Group 1 - Project Proposal: Predicting Loan Default

Team members: Luca Matteucci, Santiago Mazzei, Srithijaa Sankepally, and Victor Floriano

Problem Statement:

Our goal is to predict what customers are more likely to default on their loan payments. By analyzing the Lending Club Loan dataset, we aim to understand the factors that contribute to loan defaults and late payments, as well as why some borrowers can't repay their loans on time, and to find out what helps borrowers succeed. We want to make the lending process better, reduce default risk, and increase profitability for lenders.

Data Source:

We will use the Lending Club Loan dataset, which includes complete loan data for loans issued from 2007 to 2015. The Lending Club is a peer-to-peer lending company that matches people looking to invest money with people looking to borrow money. The dataset provides information about borrowers, loan characteristics, and loan performance. The data is freely available on Kaggle at: https://www.kaggle.com/datasets/adarshsng/lending-club-loan-data-csv

Data Description:

The dataset contains approximately 2,260,668 observations and 145 variables (columns). The variables include information, such as borrower characteristics (e.g., credit scores, income, employment details), loan characteristics (e.g., loan amount, interest rate, purpose), and loan performance data (e.g., current loan status, delinquency history). Some of the variables we expect to be useful for our analysis include: `annual_inc`, `loan_amnt`, `int_rate`, `delinq_2yrs`, `purpose`, among others.

The dataset contains multiple data types, including float64 (105 instances), int64 (4 instances), and object (36 instances). Additionally, it includes several variables that necessitate preprocessing, such as:

1. Binary values assigned to 'Y'/'N'. (i.e. `hardship_flag`)
2. Object columns would work better as datetime. (i.e. `settlement_date`)
3. Variables with text input. (i.e. `desc`)

Loading the data and brief exploration:

In [1]:
```
#Importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
from google.colab import drive
drive.mount('/content/drive')
loan_df = pd.read_csv('/content/drive/MyDrive/BU_MSBA/BA810/Data/loan.csv')
```

```
Mounted at /content/drive
<ipython-input-2-f5231390c1e7>:3: DtypeWarning: Columns (19,47,55,112,123,124,125,128,129,130,133,139,140,141) have mixed types. Specify dtype option on import or set low_memory=False.
  loan_df = pd.read_csv('/content/drive/MyDrive/BU_MSBA/BA810/Data/loan.csv')
```

In [3]:
```
#Display the number of entries, columns, and dtypes
loan_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2260668 entries, 0 to 2260667
Columns: 145 entries, id to settlement_term
dtypes: float64(105), int64(4), object(36)
memory usage: 2.4+ GB
```

In [4]:
```
#Our target variable contains multiple values, we will
#transform this column into a binary 0/1 variable
loan_df['loan_status'].value_counts()
```

Out[4]:
```
Fully Paid                                             1041952
Current                                                 919695
Charged Off                                             261655
Late (31-120 days)                                       21897
In Grace Period                                           8952
Late (16-30 days)                                         3737
Does not meet the credit policy. Status:Fully Paid        1988
Does not meet the credit policy. Status:Charged Off        761
Default                                                     31
Name: loan_status, dtype: int64
```

In [5]:
```
#Summary Statstics of selected varibles
loan_df[['annual_inc', 'loan_amnt', 'int_rate', 'delinq_2yrs']].describe()
```

Out[5]:

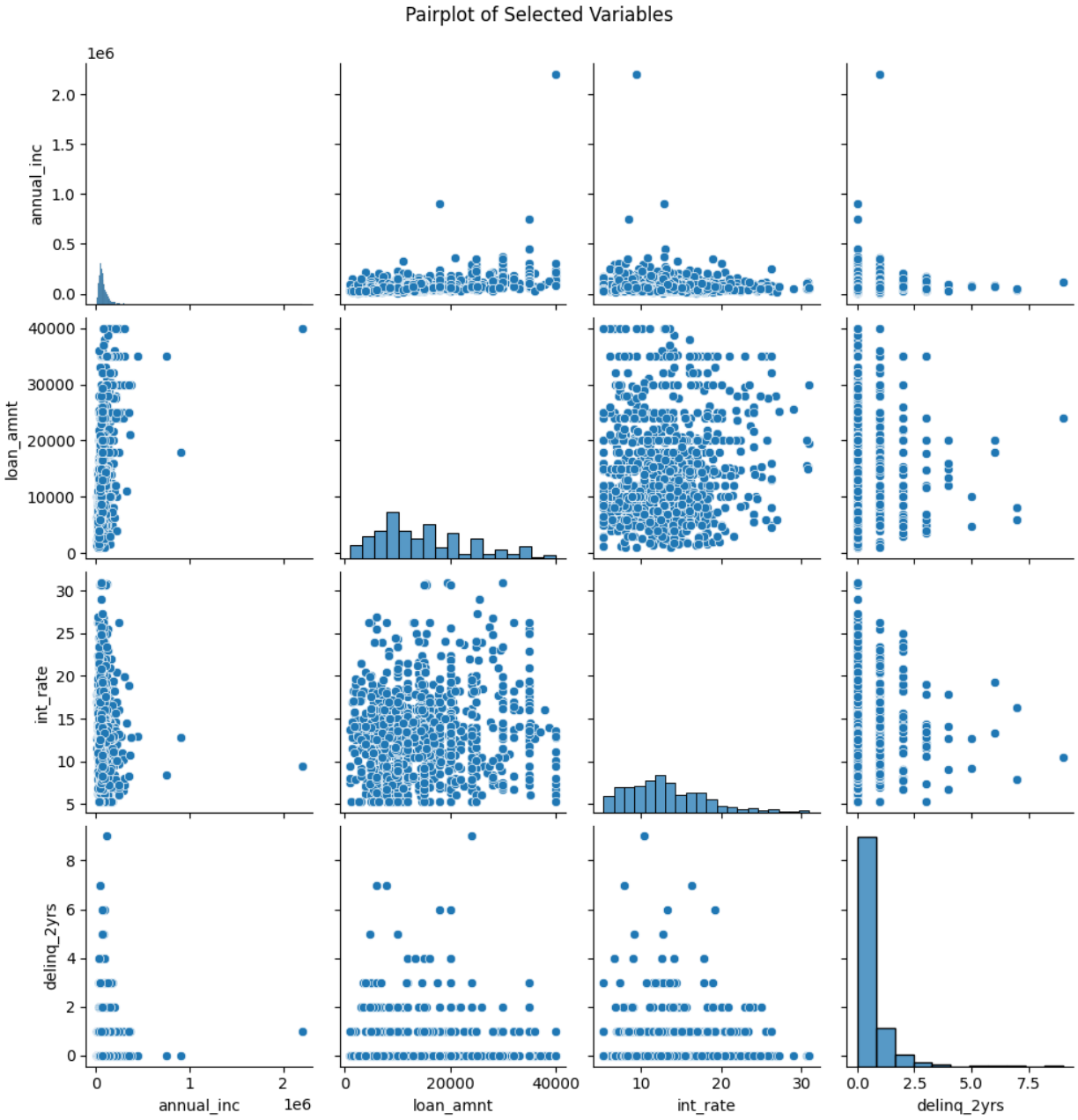
	annual_inc	loan_amnt	int_rate	delinq_2yrs
count	2.260664e+06	2.260668e+06	2.260668e+06	2.260639e+06
mean	7.799243e+04	1.504693e+04	1.309291e+01	3.068792e-01
std	1.126962e+05	9.190245e+03	4.832114e+00	8.672303e-01
min	0.000000e+00	5.000000e+02	5.310000e+00	0.000000e+00
25%	4.600000e+04	8.000000e+03	9.490000e+00	0.000000e+00
50%	6.500000e+04	1.290000e+04	1.262000e+01	0.000000e+00
75%	9.300000e+04	2.000000e+04	1.599000e+01	0.000000e+00
max	1.100000e+08	4.000000e+04	3.099000e+01	5.800000e+01

In [6]:
```
#Correlation matrix of selected key variables
loan_df[['annual_inc', 'loan_amnt', 'int_rate', 'delinq_2yrs']].corr()
```

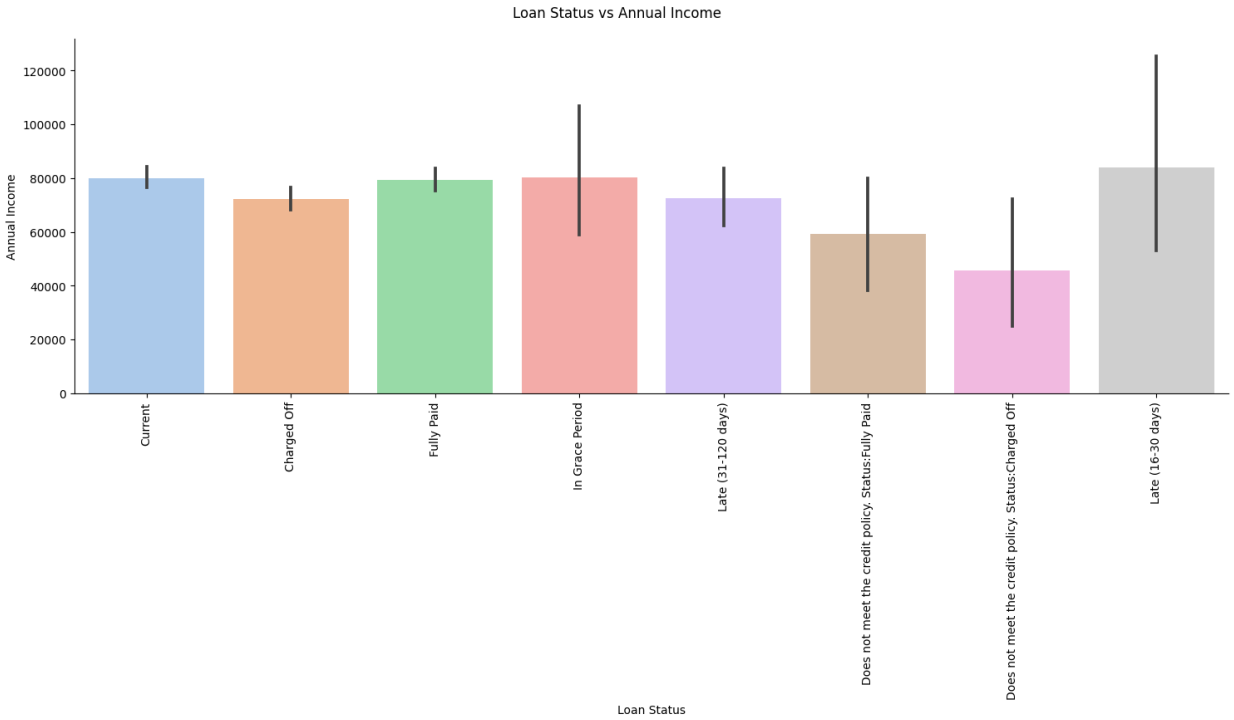
Out[6]:

	annual_inc	loan_amnt	int_rate	delinq_2yrs
annual_inc	1.000000	0.197246	-0.050585	0.026134
loan_amnt	0.197246	1.000000	0.098083	-0.009277
int_rate	-0.050585	0.098083	1.000000	0.058899
delinq_2yrs	0.026134	-0.009277	0.058899	1.000000

In [7]:
```
#For the pairplot we had to select a smaller set of the database because
#using the entire dataset was making colab run out of RAM, so here we added
#the pairplot as an example of what we intend to do. In our project we will either
#use google cloud or work wil a smaller set of the original dataset.
loan_sample = loan_df[['annual_inc', 'loan_amnt', 'int_rate',\
                       'delinq_2yrs']].sample(1000, replace=False)

pairplot = sns.pairplot(loan_sample)
pairplot.fig.suptitle('Pairplot of Selected Variables', y=1.02);
```



In [15]:
```
#Graph of the Loan Status vs Annual Income

#Create a new sample of the data
loan_sample_2 = loan_df.sample(5000, replace=False)

catplot = sns.catplot(kind = 'bar', x = 'loan_status',y = 'annual_inc', data = loan_sa
plt.suptitle('Loan Status vs Annual Income', y=1.05)
catplot.set_axis_labels('Loan Status', 'Annual Income')
catplot.set_xticklabels(rotation=90)
plt.show();
```



Anticipated Results:

Our target variable will be whether a customer was charged off/defaulted on the loan or not. Therefore, we will be dealing with a classification problem and our models will return a class 1(default) or 0(no default) based on each record's characterists. We expect to apply pre-processing steps such as data cleaning/imputation of missing values, normalization, and feature selection due to the high dimensionality of our data. For our feature selection we plan to experiment with Sequential Feature Selection and Tree-based Feature Selection (available on scikit-learn)

In order to predict which customers will default on their loans we plan to test the following models:

- Logistic Regression (sklearn-LogisticRegression) (params)
- k-Nearest Neighbors (sklearn-KNeighborsClassifier) (params)
- Decision Tree (sklearn-DecisionTreeClassifier) (params)
- Random Forest (sklearn-RandomForestClassifier) (params) Assessing the influence of loan amount and term on defaults and late payments. Exploring patterns in delinquency history and their impact on loan performance.

We will then fine-tune our models hyperparameters to achieve the best performance based on selected metrics such as accuracy, precision, recall, and f1. In order to try multiple hyperparameters we anticipate using parameter search methods such as GridSearchCV and RandomizedSearchCV. Our final step will then be to compare the models and select the best one.

Potential Implications:

With the results of this analysis, we expect to be able to identify the biggest drivers of loan default, and to generate a model with relatively accurate predictions.

1. **Risk Assessment and Mitigation:** Identifying the factors contributing to loan defaults and late payments allows lending institutions to assess the risk associated with potential borrowers more effectively. This can lead to improved risk management practices.

1. **Market Competitiveness:** Borrowers can benefit from more competitive loan terms and rates based on their credit profiles and purposes, resulting in higher customer satisfaction. Investors will also be more attracted because understanding the factors that contribute to loan defaults can help them diversify their investments and potentially earn higher returns.

2. **Compliance with Ethics and Regulations:** Ensuring fair and transparent lending practices, supported by data-driven decisions, can help lending institutions comply with financial regulations and avoid legal issues. This also contributes to the promotion of ethical lending practices by reducing the likelihood of lending discrimination and bias.