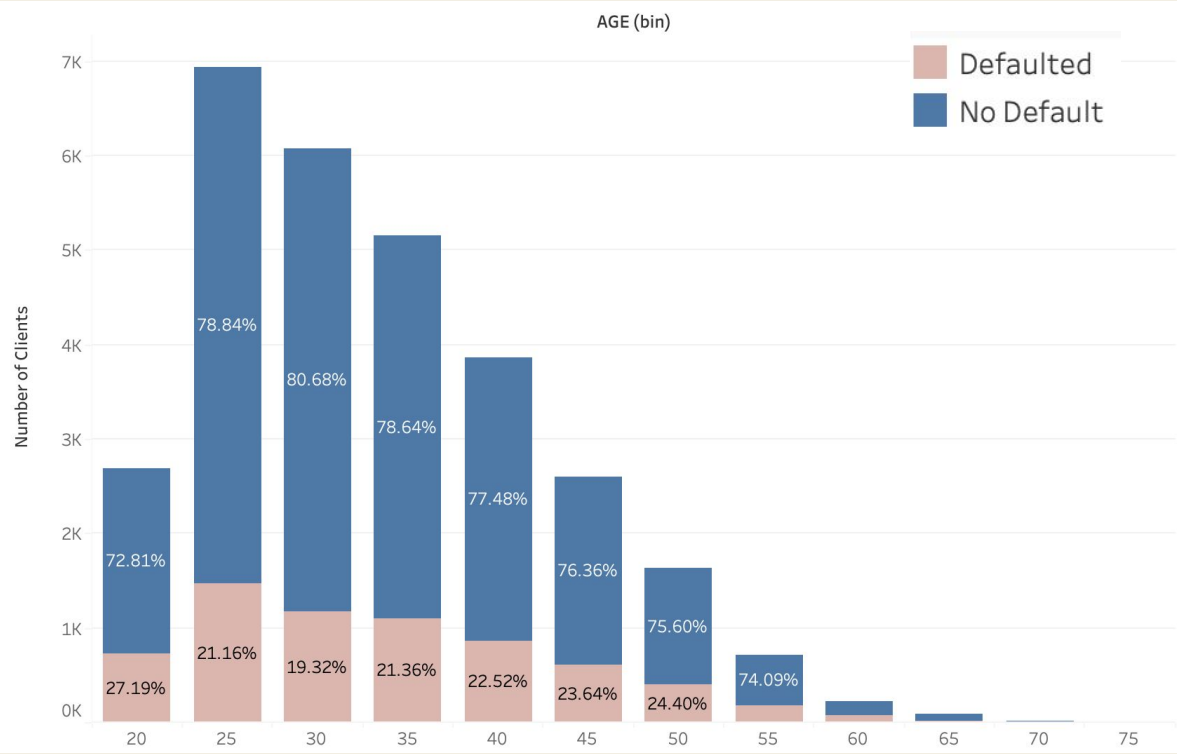
Target Variable:	**Default Rate**
Independent Variable 1:	**Credit Limit**
Independent Variable 2:	**Age**
Independent Variable 3:	**Gender**
Independent Variable 4:	**Education**
Independent Variable 5:	**Outstanding Debt**

Data Visualisation - Credit Limit



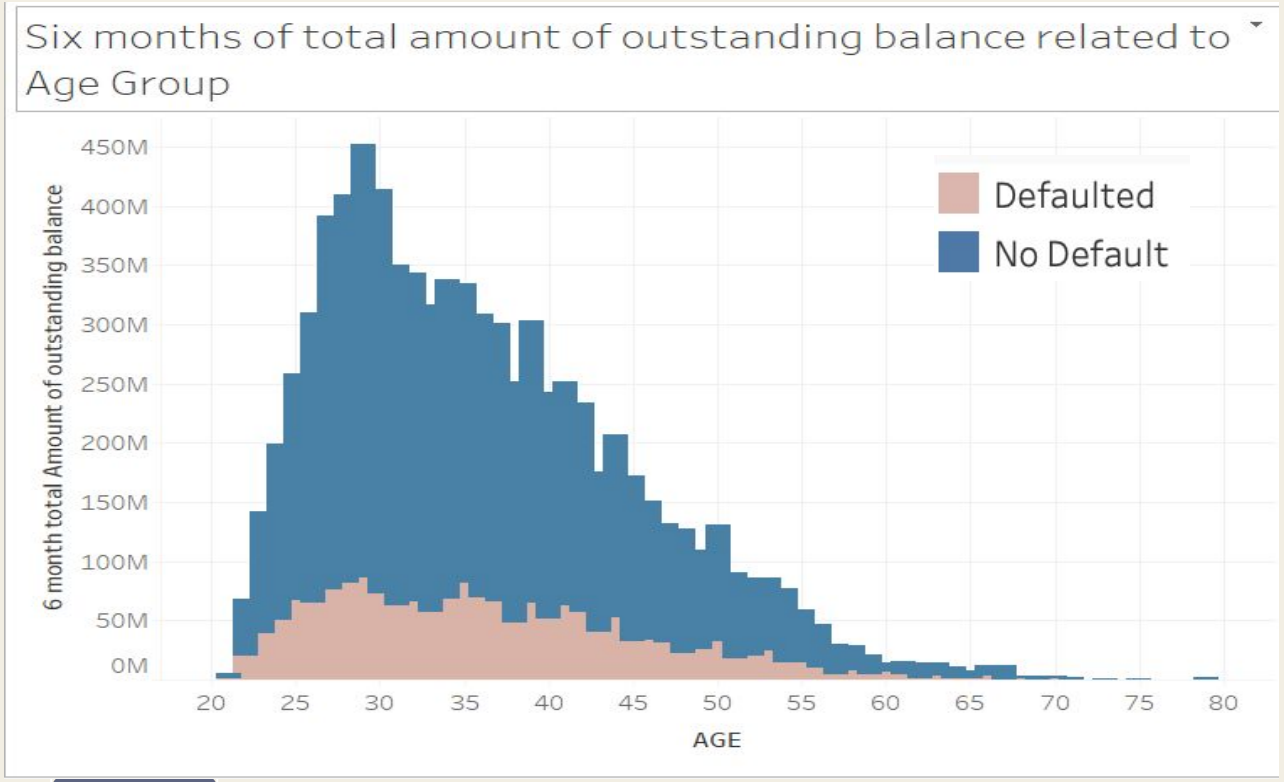
- Clients with <50K credit limit have the **highest default rate at 36.07%**
- This is **10% higher** than the next closest bucket of clients with 50K - 100K credit available

Data Visualisation - Age



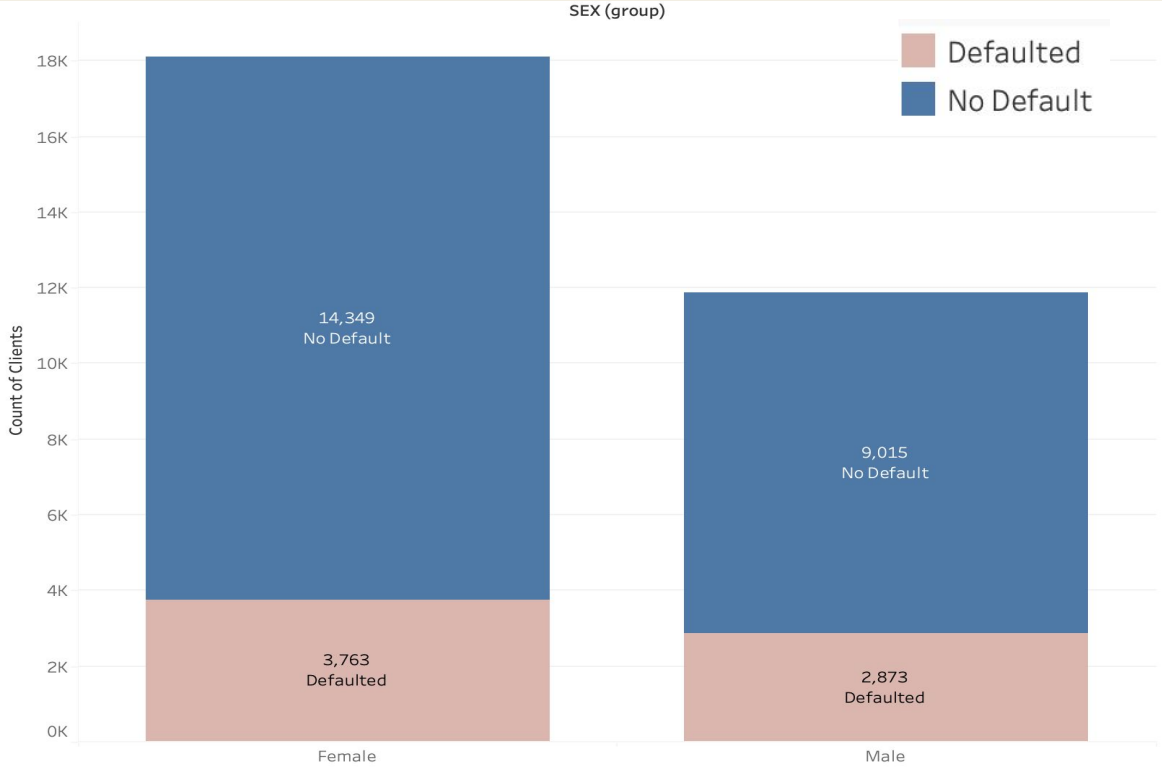
- 22.12% of customers in the dataset defaulted on their loans within the 7 month period on record.
- Age group 25-29 have the highest defaulted.

Data Visualisation - Six months Outstanding Debt



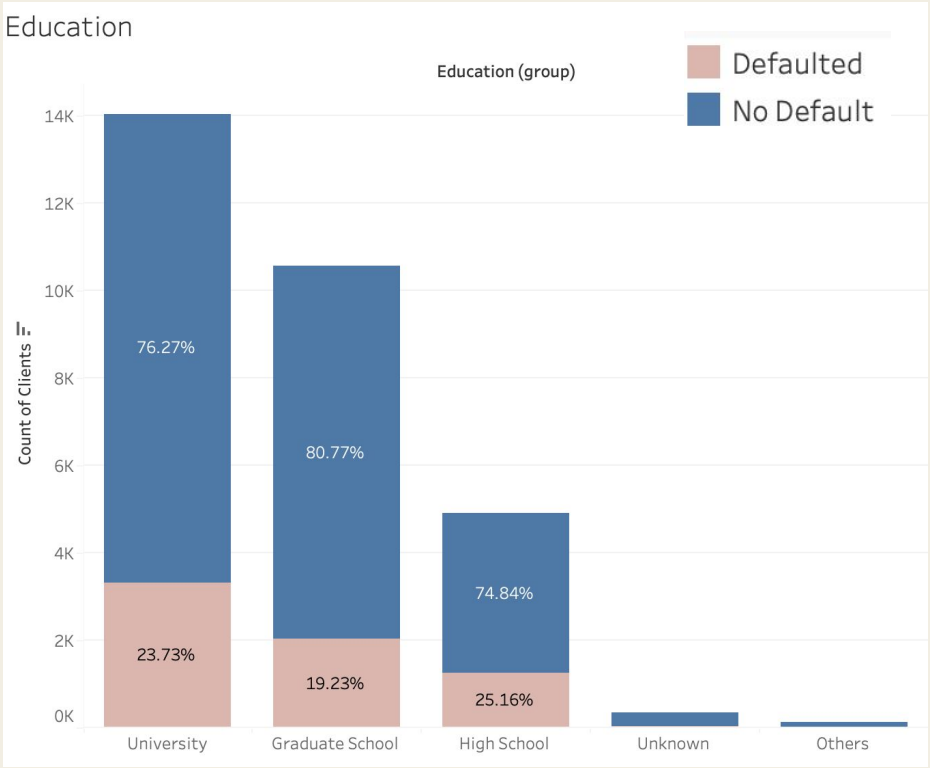
- Clients age group between 28 to 30 tend to have higher 6 months outstanding loan than the rest of the age group.
- It shown that clients spending decline after age 30.
- Credit card default amount also start to **decline** after age 30.

Data Visualisation - Gender



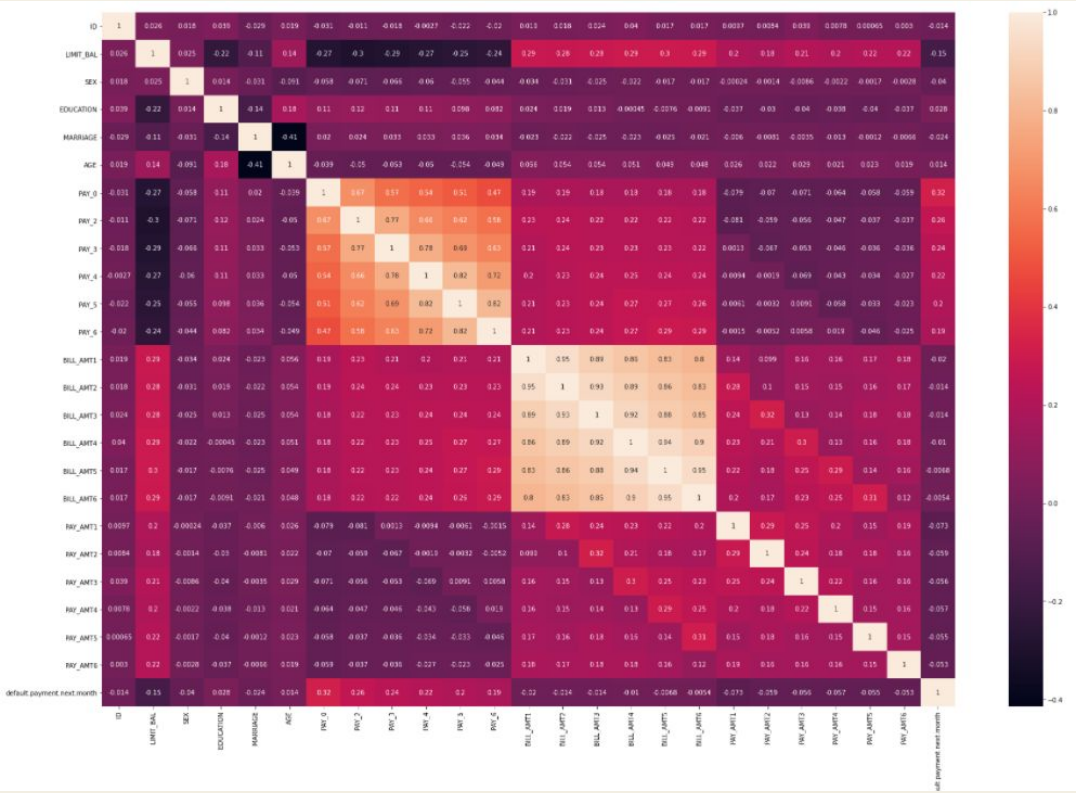
- Females make up 60.37% of the clients in this dataset, despite that, they have a 3.37% lower default rate out of the set.
- Gender is an important variable to consider when determining the overall chance of default.

Data Visualisation - Education



- Highest default percentage coming from high school background (25.16%)
- Lowest coming from graduate school (19.23%).
- Small percentage of unknown variables to be ignored for final result output.
- Education has a significant contribution to Credit card default.

EDA - Data Cleaning and Correlation Heatmap



- **Repayment Status** (PAY_0 to PAY 6) and **Outstanding Balance** (BILL_AMT 1 to BILL_AMT6) have the highest correlation within the dataset
- This shows the importance of **consistency of repayments** when attaining the likelihood of default

Feature Engineering

Remap Education

Remap EDUCATION column: **Unknown** (0, 5, 6) with **others** (4) to group undefined values together (1.56% of dataset)

```
# remap the education data to back in range with the data dictionary (0,5,6 -> 4)

def rep(x):
    if x in [0,4,5,6]:
        return 4
    else:
        return x
df['EDUCATION']=df.EDUCATION.apply(rep)

print(df['EDUCATION'].value_counts())
```

Remap Pay

Remap PAY_0 - PAY6 column: **Undefined values** (0, -2) with **pay duly** (-1) to group successful payments together

```
def remap(value):
    if value in [-2, -1, 0]:
        return -1
    else:
        return value

for col in df[['PAY_0','PAY_2','PAY_3','PAY_4','PAY_5','PAY_6']]:
    df[col]=df[col].apply(remap)
    print(df[col].value_counts())
```

Standard Scalar

Normalising and scaling values to a standard range for ease of modeling

```
# Normalisation (Standard Scaler)
from sklearn.preprocessing import StandardScaler

# initialise scalar
scaler = StandardScaler()
df[col_to_norm] = scaler.fit_transform(df[col_to_norm])
df.head()
```

Feature Engineering

Feature Selection

Identify our **target** and **independent** variables to build our models on, drop unnecessary column **ID**

```
# Features
newdf = df.drop(["ID", "default.payment.next.month"], axis = 1)
X = newdf
print(X)

# Target
Y = df.loc[:, ["default.payment.next.month"]]
print(Y)
```

Train-test Split

Setting aside **30%** of the total dataset (9,000 entries) to train our machine learning model

```
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.3, stratif
y=Y, random_state=42)
```

SMOTE

Increase sample size on training set to balance the minority class

```
# SMOTE (if we want to use to balance the minority class)
# only on train set
# increase minority class sample size to the same as majority class
# did not run on models for SMOTE dataset

smote = SMOTE(sampling_strategy='minority')

X_train_sm , Y_train_sm = smote.fit_resample(X_train, Y_train)
Y_train_sm.value_counts()
```

Weaker Classifier

Neural Network

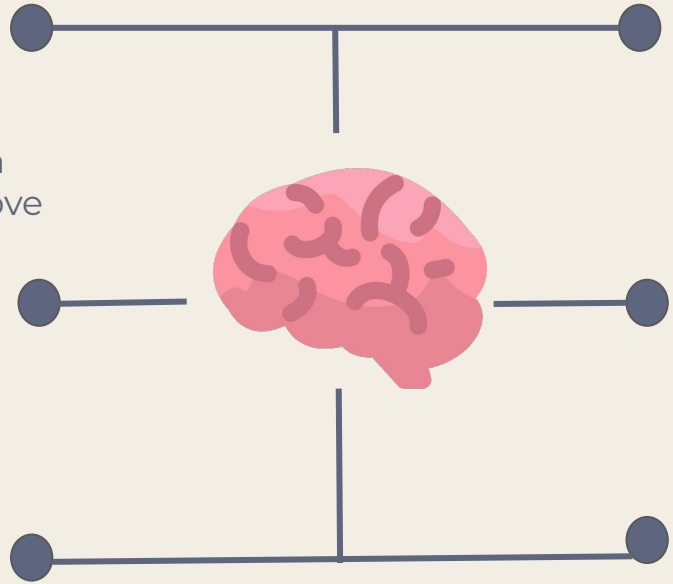
To recognise hidden patterns and correlation in raw data, much like the brain, and then continuously learn and improve

Decision Tree

Tree-like model that makes predictions based on what was answered previously

K-Nearest Neighbours

A non-parametric supervised learning method used for classification and regression.



Stronger Classifier

Logistic Regression

To model the probability of certain class existing in binary or multiclass outputs

Random Forest

Collection of multiple independent decision trees with single, aggregated result by majority voting

XGBoost

Additive (sequential ensemble) model where a weak learner improves on past existing weak learners

Model Comparison

	Logistic Regression	Decision Tree	Random Forest	XGBoost	Neural Network	K-Nearest Neighbours
F1 Score	0.51	0.46	**0.52**	0.49	0.47	0.40
Precision	0.47	0.40	0.50	**0.56**	0.39	0.35
Recall	0.57	0.53	0.54	0.43	**0.59**	0.48

- Random Forest is the best model based on F1 Score
- XGBoost is the best model based on Precision
- Neural Network is the best model based on Recall
- Overall: Random Forest is the best model considering Precision VS Recall tradeoff

**F1 Score = 2 x (Precision x Recall) / (Precision + Recall)*

Model Comparison

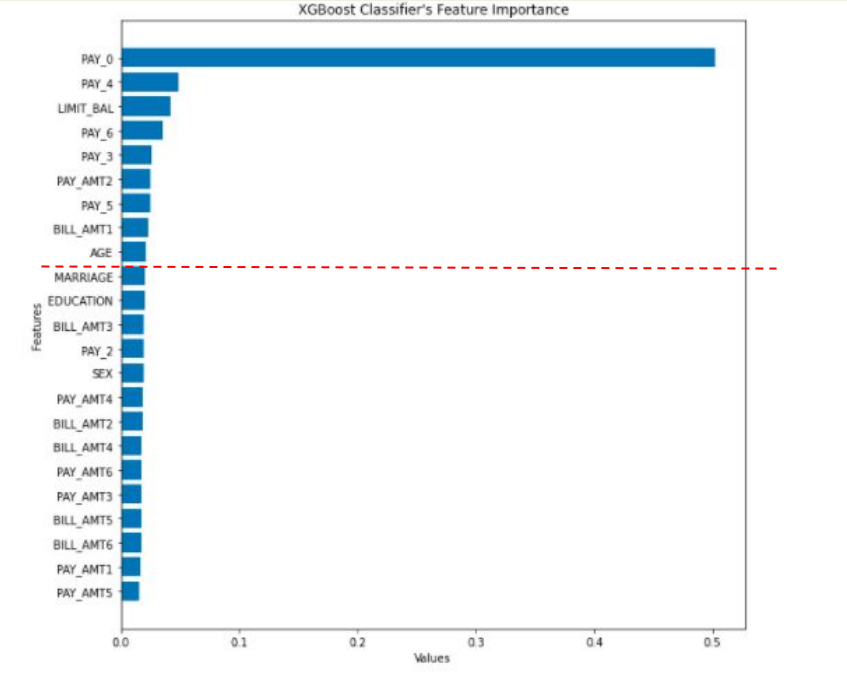
	Logistic Regression	Decision Tree	Random Forest	XGBoost	Neural Network	K-Nearest Neighbours
Accuracy (Train)	0.708	0.883	**0.893**	0.995	0.796	1.000
Accuracy (Test)	0.762	0.720	0.779	**0.799**	0.706	0.688

- Overfitting in all Tree-based models and K-Nearest Neighbours model
  - Due to max depths of tree models (best parameter: max_depth = 14)
  - Different data distribution of Train-Test datasets after SMOTE

- Logistic Regression & Neural Network model generalise better to unseen dataset
- Logistic Regression model performs better on Test than Train set
- XGBoost is the best model based on Accuracy (Test)

Feature Importance

XGBoost



Importance Feature

- **Pay_0:** Repayment Status in September
- **LIMIT_BAL:** Amount of credits given in NT dollars
- **PAY_AMT2:** Amount of previous payment in August 2005
- **BILL_AMT_1:** Amount of bill statement in Sept 2005
- **Age:** Age in years

Because Pay_0, Pay_4, Pay_6 and Pay_5 are highly correlated

Business Insights

1 Bill Payment and Repayment Status

- Clients age group between 28 to 30 tend to have higher outstanding loan than the rest of the age group

2 Credit Limit

- Most clients undertake a loan credit of 50k
- Clients with less than 50K have the highest defaulting rate

3 Age

- Majority of the clients tend to be within the late 20s and early 30s range
- Clients age 20-25 have the highest rate of defaulting

Business Solution

More Stringent Checks

- Given that clients age 20-25 have the highest rate of defaulting, **more checks** should be done to determine their ability to repay the loans
- Ascertain whether clients within this range have a **steady flow of income** to repay the loans
- **Add more variables** such as asset ownership and occupation

Timely Reminders

- **Send more notifications** to remind customers to pay

Offer incentives

- Once customers are flagged out to have a risk of default, provide **incentives/reduce repayment amount** to encourage early repayment
- Offer **debt restructuring plan**

The End