Final Project-2

Bank Loan Case Study

PLEASE REFER TO THE COLLAB FILE IN THE SAME FOLDER NAMED “Final Project 2: Bank Loan Case Study” FOR IN DEPTH DETAILS. IT WAS IMPOSSIBLE TO WORK ON EXCEL WITH SUCH A HUGE DATASET AS EXCEL HAD ITS LIMITATIONS. I HOPE GOOGLE COLLAB WAS THE ONLY WAY TO GET THROUGH THE TASK.

# Project Description:

The Final Project-2 is based on a case study of Bank Loans. It is obvious that, when a loan application is received, the business must evaluate whether to approve the loan based on the applicant's profile. The bank's may be subjected to two different kinds of risks when a loan is approved: approving loans for those who cannot pay them back, not approving loans who can repay the loan. This project comprehends the same. We need to consider the previous data that the bank has recorded for the past years, analyze the data and then come up with an approach or a way to consider the details from the application and decide if we can approve the loan.

# Tech Stack:

Software used: Microsoft Excel, Google Colaboratory

Version: 2103 (16.0.13901.20400)

Developer: Microsoft

Latest Stable Release: April 13, 2021

Purpose: To perform data analysis.

Written in: C++(back-end)

# Problem Statement and Approach:

The project’s problem is that because of the people's weak or nonexistent credit histories, loan providers find it challenging to grant loans to them. Due to this, some customers take advantage of it by defaulting. Imagine you work for a consumer finance company that specializes in providing urban customers with several kinds of loans. To analyze the patterns found in the data, employing EDA - Exploratory Data Analysis as suggested would be my approach for finding a solution. Since the data is enormous and it is impossible to work with this data on Excel, Google’s Colaboratory is being used for quicker results.

# Task/Solution:

**A: Clean the data (Handling Missing Data)**

Sol: the following techniques can be used to clean the data

* Remove Duplicates: All the duplicate values or repeated records are removed
* Data Parsing from Text to Column: Compound words which are appended using a particular delimiter can be split into separate columns using this feature.
* Delete All Formatting: A professional should have minimal data formatting and no conditional formatting to increase its durability. Hence avoid unnecessary formatting.
* Spell Check: Spell check is one other amazing built-in feature of Excel which I have utilized to correct errors in spellings.
* Change Case - Lower/Upper/Proper: Not used in our task but can be used if needed.
* Highlight Errors: The conditional formatting can be used to show specific exceptions or errors made in the data collected based on a particular condition.
* TRIM Function: Trim() built-in function can be used to remove unnecessary spaces between words.
* Find and Replace: Find and Replace
* Handle Missing Data: Here since it is a large dataset, some of the records have more than two missing values ergo chose to remove all such records to have our cleaned dataset.

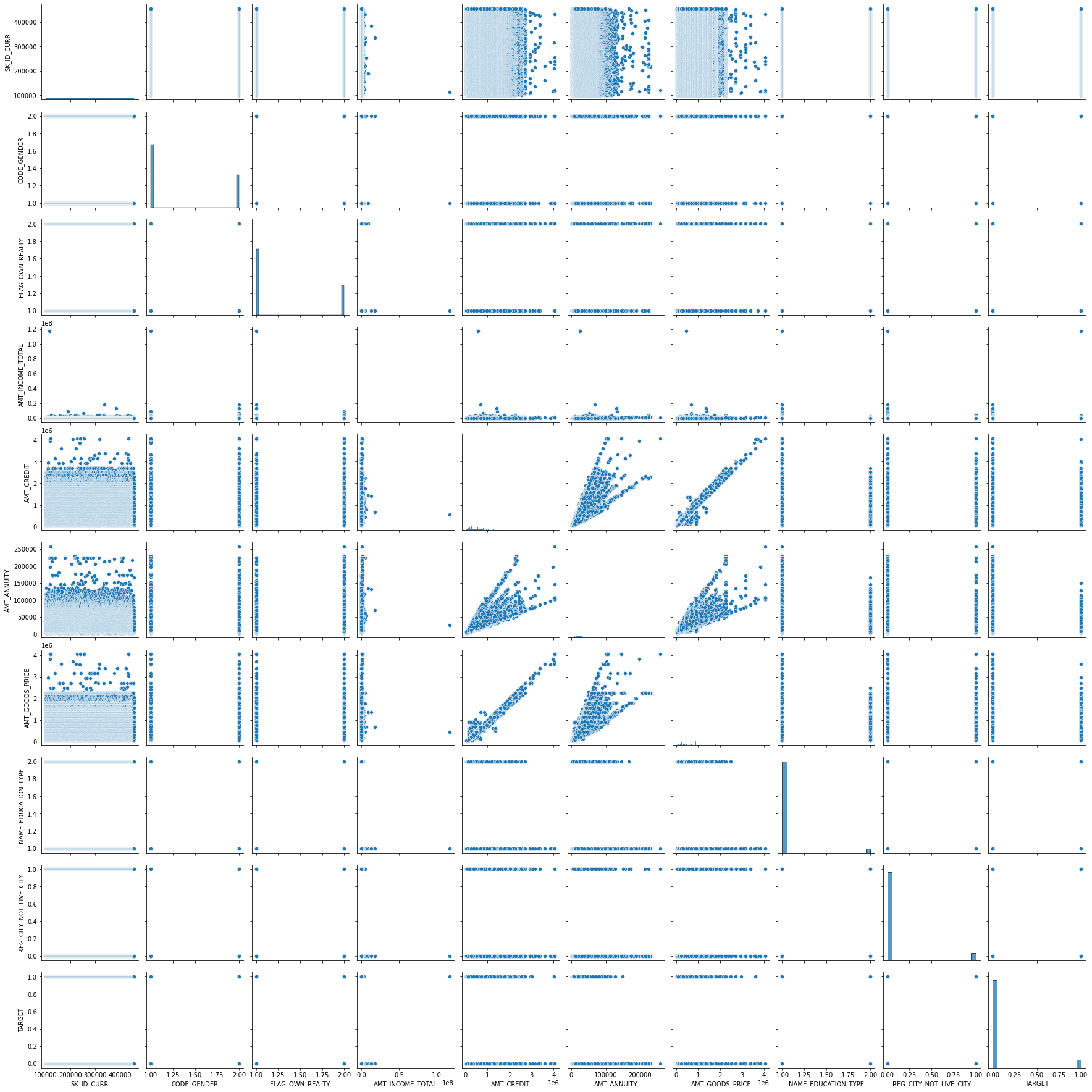
**NOTE: For easy and better understanding only essential attributes of the dataset are chosen and processed on. Please refer to the Colab file in the same folder as this.**

**B. Outliers**

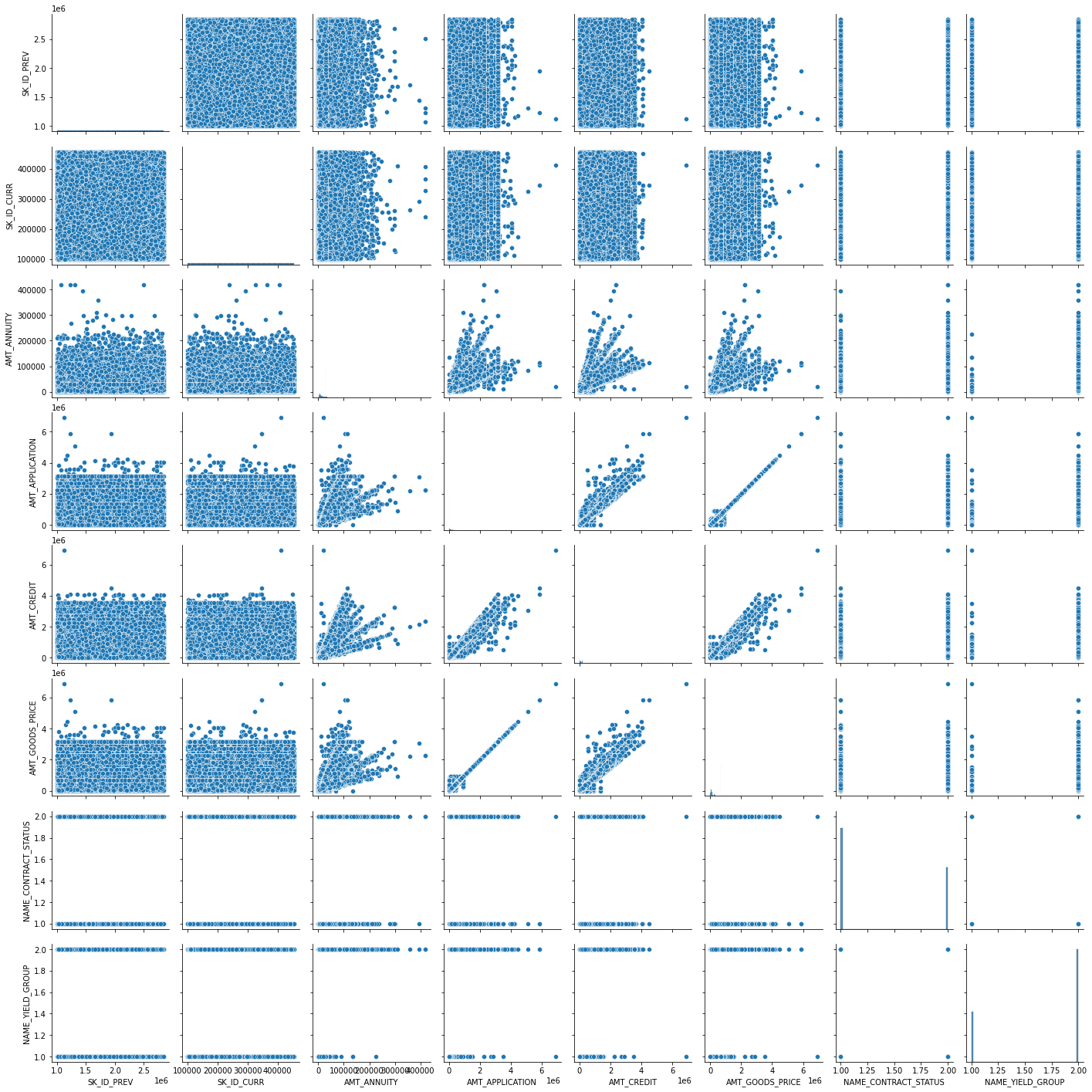
Outliers can be defined as a value in the dataset that is very different from all the other values. Outliers can be clearly spotted in these plots. (These plots are plotted using the seaborn library from python. Refer the colab file in the same folder)

import seaborn as sns

sns.pairplot(df1\_cleaned) #application\_data

****

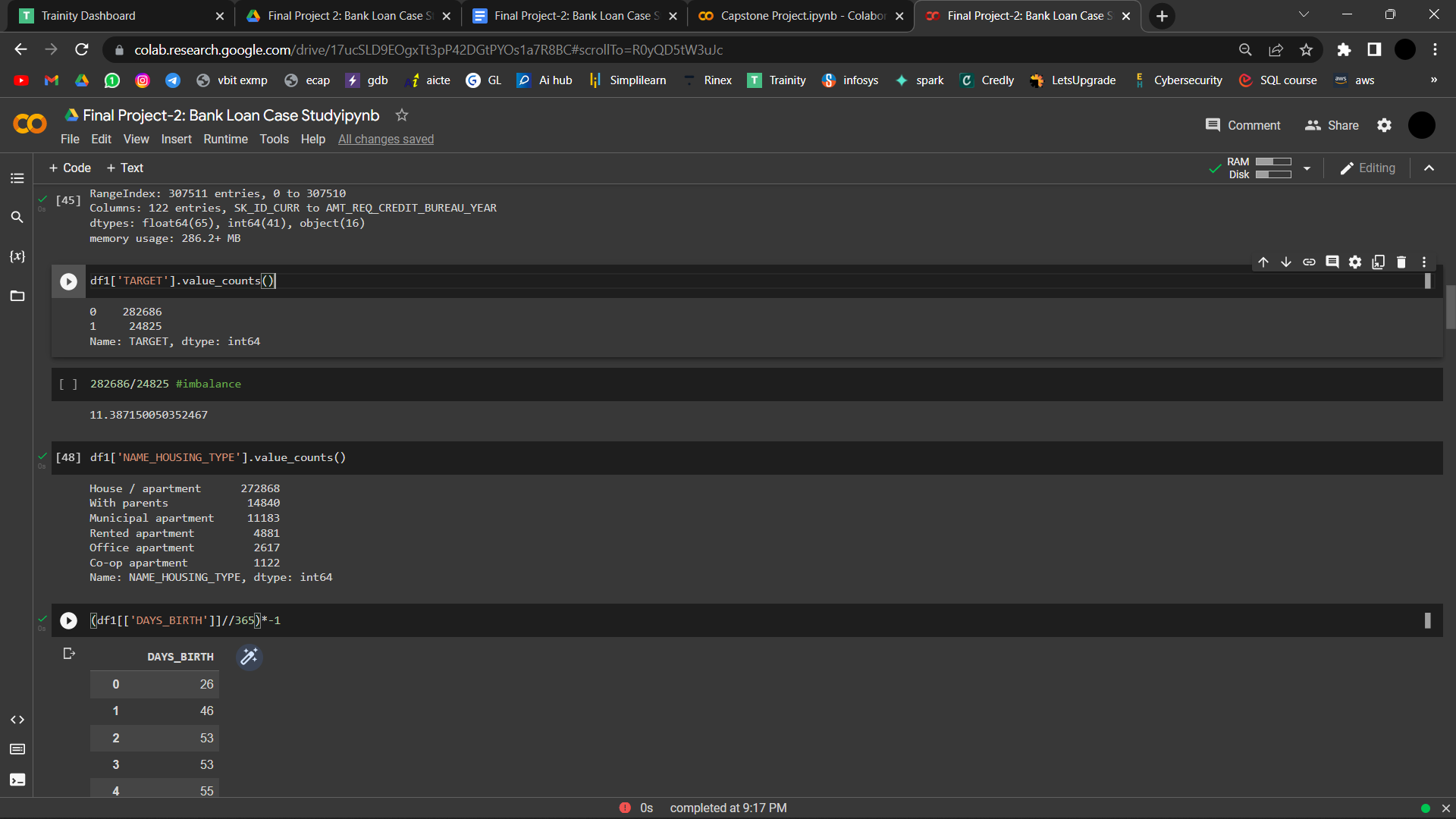
**sns.pairplot(df2\_cleaned) #previous\_application**

****

**One can clearly see that even when most of the data is grouped together there are some data points that are away from the clusters which might affect the bias and variance in the dataset.**

**C. Imbalance in data:**

A balanced dataset refers to a dataset where the 2 classes have the same amount of data i.e. 50% each whereas, on the other hand, Imbalance means that the number of data points available for different classes is different. The ratio of imbalance in the data is as follows:



Therefore, the ratio of imbalance is around 11.3.

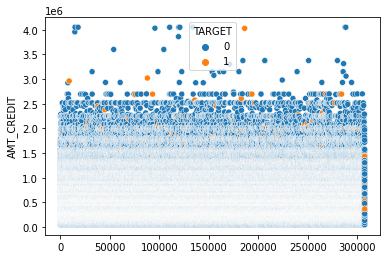
The graphical view of the same is shown below.

**D. ANALYSIS:**

**UNIVARIATE ANALYSIS: Analysis of one variable at a time.**

Univariate analysis is basically the simplest form to analyze data. Uni means one and this means that the data has only one kind of variable. The major reason for univariate analysis is to use the data to describe. The analysis will take data, summarize it, and then find some pattern in the data. There are various ways to perform univariate analysis using python one of which is as follows:

**sns.scatterplot(df1\_cleaned.index,df1\_cleaned['AMT\_CREDIT'],hue=df1\_cleaned['TARGET'])**

****

The **imbalance** in the data can easily be seen here since, 0-Blue represents people with payment difficulties and 1-orange represents all the other cases

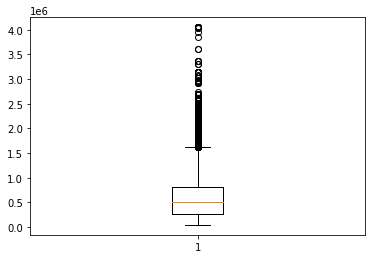
**SEGMENTED UNIVARIATE ANALYSIS:**

As far as I understood, Segmented Univariate analysis can be used to find a summary of a single data variable in the form of segments. The dataset variable is divided into subsets and patterns can be observed across the segments. One of the ways to perform segmented univariate analysis is using a boxplot since it segments the data in the quartile 1 and quartile 3 of the interquartile range and also shows other important features like median and outliers.

**import matplotlib.pyplot as plt**

**plt.boxplot(df1\_cleaned['AMT\_CREDIT'])**

**plt.show()**

****

**BIVARIATE ANALYSIS:**

Bivariate analysis is one of the statistical analyses where two variables are observed. One variable here is dependent while the other is independent. These variables are usually denoted by X and Y. So, here we analyze the changes occured between the two variables and to what extent.

**import statsmodels.api as sm**

**#dependent variable**

**y = df1\_cleaned['TARGET']**

**#define explanatory variable**

**x = df1\_cleaned['AMT\_CREDIT']**

**#add constant to predictor variables**

**x = sm.add\_constant(x)**

**#fit linear regression model**

**model = sm.OLS(y,x).fit()**

**#view model summary**

**print(model.summary())**

****

**From the above summary:**

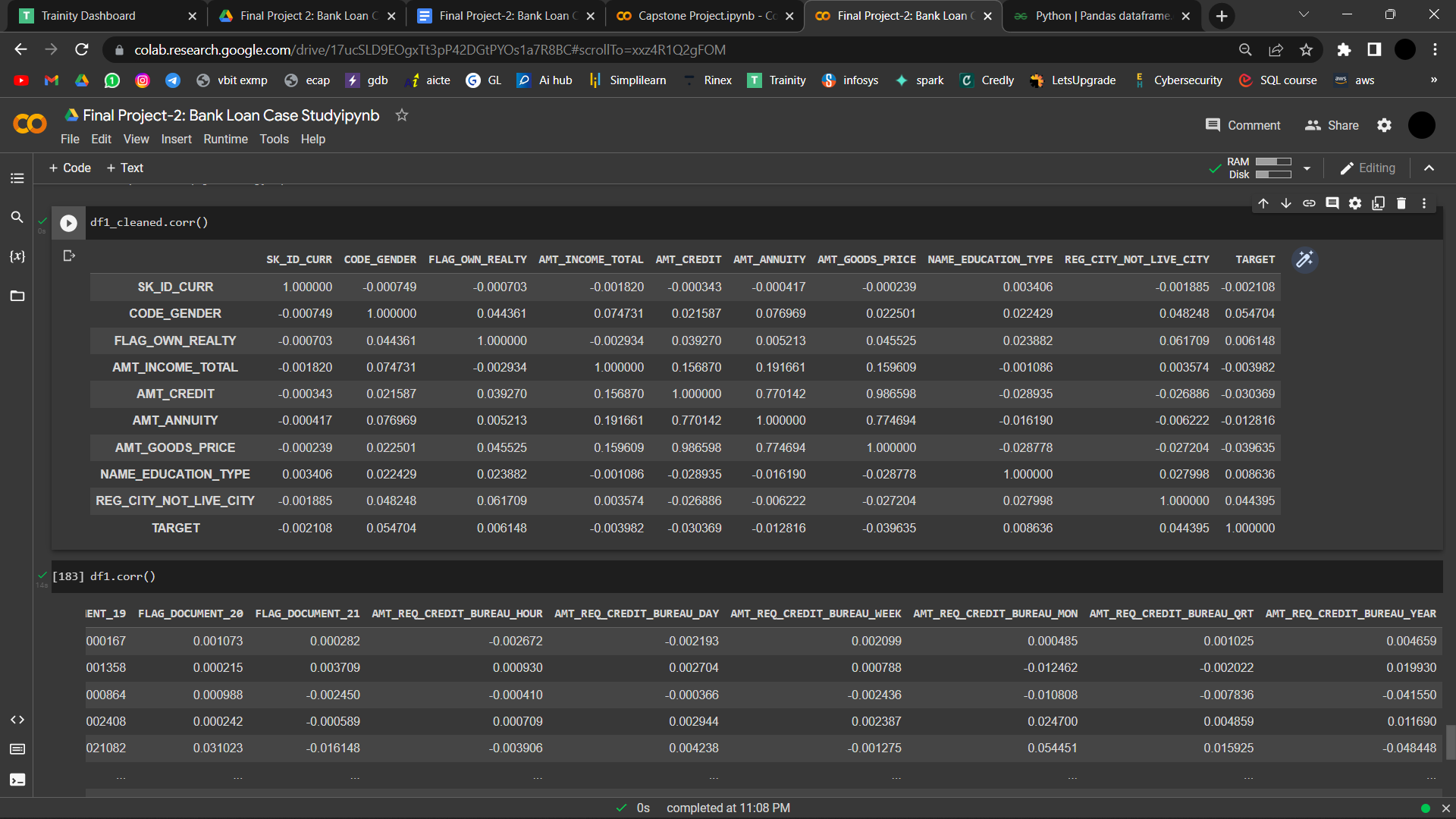
**R-squared:**

R2 is the coefficient of determination that tells us how much percentage variation of an independent variable can be explained by an independent variable. Here, 0.1% variation in Y can be explained by X.

**E. CORRELATION:**

The correlation for some of the attributes v/s the ‘TARGET’ is shown below. Please refer to the colab file for all the attributes.

**df1.corr()**

****

**F. VISUALIZATIONS:**

Many graphs are plotted using matplotlib and seaborn in the colab file. Please refer to the same.

# 

# Insights:

Insights refers to accurate understanding of something. These points helps in an insightful understanding of the problems:

* All the problems refer to real-life situations which any data analyst would face while dealing with data. The attributes may differ but the application or approach will not change.
* Here we are dealing with a Bank Loan case study where banks usually find it challenging to decide whether to approve the bank loan or not.
* There three different datasets which are given:
  + application\_data:details of the application
  + previous\_application: details of previous application
  + columns\_description: Since both the datasets together have around 160 attributes this dataset contains the description of the attributes briefly.
* Since the dataset “previous\_application” has around 1.6 million records it was quite a task to work on EXCEL or Google sheets as their limit stops at 1million.
* Google’s Colaboratory was used to solve the entire task with ease.
* Processing and working on such a huge dataset taught me alot

# 

# Result:

To recapitulate, the results are elaborately discussed above, moreover this project/task helped me in better understanding of Excel and its limitations, most of the Google Colaboratory and many python built-in functions. It also enhanced my Critical Thinking and Problem-Solving skills. (I could not solve all the questions by using joins. However I managed to draw conclusions using other concepts which are hopefully right).

Thank You.