Problem Statement:

Home Credit is an organization that serves the unbanked population with access to loans. Such individuals that do not have a built-up credit score have a challenging time securing loans from financial institutions.

Home Credit strives to broaden financial inclusion for the unbanked population by providing a positive and safe borrowing experience. In order to make sure this underserved population has a positive loan experience; Home Credit makes use of a variety of alternative data--including telco and transactional information--to predict their clients' repayment abilities

So, problem statement would be predicting how likely each applicant is of repaying a loan?

Dataset Description:

There are seven data sets that are at the disposal of Home Credit:

Application_train.csv - this is the principal table and presents all the application information. There is a single row per application, which has a unique identifier.

previous_application.csv - this file presents previous applications for people in the sample through Home Credit. There is a row per each application.

installments_payments.csv - this is the repayment history on loans given out through Home Credit for people in the sample. Each row is a made or missed payment.

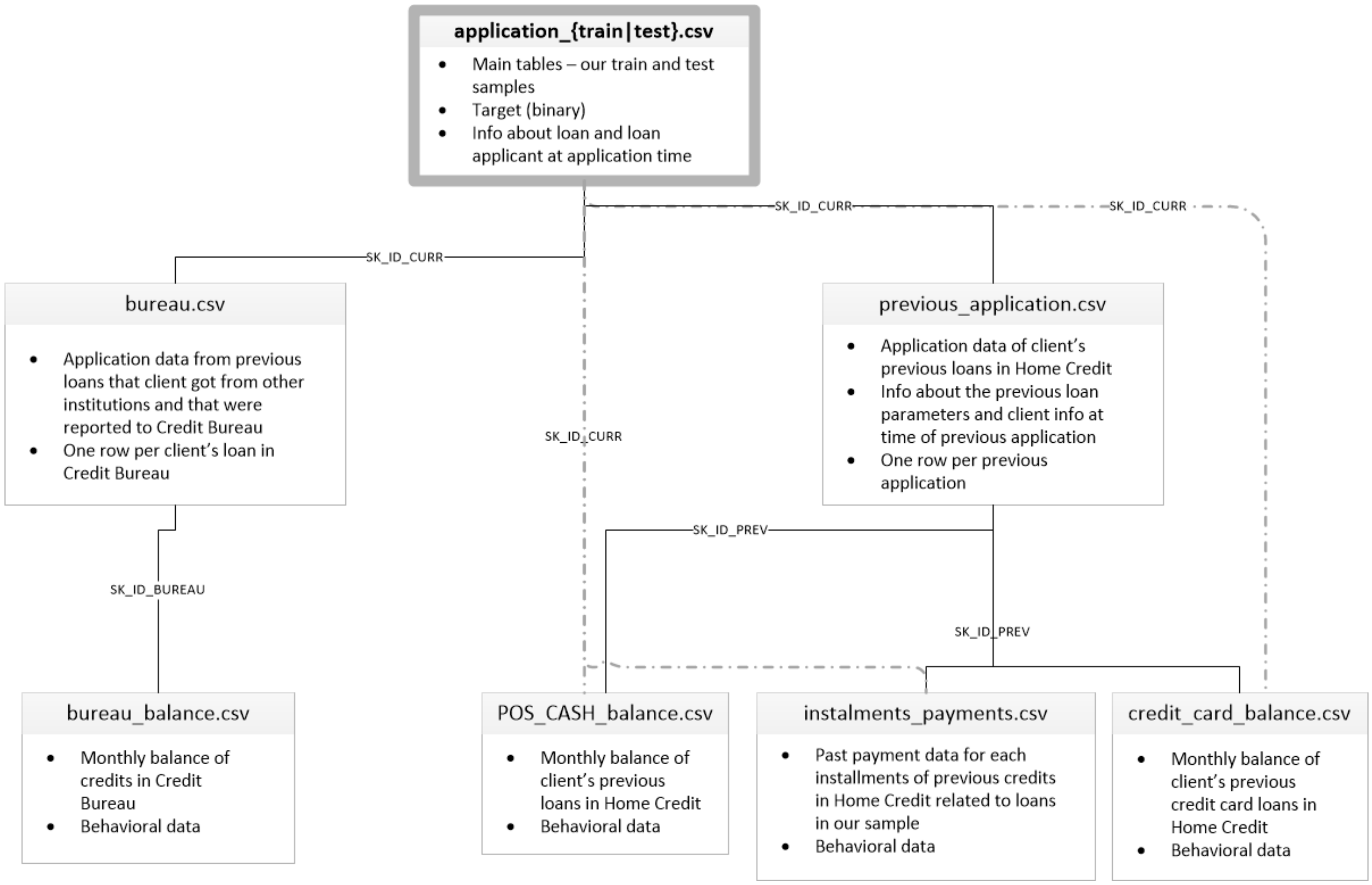
bureau.csv - credit information from other financial institutions that were reported to Home Credit. Each row represents a credit that was given to an individual in the sample.

bureau_balance.csv - monthly information per credit, per loan for users in the sample..

POS_CASH_balance.csv - like bureau_balance.csv, this data set is the internal version of the previous monthly breakdown of balances for consumer credit and cash loans that were taken out through Home Credit.

credit_card_calance.csv - each row in this data set represents a monthly balance of credit cards that were issued to applicants in the sample through Home Credit.

Links between the Data sets:



Data Wrangling and Cleaning:

A) bureau_balance.csv :

- Analyzed dataset with few rows & identified categorical column
- "STATUS " column being used to denote "**Status of Credit Bureau loan during the month (active, closed, DPD0-30,… [C means closed, X means status unknown, 0 means no DPD, 1 means maximal did during month between 1-30, 2 means DPD 31-60,… 5 means DPD 120+ or sold or written off])**" which typically denotes on which month Paid , Not paid , closed . This column dropped & instead dummy variables created for Numerical Conversion.
- Then it summing based on '**SK_ID_BUREAU**' , which will be representing '**Month_Balance_Count' .** Also, we are dropping **'MONTHS_BALANCE'** original column
- Grouped Data set is ready for Merge

B) bureau.csv

- Transforming Categorical variables to Numerical variables & Dropping actual columns
- Merging with **bureau_balance** dataset based on '**SK_ID_BUREAU'**
- Now Merged Data set is ready

C) credit_card_balance.csv

- It has Each month credit record
- Transforming Categorical variables to Numerical variables & Dropping actual columns
- Step 2: Creating new unique DF for **'SK_ID_PREV'** & **'SK_ID_CURR'**
- Dropping **''SK_ID_CURR''** from **credit_card_balance** data frame**.**
- Grouping DF based on **'SK_ID_PREV' –** Summing past Installement payment information and grouping by previous ID.
- Merging with Step 2 DF, based on **'SK_ID_PREV'**
- So it has unique **'SK_ID_PREV'** & **'SK_ID_CURR'** with all data's Merged together

D) previous_application.csv

- Transforming Categorical variables to Numerical variables & Dropping actual columns

E) POS_CASH_BALANCE.csv

- Transforming Categorical variables to Numerical variables & Dropping actual columns

F) installments_payments.csv

- Transforming Categorical variables to Numerical variables & Dropping actual columns
- Dropping **''SK_ID_CURR''** from **installments_payments** data frame**.**
- New Variable Created to know whether Payment Made on date or Late
- Grouping DF based on **'SK_ID_PREV' –** Summing past Installement payment information and grouping by previous ID.
- Merging with Step 2 DF, based on **'SK_ID_PREV'** from **credit_card_balance.csv**
- So, it has unique **'SK_ID_PREV'** & **'SK_ID_CURR'** with all data's Merged together

General Merging Strategy:

- Dropping **''SK_ID_CURR''** from **installments_payments, pos_cash_balance, cc_balance** data frame**.**
- **previous_application** & **installments_payments** Merged based on **'SK_ID_PREV'**
- Above Merged to **cc_balance** based on **'SK_ID_PREV'**
- 1-1 Mapping enabled for **'SK_ID_PREV'** & **'SK_ID_CURR'.** Then Merged with above 4 data set
- So Right side of above Image is completely Merged.
- All Previous Data is Grouped By **'SK_ID_CURR'** then dropped **'SK_ID_PREV',** to merge with training data set
- Merged Previous Data sets & bureau grouped datasets with training data set through '**SK_ID_CURR'**
- All data sets Merged with Training Data set. Merging Process completed.

Missing Variables Handling:

- There are 283 columns with missing variables out of 339 columns in the data frame.
- Above 35 % Missing variables columns dropped as it is not going to impact predictions.

- np.isfinite() Method being used to drop few rows from the data set which depends on ‘late’ & ‘closed’ Which is being referred from Missing variable % table . which shares same % of missing values.
- **‘OCCUPATION_TYPE’** is important feature though it has 31 % of missing values, so NAN marked as Unemployment
- Rest of below 10 % missing variables being filled with wither 0 or Mean, based on feature reference from excel.

After Handling Missing variables, data set stored to CSV as a single Merged data set .

Outlier Handling:

- Have not gone through complete data set columns, so looked at few important columns
- DOB has not any outliers
- Days Employed has outlier, it’s been handling after divided by -365 then > 0’s will be marked & masked as 0.
- Could see few outliers in ‘**AMT_INCOME_TOTAL’** it’ been handled by 3 standard deviations of the mean.

Application_train.csv – Preparation:

- Identified categorial variables
- Transforming Categorical variables to Numerical variables & Dropping actual columns.
- Replaced all negative values to 0
- X_train & y_train identified for feature selection
- Training set and validation set are split in following percentages: 66.66% : 33.33%.
- Top 10 features identified based on ‘**mutual_info_classif’**