Phase 2: Exploratory Data Analysis

Home Credit Default Risk

Group 10

Table of Contents

- Authors

1. Phase Leader Table
2. Credit Assignment Plan
3. Abstract
4. Introduction
5. Data Description
6. Exploratory Data Analysis
7. Visual EDA
8. Baseline Machine Learning Pipelines
9. Results and Discussion
10. Conclusion
11. Bibliography
12. Attachment 1 - Data Dictionary

Authors

Mark Green	Pragat Wagle
margree@iu.edu	pwagle@iupui.edu

Sahil Dhingra

sahdhin@iu.edu

Kunal Singh

singhku@iu.edu

Phase Leader Table

Phase	Leader	Phase Description	Completed
1	Mark	**Project Proposal**: Project Plan, Data description, algorithms and metrics, baseline models, pipelines	*11/15/22*
2	Kunal	**Baseline Report**: EDA, baseline pipeline, feature engineering, video presentation	*11/29/22*
3	Pragat	**Finetuning Model**: Hyperparameter tuning, feature selection, ensemble methods, video presentation	TBD
4	Sahil	**Final Report**: Implement neural network, advanced models and loss functions, video presentation	TBD

Credit Assignment Table

Phase	Person	Hours	Effort
2	Mark		
2	Sahil		
2	Pragat		
2	Kunal		
1	Mark	16	Mark's goals were to accomplish exploration and write about the data description, combine project elements developed by team members into a singular document, organize phase leader table, edit document appearance and content, guide discussion, goals, and organize team and resources as Phase 1 Leader, and to collaborate with team members where needed by November 15, 2022. Mark accomplished these goals by stitching together elements into a jupyter notebook and formatting the markdown into a report style, setting up a project Github resource and downloading/democratizing datasource, exploring and describing the data structure, editing the jupyter notebook submittal, being an active leader on team discussions, consulting other team members on writing in the Project Execution Plan, ML Algorithms, and ML Pipelines sections. Accomplishing these goals is key to project delivery in Phase 1 because these tasks complete the logistical elements of report development. They also contribute to the group understanding of the data structure and context. This work also sets precedents and tools for delivering future reports.
1	Sahil	14	Sahil’s goal for this phase was to do EDA, feature engineering, create pipelines, create Block Diagram, conclude results and publish results for the same in Kaggle by 15-NOV-2022. To achieve this, Sahil went through the initial notebook and worked from there to do EDA, feature selection and create pipeline with different attribute list using the knowledge from prior assignments. Accomplishing this goal will help us get us to baseline test score and AUC score which will give us a good idea on our predictions and what more we can do to make better predictions.
1	Pragat	12	Pragat’s goal was to describes the algorithms to be used and work on the getting it done prior to the November 15th deadline. Pragat will accomplish this goal by looking at the implementations on sklearn and the prioritize the most important parameters for each algorithm by looking at existing implementations seen in the documentation. Also by looking at existing implementations he will determine the loss functions and metrics and analysis to be used to measure the accuracy and overall quality of the algorithm. This goal will help to prioritize the most important arguments to consider for hyper-parameter tuning, optimizing the model, and in measuring the overall quality of results.
1	Kunal	10	Kunal's was to develop the Abstract and Project Execution plan. He accomplished this by collaborating with team members on regular team calls. This is important for the project as it helps to outline a plan of expectations and summarizes work done this week.

Abstract

The Home Credit Group needs a machine learning-based classification model to make accurate lending decisions for individuals by predicting loan default risk using a scope of data beyond just traditional credit history, including loan applicant data encompassing their demographics, social status, employment status, and previous credit history. The seven data tables comprising these information use custom EDA and ETL functions to numerically and visually explore characteristics such as input variable distributions and skewness, inter-variable and target-variable correlation, and multicollinearity. Insights from the data exploration inform the process and decisions necessary to responsibly and accurately denormalize these tables into an ML-model ready input variable. This input variable - now comprised of the entire 2.7 gigabyte dataset - is run through a baseline logistic regression model in `SKLearn` using 1277 input features. Results from modelling on test data indicate the *Wagle-Dhingra-Singh-Green* baseline model achieves AUC-ROC score of approximately 0.7709, with prediction accuracy of approximately 92%.

Introduction

Loans have always been an important part of people's lives. Each individual has different reasons for borrowing a loan. It could be to buy a dream car or a home, to set up a business, or to buy some products. A large part of the population finds it difficult to get their home loans approved due to insufficient or absent credit history. It is a major challenge for banks and other finance lending agencies to decide for which candidates to approve housing loans.

The Home Credit organization's mission is to provide responsible lending solutions to people with little to no credit history. Home Credit wants to determine their customer's eligibility for their financial products by determining their risk of default - the risk they will miss payments. **Group 10** has been tasked to evaluate their data and develop a machine learning ("ML") classification tool to predict loan default risk which uses more features than just the traditional credit history. Once developed, the *Wagle-Dhingra-Singh-Green model* will eventaully be used as a tool to evaluate future loan candidates.

This report discusses the integration of the full normalized dataset into a machine learning model. The Exploratory Data Analysis (*EDA*) performed throughout this process is also presented as a means to familiarize the data and to make informed decisions throughout the data engineering and model integration process.

A note on the styling of this report. After Section 5, many sections will be broken up by inline code blocks as the aim of this report is to highlight the exploratory data analysis and the work we are performing to develop t

Phase 2 - Task Description

The focus of Phase 2 is Exploratory Data Analysis and integration of the full dataset into a baseline machine learning model. A generalized machine learning project pipeline is outlined in figure 3 below and each step is described in detail in the Phase 1 Project report. The steps relevant to this phase (2, 3, 4) are outlined in detail below.

1. **Data Logging and ETL**
    - The data is segregated in multiple csv files related to each other by primary and foreign keys. Extract-Transform-Load operations must be performed on the dataset to consolidate the datasources into single denormalized table for efficient analysis.
2. **Exploratory Data Analysis**
    - While integrating the data, we explore the data with the goal of answering questions like:
        - How many features are present?
        - How are they correlated to one another and the target?
        - How is the data quality? Are there any problems such as high leverage points or multicollinearity?

multicollinearity?
    - What are the data types, missing values, or non-numeric features?
    - Are there any patterns between the predictor and response features?
  - The denormalized data is explored to find features which have the most impact on predicting the default risk of a loan applicant. Additional feature characteristics such as correlation between features and skewness are also observed and noted at this stage.
3. **Feature Engineering**
  - Using unnecessary features to train a machine learning model reduces the overall model accuracy and increases the model complexity and bias. To counter these issues, Feature Selection is implemented to reduce the number of input variables for the model by using only features with a relevant effect on predicting the target variable. Some feature selection methods include finding correlation coefficients, forward selection, backward selections, P-value tests, regularization, gradient boosting, and principal components analysis.
  - The features also need to be re-engineered for optimal learning. This is accomplished by restructuring the input data such that the gradient surface of the model's objective function is sufficiently convex and of a form sympathetic to gradient descent optimization. Such feature engineering tasks include transforming categorical features into a numerical space, transforming skewed features into a normalized space, or extracting relevant information "hidden" within text strings.

Figure 3: Visualization of key project steps to successfully build a machine learning model application.

Challenges

Challenges encountered during this phase include the following:

1. Developing a method to consistently and appropriately aggregate and summarize normalized data into a denormalized input variable. The challenge here was to do this in such a way as to not artificially dilute the data, to avoid choosing "winners" and "losers" based on arbitrary decisions, and to be systematic and reasoned throughout the process.
2. Running the model was challenging given the full raw dataset is approximately 2.7 gb of data. The process of denormalizing this dataset results in a high-dimensional dataset which can be cumbersome for complex models to work on. Local machines were having difficulty running the pipeline on the unrefined denormalized dataset.

Data Description

Although the Data Description is also discussed in the Phase 1 Proposal Report, a review of the underlying data structure will aid understanding the process by which the data is denormalized and prepared for the ML pipeline.

The data provided in the loan applications give insights into an applicant's current demographic and social status. Data from past credit accounts also give insight into an applicant's past financial behaviors. Notably, the primary key for each sample (loan application) is the Loan ID `SK_ID_CURR` . This relates the current loan application to associated loans from Home Credit's past account records, and to associated loans from other institution's that were reported to the Credit Bureaus. The data are stored in 7 different table schemas shown on Figure 2 below. The specific attributes in each table, along with their descriptions, are stored in an addendum table named `HomeCredit_columns_description.csv` .

Figure 2: Entity Relation Diagram for Home Credit Applicant Datatables

Home Credit provides three main sources of data upon which to learn the ML model: applicant information, credit bureau information, and Home Credit's account records. The sections below describe the data from these various sources, and some of the key variables.

Background on the dataset

> Home Credit is a non-banking financial institution, founded in 1997 in the Czech Republic.
>
> The company operates in 14 countries (including United States, Russia, Kazahstan, Belarus, China, India) and focuses on lending primarily to people with little or no credit history which will either not obtain loans or became victims of untrustworthy lenders.
>
> Home Credit group has over 29 million customers, total assests of 21 billions Euro, over 160 millions loans, with the majority in Asia and and almost half of them in China (as of 19-05-2018).
>
> While Home Credit is currently using various statistical and machine learning methods to make these predictions, they're challenging Kagglers to help them unlock the full potential of their data. Doing so will ensure that clients capable of repayment are not rejected and that loans are given with a principal, maturity, and repayment calendar that will empower their clients to be successful.
>
> *-- From original Jupyter Notebook*

Data files overview

> There are 7 different sources of data:
>
> - **application_train/application_test:** the main training and testing data with information about each loan application at Home Credit. Every loan has its own row and is identified by the feature SK_ID_CURR. The training application data comes with the TARGET indicating **0: the loan was repaid** or **1: the loan was not repaid**. The target variable defines if the client had payment difficulties meaning he/she had late payment more than X days on at least one of the first Y installments of the loan. Such case is marked as 1 while other all other cases as 0.
> - **bureau:** data concerning client's previous credits from other financial institutions. Each previous credit has its own row in bureau, but one loan in the application data can have multiple previous credits.
> - **bureau_balance:** monthly data about the previous credits in bureau. Each row is one month of a previous credit, and a single previous credit can have multiple rows, one for each month of the credit length.
> - **previous_application:** previous applications for loans at Home Credit of clients who have loans in the application data. Each current loan in the application data can have multiple previous loans. Each previous application has one row and is identified by the feature SK_ID_PREV.
> - **POS_CASH_BALANCE:** monthly data about previous point of sale or cash loans clients have had with Home Credit. Each row is one month of a previous point of sale or cash loan, and a single previous loan can have many rows.
> - **credit_card_balance:** monthly data about previous credit cards clients have had with Home Credit. Each row is one month of a credit card balance, and a single credit card can have many rows.
> - **installments_payment:** payment history for previous loans at Home Credit. There is one row for every made payment and one row for every missed payment.
>
> *-- From original Jupyter Notebook*

Loan Application Information

This is a denormalized pair of tables `application_test|train.csv` with attribute information about the current loan. These attributes describe both the financial instrument itself (the contract type, rate, size of loan, annuity, etc.) and also the demographic information for the applicant (age, gender, income, education, etc.). In addition to these basic demographic data on the applicant, a suite of other very detailed information about the applicant are provided including details on the building/property in which they live, differences in work and home addresses, car ownership, if the applicant's friends recently defaulted on

any loans, which documents were turned in, and many more. This table also contains the target variable `TARGET`. This is a binary variable indicating whether or not the client had payment difficulties on the loan.

These data are explored and a baseline model is created in the supplementary Baseline Jupyter Notebook in Section 10.

Credit Bureau Information

Some applicants have had loans through other financial institutions which were reported to the Credit Bureaus. Data from these loan applications are available from `bureau.csv` and are related to the current loan application IDs. A monthly time series describing the payment status through loan term is available for each of these related outside institution loans in `bureau_balance.csv`, related to `bureau.csv` by `SK_ID_BUREAU`. The `STATUS` attribute in this table categorically indicates whether a loan has past-due payments on a given statement. These data can be used to evaluate an applicant's past payment behaviors.

Home Credit Previous Records

Some applicants have had loans or other financial products through Home Credit in the past. These related financial product applications are made available under `previous_application.csv`. The `SK_ID_PREV` relates these applications to payment and balance information depending on the type of financial product (credit card balances `credit_card_balance.csv`, installment payments `installments_payments.csv`, and loan balances `POS_CASH_balance.csv`). These data can be used to evaluate an applicant's past payment behaviors.

The credit card balances table shows detailed statement accounts of an applicants credit card withdrawals, credit limit, and days past due. Similarly, the installment payments and loan balances tables detail the amount prescribed and paid for various loan installments as well as the days past due for any late payments.

Data Analysis Environment Setup

This section is composed of code that sets up the data for exploration, denormalization, and integration with an ML pipeline. It assumes the `*.csv` files from the Kaggle dataset are stored in a folder at the following directory relative to the path of this notebook:

```
DATA_DIR = "../Data/"
```

Import Required Packages

In [1]:

```
# import packages

import os
import time
import warnings
import zipfile
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

from sklearn.base import BaseEstimator, TransformerMixin
from sklearn.pipeline import Pipeline, FeatureUnion, make_pipeline
from sklearn.compose import ColumnTransformer
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import MinMaxScaler
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import OneHotEncoder
from sklearn.feature_selection import SelectKBest
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.model_selection import GridSearchCV
from sklearn.linear_model import LogisticRegression
from pandas.plotting import scatter_matrix
import missingno as msno
#!pip install msno

warnings.filterwarnings('ignore')
```

In [2]:

```
!pwd
```

```
/root/shared/Nextcloud/GradSchool/Coursework/CSCI_556/AML556FinalProjectFall
2022/Phase2
```

Loading Data

Creating dictionary for datasets so that we can keep track of datsets and also make them callable to functions.

In [ ]:

```
# load data

DATA_DIR =  "/../Data/"

ds_names = ("application_train", "application_test", "bureau","bureau_balan
            "credit_card_balance", "installments_payments","previous_applic
            "POS_CASH_balance")

datasets = {}
datasets_transformed = {}

for ds_name in ds_names:
    datasets[ds_name] = pd.read_csv(os.getcwd() + DATA_DIR + f'{ds_name}.cs
```

Data Dictionary, Size, and Overview

As part of the data download comes a *Data Dictionary* named `HomeCredit_columns_description.csv` and it is summarized in Section 11 as an attachment to this report. The subsections below give brief table descriptions, variable name overviews, and head views for the attributes in the normalized data tables. Additionally included are information on the size (memory usage) of each table. All together summed the dataset is approximately 2.7 gigabytes large.

Application train

- **application_train/application_test:** the main training and testing data with information about each loan application at Home Credit. Every loan has its own row and is identified by the feature SK_ID_CURR. The training application data comes with the TARGET indicating **0: the loan was repaid** or **1: the loan was not repaid**. The target variable defines if the client had payment difficulties meaning he/she had late payment more than X days on at least one of the first Y installments of the loan. Such case is marked as 1 while other all other cases as 0.

In [5]:

```
def load_data(df,name):
    #df = pd.read_csv(in_path)
    print(f"{name}: shape is {df.shape}")
    print(df.info())
    display(df.head(5))
    return df

ds_name = 'application_train'
datasets[ds_name]= load_data(datasets[ds_name],ds_name)
```

```
application_train: shape is (307511, 122)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307511 entries, 0 to 307510
Columns: 122 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(41), object(16)
memory usage: 286.2+ MB
None
```

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_O
0	100002	1	Cash loans	M	N	
1	100003	0	Cash loans	F	N	
2	100004	0	Revolving loans	M	Y	
3	100006	0	Cash loans	F	N	
4	100007	0	Cash loans	M	N	

5 rows × 122 columns

Application test

In [6]:

```
ds_name = 'application_test'
datasets[ds_name]= load_data(datasets[ds_name],ds_name)
```

```
application_test: shape is (48744, 121)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48744 entries, 0 to 48743
Columns: 121 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(40), object(16)
memory usage: 45.0+ MB
None
```

	SK_ID_CURR	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALT
0	100001	Cash loans	F	N	
1	100005	Cash loans	M	N	
2	100013	Cash loans	M	Y	
3	100028	Cash loans	F	N	
4	100038	Cash loans	M	Y	

5 rows × 121 columns

The application dataset has the most information about the client: Gender, Income, Family Status, Education, and others.

The Other Datasets

- **bureau:** data concerning client's previous credits from other financial institutions. Each previous credit has its own row in bureau but one loan in the application data can have multiple previous credits.
- **bureau_balance:** monthly data about previous credits in bureau. Each row is one month of a previous credit and a single previous credit can have multiple rows, one for each month of the credit length.
- **previous_application:** previous applications for loans at Home Credit of clients who have loans in the application train data. Each current loan in the application train data can have multiple previous loans. Each previous application has one row and is identified by the feature SK_ID_PREV.
- **POS_CASH_BALANCE:** monthly data about previous point of sale or cash loans clients have had with Home Credit. Each row is one month of a previous point of sale or cash loan and a single previous loan can have many rows.
- **Credit_card_balance:** monthly data about previous credit cards clients have had with Home Credit. Each row is one month of a credit card balance and a single credit card can have many rows.
- **installments_payment:** payment history for previous loans at Home Credit. There is one row for every made payment and one row for every missed payment.

In [7]:

```
%%time
ds_names = ("application_train", "application_test", "bureau","credit_card_
            "previous_application","POS_CASH_balance")

for ds_name in ds_names:
    datasets[ds_name] = load_data(datasets[ds_name], ds_name)
```

```
application_train: shape is (307511, 122)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 307511 entries, 0 to 307510
Columns: 122 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(41), object(16)
memory usage: 286.2+ MB
None
```

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_O
0	100002	1	Cash loans	M	N	
1	100003	0	Cash loans	F	N	
2	100004	0	Revolving loans	M	Y	
3	100006	0	Cash loans	F	N	
4	100007	0	Cash loans	M	N	

5 rows × 122 columns

```
application_test: shape is (48744, 121)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48744 entries, 0 to 48743
Columns: 121 entries, SK_ID_CURR to AMT_REQ_CREDIT_BUREAU_YEAR
dtypes: float64(65), int64(40), object(16)
memory usage: 45.0+ MB
None
```

	SK_ID_CURR	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FLAG_OWN_REALT
0	100001	Cash loans	F	N	
1	100005	Cash loans	M	N	
2	100013	Cash loans	M	Y	
3	100028	Cash loans	F	N	
4	100038	Cash loans	M	Y	

5 rows × 121 columns

```
bureau: shape is (1716428, 17)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1716428 entries, 0 to 1716427
Data columns (total 17 columns):
 #   Column                  Dtype
---  ------                  -----
 0   SK_ID_CURR              int64
 1   SK_ID_BUREAU            int64
 2   CREDIT_ACTIVE           object
 3   CREDIT_CURRENCY         object
 4   DAYS_CREDIT             int64
 5   CREDIT_DAY_OVERDUE      int64
 6   DAYS_CREDIT_ENDDATE     float64
 7   DAYS_ENDDATE_FACT       float64
 8   AMT_CREDIT_MAX_OVERDUE  float64
 9   CNT_CREDIT_PROLONG      int64
 10  AMT_CREDIT_SUM          float64
 11  AMT_CREDIT_SUM_DEBT     float64
 12  AMT_CREDIT_SUM_LIMIT    float64
 13  AMT_CREDIT_SUM_OVERDUE  float64
 14  CREDIT_TYPE             object
 15  DAYS_CREDIT_UPDATE      int64
 16  AMT_ANNUITY             float64
dtypes: float64(8), int64(6), object(3)
memory usage: 222.6+ MB
None
```

	SK_ID_CURR	SK_ID_BUREAU	CREDIT_ACTIVE	CREDIT_CURRENCY	DAYS_CREDIT	CREDIT_
0	215354	5714462	Closed	currency 1	-497	
1	215354	5714463	Active	currency 1	-208	
2	215354	5714464	Active	currency 1	-203	
3	215354	5714465	Active	currency 1	-203	
4	215354	5714466	Active	currency 1	-629	

```
credit_card_balance: shape is (3840312, 23)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3840312 entries, 0 to 3840311
Data columns (total 23 columns):
 #   Column                      Dtype
---  ------                      -----
 0   SK_ID_PREV                  int64
 1   SK_ID_CURR                  int64
 2   MONTHS_BALANCE              int64
 3   AMT_BALANCE                 float64
 4   AMT_CREDIT_LIMIT_ACTUAL     int64
 5   AMT_DRAWINGS_ATM_CURRENT    float64
 6   AMT_DRAWINGS_CURRENT        float64
 7   AMT_DRAWINGS_OTHER_CURRENT  float64
 8   AMT_DRAWINGS_POS_CURRENT    float64
 9   AMT_INST_MIN_REGULARITY     float64
 10  AMT_PAYMENT_CURRENT         float64
 11  AMT_PAYMENT_TOTAL_CURRENT   float64
 12  AMT_RECEIVABLE_PRINCIPAL    float64
 13  AMT_RECIVABLE               float64
 14  AMT_TOTAL_RECEIVABLE        float64
 15  CNT_DRAWINGS_ATM_CURRENT    float64
 16  CNT_DRAWINGS_CURRENT        int64
 17  CNT_DRAWINGS_OTHER_CURRENT  float64
 18  CNT_DRAWINGS_POS_CURRENT    float64
 19  CNT_INSTALMENT_MATURE_CUM   float64
 20  NAME_CONTRACT_STATUS        object
 21  SK_DPD                      int64
 22  SK_DPD_DEF                  int64
dtypes: float64(15), int64(7), object(1)
memory usage: 673.9+ MB
None
```

	SK_ID_PREV	SK_ID_CURR	MONTHS_BALANCE	AMT_BALANCE	AMT_CREDIT_LIMIT_ACTUAL
0	2562384	378907	-6	56.970	135000
1	2582071	363914	-1	63975.555	45000
2	1740877	371185	-7	31815.225	450000
3	1389973	337855	-4	236572.110	225000
4	1891521	126868	-1	453919.455	450000

5 rows × 23 columns

```
installments_payments: shape is (13605401, 8)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 13605401 entries, 0 to 13605400
Data columns (total 8 columns):
 #   Column                  Dtype
---  ------                  -----
 0   SK_ID_PREV              int64
 1   SK_ID_CURR              int64
 2   NUM_INSTALMENT_VERSION  float64
 3   NUM_INSTALMENT_NUMBER   int64
 4   DAYS_INSTALMENT         float64
 5   DAYS_ENTRY_PAYMENT      float64
 6   AMT_INSTALMENT          float64
 7   AMT_PAYMENT             float64
dtypes: float64(5), int64(3)
memory usage: 830.4 MB
None
```

	SK_ID_PREV	SK_ID_CURR	NUM_INSTALMENT_VERSION	NUM_INSTALMENT_NUMBER	DAYS_
0	1054186	161674	1.0	6	
1	1330831	151639	0.0	34	
2	2085231	193053	2.0	1	
3	2452527	199697	1.0	3	
4	2714724	167756	1.0	2	

```
previous_application: shape is (1670214, 37)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1670214 entries, 0 to 1670213
Data columns (total 37 columns):
 #   Column                       Non-Null Count    Dtype
---  ------                       --------------    -----
 0   SK_ID_PREV                   1670214 non-null  int64
 1   SK_ID_CURR                   1670214 non-null  int64
 2   NAME_CONTRACT_TYPE           1670214 non-null  object
 3   AMT_ANNUITY                  1297979 non-null  float64
 4   AMT_APPLICATION              1670214 non-null  float64
 5   AMT_CREDIT                   1670213 non-null  float64
 6   AMT_DOWN_PAYMENT             774370 non-null   float64
 7   AMT_GOODS_PRICE              1284699 non-null  float64
 8   WEEKDAY_APPR_PROCESS_START   1670214 non-null  object
 9   HOUR_APPR_PROCESS_START      1670214 non-null  int64
 10  FLAG_LAST_APPL_PER_CONTRACT  1670214 non-null  object
 11  NFLAG_LAST_APPL_IN_DAY       1670214 non-null  int64
 12  RATE_DOWN_PAYMENT            774370 non-null   float64
 13  RATE_INTEREST_PRIMARY        5951 non-null     float64
 14  RATE_INTEREST_PRIVILEGED     5951 non-null     float64
 15  NAME_CASH_LOAN_PURPOSE       1670214 non-null  object
 16  NAME_CONTRACT_STATUS         1670214 non-null  object
 17  DAYS_DECISION                1670214 non-null  int64
 18  NAME_PAYMENT_TYPE            1670214 non-null  object
 19  CODE_REJECT_REASON           1670214 non-null  object
 20  NAME_TYPE_SUITE              849809 non-null   object
 21  NAME_CLIENT_TYPE             1670214 non-null  object
 22  NAME_GOODS_CATEGORY          1670214 non-null  object
 23  NAME_PORTFOLIO               1670214 non-null  object
 24  NAME_PRODUCT_TYPE            1670214 non-null  object
 25  CHANNEL_TYPE                 1670214 non-null  object
 26  SELLERPLACE_AREA             1670214 non-null  int64
 27  NAME_SELLER_INDUSTRY         1670214 non-null  object
 28  CNT_PAYMENT                  1297984 non-null  float64
 29  NAME_YIELD_GROUP             1670214 non-null  object
 30  PRODUCT_COMBINATION          1669868 non-null  object
 31  DAYS_FIRST_DRAWING           997149 non-null   float64
 32  DAYS_FIRST_DUE               997149 non-null   float64
 33  DAYS_LAST_DUE_1ST_VERSION    997149 non-null   float64
 34  DAYS_LAST_DUE                997149 non-null   float64
 35  DAYS_TERMINATION             997149 non-null   float64
 36  NFLAG_INSURED_ON_APPROVAL    997149 non-null   float64
dtypes: float64(15), int64(6), object(16)
memory usage: 471.5+ MB
None
```

	SK_ID_PREV	SK_ID_CURR	NAME_CONTRACT_TYPE	AMT_ANNUITY	AMT_APPLICATION	AM
0	2030495	271877	Consumer loans	1730.430	17145.0	
1	2802425	108129	Cash loans	25188.615	607500.0	
2	2523466	122040	Cash loans	15060.735	112500.0	
3	2819243	176158	Cash loans	47041.335	450000.0	
4	1784265	202054	Cash loans	31924.395	337500.0	

5 rows × 37 columns

```
POS_CASH_balance: shape is (10001358, 8)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 10001358 entries, 0 to 10001357
Data columns (total 8 columns):
 #   Column                 Dtype
---  ------                 -----
 0   SK_ID_PREV             int64
 1   SK_ID_CURR             int64
 2   MONTHS_BALANCE         int64
 3   CNT_INSTALMENT         float64
 4   CNT_INSTALMENT_FUTURE  float64
 5   NAME_CONTRACT_STATUS   object
 6   SK_DPD                 int64
 7   SK_DPD_DEF             int64
dtypes: float64(2), int64(5), object(1)
memory usage: 610.4+ MB
None
```

	SK_ID_PREV	SK_ID_CURR	MONTHS_BALANCE	CNT_INSTALMENT	CNT_INSTALMENT_FUTUR
0	1803195	182943	-31	48.0	45
1	1715348	367990	-33	36.0	35
2	1784872	397406	-32	12.0	9
3	1903291	269225	-35	48.0	42
4	2341044	334279	-35	36.0	35

```
CPU times: user 3.55 s, sys: 3.64 ms, total: 3.55 s
Wall time: 3.54 s
```

In [8]:

```
for ds_name in datasets.keys():
    print(f'dataset {ds_name:24}: [ {datasets[ds_name].shape[0]:10,}, {data
```

```
dataset application_train       : [    307,511, 122]
dataset application_test        : [     48,744, 121]
dataset bureau                  : [  1,716,428, 17]
dataset bureau_balance          : [ 27,299,925, 3]
dataset credit_card_balance     : [  3,840,312, 23]
dataset installments_payments   : [ 13,605,401, 8]
dataset previous_application    : [  1,670,214, 37]
dataset POS_CASH_balance        : [ 10,001,358, 8]
```

Exploratory Data Analysis

Exploratory Data Analysis (*EDA*) on the input tables is necessary to understand the data and to make decisions on transforming the data into a format that is ready for the machine learning pipeline. Custom functions are designed and used for each normalized table to explore the data distribution and transform the table. The tables are transformed by aggregating the data into summary statistics for the table so the child tables can be joined to their parent tables, and ultimately rolled into the application_test_train data based on their ID keys (`SK_ID_CURR` , `SK_ID_PREV` , `SK_ID_BUREAU`). The subsections below outline the main functions designed to accomplish the EDA and data transformation, and also the individual steps applied to each table to accomplish this process.

EDA Functions

A number of custom functions were developed to explore and transform the data. Names of the functions and brief descriptions of their usefulness are summarized here:

1. Summary Statistics
    - This provides an overview of summary statistics on each variable of each table
2. Feature Type Extraction
    - This function describes numerical and categorical feature types and provides reference lists for other functions in the EDA process
3. Missing Data Analysis
    - Looks at the missing data rate for a table's columns
4. Feature Aggregating
    - A function to group the tables by their ID key and aggregate them into their summary statistics
5. One-Hot-Encoding Extract-Transform-Load
    - An ETL function that one-hot encodes categorical variables so they can be aggregated, and applies the transformation to the table

Summary Statistics for All tables

Function Feature Summary:

A summary of each table is provided using the described function below. This function will take a dataframe as input and provide summary statistics for the table. This generic function will help us evaluate each table with just one line of code, improving the reusablity of the code. Summary statistics will show statistics on each dataframe and through that meaningful interpretations were made about the data to further plan EDA on each dataset.

Function name: feature_summary

Input: Dataframe

Output: Dataframe Summary

Function output:

1. Provide table name for which analysis in progress.
2. Shape - provides rows and columns in the data set.
3. NULLS - Number of nulls in the column.
4. %_NULL - NULL percentage. (nulls/attributes count).
5. Unique values - No of unique values for the feature.
6. Data Type - Id column numerical or categorical.
7. Max/Min - Max and min for the column. This provides idea about scale. helps if need log transformation.
8. Mean - Mean of the attribute.
9. Std - Standard Deviation of the atttibute.
10. Skewness - Skewness, data distribution of transformations.
11. Sample values - Sample values for the attribute.

In [9]:

```
def feature_summary(df_fa):
    print('DataFrame shape')
    print('Rows:',df_fa.shape[0])
    print('Cols:',df_fa.shape[1])
    print("-----------------------------------------------------------------

    col_list=['Null','%_Null','Unique_Count','Data_type','Max/Min','Mean','S
    df=pd.DataFrame(index=df_fa.columns,columns=col_list)
    df['Null']=list([len(df_fa[col][df_fa[col].isnull()]) for i,col in enume
    df['%_Null']=round((df_fa.isnull().sum()/df_fa.isnull().count()*100),3)
    df['Unique_Count']=list([len(df_fa[col].unique()) for i,col in enumerate
    df['Data_type']=list([df_fa[col].dtype for i,col in enumerate(df_fa.colu
    for i,col in enumerate(df_fa.columns):
        if 'float' in str(df_fa[col].dtype) or 'int' in str(df_fa[col].dtype
            df.at[col,'Max/Min']=str(round(df_fa[col].max(),2))+'/'+str(roun
            df.at[col,'Mean']=df_fa[col].mean()
            df.at[col,'Std']=df_fa[col].std()
            df.at[col,'Skewness']=df_fa[col].skew()
        df.at[col,'Sample_values']=list(df_fa[col].unique())
    print("Table Statistics")
    print("-----------------------------------------------------------------
    display(df.fillna('-'))
```

In [10]:

```
for i,ds_name in enumerate(datasets.keys()):
    print("Table under consideration:",ds_name.upper())
    print("-----------------------------------------------------------------
    ds = feature_summary(datasets[ds_name])
    print("-----------------------------------------------------------------
```

```
Table under consideration: APPLICATION_TRAIN
-----------------------------------------------------------------------
DataFrame shape
Rows: 307511
Cols: 122
-----------------------------------------------------------------------
Table Statistics
-----------------------------------------------------------------------
```

	Null	%_Null	Unique_Count	Data_type	Max/Min	
SK_ID_CURR	0	0.000	307511	int64	456255/100002	27
TARGET	0	0.000	2	int64	1/0	
NAME_CONTRACT_TYPE	0	0.000	2	object	-	
CODE_GENDER	0	0.000	3	object	-	
FLAG_OWN_CAR	0	0.000	2	object	-	
...	...	...	...	...	...	
AMT_REQ_CREDIT_BUREAU_DAY	41519	13.502	10	float64	9.0/0.0	
AMT_REQ_CREDIT_BUREAU_WEEK	41519	13.502	10	float64	8.0/0.0	
AMT_REQ_CREDIT_BUREAU_MON	41519	13.502	25	float64	27.0/0.0	
AMT_REQ_CREDIT_BUREAU_QRT	41519	13.502	12	float64	261.0/0.0	
AMT_REQ_CREDIT_BUREAU_YEAR	41519	13.502	26	float64	25.0/0.0	

122 rows × 9 columns

```
----------------------------------------------------------------------
Table under consideration: APPLICATION_TEST
----------------------------------------------------------------------
DataFrame shape
Rows: 48744
Cols: 121
----------------------------------------------------------------------
Table Statistics
----------------------------------------------------------------------
```

	Null	%_Null	Unique_Count	Data_type	Max/Min	
SK_ID_CURR	0	0.00	48744	int64	456250/100001	277
NAME_CONTRACT_TYPE	0	0.00	2	object	-	
CODE_GENDER	0	0.00	2	object	-	
FLAG_OWN_CAR	0	0.00	2	object	-	
FLAG_OWN_REALTY	0	0.00	2	object	-	
...	...	...	...	...	...	
AMT_REQ_CREDIT_BUREAU_DAY	6049	12.41	4	float64	2.0/0.0	
AMT_REQ_CREDIT_BUREAU_WEEK	6049	12.41	4	float64	2.0/0.0	
AMT_REQ_CREDIT_BUREAU_MON	6049	12.41	8	float64	6.0/0.0	
AMT_REQ_CREDIT_BUREAU_QRT	6049	12.41	9	float64	7.0/0.0	
AMT_REQ_CREDIT_BUREAU_YEAR	6049	12.41	17	float64	17.0/0.0	

121 rows × 9 columns

```
-----------------------------------------------------------------------
Table under consideration: BUREAU
-----------------------------------------------------------------------
DataFrame shape
Rows: 1716428
Cols: 17
-----------------------------------------------------------------------
Table Statistics
-----------------------------------------------------------------------
```

	Null	%_Null	Unique_Count	Data_type	Max/M
SK_ID_CURR	0	0.000	305811	int64	456255/1000(
SK_ID_BUREAU	0	0.000	1716428	int64	6843457/50000(
CREDIT_ACTIVE	0	0.000	4	object	
CREDIT_CURRENCY	0	0.000	4	object	
DAYS_CREDIT	0	0.000	2923	int64	0/-29
CREDIT_DAY_OVERDUE	0	0.000	942	int64	2792
DAYS_CREDIT_ENDDATE	105553	6.150	14097	float64	31199.0/-42060
DAYS_ENDDATE_FACT	633653	36.917	2918	float64	0.0/-42023
AMT_CREDIT_MAX_OVERDUE	1124488	65.513	68252	float64	115987185.0/0
CNT_CREDIT_PROLONG	0	0.000	10	int64	9
AMT_CREDIT_SUM	13	0.001	236709	float64	585000000.0/0
AMT_CREDIT_SUM_DEBT	257669	15.012	226538	float64	170100000.0/-4705600.3
AMT_CREDIT_SUM_LIMIT	591780	34.477	51727	float64	4705600.32/-586406.1

	Null	%_Null	Unique_Count	Data_type	Max/M
AMT_CREDIT_SUM_OVERDUE	0	0.000	1616	float64	3756681.0/0
CREDIT_TYPE	0	0.000	15	object	
DAYS_CREDIT_UPDATE	0	0.000	2982	int64	372/-419
AMT_ANNUITY	1226791	71.473	40322	float64	118453423.5/0

```
----------------------------------------------------------------------
Table under consideration: BUREAU_BALANCE
----------------------------------------------------------------------
DataFrame shape
Rows: 27299925
Cols: 3
----------------------------------------------------------------------
Table Statistics
----------------------------------------------------------------------
```

	Null	%_Null	Unique_Count	Data_type	Max/Min	Mean
SK_ID_BUREAU	0	0.0	817395	int64	6842888/5001709	6036297.332974
MONTHS_BALANCE	0	0.0	97	int64	0/-96	-30.741687
STATUS	0	0.0	8	object	-	-

```
----------------------------------------------------------------------
Table under consideration: CREDIT_CARD_BALANCE
----------------------------------------------------------------------
DataFrame shape
Rows: 3840312
Cols: 23
----------------------------------------------------------------------
Table Statistics
----------------------------------------------------------------------
```

	Null	%_Null	Unique_Count	Data_type	Max
SK_ID_PREV	0	0.000	104307	int64	2843496/100
SK_ID_CURR	0	0.000	103558	int64	456250/10
MONTHS_BALANCE	0	0.000	96	int64	-
AMT_BALANCE	0	0.000	1347904	float64	1505902.19/-42025
AMT_CREDIT_LIMIT_ACTUAL	0	0.000	181	int64	1350(
AMT_DRAWINGS_ATM_CURRENT	749816	19.525	2268	float64	2115000.0/-682
AMT_DRAWINGS_CURRENT	0	0.000	187005	float64	2287098.31/-621
AMT_DRAWINGS_OTHER_CURRENT	749816	19.525	1833	float64	1529847.
AMT_DRAWINGS_POS_CURRENT	749816	19.525	168749	float64	2239274.1
AMT_INST_MIN_REGULARITY	305236	7.948	312267	float64	202882.0
AMT_PAYMENT_CURRENT	767988	19.998	163210	float64	4289207.4

	Null	%_Null	Unique_Count	Data_type	Max
AMT_PAYMENT_TOTAL_CURRENT	0	0.000	182957	float64	4278315.6
AMT_RECEIVABLE_PRINCIPAL	0	0.000	1195839	float64	1472316.79/-4233(
AMT_RECIVABLE	0	0.000	1338878	float64	1493338.19/-42025
AMT_TOTAL_RECEIVABLE	0	0.000	1339008	float64	1493338.19/-42025
CNT_DRAWINGS_ATM_CURRENT	749816	19.525	45	float64	51.
CNT_DRAWINGS_CURRENT	0	0.000	129	int64	1
CNT_DRAWINGS_OTHER_CURRENT	749816	19.525	12	float64	12.
CNT_DRAWINGS_POS_CURRENT	749816	19.525	134	float64	165.
CNT_INSTALMENT_MATURE_CUM	305236	7.948	122	float64	120.
NAME_CONTRACT_STATUS	0	0.000	7	object	
SK_DPD	0	0.000	917	int64	32
SK_DPD_DEF	0	0.000	378	int64	32

```
----------------------------------------------------------------------
Table under consideration: INSTALLMENTS_PAYMENTS
----------------------------------------------------------------------
DataFrame shape
Rows: 13605401
Cols: 8
----------------------------------------------------------------------
Table Statistics
----------------------------------------------------------------------
```

	Null	%_Null	Unique_Count	Data_type	Max/Min	
SK_ID_PREV	0	0.000	997752	int64	2843499/1000001	1.90336
SK_ID_CURR	0	0.000	339587	int64	456255/100001	2.78444
NUM_INSTALMENT_VERSION	0	0.000	65	float64	178.0/0.0	8.56637
NUM_INSTALMENT_NUMBER	0	0.000	277	int64	277/1	1.88709
DAYS_INSTALMENT	0	0.000	2922	float64	-1.0/-2922.0	-1.04227
DAYS_ENTRY_PAYMENT	2905	0.021	3040	float64	-1.0/-4921.0	-1.05111
AMT_INSTALMENT	0	0.000	902539	float64	3771487.85/0.0	1.70509
AMT_PAYMENT	2905	0.021	944236	float64	3771487.85/0.0	1.72382

```
----------------------------------------------------------------------
Table under consideration: PREVIOUS_APPLICATION
----------------------------------------------------------------------
DataFrame shape
Rows: 1670214
Cols: 37
----------------------------------------------------------------------
Table Statistics
----------------------------------------------------------------------
```

	Null	%_Null	Unique_Count	Data_type	Max/Mir
SK_ID_PREV	0	0.000	1670214	int64	2845382/1000001
SK_ID_CURR	0	0.000	338857	int64	456255/100001
NAME_CONTRACT_TYPE	0	0.000	4	object	-
AMT_ANNUITY	372235	22.287	357960	float64	418058.15/0.0
AMT_APPLICATION	0	0.000	93885	float64	6905160.0/0.0
AMT_CREDIT	1	0.000	86804	float64	6905160.0/0.0
AMT_DOWN_PAYMENT	895844	53.636	29279	float64	3060045.0/-0.9
AMT_GOODS_PRICE	385515	23.082	93886	float64	6905160.0/0.0
WEEKDAY_APPR_PROCESS_START	0	0.000	7	object	-
HOUR_APPR_PROCESS_START	0	0.000	24	int64	23/0
FLAG_LAST_APPL_PER_CONTRACT	0	0.000	2	object	-
NFLAG_LAST_APPL_IN_DAY	0	0.000	2	int64	1/0
RATE_DOWN_PAYMENT	895844	53.636	207034	float64	1.0/-0.0
RATE_INTEREST_PRIMARY	1664263	99.644	149	float64	1.0/0.03
RATE_INTEREST_PRIVILEGED	1664263	99.644	26	float64	1.0/0.37
NAME_CASH_LOAN_PURPOSE	0	0.000	25	object	-
NAME_CONTRACT_STATUS	0	0.000	4	object	-

	Null	%_Null	Unique_Count	Data_type	Max/Mir
DAYS_DECISION	0	0.000	2922	int64	-1/-2922
NAME_PAYMENT_TYPE	0	0.000	4	object	-
CODE_REJECT_REASON	0	0.000	9	object	-
NAME_TYPE_SUITE	820405	49.120	8	object	-
NAME_CLIENT_TYPE	0	0.000	4	object	-
NAME_GOODS_CATEGORY	0	0.000	28	object	-
NAME_PORTFOLIO	0	0.000	5	object	-
NAME_PRODUCT_TYPE	0	0.000	3	object	-
CHANNEL_TYPE	0	0.000	8	object	-
SELLERPLACE_AREA	0	0.000	2097	int64	4000000/-1
NAME_SELLER_INDUSTRY	0	0.000	11	object	-
CNT_PAYMENT	372230	22.286	50	float64	84.0/0.0
NAME_YIELD_GROUP	0	0.000	5	object	-
PRODUCT_COMBINATION	346	0.021	18	object	-
DAYS_FIRST_DRAWING	673065	40.298	2839	float64	365243.0/-2922.0
DAYS_FIRST_DUE	673065	40.298	2893	float64	365243.0/-2892.0
DAYS_LAST_DUE_1ST_VERSION	673065	40.298	4606	float64	365243.0/-2801.0
DAYS_LAST_DUE	673065	40.298	2874	float64	365243.0/-2889.0

	Null	%_Null	Unique_Count	Data_type	Max/Mir
DAYS_TERMINATION	673065	40.298	2831	float64	365243.0/-2874.(
NFLAG_INSURED_ON_APPROVAL	673065	40.298	3	float64	1.0/0.(

```
----------------------------------------------------------------------
Table under consideration: POS_CASH_BALANCE
----------------------------------------------------------------------
DataFrame shape
Rows: 10001358
Cols: 8
----------------------------------------------------------------------
Table Statistics
----------------------------------------------------------------------
```

	Null	%_Null	Unique_Count	Data_type	Max/Min	
SK_ID_PREV	0	0.000	936325	int64	2843499/1000001	1903216.
SK_ID_CURR	0	0.000	337252	int64	456255/100001	278403.
MONTHS_BALANCE	0	0.000	96	int64	-1/-96	-35.
CNT_INSTALMENT	26071	0.261	74	float64	92.0/1.0	1
CNT_INSTALMENT_FUTURE	26087	0.261	80	float64	85.0/0.0	1(
NAME_CONTRACT_STATUS	0	0.000	9	object	-	
SK_DPD	0	0.000	3400	int64	4231/0	11.
SK_DPD_DEF	0	0.000	2307	int64	3595/0	0.

```
----------------------------------------------------------------------
```

Feature extraction based on Type

Function id_num_cat_feature:

This function will take a dataframe as input and return 4 lists that contain ID columns, numerical features, categorical features, and numerical features without the ID cols.

Function name: id_num_cat_feature Input : Dataframe Output : 4 Lists

Function output:

1. ID columns
2. Numerical features
3. Categorical features
4. Numerical features excluding the ID columns.

Using this function gives us mutiple lists of the column types which are then used in other functions. The function returns clear separations on what columns are categorical and numerical. By having the distinctive categorical columns and numerical columns lists, different transformations are performed on them based on if they are categorical or numerical.

In [6]:

```
def id_num_cat_feature(df,text = True):
    numerical = df.select_dtypes(include=['int64', 'float64']).columns
    categorical = df.select_dtypes(include=['object', 'bool']).columns
    feat_num = list(numerical)
    feat_cat = list(categorical)

    id_cols = ['SK_ID_CURR','SK_ID_BUREAU']

    id_cols = [cols for cols in  list(df.columns.intersection(id_cols))]
    features = list(set(df.columns) - set(id_cols))

    if text == True:
          # print eda
        print('--------')
        print(f"# of ID's: {len(id_cols)}")
        print(f" ID's:")
        print(id_cols)
        print('')
        print('--------')
        print(f"# All features: {len(features)}")
        print(f"All features:")
        print(features)
        print('')
        print(f"Missing data:")
        print(missing_data(df[features]))
        print('')
        print('--------')
        print(f"# of Numerical features: {len(feat_num)}")
        print(f"Numerical features:")
        print(feat_num)
        print('')
        print(f"Numerical Statistical Summary:")
        print('')
        print(df[feat_num].describe())
        print('')
        print('--------')
        print(f"# of Categorical features: {len(feat_cat)}")
        print(f"Categorical features:")
        print(feat_cat)
        print('')
        print(f"Categorical Statistical Summary:")
        print('')
        #print(df[feat_cat].describe(include='all'))
        print('')
        print("Categories:")
        print('')
        print(df[feat_cat].apply(lambda col: col.unique()))
        print('')
        print('--------')
    return id_cols,feat_num,feat_cat,features
```

Missing Count and percentage

Function missing_data:

This function will take a dataframe as input and provide null count and % of nulls for a dataframe.

Function name: missing_data

Input: Dataframe

Output: NULL count and NULL %

Function output:

1. NULL Count
2. NULL Percentage.

In [7]:

```
def missing_data(data):
    total = data.isnull().sum().sort_values(ascending = False)
    percent = (data.isnull().sum()/data.isnull().count()*100).sort_values(a
    return pd.concat([total, percent], axis=1, keys=['Total', 'Percent'])
```

Feature Aggregating

Class: FeatureSummarizer

Class FeatureSummarizer has following aggregation parameters:

1. "min"
2. "max"
3. "count"
4. "sum"
5. "median"
6. "mean"
7. "var"

Based on the keys of the dataframe, transformation function will enable grouping of feature variables on ID and then aggregate the feature variables into their predefined statistical summaries for each grouping.

Function: runFeatureSummarizer

This function will take dataframe and features as input and using the class above, get aggregated features. The output of the above function will be the transformed aggregated features for the dataframe passed into the function argument.

In [8]:

```
class FeatureSummarizer(BaseEstimator, TransformerMixin):
    def __init__(self, features=None):
        self.features = features
        self.agg_ops = ["min", "max", "count", "sum", "median", "mean", "va

    def fit(self, X, y=None):
        return self

    def transform(self, X, y=None):
        keys = list(set(X.columns) - set(self.features))

        result = X.groupby(keys, as_index=False).agg({ft:self.agg_ops for f
        result.columns = result.columns.map(lambda ct: '_'.join([x for x in
        result.reset_index()
        return result
```

In [9]:

```
def runFeatureSummarizer(df, features):
    print(f"df.shape: {df.shape}\n")
    print(f"Aggregated Features:\ndf[{features}][0:5]: \n{df[features][0:5]
    pipeline = make_pipeline(FeatureSummarizer(features))
    return(pipeline.fit_transform(df))
```

One-Hot-Encoded Extract-Transform-Load

Function eda_transformation:

Function input: Dataframe, n. Where n is a parameter for feature selection.

- This function calls id_num_cat_feature to put all features types id columns, numerical features, categorical features, and numerical features excluding id columns into 4 respectively lists.
- Categorical variables are one hot encoded into some numerical value, to allow the pipeline to make interpretations from categorical varibles more easily.
- run FeatureSummarizer function is called to get all aggregated features.
- Final features are selected through feature selection from the transformed tables.
- Output of this function will be a dataframe with all aggregated features selected, which will eventually be used in feature selection.
- The function will also prints aggregated features and aggregated data.

Feature Selection:

```
- Once we have completed OHE, we can remove some attributes
which are redundant, for example:
- count of previous application is good to have, but mean,max
is not required.
- Features with DAYS, count suffice the aggregated feature, we
can remove all other min, max, etc. feature forthat column.
- Based on above logic, we removed certain columns from the
supporting tables to remove the redundant attributes and save
on execution time.
```

In [10]:

```
def eda_transformation(df,n):

    id_cols, feat_num, feat_cat, features =  id_num_cat_feature(df)

    df = pd.get_dummies(data=df, columns=feat_cat)

    features = list(set(df.columns) - set(id_cols))
    feat_ohe = list(set(features) - set(feat_num))

    print(f"# of OHE categorical features: {len(feat_ohe)}")
    print(f"OHE Categorical features: {feat_ohe}")
    print('--------')

    df = runFeatureSummarizer(df, features)

    if n == 0:
        # bureau_balance
        feature_selection = [
            df[id_cols],
            df[[column for column in df.columns if column.startswith('MONTH
            df[[column for column in df.columns if column.startswith('STATU
        ]
    elif n == 1:
        # bureau
        feature_selection = [
            df[[column for column in df.columns if not column.startswith(tu
            df[[column for column in df.columns if column.startswith('DAYS_
            df[[column for column in df.columns if column.startswith(tuple(
        ]
    elif n ==3:

        feature_selection = [
            df[[column for column in df.columns if not column.startswith('S
            df[[column for column in df.columns if column.startswith('DAYS'
            df[[column for column in df.columns if column.startswith('SK_ID_
            df[[column for column in df.columns if column.startswith(tuple(
        ]

    elif n ==4:

        feature_selection = [
            df[[column for column in df.columns if not column.startswith('S
            df[[column for column in df.columns if column.startswith('SK_ID_
            df[[column for column in df.columns if column.startswith(tuple(
        ]
    else:
        print('ERROR: Invalid feature method. ')

    df = pd.concat(feature_selection, axis=1)

    features = list(set(df.columns) - set(id_cols))

    print('--------')
    print('Aggregated Features:')
    print('\n'.join(map(str, sorted(features))))
    print('')
    print('Aggregated Data:')
    print('')
```

```
    print(df[features].describe().T)
    return df
```

EDA on Tables

Since each table is structured slightly differently and they are all composed of separate variables, It is necessary to perform different steps on each of the tables as necessary. While all the tables can be explored and transformed by the `eda_transformation()` function,

Bureau Balance

1. This function is specifically for Bureau Balance table. Below are the intial pre-processing steps done before passing this table into the pipeline.

    - Since this table has only 2 features, no features are dropped or created for this table.
    - Take absolute of the months balance attribute. which was provided as negative values, as it is relative to application date.
    - Any column or row with more than 70% of its data as null will be deleted from the dataframe, as the threshold is set to .7.
    - Once processed, store the transformed data into a csv file. Benefit of this is that data can be passed ed csv file. directly to model for merging into application train/test table. We do not have to repeatedly perform expensive EDA/ETL/Transformation.

    - Note that this table does not contain SK_ID_CURR. It will be rolled up to Bureau table which contains the SK_ID_CURR.

In [15]:

```
def bbal_eda(df):
    print("bureau_bal :: EDA and transformation")
    print('')
    bbal = df
    #bureau balance table contains all the data so no need to drop any colu
    #Adding new features, take the abs for the monthly balance attribute.
    bbal['MONTHS_BALANCE'] = bbal['MONTHS_BALANCE'].abs()
    return (eda_transformation(bbal,0))
```

In [16]:

```
bbal = datasets['bureau_balance']
bbal = bbal_eda(bbal)
datasets_transformed['bureau_balance'] = bbal
```

```
bureau_bal :: EDA and transformation

--------
# of ID's: 1
 ID's:
['SK_ID_BUREAU']

--------
# All features: 2
All features:
['MONTHS_BALANCE', 'STATUS']

Missing data:
                Total  Percent
MONTHS_BALANCE      0      0.0
STATUS              0      0.0

--------
# of Numerical features: 2
Numerical features:
['SK_ID_BUREAU', 'MONTHS_BALANCE']

Numerical Statistical Summary:

       SK_ID_BUREAU  MONTHS_BALANCE
count  2.729992e+07    2.729992e+07
mean   6.036297e+06    3.074169e+01
std    4.923489e+05    2.386451e+01
min    5.001709e+06    0.000000e+00
25%    5.730933e+06    1.100000e+01
50%    6.070821e+06    2.500000e+01
75%    6.431951e+06    4.600000e+01
max    6.842888e+06    9.600000e+01

--------
# of Categorical features: 1
Categorical features:
['STATUS']

Categorical Statistical Summary:

          STATUS
count   27299925
unique         8
top            C
freq    13646993

Categories:

  STATUS
0      C
1      0
2      X
3      1
4      2
5      3
6      5
7      4
```

```
--------
# of OHE categorical features: 8
OHE Categorical features: ['STATUS_1', 'STATUS_4', 'STATUS_C', 'STATUS_3', '
STATUS_X', 'STATUS_5', 'STATUS_2', 'STATUS_0']
--------
df.shape: (27299925, 10)

Aggregated Features:
df[['STATUS_1', 'STATUS_4', 'STATUS_C', 'STATUS_3', 'STATUS_X', 'STATUS_5',
'STATUS_2', 'MONTHS_BALANCE', 'STATUS_0']][0:5]:
   STATUS_1  STATUS_4  STATUS_C  STATUS_3  STATUS_X  STATUS_5  STATUS_2  \
0         0         0         1         0         0         0         0
1         0         0         1         0         0         0         0
2         0         0         1         0         0         0         0
3         0         0         1         0         0         0         0
4         0         0         1         0         0         0         0

   MONTHS_BALANCE  STATUS_0
0               0         0
1               1         0
2               2         0
3               3         0
4               4         0
--------
Aggregated Features:
MONTHS_BALANCE_count
STATUS_0_mean
STATUS_0_median
STATUS_0_var
STATUS_1_mean
STATUS_1_median
STATUS_1_var
STATUS_2_mean
STATUS_2_median
STATUS_2_var
STATUS_3_mean
STATUS_3_median
STATUS_3_var
STATUS_4_mean
STATUS_4_median
STATUS_4_var
STATUS_5_mean
STATUS_5_median
STATUS_5_var
STATUS_C_mean
STATUS_C_median
STATUS_C_var
STATUS_X_mean
STATUS_X_median
STATUS_X_var

Aggregated Data:

                        count          mean        std  min        25%  \
STATUS_2_median       817395.0  8.563791e-06   0.002594  0.0   0.000000
STATUS_1_var          811206.0  9.649919e-03   0.035212  0.0   0.000000
STATUS_X_var          811206.0  6.244586e-02   0.096936  0.0   0.000000
STATUS_2_mean         817395.0  7.070824e-04   0.008025  0.0   0.000000
STATUS_4_median       817395.0  6.116994e-07   0.000553  0.0   0.000000
```

STATUS_4_var	811206.0	1.601024e-04	0.002804	0.0	0.000000
STATUS_C_mean	817395.0	3.710395e-01	0.379741	0.0	0.000000
STATUS_5_median	817395.0	1.025208e-03	0.031811	0.0	0.000000
STATUS_5_var	811206.0	7.843847e-04	0.011837	0.0	0.000000
STATUS_3_var	811206.0	2.446794e-04	0.003681	0.0	0.000000
STATUS_0_mean	817395.0	3.994569e-01	0.347193	0.0	0.086957
STATUS_5_mean	817395.0	1.492400e-03	0.027131	0.0	0.000000
STATUS_2_var	811206.0	6.811640e-04	0.007217	0.0	0.000000
STATUS_C_median	817395.0	4.211721e-01	0.490946	0.0	0.000000
STATUS_X_median	817395.0	1.815352e-01	0.380877	0.0	0.000000
STATUS_X_mean	817395.0	2.157315e-01	0.337008	0.0	0.000000
STATUS_0_var	811206.0	1.298257e-01	0.103613	0.0	0.013514
MONTHS_BALANCE_count	817395.0	3.339869e+01	25.794666	1.0	13.000000
STATUS_1_mean	817395.0	1.116493e-02	0.046621	0.0	0.000000
STATUS_1_median	817395.0	1.404462e-03	0.035453	0.0	0.000000
STATUS_C_var	811206.0	9.398683e-02	0.102171	0.0	0.000000
STATUS_3_median	817395.0	0.000000e+00	0.000000	0.0	0.000000
STATUS_3_mean	817395.0	2.469056e-04	0.003800	0.0	0.000000
STATUS_4_mean	817395.0	1.608014e-04	0.002889	0.0	0.000000
STATUS_0_median	817395.0	3.616085e-01	0.474283	0.0	0.000000

	50%	75%	max
STATUS_2_median	0.000000	0.000000	1.000000
STATUS_1_var	0.000000	0.000000	0.500000
STATUS_X_var	0.000000	0.100000	0.500000
STATUS_2_mean	0.000000	0.000000	0.714286
STATUS_4_median	0.000000	0.000000	0.500000
STATUS_4_var	0.000000	0.000000	0.333333
STATUS_C_mean	0.285714	0.764045	1.000000
STATUS_5_median	0.000000	0.000000	1.000000
STATUS_5_var	0.000000	0.000000	0.333333
STATUS_3_var	0.000000	0.000000	0.333333
STATUS_0_mean	0.307692	0.705882	1.000000
STATUS_5_mean	0.000000	0.000000	1.000000
STATUS_2_var	0.000000	0.000000	0.333333
STATUS_C_median	0.000000	1.000000	1.000000
STATUS_X_median	0.000000	0.000000	1.000000
STATUS_X_mean	0.024390	0.304348	1.000000
STATUS_0_var	0.128571	0.226129	0.500000
MONTHS_BALANCE_count	26.000000	48.000000	97.000000
STATUS_1_mean	0.000000	0.000000	1.000000
STATUS_1_median	0.000000	0.000000	1.000000
STATUS_C_var	0.058634	0.194444	0.500000
STATUS_3_median	0.000000	0.000000	0.000000
STATUS_3_mean	0.000000	0.000000	0.400000
STATUS_4_mean	0.000000	0.000000	0.500000
STATUS_0_median	0.000000	1.000000	1.000000

Bureau

1. This function is specifically for Bureau table. Below are the intial pre-processing steps done before passing this table into the pipeline.

    - For columns with DAYS in name, there are two with negative values, we will take the absolute for those.
    - Data from Buereau balance table is rolled up in Bureau table befor any EDA. This will enable using all the featues from buereau balance table as well before rolling up all data to main table, i.e. application train.
    - while doing left join, we updated OHE column names to more readale forms by removing any space or spcl charater to "_" which is widely used in column names.
    - Any column or row with more than 70% of its data as null will be deleted from the dataframe, as the threshold is set to .7.
    - Once processed, store the transformed data into a csv file. Benefit of this is that data can be passed ed csv file. directly to model for merging into application train/test table. We do not have to repeatedly perform expensive EDA/ETL/Transformation.

In [17]:

```
def bureau_eda(df):
    bureau = df
    drop_list_bureau = []

    #Adding new features
    #bureau['MONTHS_BALANCE'] = bureau['MONTHS_BALANCE'].abs()
    for c in [co for co in bureau.columns if 'DAYS' in co]:
        bureau[c] = bureau[c].replace({365243.0: np.nan})
        bureau[c] = bureau[c].abs()
    # Drop elements in drop list
    threshold = 0.7

    #Dropping rows with missing value rate higher than threshold
    bureau = bureau.loc[bureau.isnull().mean(axis=1) < threshold]

    return (eda_transformation(bureau,1))
```

In [18]:

```
bureau = datasets['bureau']

# rollup bureau_bal
# gets rid of the unwanted characters in categorical columns entries - make
bureau = bureau.merge(bbal, on='SK_ID_BUREAU', how='left') \
               .replace(to_replace='\s+', value='_', regex=True) \
               .replace(to_replace='\-', value='_', regex=True) \
               .replace(to_replace='\(', value='', regex=True) \
               .replace(to_replace='\)', value='', regex=True) \
               .drop('SK_ID_BUREAU', axis=1)
bureau = bureau_eda(bureau)
datasets_transformed['bureau'] = bureau
#bureau.to_csv(os.getcwd() + DATA_DIR + 'bureau_agg_data.csv')
```

```
--------
# of ID's: 1
 ID's:
['SK_ID_CURR']

--------
# All features: 40
All features:
['STATUS_2_median', 'AMT_CREDIT_SUM_LIMIT', 'DAYS_CREDIT_UPDATE', 'STATUS_4_
median', 'STATUS_4_var', 'STATUS_5_median', 'STATUS_C_mean', 'STATUS_5_var',
'STATUS_3_var', 'STATUS_2_var', 'AMT_ANNUITY', 'CREDIT_TYPE', 'STATUS_C_medi
an', 'CREDIT_DAY_OVERDUE', 'STATUS_0_var', 'MONTHS_BALANCE_count', 'STATUS_1
_mean', 'DAYS_CREDIT_ENDDATE', 'STATUS_3_mean', 'DAYS_ENDDATE_FACT', 'STATUS
_0_median', 'STATUS_1_var', 'STATUS_X_var', 'STATUS_2_mean', 'AMT_CREDIT_SUM
_OVERDUE', 'STATUS_0_mean', 'STATUS_5_mean', 'AMT_CREDIT_SUM', 'CREDIT_ACTIV
E', 'DAYS_CREDIT', 'STATUS_X_median', 'STATUS_X_mean', 'CNT_CREDIT_PROLONG',
'STATUS_1_median', 'STATUS_C_var', 'STATUS_3_median', 'CREDIT_CURRENCY', 'ST
ATUS_4_mean', 'AMT_CREDIT_SUM_DEBT', 'AMT_CREDIT_MAX_OVERDUE']

Missing data:
                          Total    Percent
AMT_ANNUITY             1064802  68.507601
AMT_CREDIT_MAX_OVERDUE   976510  62.827040
STATUS_5_var             785833  50.559197
STATUS_0_var             785833  50.559197
STATUS_1_var             785833  50.559197
STATUS_3_var             785833  50.559197
STATUS_2_var             785833  50.559197
STATUS_C_var             785833  50.559197
STATUS_4_var             785833  50.559197
STATUS_X_var             785833  50.559197
STATUS_1_median          779929  50.179343
STATUS_3_median          779929  50.179343
STATUS_X_mean            779929  50.179343
STATUS_X_median          779929  50.179343
STATUS_5_mean            779929  50.179343
STATUS_4_mean            779929  50.179343
STATUS_0_mean            779929  50.179343
STATUS_2_mean            779929  50.179343
STATUS_2_median          779929  50.179343
STATUS_0_median          779929  50.179343
STATUS_C_median          779929  50.179343
STATUS_3_mean            779929  50.179343
STATUS_4_median          779929  50.179343
STATUS_1_mean            779929  50.179343
STATUS_5_median          779929  50.179343
MONTHS_BALANCE_count     779929  50.179343
STATUS_C_mean            779929  50.179343
DAYS_ENDDATE_FACT        542773  34.921118
AMT_CREDIT_SUM_LIMIT     430737  27.712907
AMT_CREDIT_SUM_DEBT      144124   9.272700
DAYS_CREDIT_ENDDATE       81383   5.236048
AMT_CREDIT_SUM                5   0.000322
CREDIT_ACTIVE                 0   0.000000
CREDIT_TYPE                   0   0.000000
CNT_CREDIT_PROLONG            0   0.000000
CREDIT_DAY_OVERDUE            0   0.000000
AMT_CREDIT_SUM_OVERDUE        0   0.000000
CREDIT_CURRENCY               0   0.000000
```

```
DAYS_CREDIT_UPDATE            0   0.000000
DAYS_CREDIT                   0   0.000000

--------
# of Numerical features: 38
Numerical features:
['SK_ID_CURR', 'DAYS_CREDIT', 'CREDIT_DAY_OVERDUE', 'DAYS_CREDIT_ENDDATE', '
DAYS_ENDDATE_FACT', 'AMT_CREDIT_MAX_OVERDUE', 'CNT_CREDIT_PROLONG', 'AMT_CRE
DIT_SUM', 'AMT_CREDIT_SUM_DEBT', 'AMT_CREDIT_SUM_LIMIT', 'AMT_CREDIT_SUM_OVE
RDUE', 'DAYS_CREDIT_UPDATE', 'AMT_ANNUITY', 'MONTHS_BALANCE_count', 'STATUS_
1_median', 'STATUS_1_mean', 'STATUS_1_var', 'STATUS_4_median', 'STATUS_4_mea
n', 'STATUS_4_var', 'STATUS_C_median', 'STATUS_C_mean', 'STATUS_C_var', 'STA
TUS_3_median', 'STATUS_3_mean', 'STATUS_3_var', 'STATUS_X_median', 'STATUS_X
_mean', 'STATUS_X_var', 'STATUS_5_median', 'STATUS_5_mean', 'STATUS_5_var',
'STATUS_2_median', 'STATUS_2_mean', 'STATUS_2_var', 'STATUS_0_median', 'STAT
US_0_mean', 'STATUS_0_var']

Numerical Statistical Summary:

         SK_ID_CURR   DAYS_CREDIT  CREDIT_DAY_OVERDUE  DAYS_CREDIT_ENDDATE
\
count  1.554283e+06  1.554283e+06        1.554283e+06         1.472900e+06
mean   2.781766e+05  1.153320e+03        7.180353e-01         1.724138e+03
std    1.029210e+05  7.931345e+02        3.417140e+01         4.470893e+03
min    1.000010e+05  0.000000e+00        0.000000e+00         0.000000e+00
25%    1.888350e+05  4.900000e+02        0.000000e+00         3.750000e+02
50%    2.780340e+05  1.003000e+03        0.000000e+00         8.460000e+02
75%    3.673420e+05  1.673000e+03        0.000000e+00         1.478000e+03
max    4.562550e+05  2.922000e+03        2.792000e+03         4.206000e+04

       DAYS_ENDDATE_FACT  AMT_CREDIT_MAX_OVERDUE  CNT_CREDIT_PROLONG  \
count       1.011510e+06            5.777730e+05        1.554283e+06
mean        1.021003e+03            3.707606e+03        6.313522e-03
std         7.162829e+02            2.084828e+05        9.518150e-02
min         0.000000e+00            0.000000e+00        0.000000e+00
25%         4.250000e+02            0.000000e+00        0.000000e+00
50%         8.910000e+02            0.000000e+00        0.000000e+00
75%         1.501000e+03            0.000000e+00        0.000000e+00
max         4.202300e+04            1.159872e+08        9.000000e+00

       AMT_CREDIT_SUM  AMT_CREDIT_SUM_DEBT  AMT_CREDIT_SUM_LIMIT  ...  \
count    1.554278e+06         1.410159e+06          1.123546e+06  ...
mean     3.438281e+05         1.296205e+05          6.235625e+03  ...
std      1.137067e+06         6.581787e+05          4.505369e+04  ...
min      0.000000e+00        -4.705600e+06         -5.864061e+05  ...
25%      4.945500e+04         0.000000e+00          0.000000e+00  ...
50%      1.214910e+05         0.000000e+00          0.000000e+00  ...
75%      2.942775e+05         2.977650e+04          0.000000e+00  ...
max      5.850000e+08         1.701000e+08          4.705600e+06  ...

        STATUS_X_var  STATUS_5_median  STATUS_5_mean   STATUS_5_var  \
count  768450.000000    774354.000000  774354.000000  768450.000000
mean        0.063504         0.000995       0.001393       0.000729
std         0.097617         0.031328       0.026312       0.011508
min         0.000000         0.000000       0.000000       0.000000
25%         0.000000         0.000000       0.000000       0.000000
50%         0.000000         0.000000       0.000000       0.000000
75%         0.102564         0.000000       0.000000       0.000000
max         0.500000         1.000000       1.000000       0.333333
```

```
       STATUS_2_median  STATUS_2_mean   STATUS_2_var  STATUS_0_median  \
count    774354.000000  774354.000000  768450.000000    774354.000000
mean          0.000009       0.000708       0.000682         0.375246
std           0.002665       0.008136       0.007312         0.477813
min           0.000000       0.000000       0.000000         0.000000
25%           0.000000       0.000000       0.000000         0.000000
50%           0.000000       0.000000       0.000000         0.000000
75%           0.000000       0.000000       0.000000         1.000000
max           1.000000       0.714286       0.333333         1.000000

       STATUS_0_mean   STATUS_0_var
count  774354.000000  768450.000000
mean        0.413294       0.133522
std         0.346153       0.103150
min         0.000000       0.000000
25%         0.103448       0.027027
50%         0.333333       0.135976
75%         0.727273       0.229167
max         1.000000       0.500000

[8 rows x 38 columns]

--------
# of Categorical features: 3
Categorical features:
['CREDIT_ACTIVE', 'CREDIT_CURRENCY', 'CREDIT_TYPE']

Categorical Statistical Summary:

       CREDIT_ACTIVE CREDIT_CURRENCY      CREDIT_TYPE
count        1554283         1554283          1554283
unique             4               4               15
top           Closed      currency_1  Consumer_credit
freq         1007961         1553026          1141522

Categories:

CREDIT_ACTIVE                        [Closed, Active, Sold, Bad_debt]
CREDIT_CURRENCY     [currency_1, currency_2, currency_4, currency_3]
CREDIT_TYPE        [Consumer_credit, Credit_card, Car_loan, Mortg...
dtype: object

--------
# of OHE categorical features: 23
OHE Categorical features: ['CREDIT_TYPE_Mobile_operator_loan', 'CREDIT_ACTIV
E_Bad_debt', 'CREDIT_CURRENCY_currency_3', 'CREDIT_CURRENCY_currency_4', 'CR
EDIT_TYPE_Real_estate_loan', 'CREDIT_TYPE_Credit_card', 'CREDIT_TYPE_Loan_fo
r_the_purchase_of_equipment', 'CREDIT_ACTIVE_Sold', 'CREDIT_CURRENCY_currenc
y_2', 'CREDIT_TYPE_Unknown_type_of_loan', 'CREDIT_TYPE_Microloan', 'CREDIT_T
YPE_Cash_loan_non_earmarked', 'CREDIT_TYPE_Car_loan', 'CREDIT_TYPE_Loan_for_
working_capital_replenishment', 'CREDIT_TYPE_Another_type_of_loan', 'CREDIT_
ACTIVE_Active', 'CREDIT_TYPE_Interbank_credit', 'CREDIT_ACTIVE_Closed', 'CRE
DIT_TYPE_Loan_for_purchase_of_shares_margin_lending', 'CREDIT_TYPE_Mortgage
', 'CREDIT_TYPE_Consumer_credit', 'CREDIT_CURRENCY_currency_1', 'CREDIT_TYPE
_Loan_for_business_development']
--------
df.shape: (1554283, 61)
```

```
Aggregated Features:
df[['STATUS_2_median', 'AMT_CREDIT_SUM_LIMIT', 'CREDIT_ACTIVE_Bad_debt', 'CR
EDIT_TYPE_Mobile_operator_loan', 'DAYS_CREDIT_UPDATE', 'STATUS_4_median', 'S
TATUS_4_var', 'STATUS_5_median', 'STATUS_C_mean', 'STATUS_5_var', 'STATUS_3_
var', 'CREDIT_CURRENCY_currency_3', 'CREDIT_CURRENCY_currency_4', 'STATUS_2_
var', 'CREDIT_TYPE_Real_estate_loan', 'CREDIT_TYPE_Credit_card', 'CREDIT_TYP
E_Loan_for_the_purchase_of_equipment', 'AMT_ANNUITY', 'STATUS_C_median', 'CR
EDIT_DAY_OVERDUE', 'CREDIT_ACTIVE_Sold', 'STATUS_0_var', 'MONTHS_BALANCE_cou
nt', 'STATUS_1_mean', 'CREDIT_CURRENCY_currency_2', 'DAYS_CREDIT_ENDDATE', '
STATUS_3_mean', 'DAYS_ENDDATE_FACT', 'CREDIT_TYPE_Unknown_type_of_loan', 'CR
EDIT_TYPE_Microloan', 'STATUS_0_median', 'CREDIT_TYPE_Cash_loan_non_earmarke
d', 'STATUS_1_var', 'STATUS_X_var', 'CREDIT_TYPE_Car_loan', 'STATUS_2_mean',
'CREDIT_TYPE_Loan_for_working_capital_replenishment', 'AMT_CREDIT_SUM_OVERDU
E', 'CREDIT_TYPE_Another_type_of_loan', 'STATUS_0_mean', 'CREDIT_ACTIVE_Acti
ve', 'STATUS_5_mean', 'CREDIT_TYPE_Interbank_credit', 'AMT_CREDIT_SUM', 'DAY
S_CREDIT', 'STATUS_X_median', 'STATUS_X_mean', 'CREDIT_ACTIVE_Closed', 'CRED
IT_TYPE_Loan_for_purchase_of_shares_margin_lending', 'CNT_CREDIT_PROLONG', '
CREDIT_TYPE_Mortgage', 'STATUS_1_median', 'STATUS_C_var', 'STATUS_3_median',
'STATUS_4_mean', 'CREDIT_TYPE_Consumer_credit', 'CREDIT_CURRENCY_currency_1
', 'AMT_CREDIT_SUM_DEBT', 'CREDIT_TYPE_Loan_for_business_development', 'AMT_
CREDIT_MAX_OVERDUE']][0:5]:
   STATUS_2_median  AMT_CREDIT_SUM_LIMIT  CREDIT_ACTIVE_Bad_debt  \
0              NaN                   NaN                       0
5              NaN             108982.62                       0
6              NaN                  0.00                       0
7              NaN                  0.00                       0
8              NaN                  0.00                       0

   CREDIT_TYPE_Mobile_operator_loan  DAYS_CREDIT_UPDATE  STATUS_4_median  \
0                                 0                 131              NaN
5                                 0                  31              NaN
6                                 0                  22              NaN
7                                 0                1710              NaN
8                                 0                 840              NaN

   STATUS_4_var  STATUS_5_median  STATUS_C_mean  STATUS_5_var  ...  \
0           NaN              NaN            NaN           NaN  ...
5           NaN              NaN            NaN           NaN  ...
6           NaN              NaN            NaN           NaN  ...
7           NaN              NaN            NaN           NaN  ...
8           NaN              NaN            NaN           NaN  ...

   CREDIT_TYPE_Mortgage  STATUS_1_median  STATUS_C_var  STATUS_3_median  \
0                     0              NaN           NaN              NaN
5                     0              NaN           NaN              NaN
6                     0              NaN           NaN              NaN
7                     0              NaN           NaN              NaN
8                     0              NaN           NaN              NaN

   STATUS_4_mean  CREDIT_TYPE_Consumer_credit  CREDIT_CURRENCY_currency_1  \
0            NaN                            1                           1
5            NaN                            0                           1
6            NaN                            1                           1
7            NaN                            1                           1
8            NaN                            1                           1

   AMT_CREDIT_SUM_DEBT  CREDIT_TYPE_Loan_for_business_development  \
0                 0.00                                          0
5             71017.38                                          0
```

```
6             42103.80                                          0
7                 0.00                                          0
8                 0.00                                          0

   AMT_CREDIT_MAX_OVERDUE
0                     NaN
5                     0.0
6                     0.0
7                 14985.0
8                     0.0

[5 rows x 60 columns]
--------
Aggregated Features:
AMT_ANNUITY_max
AMT_ANNUITY_mean
AMT_ANNUITY_median
AMT_ANNUITY_min
AMT_ANNUITY_sum
AMT_ANNUITY_var
AMT_CREDIT_MAX_OVERDUE_max
AMT_CREDIT_MAX_OVERDUE_mean
AMT_CREDIT_MAX_OVERDUE_median
AMT_CREDIT_MAX_OVERDUE_min
AMT_CREDIT_MAX_OVERDUE_sum
AMT_CREDIT_MAX_OVERDUE_var
AMT_CREDIT_SUM_DEBT_max
AMT_CREDIT_SUM_DEBT_mean
AMT_CREDIT_SUM_DEBT_median
AMT_CREDIT_SUM_DEBT_min
AMT_CREDIT_SUM_DEBT_sum
AMT_CREDIT_SUM_DEBT_var
AMT_CREDIT_SUM_LIMIT_max
AMT_CREDIT_SUM_LIMIT_mean
AMT_CREDIT_SUM_LIMIT_median
AMT_CREDIT_SUM_LIMIT_min
AMT_CREDIT_SUM_LIMIT_sum
AMT_CREDIT_SUM_LIMIT_var
AMT_CREDIT_SUM_OVERDUE_max
AMT_CREDIT_SUM_OVERDUE_mean
AMT_CREDIT_SUM_OVERDUE_median
AMT_CREDIT_SUM_OVERDUE_min
AMT_CREDIT_SUM_OVERDUE_sum
AMT_CREDIT_SUM_OVERDUE_var
AMT_CREDIT_SUM_max
AMT_CREDIT_SUM_mean
AMT_CREDIT_SUM_median
AMT_CREDIT_SUM_min
AMT_CREDIT_SUM_sum
AMT_CREDIT_SUM_var
CNT_CREDIT_PROLONG_max
CNT_CREDIT_PROLONG_mean
CNT_CREDIT_PROLONG_median
CNT_CREDIT_PROLONG_min
CNT_CREDIT_PROLONG_sum
CNT_CREDIT_PROLONG_var
CREDIT_ACTIVE_Active_mean
CREDIT_ACTIVE_Active_median
CREDIT_ACTIVE_Active_var
```

CREDIT_ACTIVE_Bad_debt_mean
CREDIT_ACTIVE_Bad_debt_median
CREDIT_ACTIVE_Bad_debt_var
CREDIT_ACTIVE_Closed_mean
CREDIT_ACTIVE_Closed_median
CREDIT_ACTIVE_Closed_var
CREDIT_ACTIVE_Sold_mean
CREDIT_ACTIVE_Sold_median
CREDIT_ACTIVE_Sold_var
CREDIT_CURRENCY_currency_1_mean
CREDIT_CURRENCY_currency_1_median
CREDIT_CURRENCY_currency_1_var
CREDIT_CURRENCY_currency_2_mean
CREDIT_CURRENCY_currency_2_median
CREDIT_CURRENCY_currency_2_var
CREDIT_CURRENCY_currency_3_mean
CREDIT_CURRENCY_currency_3_median
CREDIT_CURRENCY_currency_3_var
CREDIT_CURRENCY_currency_4_mean
CREDIT_CURRENCY_currency_4_median
CREDIT_CURRENCY_currency_4_var
CREDIT_DAY_OVERDUE_max
CREDIT_DAY_OVERDUE_mean
CREDIT_DAY_OVERDUE_median
CREDIT_DAY_OVERDUE_min
CREDIT_DAY_OVERDUE_sum
CREDIT_DAY_OVERDUE_var
CREDIT_TYPE_Another_type_of_loan_mean
CREDIT_TYPE_Another_type_of_loan_median
CREDIT_TYPE_Another_type_of_loan_var
CREDIT_TYPE_Car_loan_mean
CREDIT_TYPE_Car_loan_median
CREDIT_TYPE_Car_loan_var
CREDIT_TYPE_Cash_loan_non_earmarked_mean
CREDIT_TYPE_Cash_loan_non_earmarked_median
CREDIT_TYPE_Cash_loan_non_earmarked_var
CREDIT_TYPE_Consumer_credit_mean
CREDIT_TYPE_Consumer_credit_median
CREDIT_TYPE_Consumer_credit_var
CREDIT_TYPE_Credit_card_mean
CREDIT_TYPE_Credit_card_median
CREDIT_TYPE_Credit_card_var
CREDIT_TYPE_Interbank_credit_mean
CREDIT_TYPE_Interbank_credit_median
CREDIT_TYPE_Interbank_credit_var
CREDIT_TYPE_Loan_for_business_development_mean
CREDIT_TYPE_Loan_for_business_development_median
CREDIT_TYPE_Loan_for_business_development_var
CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean
CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_median
CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_var
CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean
CREDIT_TYPE_Loan_for_the_purchase_of_equipment_median
CREDIT_TYPE_Loan_for_the_purchase_of_equipment_var
CREDIT_TYPE_Loan_for_working_capital_replenishment_mean
CREDIT_TYPE_Loan_for_working_capital_replenishment_median
CREDIT_TYPE_Loan_for_working_capital_replenishment_var
CREDIT_TYPE_Microloan_mean
CREDIT_TYPE_Microloan_median

CREDIT_TYPE_Microloan_var
CREDIT_TYPE_Mobile_operator_loan_mean
CREDIT_TYPE_Mobile_operator_loan_median
CREDIT_TYPE_Mobile_operator_loan_var
CREDIT_TYPE_Mortgage_mean
CREDIT_TYPE_Mortgage_median
CREDIT_TYPE_Mortgage_var
CREDIT_TYPE_Real_estate_loan_mean
CREDIT_TYPE_Real_estate_loan_median
CREDIT_TYPE_Real_estate_loan_var
CREDIT_TYPE_Unknown_type_of_loan_mean
CREDIT_TYPE_Unknown_type_of_loan_median
CREDIT_TYPE_Unknown_type_of_loan_var
DAYS_CREDIT_ENDDATE_count
DAYS_CREDIT_ENDDATE_max
DAYS_CREDIT_ENDDATE_mean
DAYS_CREDIT_ENDDATE_median
DAYS_CREDIT_ENDDATE_min
DAYS_CREDIT_ENDDATE_sum
DAYS_CREDIT_ENDDATE_var
DAYS_CREDIT_UPDATE_count
DAYS_CREDIT_UPDATE_max
DAYS_CREDIT_UPDATE_mean
DAYS_CREDIT_UPDATE_median
DAYS_CREDIT_UPDATE_min
DAYS_CREDIT_UPDATE_sum
DAYS_CREDIT_UPDATE_var
DAYS_CREDIT_count
DAYS_CREDIT_max
DAYS_CREDIT_mean
DAYS_CREDIT_median
DAYS_CREDIT_min
DAYS_CREDIT_sum
DAYS_CREDIT_var
DAYS_ENDDATE_FACT_max
DAYS_ENDDATE_FACT_mean
DAYS_ENDDATE_FACT_median
DAYS_ENDDATE_FACT_min
DAYS_ENDDATE_FACT_sum
DAYS_ENDDATE_FACT_var
MONTHS_BALANCE_count_max
MONTHS_BALANCE_count_mean
MONTHS_BALANCE_count_median
MONTHS_BALANCE_count_min
MONTHS_BALANCE_count_sum
MONTHS_BALANCE_count_var
STATUS_0_mean_max
STATUS_0_mean_mean
STATUS_0_mean_median
STATUS_0_mean_min
STATUS_0_mean_sum
STATUS_0_mean_var
STATUS_0_median_max
STATUS_0_median_mean
STATUS_0_median_median
STATUS_0_median_min
STATUS_0_median_sum
STATUS_0_median_var
STATUS_0_var_max

STATUS_0_var_mean
STATUS_0_var_median
STATUS_0_var_min
STATUS_0_var_sum
STATUS_0_var_var
STATUS_1_mean_max
STATUS_1_mean_mean
STATUS_1_mean_median
STATUS_1_mean_min
STATUS_1_mean_sum
STATUS_1_mean_var
STATUS_1_median_max
STATUS_1_median_mean
STATUS_1_median_median
STATUS_1_median_min
STATUS_1_median_sum
STATUS_1_median_var
STATUS_1_var_max
STATUS_1_var_mean
STATUS_1_var_median
STATUS_1_var_min
STATUS_1_var_sum
STATUS_1_var_var
STATUS_2_mean_max
STATUS_2_mean_mean
STATUS_2_mean_median
STATUS_2_mean_min
STATUS_2_mean_sum
STATUS_2_mean_var
STATUS_2_median_max
STATUS_2_median_mean
STATUS_2_median_median
STATUS_2_median_min
STATUS_2_median_sum
STATUS_2_median_var
STATUS_2_var_max
STATUS_2_var_mean
STATUS_2_var_median
STATUS_2_var_min
STATUS_2_var_sum
STATUS_2_var_var
STATUS_3_mean_max
STATUS_3_mean_mean
STATUS_3_mean_median
STATUS_3_mean_min
STATUS_3_mean_sum
STATUS_3_mean_var
STATUS_3_median_max
STATUS_3_median_mean
STATUS_3_median_median
STATUS_3_median_min
STATUS_3_median_sum
STATUS_3_median_var
STATUS_3_var_max
STATUS_3_var_mean
STATUS_3_var_median
STATUS_3_var_min
STATUS_3_var_sum
STATUS_3_var_var

STATUS_4_mean_max
STATUS_4_mean_mean
STATUS_4_mean_median
STATUS_4_mean_min
STATUS_4_mean_sum
STATUS_4_mean_var
STATUS_4_median_max
STATUS_4_median_mean
STATUS_4_median_median
STATUS_4_median_min
STATUS_4_median_sum
STATUS_4_median_var
STATUS_4_var_max
STATUS_4_var_mean
STATUS_4_var_median
STATUS_4_var_min
STATUS_4_var_sum
STATUS_4_var_var
STATUS_5_mean_max
STATUS_5_mean_mean
STATUS_5_mean_median
STATUS_5_mean_min
STATUS_5_mean_sum
STATUS_5_mean_var
STATUS_5_median_max
STATUS_5_median_mean
STATUS_5_median_median
STATUS_5_median_min
STATUS_5_median_sum
STATUS_5_median_var
STATUS_5_var_max
STATUS_5_var_mean
STATUS_5_var_median
STATUS_5_var_min
STATUS_5_var_sum
STATUS_5_var_var
STATUS_C_mean_max
STATUS_C_mean_mean
STATUS_C_mean_median
STATUS_C_mean_min
STATUS_C_mean_sum
STATUS_C_mean_var
STATUS_C_median_max
STATUS_C_median_mean
STATUS_C_median_median
STATUS_C_median_min
STATUS_C_median_sum
STATUS_C_median_var
STATUS_C_var_max
STATUS_C_var_mean
STATUS_C_var_median
STATUS_C_var_min
STATUS_C_var_sum
STATUS_C_var_var
STATUS_X_mean_max
STATUS_X_mean_mean
STATUS_X_mean_median
STATUS_X_mean_min
STATUS_X_mean_sum

```
STATUS_X_mean_var
STATUS_X_median_max
STATUS_X_median_mean
STATUS_X_median_median
STATUS_X_median_min
STATUS_X_median_sum
STATUS_X_median_var
STATUS_X_var_max
STATUS_X_var_mean
STATUS_X_var_median
STATUS_X_var_min
STATUS_X_var_sum
STATUS_X_var_var
index

Aggregated Data:

                                 count         mean            std  min  \
AMT_CREDIT_MAX_OVERDUE_max    210884.0  8698.453793  344794.250667  0.0
STATUS_5_mean_var             116902.0     0.000540       0.007895  0.0
STATUS_5_mean_max             134542.0     0.005455       0.052524  0.0
STATUS_X_mean_max             134542.0     0.486669       0.406290  0.0
CREDIT_TYPE_Microloan_median  299803.0     0.002510       0.047357  0.0
...                                ...          ...            ...  ...
AMT_ANNUITY_min               118215.0  6240.906051   88732.428655  0.0
STATUS_5_var_var              116504.0     0.000107       0.001161  0.0
STATUS_2_mean_var             116902.0     0.000058       0.000906  0.0
AMT_CREDIT_SUM_OVERDUE_max    299803.0   128.489555   11057.581175  0.0
STATUS_1_var_median           134253.0     0.007160       0.028318  0.0

                                 25%       50%        75%           max
AMT_CREDIT_MAX_OVERDUE_max    0.0000  0.000000  5445.0000  1.159872e+08
STATUS_5_mean_var             0.0000  0.000000     0.0000  4.026956e-01
STATUS_5_mean_max             0.0000  0.000000     0.0000  1.000000e+00
STATUS_X_mean_max             0.0625  0.444444     1.0000  1.000000e+00
CREDIT_TYPE_Microloan_median  0.0000  0.000000     0.0000  1.000000e+00
...                              ...       ...        ...           ...
AMT_ANNUITY_min               0.0000  0.000000  4221.3825  2.401604e+07
STATUS_5_var_var              0.0000  0.000000     0.0000  3.445313e-02
STATUS_2_mean_var             0.0000  0.000000     0.0000  1.214772e-01
AMT_CREDIT_SUM_OVERDUE_max    0.0000  0.000000     0.0000  3.756681e+06
STATUS_1_var_median           0.0000  0.000000     0.0000  5.000000e-01

[295 rows x 8 columns]
```

Collinearity Analysis

The collinearity of variable pairs are compared via correlation and can be iteratively dropped from combined dataset based on which variables are least correlated to the target variable. Below is an example of the collinearity analysis. This step will be added to the pre-processing pipeline as a method of feature selection in future phases.

In [10]:

```
def colinearityReducer(dataframe, threshold=0.5):

    '''
    This function explores the correlation between each variable pair and tl
    Of the var1iable pairs with absolute correlations above the threshold va
    ...the variable with the lowest target variable correlation is dropped t
    The process is repeated until there are no more colinear pairs with abs

    NOTE! The function receives a dataframe structured with the target varia
    '''

    print('-------------------------------------')
    print('BEGIN COLINEAR FEATURE REDUCTION')
    print('-------------------------------------')

    i = 1
    dropped_variables = list()
    while True:

        # read-in and assign columns
        # gets correlation matrix between variables and pivots to a longer
        # identify target variable
        # drop same-name and target correlations

        print('-------------------------------------')
        print(f"Colinearity Reduction Iteration {i}\n")

        df = dataframe
        features = df.iloc[:,1:].columns
        target_name = df.iloc[:,0].name

        print('')
        print(f'Dataframe Features ({len(features)}):')
        print(features)

        df = pd.melt(abs(df.corr()).reset_index(), id_vars='index', value_va
        targets = df[df['index']==target_name]
        df = df[(df['index'] != df['variable']) & (df['index'] != target_nar

        # combine the correlated variables into ordered string
        # aggregate the max correlation and sort pairs
        # split out the variables from the original string
        # join the target variable correlations for each variable pair, rena

        df['joined'] = df[['index', 'variable']].apply(lambda row: '::'.joi

        df = df.groupby('joined', as_index=False) \
               .agg({'value':'max'}) \
               .sort_values(by='value', ascending=False)

        df[['var_1','var_2']] = df['joined'].str.split("::",expand=True)

        df = df.merge(targets, how='left', left_on='var_1', right_on='varial
               .merge(targets, how='left', left_on='var_2', right_on='varial
        df.rename(columns = {'value_x':'var_pair_corr', 'value_y':'var_1_ta

        # This section takes all variable pairs with a correlation greater
        # test to determine which has a higher correlation with the target.
```

```
        # The higher of the two gets marked as a win
        # While the other gets marked as a loss
        # the wins and losses for each variable are then grouped and summed

        exceeds = df[df['var_pair_corr']>threshold]

        # break if none above threshold
        if len(exceeds['var_pair_corr'])==0:
            print('-------------------------------------')
            print(f"NO VARIABLE PAIRS WITH CORRELATION > {threshold}")
            break

        exceeds['var_1_win'] = exceeds.apply(lambda row: 1 if row["var_1_ta
        exceeds['var_1_loss'] = exceeds.apply(lambda row: 1 if row["var_2_t
        exceeds['var_2_win'] = exceeds.apply(lambda row: 1 if row["var_1_ta
        exceeds['var_2_loss'] = exceeds.apply(lambda row: 1 if row["var_2_t

        var1 = exceeds[['var_1', 'var_1_win', 'var_1_loss']].groupby('var_1
                                                              .agg({'var_1_wi
        var1.rename(columns = {'var_1':'var', 'var_1_win':'win', 'var_1_los

        var2 = exceeds[['var_2', 'var_2_win', 'var_2_loss']].groupby('var_2
                                                              .agg({'var_2_wi
        var2.rename(columns = {'var_2':'var', 'var_2_win':'win', 'var_2_los

        corrcomps = pd.concat([var1,var2], axis=0).groupby('var', as_index=
                                                  .agg({'win':'sum', 'loss'

        # drop variables which had 0 wins - IE collinear variables which we
        dropvars = corrcomps[corrcomps['win']==0]['var']

        dropped_variables.extend(list(dropvars))

        dropvarsummary = targets[targets['variable'].isin(dropvars)].iloc[:
        dropvarsummary.rename(columns={'variable':'Dropped Variable', 'valu

        print('')
        print('Dropped Variables:')
        print(dropvarsummary)
        # print('-------------------------------------')
        # print('Exceedances:')
        # print(exceeds)

        dataframe = dataframe.drop(dropvars, axis=1)

        i += 1

    print('-------------------------------------')
    print('Final Dropped Variable List:')
    print(dropped_variables)
    print('-------------------------------------')
    print('END COLINEAR FEATURE REDUCTION')
    print('-------------------------------------')

    return dataframe


# testing
df = pd.concat([y_train,X_train[X_feat_num]],axis=1)
```

```
colinearityReducer(df, 0.5)
```

```
-------------------------------------
BEGIN COLINEAR FEATURE REDUCTION
-------------------------------------
-------------------------------------
Colinearity Reduction Iteration 1


Dataframe Features (205):
Index(['AMT_CREDIT_SUM_sum', 'FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean',
       'STATUS_2_mean_sum', 'AMT_CREDIT_SUM_max', 'APARTMENTS_MODE',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       ...
       'CREDIT_CURRENCY_currency_4_mean', 'DAYS_ENDDATE_FACT_max',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'BASEMENTAREA_MEDI
',
       'DAYS_CREDIT_max', 'LIVINGAREA_MODE', 'DAYS_REGISTRATION',
       'LIVE_CITY_NOT_WORK_CITY', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object', length=205)

Dropped Variables:
                       Dropped Variable  Target Variable Correlation
5150                     DAYS_CREDIT_var                     0.037743
5974             CREDIT_DAY_OVERDUE_min                     0.005965
6386       YEARS_BEGINEXPLUATATION_MODE                     0.006449
6592                      LANDAREA_MODE                     0.008926
8034                     FLOORSMIN_MODE                     0.033014
8858          DAYS_CREDIT_ENDDATE_count                     0.000834
9476           AMT_CREDIT_SUM_LIMIT_min                     0.005163
10506          DEF_30_CNT_SOCIAL_CIRCLE                     0.032452
12360                   COMMONAREA_MODE                     0.015697
13184           DAYS_CREDIT_ENDDATE_var                     0.035183
13802                 STATUS_C_mean_sum                     0.007139
14008                AMT_CREDIT_SUM_var                     0.001490
15038          OBS_60_CNT_SOCIAL_CIRCLE                     0.010328
17510            DAYS_CREDIT_UPDATE_var                     0.003128
19158          AMT_CREDIT_SUM_LIMIT_var                     0.007225
19364                   YEARS_BUILD_MODE                     0.021269
19776   CREDIT_CURRENCY_currency_1_mean                     0.005527
19982        AMT_CREDIT_SUM_OVERDUE_var                     0.000257
20600                   FLAG_DOCUMENT_6                     0.028109
20806        AMT_CREDIT_SUM_OVERDUE_min                     0.000337
21630          NONLIVINGAPARTMENTS_MODE                     0.000117
23072        REGION_POPULATION_RELATIVE                     0.037599
24720       LIVE_REGION_NOT_WORK_REGION                     0.001962
25338            CREDIT_DAY_OVERDUE_var                     0.002149
28840                NONLIVINGAREA_MODE                     0.011677
29046        AMT_CREDIT_MAX_OVERDUE_var                     0.001026
29664            CNT_CREDIT_PROLONG_var                     0.000288
30076                 STATUS_X_mean_sum                     0.011984
31930                 STATUS_3_mean_sum                     0.009778
32754           DAYS_CREDIT_ENDDATE_min                     0.030961
33578                   CNT_FAM_MEMBERS                     0.008963
35844        AMT_CREDIT_SUM_DEBT_median                     0.000126
38522                       AMT_ANNUITY                     0.013680
```

```
39346  CREDIT_TYPE_Consumer_credit_mean                    0.024842
40376             DAYS_ENDDATE_FACT_max                    0.019857
41818           LIVE_CITY_NOT_WORK_CITY                    0.032414
------------------------------------
Colinearity Reduction Iteration 2


Dataframe Features (169):
Index(['AMT_CREDIT_SUM_sum', 'FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean',
       'STATUS_2_mean_sum', 'AMT_CREDIT_SUM_max', 'APARTMENTS_MODE',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       ...
       'ENTRANCES_MEDI', 'YEARS_BEGINEXPLUATATION_MEDI',
       'CREDIT_CURRENCY_currency_4_mean', 'CREDIT_ACTIVE_Bad_debt_mean',
       'FLAG_DOCUMENT_12', 'BASEMENTAREA_MEDI', 'DAYS_CREDIT_max',
       'LIVINGAREA_MODE', 'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean
'],
      dtype='object', length=169)

Dropped Variables:
                    Dropped Variable  Target Variable Correlation
2210               DAYS_CREDIT_count                     0.007057
3570       CREDIT_DAY_OVERDUE_median                     0.007210
3910         AMT_CREDIT_SUM_DEBT_min                     0.000812
4250                      AMT_CREDIT                     0.030522
4590   AMT_CREDIT_MAX_OVERDUE_median                     0.001604
4760          CREDIT_DAY_OVERDUE_max                     0.006302
6290        NONLIVINGAPARTMENTS_MEDI                     0.001632
7480          DAYS_CREDIT_UPDATE_max                     0.028840
8330                 YEARS_BUILD_AVG                     0.021578
10030       DAYS_ENDDATE_FACT_median                     0.050114
10200             NONLIVINGAREA_MEDI                     0.012193
11730    AMT_CREDIT_SUM_LIMIT_median                     0.007816
13260        DAYS_CREDIT_ENDDATE_max                     0.035530
16660                   LANDAREA_AVG                     0.009738
16830                COMMONAREA_MEDI                     0.018059
17510        CNT_CREDIT_PROLONG_mean                     0.001243
18530                 FLOORSMIN_MEDI                     0.033841
19040     DAYS_CREDIT_ENDDATE_median                     0.036222
19550                  DAYS_EMPLOYED                     0.045288
20230  AMT_CREDIT_SUM_OVERDUE_median                     0.003511
21590                 ENTRANCES_MODE                     0.016987
22440       MONTHS_BALANCE_count_sum                     0.012209
24140       AMT_CREDIT_SUM_LIMIT_sum                     0.009171
24650           REGION_RATING_CLIENT                     0.058802
26180    YEARS_BEGINEXPLUATATION_AVG                     0.007446
------------------------------------
Colinearity Reduction Iteration 3


Dataframe Features (144):
Index(['AMT_CREDIT_SUM_sum', 'FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean',
       'STATUS_2_mean_sum', 'AMT_CREDIT_SUM_max', 'APARTMENTS_MODE',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       ...
```

```
       'ENTRANCES_MEDI', 'YEARS_BEGINEXPLUATATION_MEDI',
       'CREDIT_CURRENCY_currency_4_mean', 'CREDIT_ACTIVE_Bad_debt_mean',
       'FLAG_DOCUMENT_12', 'BASEMENTAREA_MEDI', 'DAYS_CREDIT_max',
       'LIVINGAREA_MODE', 'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean
'],
      dtype='object', length=144)

Dropped Variables:
                  Dropped Variable  Target Variable Correlation
2030      DAYS_CREDIT_ENDDATE_mean                     0.043757
4060              YEARS_BUILD_MEDI                     0.021809
5510   AMT_CREDIT_SUM_OVERDUE_mean                     0.004682
8700        CREDIT_DAY_OVERDUE_sum                     0.007202
9280                FLAG_EMP_PHONE                     0.046327
10295                LANDAREA_MEDI                     0.009983
10875            BASEMENTAREA_MODE                     0.018166
11455  AMT_CREDIT_MAX_OVERDUE_mean                     0.001737
12905     DAYS_CREDIT_UPDATE_count                     0.007057
13050     AMT_CREDIT_SUM_LIMIT_max                     0.010282
13630                FLOORSMIN_AVG                     0.034047
13920     AMT_CREDIT_SUM_DEBT_mean                     0.002173
15225       CNT_CREDIT_PROLONG_max                     0.004104
15515    CNT_CREDIT_PROLONG_median                     0.001351
18705               COMMONAREA_AVG                     0.018134
20300              DAYS_CREDIT_max                     0.050482
------------------------------------
Colinearity Reduction Iteration 4


Dataframe Features (128):
Index(['AMT_CREDIT_SUM_sum', 'FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean',
       'STATUS_2_mean_sum', 'AMT_CREDIT_SUM_max', 'APARTMENTS_MODE',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       ...
       'FLAG_DOCUMENT_15', 'ENTRANCES_MEDI', 'YEARS_BEGINEXPLUATATION_MEDI',
       'CREDIT_CURRENCY_currency_4_mean', 'CREDIT_ACTIVE_Bad_debt_mean',
       'FLAG_DOCUMENT_12', 'BASEMENTAREA_MEDI', 'LIVINGAREA_MODE',
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object', length=128)

Dropped Variables:
                  Dropped Variable  Target Variable Correlation
4128       DAYS_CREDIT_UPDATE_sum                     0.038471
5031      AMT_CREDIT_SUM_DEBT_max                     0.003124
7611                   DAYS_BIRTH                     0.078680
8256           AMT_CREDIT_SUM_min                     0.011952
10191  AMT_CREDIT_SUM_OVERDUE_max                     0.007134
12771  AMT_CREDIT_MAX_OVERDUE_min                     0.001766
15351              ENTRANCES_MEDI                     0.018663
------------------------------------
Colinearity Reduction Iteration 5


Dataframe Features (121):
Index(['AMT_CREDIT_SUM_sum', 'FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean',
       'STATUS_2_mean_sum', 'AMT_CREDIT_SUM_max', 'APARTMENTS_MODE',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
```

```
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       ...
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'BASEMENTAREA_MEDI
',
       'LIVINGAREA_MODE', 'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean
'],
      dtype='object', length=121)

Dropped Variables:
                 Dropped Variable  Target Variable Correlation
5002      AMT_CREDIT_SUM_DEBT_var                     0.005692
6954              DAYS_CREDIT_sum                     0.038594
7320                ENTRANCES_AVG                     0.019075
12078  AMT_CREDIT_MAX_OVERDUE_max                     0.001781
13054       AMT_CREDIT_SUM_median                     0.016208
------------------------------------
Colinearity Reduction Iteration 6


Dataframe Features (116):
Index(['AMT_CREDIT_SUM_sum', 'FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean',
       'STATUS_2_mean_sum', 'AMT_CREDIT_SUM_max', 'APARTMENTS_MODE',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       ...
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'BASEMENTAREA_MEDI
',
       'LIVINGAREA_MODE', 'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean
'],
      dtype='object', length=116)

Dropped Variables:
              Dropped Variable  Target Variable Correlation
1287    DAYS_CREDIT_UPDATE_min                     0.041172
10179  AMT_CREDIT_SUM_DEBT_sum                     0.006360
13104        BASEMENTAREA_MEDI                     0.019898
------------------------------------
Colinearity Reduction Iteration 7


Dataframe Features (113):
Index(['AMT_CREDIT_SUM_sum', 'FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean',
       'STATUS_2_mean_sum', 'AMT_CREDIT_SUM_max', 'APARTMENTS_MODE',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       ...
       'LIVINGAPARTMENTS_AVG', 'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_1
5',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'LIVINGAREA_MODE',
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object', length=113)
```

```
Dropped Variables:
            Dropped Variable  Target Variable Correlation
0         AMT_CREDIT_SUM_sum                     0.013066
6498  DAYS_ENDDATE_FACT_mean                     0.052414
6840        BASEMENTAREA_AVG                     0.020816
8550   DAYS_ENDDATE_FACT_sum                     0.045856
------------------------------------
Colinearity Reduction Iteration 8


Dataframe Features (109):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'AMT_CREDIT_SUM_max', 'APARTMENTS_MODE', 'DAYS_CREDIT_ENDDATE_sum',
       'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE',
       ...
       'LIVINGAPARTMENTS_AVG', 'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_1
5',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'LIVINGAREA_MODE',
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object', length=109)

Dropped Variables:
               Dropped Variable  Target Variable Correlation
330          AMT_CREDIT_SUM_max                     0.020145
1870  DAYS_CREDIT_UPDATE_median                     0.066350
6160      LIVINGAPARTMENTS_MODE                     0.022107
7370      DAYS_ENDDATE_FACT_min                     0.053906
------------------------------------
Colinearity Reduction Iteration 9


Dataframe Features (105):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'APARTMENTS_MODE', 'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9',
       'DAYS_ENDDATE_FACT_var',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum',
       ...
       'LIVINGAPARTMENTS_AVG', 'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_1
5',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'LIVINGAREA_MODE',
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object', length=105)

Dropped Variables:
             Dropped Variable  Target Variable Correlation
1060  DAYS_CREDIT_UPDATE_mean                     0.067478
1802    LIVINGAPARTMENTS_MEDI                     0.023553
6784      AMT_CREDIT_SUM_mean                     0.020684
------------------------------------
Colinearity Reduction Iteration 10


Dataframe Features (102):
```

```
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'APARTMENTS_MODE', 'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9',
       'DAYS_ENDDATE_FACT_var',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum',
       ...
       'LIVINGAPARTMENTS_AVG', 'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_1
5',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'LIVINGAREA_MODE',
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object', length=102)

Dropped Variables:
               Dropped Variable  Target Variable Correlation
1442            DAYS_CREDIT_min                     0.073098
7416  CREDIT_ACTIVE_Active_mean                     0.075027
9476       LIVINGAPARTMENTS_AVG                     0.024151
------------------------------------
Colinearity Reduction Iteration 11


Dataframe Features (99):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'APARTMENTS_MODE', 'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9',
       'DAYS_ENDDATE_FACT_var',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'APARTMENTS_MEDI',
       'OBS_30_CNT_SOCIAL_CIRCLE', 'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20',
       'REG_REGION_NOT_LIVE_REGION', 'CREDIT_TYPE_Unknown_type_of_loan_mean
',
       'AMT_CREDIT_MAX_OVERDUE_sum', 'NONLIVINGAREA_AVG',
       'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum', 'OWN_CAR_AGE',
       'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH', 'FLAG_MOBIL',
       'REG_CITY_NOT_WORK_CITY', 'APARTMENTS_AVG', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'DAYS_CREDIT_median', 'LIVINGAREA_AVG',
       'HOUR_APPR_PROCESS_START', 'ELEVATORS_MODE', 'LIVINGAREA_MEDI',
       'STATUS_0_mean_sum', 'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
```

```
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'TOTALAREA_MODE',
       'NONLIVINGAPARTMENTS_AVG', 'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'LIVINGAREA_MODE',
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
        Dropped Variable  Target Variable Correlation
300      APARTMENTS_MODE                     0.027085
7100  DAYS_CREDIT_median                     0.083502
-------------------------------------
Colinearity Reduction Iteration 12


Dataframe Features (97):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'APARTMENTS_MEDI',
       'OBS_30_CNT_SOCIAL_CIRCLE', 'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20',
       'REG_REGION_NOT_LIVE_REGION', 'CREDIT_TYPE_Unknown_type_of_loan_mean
',
       'AMT_CREDIT_MAX_OVERDUE_sum', 'NONLIVINGAREA_AVG',
       'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum', 'OWN_CAR_AGE',
       'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH', 'FLAG_MOBIL',
       'REG_CITY_NOT_WORK_CITY', 'APARTMENTS_AVG', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'LIVINGAREA_AVG', 'HOUR_APPR_PROCESS_START
',
       'ELEVATORS_MODE', 'LIVINGAREA_MEDI', 'STATUS_0_mean_sum',
       'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
```

```
       'CREDIT_TYPE_Interbank_credit_mean', 'TOTALAREA_MODE',
       'NONLIVINGAPARTMENTS_AVG', 'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'LIVINGAREA_MODE',
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
     Dropped Variable  Target Variable Correlation
2548  APARTMENTS_MEDI                     0.029036
------------------------------------
Colinearity Reduction Iteration 13


Dataframe Features (96):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'APARTMENTS_AVG',
       'EXT_SOURCE_3', 'AMT_REQ_CREDIT_BUREAU_HOUR',
       'DEF_60_CNT_SOCIAL_CIRCLE', 'CREDIT_TYPE_Microloan_mean',
       'AMT_REQ_CREDIT_BUREAU_DAY', 'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'LIVINGAREA_AVG', 'HOUR_APPR_PROCESS_START
',
       'ELEVATORS_MODE', 'LIVINGAREA_MEDI', 'STATUS_0_mean_sum',
       'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'TOTALAREA_MODE',
       'NONLIVINGAPARTMENTS_AVG', 'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
```

```
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'LIVINGAREA_MODE',
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
     Dropped Variable  Target Variable Correlation
3977   APARTMENTS_AVG                     0.029355
------------------------------------
Colinearity Reduction Iteration 14


Dataframe Features (95):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'LIVINGAREA_AVG', 'HOUR_APPR_PROCESS_START
',
       'ELEVATORS_MODE', 'LIVINGAREA_MEDI', 'STATUS_0_mean_sum',
       'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'TOTALAREA_MODE',
       'NONLIVINGAPARTMENTS_AVG', 'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'LIVINGAREA_MODE',
```

```
       'DAYS_REGISTRATION', 'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
     Dropped Variable  Target Variable Correlation
8832  LIVINGAREA_MODE                     0.031257
------------------------------------
Colinearity Reduction Iteration 15


Dataframe Features (94):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'LIVINGAREA_AVG', 'HOUR_APPR_PROCESS_START
',
       'ELEVATORS_MODE', 'LIVINGAREA_MEDI', 'STATUS_0_mean_sum',
       'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'TOTALAREA_MODE',
       'NONLIVINGAPARTMENTS_AVG', 'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION
',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')
```

```
Dropped Variables:
     Dropped Variable  Target Variable Correlation
6650   ELEVATORS_MODE                     0.032749
------------------------------------
Colinearity Reduction Iteration 16


Dataframe Features (93):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'LIVINGAREA_AVG', 'HOUR_APPR_PROCESS_START
',
       'LIVINGAREA_MEDI', 'STATUS_0_mean_sum',
       'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'TOTALAREA_MODE',
       'NONLIVINGAPARTMENTS_AVG', 'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION
',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
```

```
     Dropped Variable  Target Variable Correlation
6580  LIVINGAREA_MEDI                     0.033231
-------------------------------------
Colinearity Reduction Iteration 17


Dataframe Features (92):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'LIVINGAREA_AVG', 'HOUR_APPR_PROCESS_START
',
       'STATUS_0_mean_sum', 'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'TOTALAREA_MODE',
       'NONLIVINGAPARTMENTS_AVG', 'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION
',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
     Dropped Variable  Target Variable Correlation
7161   TOTALAREA_MODE                     0.033242
-------------------------------------
```

```
Colinearity Reduction Iteration 18


Dataframe Features (91):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'LIVINGAREA_AVG', 'HOUR_APPR_PROCESS_START
',
       'STATUS_0_mean_sum', 'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'NONLIVINGAPARTMENTS_AVG',
       'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL', 'FLAG_DOCUMENT_21',
       'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION
',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
     Dropped Variable  Target Variable Correlation
6256   LIVINGAREA_AVG                     0.033549
-------------------------------------
Colinearity Reduction Iteration 19
```

```
Dataframe Features (90):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'ELEVATORS_MEDI',
       'CNT_CREDIT_PROLONG_sum', 'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'HOUR_APPR_PROCESS_START',
       'STATUS_0_mean_sum', 'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'NONLIVINGAPARTMENTS_AVG',
       'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL', 'FLAG_DOCUMENT_21',
       'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION
',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
    Dropped Variable  Target Variable Correlation
819   ELEVATORS_MEDI                     0.034563
------------------------------------
Colinearity Reduction Iteration 20


Dataframe Features (89):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
```

```
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'CNT_CREDIT_PROLONG_su
m',
       'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean', 'ELEVATORS_AVG
',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'HOUR_APPR_PROCESS_START',
       'STATUS_0_mean_sum', 'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'NONLIVINGAPARTMENTS_AVG',
       'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL', 'FLAG_DOCUMENT_21',
       'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION
',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
     Dropped Variable  Target Variable Correlation
5130    ELEVATORS_AVG                      0.03502
-------------------------------------
Colinearity Reduction Iteration 21


Dataframe Features (88):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'CNT_CREDIT_PROLONG_su
m',
```

```
       'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'HOUR_APPR_PROCESS_START',
       'STATUS_0_mean_sum', 'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'NONLIVINGAPARTMENTS_AVG',
       'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL', 'FLAG_DOCUMENT_21',
       'AMT_REQ_CREDIT_BUREAU_QRT', 'FLOORSMAX_MODE',
       'AMT_REQ_CREDIT_BUREAU_MON', 'FLAG_DOCUMENT_15',
       'YEARS_BEGINEXPLUATATION_MEDI', 'CREDIT_CURRENCY_currency_4_mean',
       'CREDIT_ACTIVE_Bad_debt_mean', 'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION
',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
     Dropped Variable  Target Variable Correlation
7031   FLOORSMAX_MODE                     0.042919
------------------------------------
Colinearity Reduction Iteration 22


Dataframe Features (87):
Index(['FLOORSMAX_MEDI', 'CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'CNT_CREDIT_PROLONG_su
m',
       'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
```

```
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'HOUR_APPR_PROCESS_START',
       'STATUS_0_mean_sum', 'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'NONLIVINGAPARTMENTS_AVG',
       'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL', 'FLAG_DOCUMENT_21',
       'AMT_REQ_CREDIT_BUREAU_QRT', 'AMT_REQ_CREDIT_BUREAU_MON',
       'FLAG_DOCUMENT_15', 'YEARS_BEGINEXPLUATATION_MEDI',
       'CREDIT_CURRENCY_currency_4_mean', 'CREDIT_ACTIVE_Bad_debt_mean',
       'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')

Dropped Variables:
  Dropped Variable  Target Variable Correlation
0   FLOORSMAX_MEDI                     0.043601
------------------------------------
Colinearity Reduction Iteration 23


Dataframe Features (86):
Index(['CREDIT_TYPE_Car_loan_mean', 'STATUS_2_mean_sum',
       'DAYS_CREDIT_ENDDATE_sum', 'FLAG_DOCUMENT_9', 'DAYS_ENDDATE_FACT_var
',
       'CREDIT_TYPE_Loan_for_working_capital_replenishment_mean',
       'DAYS_LAST_PHONE_CHANGE', 'STATUS_1_mean_sum', 'CNT_CREDIT_PROLONG_su
m',
       'CREDIT_TYPE_Cash_loan_non_earmarked_mean',
       'AMT_CREDIT_SUM_OVERDUE_sum', 'AMT_GOODS_PRICE',
       'REGION_RATING_CLIENT_W_CITY', 'FLAG_DOCUMENT_18', 'EXT_SOURCE_1',
       'FLAG_DOCUMENT_14', 'FLAG_DOCUMENT_7', 'FLAG_DOCUMENT_4',
       'CREDIT_DAY_OVERDUE_mean', 'CREDIT_TYPE_Mobile_operator_loan_mean',
       'REG_CITY_NOT_LIVE_CITY', 'FLOORSMAX_AVG', 'AMT_CREDIT_SUM_LIMIT_mean
',
       'CREDIT_TYPE_Mortgage_mean', 'OBS_30_CNT_SOCIAL_CIRCLE',
       'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20', 'REG_REGION_NOT_LIVE_REGION',
```

```
       'CREDIT_TYPE_Unknown_type_of_loan_mean', 'AMT_CREDIT_MAX_OVERDUE_sum
',
       'NONLIVINGAREA_AVG', 'AMT_REQ_CREDIT_BUREAU_WEEK', 'STATUS_5_mean_sum
',
       'OWN_CAR_AGE', 'FLAG_DOCUMENT_17', 'CNT_CHILDREN', 'DAYS_ID_PUBLISH',
       'FLAG_MOBIL', 'REG_CITY_NOT_WORK_CITY', 'EXT_SOURCE_3',
       'AMT_REQ_CREDIT_BUREAU_HOUR', 'DEF_60_CNT_SOCIAL_CIRCLE',
       'CREDIT_TYPE_Microloan_mean', 'AMT_REQ_CREDIT_BUREAU_DAY',
       'FLAG_DOCUMENT_5', 'DAYS_CREDIT_mean',
       'CREDIT_TYPE_Loan_for_business_development_mean', 'FLAG_EMAIL',
       'FLAG_DOCUMENT_8', 'FLAG_CONT_MOBILE', 'FLAG_DOCUMENT_11',
       'CREDIT_ACTIVE_Closed_mean', 'FLAG_WORK_PHONE',
       'CREDIT_CURRENCY_currency_3_mean', 'EXT_SOURCE_2',
       'CREDIT_TYPE_Loan_for_the_purchase_of_equipment_mean',
       'AMT_REQ_CREDIT_BUREAU_YEAR', 'STATUS_4_mean_sum', 'FLAG_PHONE',
       'FLAG_DOCUMENT_3', 'CREDIT_CURRENCY_currency_2_mean',
       'FLAG_DOCUMENT_16', 'CREDIT_ACTIVE_Sold_mean', 'FLAG_DOCUMENT_10',
       'CNT_CREDIT_PROLONG_min', 'HOUR_APPR_PROCESS_START',
       'STATUS_0_mean_sum', 'CREDIT_TYPE_Another_type_of_loan_mean',
       'CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_mean',
       'REG_REGION_NOT_WORK_REGION', 'FLAG_DOCUMENT_2',
       'CREDIT_TYPE_Real_estate_loan_mean',
       'CREDIT_TYPE_Interbank_credit_mean', 'NONLIVINGAPARTMENTS_AVG',
       'FLAG_DOCUMENT_13', 'AMT_INCOME_TOTAL', 'FLAG_DOCUMENT_21',
       'AMT_REQ_CREDIT_BUREAU_QRT', 'AMT_REQ_CREDIT_BUREAU_MON',
       'FLAG_DOCUMENT_15', 'YEARS_BEGINEXPLUATATION_MEDI',
       'CREDIT_CURRENCY_currency_4_mean', 'CREDIT_ACTIVE_Bad_debt_mean',
       'FLAG_DOCUMENT_12', 'DAYS_REGISTRATION',
       'CREDIT_TYPE_Credit_card_mean'],
      dtype='object')
------------------------------------
NO VARIABLE PAIRS WITH CORRELATION > 0.5
------------------------------------
Final Dropped Variable List:
['AMT_ANNUITY', 'AMT_CREDIT_MAX_OVERDUE_var', 'AMT_CREDIT_SUM_DEBT_median',
'AMT_CREDIT_SUM_LIMIT_min', 'AMT_CREDIT_SUM_LIMIT_var', 'AMT_CREDIT_SUM_OVER
DUE_min', 'AMT_CREDIT_SUM_OVERDUE_var', 'AMT_CREDIT_SUM_var', 'CNT_CREDIT_PR
OLONG_var', 'CNT_FAM_MEMBERS', 'COMMONAREA_MODE', 'CREDIT_CURRENCY_currency_
1_mean', 'CREDIT_DAY_OVERDUE_min', 'CREDIT_DAY_OVERDUE_var', 'CREDIT_TYPE_Co
nsumer_credit_mean', 'DAYS_CREDIT_ENDDATE_count', 'DAYS_CREDIT_ENDDATE_min',
'DAYS_CREDIT_ENDDATE_var', 'DAYS_CREDIT_UPDATE_var', 'DAYS_CREDIT_var', 'DAY
S_ENDDATE_FACT_max', 'DEF_30_CNT_SOCIAL_CIRCLE', 'FLAG_DOCUMENT_6', 'FLOORSM
IN_MODE', 'LANDAREA_MODE', 'LIVE_CITY_NOT_WORK_CITY', 'LIVE_REGION_NOT_WORK_
REGION', 'NONLIVINGAPARTMENTS_MODE', 'NONLIVINGAREA_MODE', 'OBS_60_CNT_SOCIA
L_CIRCLE', 'REGION_POPULATION_RELATIVE', 'STATUS_3_mean_sum', 'STATUS_C_mean
_sum', 'STATUS_X_mean_sum', 'YEARS_BEGINEXPLUATATION_MODE', 'YEARS_BUILD_MOD
E', 'AMT_CREDIT', 'AMT_CREDIT_MAX_OVERDUE_median', 'AMT_CREDIT_SUM_DEBT_min
', 'AMT_CREDIT_SUM_LIMIT_median', 'AMT_CREDIT_SUM_LIMIT_sum', 'AMT_CREDIT_SU
M_OVERDUE_median', 'CNT_CREDIT_PROLONG_mean', 'COMMONAREA_MEDI', 'CREDIT_DAY
_OVERDUE_max', 'CREDIT_DAY_OVERDUE_median', 'DAYS_CREDIT_ENDDATE_max', 'DAYS
_CREDIT_ENDDATE_median', 'DAYS_CREDIT_UPDATE_max', 'DAYS_CREDIT_count', 'DAY
S_EMPLOYED', 'DAYS_ENDDATE_FACT_median', 'ENTRANCES_MODE', 'FLOORSMIN_MEDI',
'LANDAREA_AVG', 'MONTHS_BALANCE_count_sum', 'NONLIVINGAPARTMENTS_MEDI', 'NON
LIVINGAREA_MEDI', 'REGION_RATING_CLIENT', 'YEARS_BEGINEXPLUATATION_AVG', 'YE
ARS_BUILD_AVG', 'AMT_CREDIT_MAX_OVERDUE_mean', 'AMT_CREDIT_SUM_DEBT_mean', '
AMT_CREDIT_SUM_LIMIT_max', 'AMT_CREDIT_SUM_OVERDUE_mean', 'BASEMENTAREA_MODE
', 'CNT_CREDIT_PROLONG_max', 'CNT_CREDIT_PROLONG_median', 'COMMONAREA_AVG',
'CREDIT_DAY_OVERDUE_sum', 'DAYS_CREDIT_ENDDATE_mean', 'DAYS_CREDIT_UPDATE_co
unt', 'DAYS_CREDIT_max', 'FLAG_EMP_PHONE', 'FLOORSMIN_AVG', 'LANDAREA_MEDI',
```

```
'YEARS_BUILD_MEDI', 'AMT_CREDIT_MAX_OVERDUE_min', 'AMT_CREDIT_SUM_DEBT_max',
'AMT_CREDIT_SUM_OVERDUE_max', 'AMT_CREDIT_SUM_min', 'DAYS_BIRTH', 'DAYS_CRED
IT_UPDATE_sum', 'ENTRANCES_MEDI', 'AMT_CREDIT_MAX_OVERDUE_max', 'AMT_CREDIT_
SUM_DEBT_var', 'AMT_CREDIT_SUM_median', 'DAYS_CREDIT_sum', 'ENTRANCES_AVG',
'AMT_CREDIT_SUM_DEBT_sum', 'BASEMENTAREA_MEDI', 'DAYS_CREDIT_UPDATE_min', 'A
MT_CREDIT_SUM_sum', 'BASEMENTAREA_AVG', 'DAYS_ENDDATE_FACT_mean', 'DAYS_ENDD
ATE_FACT_sum', 'AMT_CREDIT_SUM_max', 'DAYS_CREDIT_UPDATE_median', 'DAYS_ENDD
ATE_FACT_min', 'LIVINGAPARTMENTS_MODE', 'AMT_CREDIT_SUM_mean', 'DAYS_CREDIT_
UPDATE_mean', 'LIVINGAPARTMENTS_MEDI', 'CREDIT_ACTIVE_Active_mean', 'DAYS_CR
EDIT_min', 'LIVINGAPARTMENTS_AVG', 'APARTMENTS_MODE', 'DAYS_CREDIT_median',
'APARTMENTS_MEDI', 'APARTMENTS_AVG', 'LIVINGAREA_MODE', 'ELEVATORS_MODE', 'L
IVINGAREA_MEDI', 'TOTALAREA_MODE', 'LIVINGAREA_AVG', 'ELEVATORS_MEDI', 'ELEV
ATORS_AVG', 'FLOORSMAX_MODE', 'FLOORSMAX_MEDI']
-------------------------------------
END COLINEAR FEATURE REDUCTION
-------------------------------------
```

Out[10]:

	TARGET	CREDIT_TYPE_Car_loan_mean	STATUS_2_mean_sum	DAYS_CREDIT_ENDDATE_
224603	0	0.0	0.0	10
57616	0	0.0	0.0	-7
197819	0	0.0	0.0	-1
278856	0	NaN	NaN	
192451	0	0.0	0.0	-1
...	...	...	...	
104304	0	0.0	0.0	-8
39206	0	NaN	NaN	
282714	0	0.0	0.0	20
125190	0	0.0	0.0	-1
223041	0	NaN	NaN	

222176 rows × 87 columns

In [ ]:

POS - POS_CASH_balance

1. This function is specifically for POS table. Below are the intial pre-processing steps done before passing this table into the pipeline.
   - Create a drop list.
     - Attributes that will be dropped are added to this list and all columns will be deleted before passing the dataframe into the eda function.
   - Create new features based on analysis. 3 new features were created:
     - Percentage of installments pending.
     - Number of installments pending.
     - Days with Tolerance.
   - Take absolute of the months balance attribute. which was provided as negative values, as it is relative to application date.
   - Replace " " with "_" for OHE columns.
   - Any column or row with more than 70% of its data as null will be deleted from the dataframe, as the threshold is set to .7.
   - Once processed, store the transformBenefit of this is that the data can be passed ed csv file. directly to model for merging into application train/test table. We do not have to repeatedly perform expensive EDA/ETL/Transformation.

In [19]:

```
def pos_eda(df):
    pos = df
    drop_list_pos = []

    #Adding new features
    pos['POS_PERC_INSTL_PNDNG']=pos['CNT_INSTALMENT_FUTURE']/pos['CNT_INSTAI
    pos['POS_CNT_INSTAL_PNDNG']=pos['CNT_INSTALMENT']-pos['CNT_INSTALMENT_FU
    pos['POS_DAYS_WTHT_TOLRNC']=pos['SK_DPD']-pos['SK_DPD_DEF']
    pos['MONTHS_BALANCE'] = pos['MONTHS_BALANCE'].abs()

    #replacing " " with _ for OHE cols.
    pos['NAME_CONTRACT_STATUS']=pos['NAME_CONTRACT_STATUS'].apply(lambda x:

    # Drop elements in drop list
    threshold = 0.7

    #Dropping rows with missing value rate higher than threshold
    pos = pos.loc[pos.isnull().mean(axis=1) < threshold]

    return (eda_transformation(pos,4))
```

In [20]:

```
pos = datasets['POS_CASH_balance']
pos = pos_eda(pos)
datasets_transformed['POS_CASH_balance'] = pos
#pos.to_csv(os.getcwd() + DATA_DIR + 'pos_agg_data.csv')
```

```
--------
# of ID's: 1
 ID's:
['SK_ID_CURR']

--------
# All features: 10
All features:
['SK_DPD', 'NAME_CONTRACT_STATUS', 'SK_ID_PREV', 'MONTHS_BALANCE', 'POS_CNT_
INSTAL_PNDNG', 'POS_DAYS_WTHT_TOLRNC', 'SK_DPD_DEF', 'CNT_INSTALMENT', 'CNT_
INSTALMENT_FUTURE', 'POS_PERC_INSTL_PNDNG']

Missing data:
                      Total   Percent
POS_CNT_INSTAL_PNDNG   26184  0.261804
POS_PERC_INSTL_PNDNG   26184  0.261804
CNT_INSTALMENT_FUTURE  26087  0.260835
CNT_INSTALMENT         26071  0.260675
SK_DPD                     0  0.000000
NAME_CONTRACT_STATUS       0  0.000000
SK_ID_PREV                 0  0.000000
MONTHS_BALANCE             0  0.000000
POS_DAYS_WTHT_TOLRNC       0  0.000000
SK_DPD_DEF                 0  0.000000

--------
# of Numerical features: 10
Numerical features:
['SK_ID_PREV', 'SK_ID_CURR', 'MONTHS_BALANCE', 'CNT_INSTALMENT', 'CNT_INSTAL
MENT_FUTURE', 'SK_DPD', 'SK_DPD_DEF', 'POS_PERC_INSTL_PNDNG', 'POS_CNT_INSTA
L_PNDNG', 'POS_DAYS_WTHT_TOLRNC']

Numerical Statistical Summary:

         SK_ID_PREV    SK_ID_CURR  MONTHS_BALANCE  CNT_INSTALMENT  \
count  1.000136e+07  1.000136e+07    1.000136e+07    9.975287e+06
mean   1.903217e+06  2.784039e+05    3.501259e+01    1.708965e+01
std    5.358465e+05  1.027637e+05    2.606657e+01    1.199506e+01
min    1.000001e+06  1.000010e+05    1.000000e+00    1.000000e+00
25%    1.434405e+06  1.895500e+05    1.300000e+01    1.000000e+01
50%    1.896565e+06  2.786540e+05    2.800000e+01    1.200000e+01
75%    2.368963e+06  3.674290e+05    5.400000e+01    2.400000e+01
max    2.843499e+06  4.562550e+05    9.600000e+01    9.200000e+01

       CNT_INSTALMENT_FUTURE        SK_DPD    SK_DPD_DEF  \
count           9.975271e+06  1.000136e+07  1.000136e+07
mean            1.048384e+01  1.160693e+01  6.544684e-01
std             1.110906e+01  1.327140e+02  3.276249e+01
min             0.000000e+00  0.000000e+00  0.000000e+00
25%             3.000000e+00  0.000000e+00  0.000000e+00
50%             7.000000e+00  0.000000e+00  0.000000e+00
75%             1.400000e+01  0.000000e+00  0.000000e+00
max             8.500000e+01  4.231000e+03  3.595000e+03

       POS_PERC_INSTL_PNDNG  POS_CNT_INSTAL_PNDNG  POS_DAYS_WTHT_TOLRNC
count          9.975174e+06          9.975174e+06          1.000136e+07
mean           5.469942e-01          6.605944e+00          1.095246e+01
std            3.303088e-01          5.923767e+00          1.286431e+02
min            0.000000e+00         -5.100000e+01          0.000000e+00
```

```
25%            2.500000e-01          2.000000e+00          0.000000e+00
50%            5.833333e-01          5.000000e+00          0.000000e+00
75%            8.333333e-01          9.000000e+00          0.000000e+00
max            6.666667e+00          7.200000e+01          4.231000e+03

--------
# of Categorical features: 1
Categorical features:
['NAME_CONTRACT_STATUS']

Categorical Statistical Summary:

       NAME_CONTRACT_STATUS
count              10001358
unique                    9
top                  Active
freq                9151119

Categories:

    NAME_CONTRACT_STATUS
0                 Active
1              Completed
2                 Signed
3               Approved
4  Returned_to_the_store
5                 Demand
6               Canceled
7                    XNA
8         Amortized_debt

--------
# of OHE categorical features: 9
OHE Categorical features: ['NAME_CONTRACT_STATUS_XNA', 'NAME_CONTRACT_STATUS
_Demand', 'NAME_CONTRACT_STATUS_Approved', 'NAME_CONTRACT_STATUS_Signed', 'N
AME_CONTRACT_STATUS_Canceled', 'NAME_CONTRACT_STATUS_Active', 'NAME_CONTRACT
_STATUS_Amortized_debt', 'NAME_CONTRACT_STATUS_Returned_to_the_store', 'NAME
_CONTRACT_STATUS_Completed']
--------
df.shape: (10001358, 19)

Aggregated Features:
df[['NAME_CONTRACT_STATUS_Approved', 'NAME_CONTRACT_STATUS_Signed', 'SK_ID_P
REV', 'NAME_CONTRACT_STATUS_Canceled', 'NAME_CONTRACT_STATUS_Active', 'NAME_
CONTRACT_STATUS_Amortized_debt', 'POS_DAYS_WTHT_TOLRNC', 'SK_DPD_DEF', 'CNT_
INSTALMENT', 'CNT_INSTALMENT_FUTURE', 'NAME_CONTRACT_STATUS_XNA', 'NAME_CONT
RACT_STATUS_Demand', 'MONTHS_BALANCE', 'POS_CNT_INSTAL_PNDNG', 'NAME_CONTRAC
T_STATUS_Completed', 'NAME_CONTRACT_STATUS_Returned_to_the_store', 'SK_DPD',
'POS_PERC_INSTL_PNDNG']][0:5]:
   NAME_CONTRACT_STATUS_Approved  NAME_CONTRACT_STATUS_Signed  SK_ID_PREV  \
0                              0                            0     1803195
1                              0                            0     1715348
2                              0                            0     1784872
3                              0                            0     1903291
4                              0                            0     2341044

   NAME_CONTRACT_STATUS_Canceled  NAME_CONTRACT_STATUS_Active  \
0                              0                            1
1                              0                            1
```

```
2                                0                                1
3                                0                                1
4                                0                                1

   NAME_CONTRACT_STATUS_Amortized_debt  POS_DAYS_WTHT_TOLRNC  SK_DPD_DEF  \
0                                    0                     0           0
1                                    0                     0           0
2                                    0                     0           0
3                                    0                     0           0
4                                    0                     0           0

   CNT_INSTALMENT  CNT_INSTALMENT_FUTURE  NAME_CONTRACT_STATUS_XNA  \
0            48.0                   45.0                         0
1            36.0                   35.0                         0
2            12.0                    9.0                         0
3            48.0                   42.0                         0
4            36.0                   35.0                         0

   NAME_CONTRACT_STATUS_Demand  MONTHS_BALANCE  POS_CNT_INSTAL_PNDNG  \
0                            0              31                   3.0
1                            0              33                   1.0
2                            0              32                   3.0
3                            0              35                   6.0
4                            0              35                   1.0

   NAME_CONTRACT_STATUS_Completed  NAME_CONTRACT_STATUS_Returned_to_the_stor
e  \
0                               0
0
1                               0
0
2                               0
0
3                               0
0
4                               0
0

   SK_DPD  POS_PERC_INSTL_PNDNG
0       0              0.937500
1       0              0.972222
2       0              0.750000
3       0              0.875000
4       0              0.972222
--------
Aggregated Features:
CNT_INSTALMENT_FUTURE_count
CNT_INSTALMENT_FUTURE_max
CNT_INSTALMENT_FUTURE_mean
CNT_INSTALMENT_FUTURE_median
CNT_INSTALMENT_FUTURE_min
CNT_INSTALMENT_FUTURE_sum
CNT_INSTALMENT_FUTURE_var
CNT_INSTALMENT_count
CNT_INSTALMENT_max
CNT_INSTALMENT_mean
CNT_INSTALMENT_median
CNT_INSTALMENT_min
CNT_INSTALMENT_sum
```

CNT_INSTALMENT_var
MONTHS_BALANCE_count
MONTHS_BALANCE_max
MONTHS_BALANCE_mean
MONTHS_BALANCE_median
MONTHS_BALANCE_min
MONTHS_BALANCE_sum
MONTHS_BALANCE_var
NAME_CONTRACT_STATUS_Active_mean
NAME_CONTRACT_STATUS_Active_median
NAME_CONTRACT_STATUS_Active_var
NAME_CONTRACT_STATUS_Amortized_debt_mean
NAME_CONTRACT_STATUS_Amortized_debt_median
NAME_CONTRACT_STATUS_Amortized_debt_var
NAME_CONTRACT_STATUS_Approved_mean
NAME_CONTRACT_STATUS_Approved_median
NAME_CONTRACT_STATUS_Approved_var
NAME_CONTRACT_STATUS_Canceled_mean
NAME_CONTRACT_STATUS_Canceled_median
NAME_CONTRACT_STATUS_Canceled_var
NAME_CONTRACT_STATUS_Completed_mean
NAME_CONTRACT_STATUS_Completed_median
NAME_CONTRACT_STATUS_Completed_var
NAME_CONTRACT_STATUS_Demand_mean
NAME_CONTRACT_STATUS_Demand_median
NAME_CONTRACT_STATUS_Demand_var
NAME_CONTRACT_STATUS_Returned_to_the_store_mean
NAME_CONTRACT_STATUS_Returned_to_the_store_median
NAME_CONTRACT_STATUS_Returned_to_the_store_var
NAME_CONTRACT_STATUS_Signed_mean
NAME_CONTRACT_STATUS_Signed_median
NAME_CONTRACT_STATUS_Signed_var
NAME_CONTRACT_STATUS_XNA_mean
NAME_CONTRACT_STATUS_XNA_median
NAME_CONTRACT_STATUS_XNA_var
POS_CNT_INSTAL_PNDNG_count
POS_CNT_INSTAL_PNDNG_max
POS_CNT_INSTAL_PNDNG_mean
POS_CNT_INSTAL_PNDNG_median
POS_CNT_INSTAL_PNDNG_min
POS_CNT_INSTAL_PNDNG_sum
POS_CNT_INSTAL_PNDNG_var
POS_DAYS_WTHT_TOLRNC_count
POS_DAYS_WTHT_TOLRNC_max
POS_DAYS_WTHT_TOLRNC_mean
POS_DAYS_WTHT_TOLRNC_median
POS_DAYS_WTHT_TOLRNC_min
POS_DAYS_WTHT_TOLRNC_sum
POS_DAYS_WTHT_TOLRNC_var
POS_PERC_INSTL_PNDNG_count
POS_PERC_INSTL_PNDNG_max
POS_PERC_INSTL_PNDNG_mean
POS_PERC_INSTL_PNDNG_median
POS_PERC_INSTL_PNDNG_min
POS_PERC_INSTL_PNDNG_sum
POS_PERC_INSTL_PNDNG_var
SK_DPD_DEF_count
SK_DPD_DEF_max
SK_DPD_DEF_mean

```
SK_DPD_DEF_median
SK_DPD_DEF_min
SK_DPD_DEF_sum
SK_DPD_DEF_var
SK_DPD_count
SK_DPD_max
SK_DPD_mean
SK_DPD_median
SK_DPD_min
SK_DPD_sum
SK_DPD_var
SK_ID_PREV_count

Aggregated Data:

                                            count          mean  \
NAME_CONTRACT_STATUS_XNA_mean              337252.0  3.672319e-08
CNT_INSTALMENT_count                       337252.0  2.957814e+01
SK_DPD_DEF_min                             337252.0  9.043682e-04
NAME_CONTRACT_STATUS_Approved_var          336880.0  5.854747e-04
POS_CNT_INSTAL_PNDNG_count                 337252.0  2.957781e+01
...                                             ...           ...
NAME_CONTRACT_STATUS_Active_median         337252.0  9.952765e-01
SK_DPD_max                                 337252.0  1.529411e+01
NAME_CONTRACT_STATUS_Amortized_debt_median 337252.0  8.895425e-06
SK_DPD_DEF_max                             337252.0  1.473355e+00
MONTHS_BALANCE_sum                         337252.0  1.038314e+03

                                                   std  min    25%    50%
\
NAME_CONTRACT_STATUS_XNA_mean                 0.000015  0.0    0.0    0.0
CNT_INSTALMENT_count                         24.527667  0.0   12.0   22.0
SK_DPD_DEF_min                                0.375754  0.0    0.0    0.0
NAME_CONTRACT_STATUS_Approved_var             0.007023  0.0    0.0    0.0
POS_CNT_INSTAL_PNDNG_count                   24.527447  0.0   12.0   22.0
...                                                ...  ...    ...    ...
NAME_CONTRACT_STATUS_Active_median            0.062951  0.0    1.0    1.0
SK_DPD_max                                  151.343806  0.0    0.0    0.0
NAME_CONTRACT_STATUS_Amortized_debt_median    0.002983  0.0    0.0    0.0
SK_DPD_DEF_max                               32.337266  0.0    0.0    0.0
MONTHS_BALANCE_sum                         1122.271733  1.0  230.0  687.0

                                            75%          max
NAME_CONTRACT_STATUS_XNA_mean               0.0      0.00625
CNT_INSTALMENT_count                       39.0    295.00000
SK_DPD_DEF_min                              0.0    176.00000
NAME_CONTRACT_STATUS_Approved_var           0.0      0.50000
POS_CNT_INSTAL_PNDNG_count                 39.0    295.00000
...                                         ...          ...
NAME_CONTRACT_STATUS_Active_median          1.0      1.00000
SK_DPD_max                                  0.0   4231.00000
NAME_CONTRACT_STATUS_Amortized_debt_median  0.0      1.00000
SK_DPD_DEF_max                              0.0   3595.00000
MONTHS_BALANCE_sum                       1427.0  14334.00000

[84 rows x 8 columns]
```

PREVAPP - Previous Application

1. This function is specifically for POS table. Below are the initial pre-processing steps done before passing this table into the pipeline.

    - Create a drop list.
        - Attributes that will be dropped will be added to this list and all columns will be deleted before passing the dataframe into the eda function.
    - Create new features. based on analysis, 6 new features were created:

        - Count of approved previous application.
        - Count of Rejected previous applications.
        - Difference: Amount requested in application - Actual credit amount.
        - Ratio - Ratio of application amount to amount credited.
        - Ratio - Ratio of amount credited to amount annuity
        - Ratio - Ratio of down payment to amount credited.
    - There are number of attributes which are in days and amount. For that, we created list of columns which ends with 'DAYS' and 'AMT'

    - Analysis on attributes with date shows that many are capped to 365243, which is 100 years. This looks to be added by the system and not user data. This will be replaced by nan and later imputed.

    - Another observation was that days columns are marked as negative and so the absolute values were calcuated and used.

    - Added below attributes to droplist:

        - WEEKDAY_APPR_PROCESS_START
        - HOUR_APPR_PROCESS_START
    - Any column or row with more than 70% of its data as null will be deleted from the dataframe as the threshold is set to .7.

    - Once processed, store the transformed csv file. Benefit of this is that we can then pass it directly to model for merging into application train/test table. We do not have to perform expensive EDA/ETL/Transformation everytime we want to process the same data.

In [21]:

```
def prevapp_eda(df):
    prevapp = df
    drop_list_pa = []

    #Day and Amount columns
    day_cols = [col for col in prevapp.columns if 'DAY' in col]

    amt_cols = [col for col in prevapp.columns if 'AMT' in col]

    #Adding new features
    prevapp['PREV_APRV_CNT'] = prevapp['NAME_CONTRACT_STATUS'].map(lambda x
    prevapp['PREV_REJ_CNT'] = prevapp['NAME_CONTRACT_STATUS'].map(lambda x:
    prevapp['PREV_APCTN_CRDT_DIFF'] = prevapp['AMT_APPLICATION'] - prevapp[
    prevapp['PREV_APCTN_CRDT_RATIO'] = prevapp['AMT_APPLICATION'] / prevapp
    prevapp['PREV_CRDT_ANNUTY_RATIO'] = prevapp['AMT_CREDIT']/prevapp['AMT_
    prevapp['PREV_DWN_PYMNT_CRDT_RATIO'] = prevapp['AMT_DOWN_PAYMENT'] / pr

    for c in [co for co in prevapp.columns if 'DAYS' in co]:
        prevapp[c] = prevapp[c].replace({365243.0: np.nan})
        prevapp[c] = prevapp[c].abs()

    drop_list_pa.append('WEEKDAY_APPR_PROCESS_START') ## weekday data is no
    drop_list_pa.append('HOUR_APPR_PROCESS_START') ## Hour application star

    # Drop elements in the drop list
    drop_list_pa.append('WEEKDAY_APPR_PROCESS_START') ## weekday data is no
    drop_list_pa.append('HOUR_APPR_PROCESS_START') ## Hour application star

    threshold = 0.7
    drop_list_pa = list(prevapp.columns[prevapp.isnull().mean() > threshold

    prevapp = prevapp.drop(columns=drop_list_pa, axis=1)

    #Dropping columns with missing value rate higher than threshold
    prevapp = prevapp[prevapp.columns[prevapp.isnull().mean() < threshold]]

    #Dropping rows with missing value rate higher than threshold
    prevapp = prevapp.loc[prevapp.isnull().mean(axis=1) < threshold]

    prevapp= eda_transformation(prevapp,3)
    return prevapp
```

In [22]:

```
prevapp = datasets['previous_application']
prevapp = prevapp_eda(prevapp)
datasets_transformed['previous_application'] = prevapp

#prevapp.to_csv(os.getcwd() + DATA_DIR + 'prevapp_agg_data.csv')
```

```
--------
# of ID's: 1
 ID's:
['SK_ID_CURR']

--------
# All features: 39
All features:
['PREV_REJ_CNT', 'NAME_PAYMENT_TYPE', 'FLAG_LAST_APPL_PER_CONTRACT', 'PREV_A
PCTN_CRDT_DIFF', 'CNT_PAYMENT', 'PREV_DWN_PYMNT_CRDT_RATIO', 'NAME_CONTRACT_
STATUS', 'NAME_PRODUCT_TYPE', 'DAYS_TERMINATION', 'RATE_DOWN_PAYMENT', 'PREV
_APRV_CNT', 'SK_ID_PREV', 'NAME_CONTRACT_TYPE', 'AMT_ANNUITY', 'AMT_APPLICAT
ION', 'NAME_YIELD_GROUP', 'HOUR_APPR_PROCESS_START', 'AMT_GOODS_PRICE', 'COD
E_REJECT_REASON', 'NAME_TYPE_SUITE', 'NAME_SELLER_INDUSTRY', 'PREV_APCTN_CRD
T_RATIO', 'NAME_GOODS_CATEGORY', 'CHANNEL_TYPE', 'DAYS_LAST_DUE', 'WEEKDAY_A
PPR_PROCESS_START', 'SELLERPLACE_AREA', 'NAME_PORTFOLIO', 'NAME_CASH_LOAN_PU
RPOSE', 'DAYS_LAST_DUE_1ST_VERSION', 'DAYS_DECISION', 'DAYS_FIRST_DUE', 'NAM
E_CLIENT_TYPE', 'NFLAG_LAST_APPL_IN_DAY', 'NFLAG_INSURED_ON_APPROVAL', 'PREV
_CRDT_ANNUTY_RATIO', 'AMT_DOWN_PAYMENT', 'AMT_CREDIT', 'PRODUCT_COMBINATION
']

Missing data:
                              Total    Percent
DAYS_TERMINATION             898978  53.824121
AMT_DOWN_PAYMENT             895844  53.636480
PREV_DWN_PYMNT_CRDT_RATIO    895844  53.636480
RATE_DOWN_PAYMENT            895844  53.636480
DAYS_LAST_DUE                884286  52.944473
NAME_TYPE_SUITE              820405  49.119754
DAYS_LAST_DUE_1ST_VERSION    766929  45.918008
DAYS_FIRST_DUE               713710  42.731650
NFLAG_INSURED_ON_APPROVAL    673065  40.298129
AMT_GOODS_PRICE              385515  23.081773
PREV_CRDT_ANNUTY_RATIO       373859  22.383898
AMT_ANNUITY                  372235  22.286665
CNT_PAYMENT                  372230  22.286366
PREV_APCTN_CRDT_RATIO        336768  20.163165
PRODUCT_COMBINATION             346   0.020716
AMT_CREDIT                        1   0.000060
PREV_APCTN_CRDT_DIFF              1   0.000060
NAME_YIELD_GROUP                  0   0.000000
HOUR_APPR_PROCESS_START           0   0.000000
FLAG_LAST_APPL_PER_CONTRACT       0   0.000000
NAME_CONTRACT_STATUS              0   0.000000
NFLAG_LAST_APPL_IN_DAY            0   0.000000
NAME_CLIENT_TYPE                  0   0.000000
NAME_PRODUCT_TYPE                 0   0.000000
DAYS_DECISION                     0   0.000000
PREV_APRV_CNT                     0   0.000000
NAME_CASH_LOAN_PURPOSE            0   0.000000
NAME_PORTFOLIO                    0   0.000000
SELLERPLACE_AREA                  0   0.000000
WEEKDAY_APPR_PROCESS_START        0   0.000000
SK_ID_PREV                        0   0.000000
CHANNEL_TYPE                      0   0.000000
NAME_GOODS_CATEGORY               0   0.000000
NAME_CONTRACT_TYPE                0   0.000000
NAME_SELLER_INDUSTRY              0   0.000000
NAME_PAYMENT_TYPE                 0   0.000000
```

```
CODE_REJECT_REASON                0   0.000000
AMT_APPLICATION                   0   0.000000
PREV_REJ_CNT                      0   0.000000

--------
# of Numerical features: 24
Numerical features:
['SK_ID_PREV', 'SK_ID_CURR', 'AMT_ANNUITY', 'AMT_APPLICATION', 'AMT_CREDIT',
'AMT_DOWN_PAYMENT', 'AMT_GOODS_PRICE', 'HOUR_APPR_PROCESS_START', 'NFLAG_LAS
T_APPL_IN_DAY', 'RATE_DOWN_PAYMENT', 'DAYS_DECISION', 'SELLERPLACE_AREA', 'C
NT_PAYMENT', 'DAYS_FIRST_DUE', 'DAYS_LAST_DUE_1ST_VERSION', 'DAYS_LAST_DUE',
'DAYS_TERMINATION', 'NFLAG_INSURED_ON_APPROVAL', 'PREV_APRV_CNT', 'PREV_REJ_
CNT', 'PREV_APCTN_CRDT_DIFF', 'PREV_APCTN_CRDT_RATIO', 'PREV_CRDT_ANNUTY_RAT
IO', 'PREV_DWN_PYMNT_CRDT_RATIO']

Numerical Statistical Summary:

         SK_ID_PREV    SK_ID_CURR   AMT_ANNUITY  AMT_APPLICATION  \
count  1.670214e+06  1.670214e+06  1.297979e+06     1.670214e+06
mean   1.923089e+06  2.783572e+05  1.595512e+04     1.752339e+05
std    5.325980e+05  1.028148e+05  1.478214e+04     2.927798e+05
min    1.000001e+06  1.000010e+05  0.000000e+00     0.000000e+00
25%    1.461857e+06  1.893290e+05  6.321780e+03     1.872000e+04
50%    1.923110e+06  2.787145e+05  1.125000e+04     7.104600e+04
75%    2.384280e+06  3.675140e+05  2.065842e+04     1.803600e+05
max    2.845382e+06  4.562550e+05  4.180581e+05     6.905160e+06

         AMT_CREDIT  AMT_DOWN_PAYMENT  AMT_GOODS_PRICE  \
count  1.670213e+06      7.743700e+05     1.284699e+06
mean   1.961140e+05      6.697402e+03     2.278473e+05
std    3.185746e+05      2.092150e+04     3.153966e+05
min    0.000000e+00     -9.000000e-01     0.000000e+00
25%    2.416050e+04      0.000000e+00     5.084100e+04
50%    8.054100e+04      1.638000e+03     1.123200e+05
75%    2.164185e+05      7.740000e+03     2.340000e+05
max    6.905160e+06      3.060045e+06     6.905160e+06

       HOUR_APPR_PROCESS_START  NFLAG_LAST_APPL_IN_DAY  RATE_DOWN_PAYMENT  \
count             1.670214e+06            1.670214e+06      774370.000000
mean              1.248418e+01            9.964675e-01           0.079637
std               3.334028e+00            5.932963e-02           0.107823
min               0.000000e+00            0.000000e+00          -0.000015
25%               1.000000e+01            1.000000e+00           0.000000
50%               1.200000e+01            1.000000e+00           0.051605
75%               1.500000e+01            1.000000e+00           0.108909
max               2.300000e+01            1.000000e+00           1.000000

       ...  DAYS_LAST_DUE_1ST_VERSION  DAYS_LAST_DUE  DAYS_TERMINATION  \
count  ...              903285.000000  785928.000000     771236.000000
mean   ...                 863.452359     996.179128        978.375222
std    ...                 752.350974     752.703178        749.134297
min    ...                   0.000000       2.000000          2.000000
25%    ...                 227.000000     353.000000        337.000000
50%    ...                 615.000000     801.000000        780.000000
75%    ...                1393.000000    1566.000000       1539.000000
max    ...                2801.000000    2889.000000       2874.000000

       NFLAG_INSURED_ON_APPROVAL  PREV_APRV_CNT  PREV_REJ_CNT  \
count              997149.000000   1.670214e+06     1670214.0
```

```
mean                    0.332570   6.207474e-01           0.0
std                     0.471134   4.852012e-01           0.0
min                     0.000000   0.000000e+00           0.0
25%                     0.000000   0.000000e+00           0.0
50%                     0.000000   1.000000e+00           0.0
75%                     1.000000   1.000000e+00           0.0
max                     1.000000   1.000000e+00           0.0

       PREV_APCTN_CRDT_DIFF  PREV_APCTN_CRDT_RATIO  PREV_CRDT_ANNUTY_RATIO
\
count          1.670213e+06           1.333446e+06            1.296355e+06
mean          -2.088006e+04                    inf                     inf
std            7.193908e+04                    NaN                     NaN
min           -1.350000e+06           0.000000e+00            1.111111e+00
25%           -1.291950e+04           8.959282e-01            8.224355e+00
50%            0.000000e+00           1.000000e+00            1.008346e+01
75%            0.000000e+00           1.011023e+00            2.000000e+01
max            3.020450e+06                    inf                     inf

       PREV_DWN_PYMNT_CRDT_RATIO
count              774370.000000
mean                    0.096739
std                     0.180674
min                    -0.000014
25%                     0.000000
50%                     0.049741
75%                     0.111111
max                    11.224490

[8 rows x 24 columns]

--------
# of Categorical features: 16
Categorical features:
['NAME_CONTRACT_TYPE', 'WEEKDAY_APPR_PROCESS_START', 'FLAG_LAST_APPL_PER_CON
TRACT', 'NAME_CASH_LOAN_PURPOSE', 'NAME_CONTRACT_STATUS', 'NAME_PAYMENT_TYPE
', 'CODE_REJECT_REASON', 'NAME_TYPE_SUITE', 'NAME_CLIENT_TYPE', 'NAME_GOODS_
CATEGORY', 'NAME_PORTFOLIO', 'NAME_PRODUCT_TYPE', 'CHANNEL_TYPE', 'NAME_SELL
ER_INDUSTRY', 'NAME_YIELD_GROUP', 'PRODUCT_COMBINATION']

Categorical Statistical Summary:

       NAME_CONTRACT_TYPE WEEKDAY_APPR_PROCESS_START  \
count             1670214                    1670214
unique                  4                          7
top            Cash loans                    TUESDAY
freq               747553                     255118

       FLAG_LAST_APPL_PER_CONTRACT NAME_CASH_LOAN_PURPOSE  \
count                      1670214                1670214
unique                           2                     25
top                              Y                    XAP
freq                       1661739                 922661

       NAME_CONTRACT_STATUS      NAME_PAYMENT_TYPE CODE_REJECT_REASON  \
count               1670214                1670214            1670214
unique                    4                      4                  9
top                Approved  Cash through the bank                XAP
freq                1036781                1033552            1353093
```

```
       NAME_TYPE_SUITE NAME_CLIENT_TYPE NAME_GOODS_CATEGORY NAME_PORTFOLIO
\
count           849809          1670214             1670214        1670214
unique               7                4                  28              5
top      Unaccompanied         Repeater                 XNA            POS
freq            508970          1231261              950809         691011

       NAME_PRODUCT_TYPE             CHANNEL_TYPE NAME_SELLER_INDUSTRY  \
count            1670214                  1670214              1670214
unique                 3                        8                   11
top                  XNA  Credit and cash offices                  XNA
freq             1063666                   719968               855720

       NAME_YIELD_GROUP PRODUCT_COMBINATION
count           1670214             1669868
unique                5                  17
top                 XNA                Cash
freq             517215              285990

Categories:

NAME_CONTRACT_TYPE             [Consumer loans, Cash loans, Revolving loans,
...
WEEKDAY_APPR_PROCESS_START     [SATURDAY, THURSDAY, TUESDAY, MONDAY, FRIDAY,
...
FLAG_LAST_APPL_PER_CONTRACT
[Y, N]
NAME_CASH_LOAN_PURPOSE         [XAP, XNA, Repairs, Everyday expenses, Car re
p...
NAME_CONTRACT_STATUS                 [Approved, Refused, Canceled, Unused of
fer]
NAME_PAYMENT_TYPE              [Cash through the bank, XNA, Non-cash from yo
u...
CODE_REJECT_REASON             [XAP, HC, LIMIT, CLIENT, SCOFR, SCO, XNA, VER
I...
NAME_TYPE_SUITE                [nan, Unaccompanied, Spouse, partner, Family,
...
NAME_CLIENT_TYPE                                [Repeater, New, Refreshed,
XNA]
NAME_GOODS_CATEGORY            [Mobile, XNA, Consumer Electronics, Construct
i...
NAME_PORTFOLIO                                    [POS, Cash, XNA, Cards, C
ars]
NAME_PRODUCT_TYPE                                        [XNA, x-sell, walk
-in]
CHANNEL_TYPE                   [Country-wide, Contact center, Credit and cas
h...
NAME_SELLER_INDUSTRY           [Connectivity, XNA, Consumer electronics, Ind
u...
NAME_YIELD_GROUP                     [middle, low_action, high, low_normal,
XNA]
PRODUCT_COMBINATION            [POS mobile with interest, Cash X-Sell: low,
C...
dtype: object

--------
# of OHE categorical features: 143
OHE Categorical features: ['NAME_GOODS_CATEGORY_Consumer Electronics', 'PROD
```

UCT_COMBINATION_POS industry with interest', 'NAME_SELLER_INDUSTRY_Construct
ion', 'CODE_REJECT_REASON_SCOFR', 'NAME_GOODS_CATEGORY_Tourism', 'NAME_GOODS
_CATEGORY_Computers', 'NAME_GOODS_CATEGORY_Additional Service', 'NAME_PORTFO
LIO_XNA', 'NAME_GOODS_CATEGORY_XNA', 'WEEKDAY_APPR_PROCESS_START_SUNDAY', 'N
AME_GOODS_CATEGORY_Homewares', 'PRODUCT_COMBINATION_POS mobile without inter
est', 'NAME_PAYMENT_TYPE_Cashless from the account of the employer', 'FLAG_L
AST_APPL_PER_CONTRACT_N', 'CODE_REJECT_REASON_VERIF', 'NAME_CONTRACT_TYPE_Re
volving loans', 'PRODUCT_COMBINATION_POS industry without interest', 'PRODUC
T_COMBINATION_Cash Street: low', 'NAME_CONTRACT_STATUS_Canceled', 'NAME_CASH
_LOAN_PURPOSE_Urgent needs', 'NAME_GOODS_CATEGORY_Furniture', 'NAME_GOODS_CA
TEGORY_Photo / Cinema Equipment', 'NAME_SELLER_INDUSTRY_Connectivity', 'CHAN
NEL_TYPE_Channel of corporate sales', 'NAME_CLIENT_TYPE_New', 'CHANNEL_TYPE_
Contact center', 'NAME_CONTRACT_TYPE_Consumer loans', 'NAME_TYPE_SUITE_Child
ren', 'CHANNEL_TYPE_Country-wide', 'NAME_CONTRACT_STATUS_Refused', 'PRODUCT_
COMBINATION_POS mobile with interest', 'NAME_CASH_LOAN_PURPOSE_Everyday expe
nses', 'NAME_GOODS_CATEGORY_House Construction', 'NAME_GOODS_CATEGORY_Other
', 'WEEKDAY_APPR_PROCESS_START_THURSDAY', 'NAME_CASH_LOAN_PURPOSE_Buying a h
ome', 'CODE_REJECT_REASON_SCO', 'NAME_CASH_LOAN_PURPOSE_Buying a used car',
'CHANNEL_TYPE_Regional / Local', 'NAME_GOODS_CATEGORY_Fitness', 'NAME_CLIENT
_TYPE_Refreshed', 'NAME_YIELD_GROUP_XNA', 'NAME_CASH_LOAN_PURPOSE_Other', 'N
AME_TYPE_SUITE_Group of people', 'NAME_PAYMENT_TYPE_XNA', 'NAME_GOODS_CATEGO
RY_Mobile', 'NAME_CASH_LOAN_PURPOSE_Gasification / water supply', 'PRODUCT_C
OMBINATION_POS household without interest', 'NAME_CASH_LOAN_PURPOSE_Car repa
irs', 'NAME_TYPE_SUITE_Unaccompanied', 'NAME_TYPE_SUITE_Other_A', 'NAME_PORT
FOLIO_Cash', 'NAME_GOODS_CATEGORY_Animals', 'NAME_CASH_LOAN_PURPOSE_Money fo
r a third person', 'NAME_GOODS_CATEGORY_Auto Accessories', 'NAME_GOODS_CATEG
ORY_Direct Sales', 'NAME_SELLER_INDUSTRY_Consumer electronics', 'NAME_SELLER
_INDUSTRY_Jewelry', 'NAME_CLIENT_TYPE_Repeater', 'NAME_CONTRACT_TYPE_XNA', '
NAME_CASH_LOAN_PURPOSE_Repairs', 'NAME_YIELD_GROUP_low_action', 'CODE_REJECT
_REASON_XNA', 'NAME_YIELD_GROUP_low_normal', 'NAME_CONTRACT_STATUS_Unused of
fer', 'NAME_CASH_LOAN_PURPOSE_Hobby', 'CHANNEL_TYPE_Stone', 'PRODUCT_COMBINA
TION_Cash Street: high', 'NAME_GOODS_CATEGORY_Gardening', 'NAME_CLIENT_TYPE_
XNA', 'PRODUCT_COMBINATION_Cash Street: middle', 'NAME_YIELD_GROUP_middle',
'WEEKDAY_APPR_PROCESS_START_WEDNESDAY', 'NAME_TYPE_SUITE_Family', 'NAME_SELL
ER_INDUSTRY_Furniture', 'NAME_GOODS_CATEGORY_Construction Materials', 'NAME_
CASH_LOAN_PURPOSE_Buying a holiday home / land', 'PRODUCT_COMBINATION_POS ot
her with interest', 'NAME_CASH_LOAN_PURPOSE_Building a house or an annex', '
NAME_GOODS_CATEGORY_Vehicles', 'NAME_TYPE_SUITE_Spouse, partner', 'CHANNEL_T
YPE_Credit and cash offices', 'NAME_SELLER_INDUSTRY_Clothing', 'WEEKDAY_APPR
_PROCESS_START_SATURDAY', 'NAME_PORTFOLIO_POS', 'NAME_CASH_LOAN_PURPOSE_Jour
ney', 'PRODUCT_COMBINATION_Cash X-Sell: low', 'NAME_PAYMENT_TYPE_Cash throug
h the bank', 'PRODUCT_COMBINATION_Card X-Sell', 'CODE_REJECT_REASON_LIMIT',
'CHANNEL_TYPE_Car dealer', 'NAME_CASH_LOAN_PURPOSE_Refusal to name the goal
', 'NAME_PORTFOLIO_Cards', 'PRODUCT_COMBINATION_Cash X-Sell: high', 'NAME_GO
ODS_CATEGORY_Clothing and Accessories', 'WEEKDAY_APPR_PROCESS_START_FRIDAY',
'NAME_CASH_LOAN_PURPOSE_XAP', 'NAME_YIELD_GROUP_high', 'NAME_TYPE_SUITE_Othe
r_B', 'NAME_GOODS_CATEGORY_Medical Supplies', 'NAME_CASH_LOAN_PURPOSE_Purcha
se of electronic equipment', 'NAME_GOODS_CATEGORY_Insurance', 'PRODUCT_COMBI
NATION_POS household with interest', 'NAME_CASH_LOAN_PURPOSE_Education', 'PR
ODUCT_COMBINATION_POS others without interest', 'PRODUCT_COMBINATION_Cash X-
Sell: middle', 'NAME_CASH_LOAN_PURPOSE_Medicine', 'NAME_CONTRACT_STATUS_Appr
oved', 'CODE_REJECT_REASON_XAP', 'CHANNEL_TYPE_AP+ (Cash loan)', 'NAME_GOODS
_CATEGORY_Jewelry', 'NAME_CASH_LOAN_PURPOSE_Furniture', 'NAME_GOODS_CATEGORY
_Weapon', 'NAME_CASH_LOAN_PURPOSE_Buying a garage', 'NAME_CASH_LOAN_PURPOSE_
Buying a new car', 'NAME_GOODS_CATEGORY_Sport and Leisure', 'CODE_REJECT_REA
SON_SYSTEM', 'NAME_SELLER_INDUSTRY_Industry', 'NAME_PRODUCT_TYPE_XNA', 'NAME
_PRODUCT_TYPE_walk-in', 'NAME_SELLER_INDUSTRY_MLM partners', 'NAME_GOODS_CAT
EGORY_Education', 'NAME_SELLER_INDUSTRY_XNA', 'CODE_REJECT_REASON_CLIENT', '
NAME_GOODS_CATEGORY_Medicine', 'NAME_CASH_LOAN_PURPOSE_Wedding / gift / holi

```
day', 'NAME_GOODS_CATEGORY_Office Appliances', 'PRODUCT_COMBINATION_Cash', '
FLAG_LAST_APPL_PER_CONTRACT_Y', 'CODE_REJECT_REASON_HC', 'NAME_GOODS_CATEGOR
Y_Audio/Video', 'NAME_CASH_LOAN_PURPOSE_Business development', 'NAME_PORTFOL
IO_Cars', 'WEEKDAY_APPR_PROCESS_START_MONDAY', 'NAME_SELLER_INDUSTRY_Tourism
', 'NAME_CASH_LOAN_PURPOSE_Payments on other loans', 'NAME_CASH_LOAN_PURPOSE
_XNA', 'WEEKDAY_APPR_PROCESS_START_TUESDAY', 'NAME_PRODUCT_TYPE_x-sell', 'NA
ME_PAYMENT_TYPE_Non-cash from your account', 'NAME_SELLER_INDUSTRY_Auto tech
nology', 'PRODUCT_COMBINATION_Card Street', 'NAME_CONTRACT_TYPE_Cash loans']
--------
df.shape: (1670214, 167)

Aggregated Features:
df[['PREV_REJ_CNT', 'CNT_PAYMENT', 'NAME_GOODS_CATEGORY_Consumer Electronics
', 'PRODUCT_COMBINATION_POS industry with interest', 'NAME_SELLER_INDUSTRY_C
onstruction', 'CODE_REJECT_REASON_SCOFR', 'NAME_GOODS_CATEGORY_Tourism', 'NA
ME_GOODS_CATEGORY_Computers', 'NAME_GOODS_CATEGORY_Additional Service', 'NAM
E_PORTFOLIO_XNA', 'NAME_GOODS_CATEGORY_XNA', 'AMT_APPLICATION', 'WEEKDAY_APP
R_PROCESS_START_SUNDAY', 'NAME_GOODS_CATEGORY_Homewares', 'PRODUCT_COMBINATI
ON_POS mobile without interest', 'NAME_PAYMENT_TYPE_Cashless from the accoun
t of the employer', 'FLAG_LAST_APPL_PER_CONTRACT_N', 'CODE_REJECT_REASON_VER
IF', 'AMT_GOODS_PRICE', 'NAME_CONTRACT_TYPE_Revolving loans', 'PRODUCT_COMBI
NATION_POS industry without interest', 'PRODUCT_COMBINATION_Cash Street: low
', 'NAME_CONTRACT_STATUS_Canceled', 'NAME_CASH_LOAN_PURPOSE_Urgent needs', '
DAYS_LAST_DUE_1ST_VERSION', 'NAME_GOODS_CATEGORY_Furniture', 'NAME_GOODS_CAT
EGORY_Photo / Cinema Equipment', 'DAYS_FIRST_DUE', 'NAME_SELLER_INDUSTRY_Con
nectivity', 'CHANNEL_TYPE_Channel of corporate sales', 'NAME_CLIENT_TYPE_New
', 'AMT_CREDIT', 'NAME_CONTRACT_TYPE_Consumer loans', 'CHANNEL_TYPE_Contact
center', 'NAME_TYPE_SUITE_Children', 'CHANNEL_TYPE_Country-wide', 'NAME_CONT
RACT_STATUS_Refused', 'PRODUCT_COMBINATION_POS mobile with interest', 'NAME_
CASH_LOAN_PURPOSE_Everyday expenses', 'DAYS_TERMINATION', 'NAME_GOODS_CATEGO
RY_House Construction', 'NAME_GOODS_CATEGORY_Other', 'RATE_DOWN_PAYMENT', 'W
EEKDAY_APPR_PROCESS_START_THURSDAY', 'NAME_CASH_LOAN_PURPOSE_Buying a home',
'CODE_REJECT_REASON_SCO', 'NAME_CASH_LOAN_PURPOSE_Buying a used car', 'HOUR_
APPR_PROCESS_START', 'CHANNEL_TYPE_Regional / Local', 'NAME_GOODS_CATEGORY_F
itness', 'NAME_CLIENT_TYPE_Refreshed', 'NAME_YIELD_GROUP_XNA', 'NAME_CASH_LO
AN_PURPOSE_Other', 'NAME_TYPE_SUITE_Group of people', 'NAME_PAYMENT_TYPE_XNA
', 'NAME_GOODS_CATEGORY_Mobile', 'PREV_APCTN_CRDT_RATIO', 'NAME_CASH_LOAN_PU
RPOSE_Gasification / water supply', 'PRODUCT_COMBINATION_POS household witho
ut interest', 'NAME_CASH_LOAN_PURPOSE_Car repairs', 'NAME_TYPE_SUITE_Unaccom
panied', 'NAME_TYPE_SUITE_Other_A', 'NAME_PORTFOLIO_Cash', 'NAME_GOODS_CATEG
ORY_Animals', 'NAME_CASH_LOAN_PURPOSE_Money for a third person', 'NAME_GOODS
_CATEGORY_Auto Accessories', 'SELLERPLACE_AREA', 'NAME_GOODS_CATEGORY_Direct
Sales', 'NAME_SELLER_INDUSTRY_Consumer electronics', 'NAME_SELLER_INDUSTRY_J
ewelry', 'NAME_CLIENT_TYPE_Repeater', 'NAME_CONTRACT_TYPE_XNA', 'NAME_CASH_L
OAN_PURPOSE_Repairs', 'NAME_YIELD_GROUP_low_action', 'CODE_REJECT_REASON_XNA
', 'NAME_YIELD_GROUP_low_normal', 'NAME_CONTRACT_STATUS_Unused offer', 'NAME
_CASH_LOAN_PURPOSE_Hobby', 'CHANNEL_TYPE_Stone', 'PRODUCT_COMBINATION_Cash S
treet: high', 'NAME_GOODS_CATEGORY_Gardening', 'NAME_CLIENT_TYPE_XNA', 'PROD
UCT_COMBINATION_Cash Street: middle', 'NAME_YIELD_GROUP_middle', 'WEEKDAY_AP
PR_PROCESS_START_WEDNESDAY', 'NAME_TYPE_SUITE_Family', 'NAME_SELLER_INDUSTRY
_Furniture', 'NAME_GOODS_CATEGORY_Construction Materials', 'PREV_APRV_CNT',
'SK_ID_PREV', 'NAME_CASH_LOAN_PURPOSE_Buying a holiday home / land', 'PRODUC
T_COMBINATION_POS other with interest', 'NAME_CASH_LOAN_PURPOSE_Building a h
ouse or an annex', 'NAME_GOODS_CATEGORY_Vehicles', 'AMT_ANNUITY', 'NAME_TYPE
_SUITE_Spouse, partner', 'CHANNEL_TYPE_Credit and cash offices', 'NAME_SELLE
R_INDUSTRY_Clothing', 'WEEKDAY_APPR_PROCESS_START_SATURDAY', 'NAME_PORTFOLIO
_POS', 'NAME_CASH_LOAN_PURPOSE_Journey', 'PRODUCT_COMBINATION_Cash X-Sell: l
ow', 'NAME_PAYMENT_TYPE_Cash through the bank', 'PRODUCT_COMBINATION_Card X-
Sell', 'CODE_REJECT_REASON_LIMIT', 'CHANNEL_TYPE_Car dealer', 'NAME_CASH_LOA
```

```
N_PURPOSE_Refusal to name the goal', 'NAME_PORTFOLIO_Cards', 'PRODUCT_COMBIN
ATION_Cash X-Sell: high', 'NAME_GOODS_CATEGORY_Clothing and Accessories', 'W
EEKDAY_APPR_PROCESS_START_FRIDAY', 'NAME_CASH_LOAN_PURPOSE_XAP', 'DAYS_LAST_
DUE', 'NAME_YIELD_GROUP_high', 'NAME_TYPE_SUITE_Other_B', 'NAME_GOODS_CATEGO
RY_Medical Supplies', 'NAME_CASH_LOAN_PURPOSE_Purchase of electronic equipme
nt', 'NAME_GOODS_CATEGORY_Insurance', 'PRODUCT_COMBINATION_POS household wit
h interest', 'NAME_CASH_LOAN_PURPOSE_Education', 'PREV_CRDT_ANNUTY_RATIO', '
PRODUCT_COMBINATION_POS others without interest', 'PRODUCT_COMBINATION_Cash
X-Sell: middle', 'PREV_APCTN_CRDT_DIFF', 'NAME_CASH_LOAN_PURPOSE_Medicine',
'PREV_DWN_PYMNT_CRDT_RATIO', 'NAME_CONTRACT_STATUS_Approved', 'CODE_REJECT_R
EASON_XAP', 'CHANNEL_TYPE_AP+ (Cash loan)', 'NAME_GOODS_CATEGORY_Jewelry', '
NAME_CASH_LOAN_PURPOSE_Furniture', 'NAME_GOODS_CATEGORY_Weapon', 'NAME_CASH_
LOAN_PURPOSE_Buying a garage', 'NAME_CASH_LOAN_PURPOSE_Buying a new car', 'N
AME_GOODS_CATEGORY_Sport and Leisure', 'CODE_REJECT_REASON_SYSTEM', 'NAME_SE
LLER_INDUSTRY_Industry', 'NAME_PRODUCT_TYPE_XNA', 'NAME_PRODUCT_TYPE_walk-in
', 'NAME_SELLER_INDUSTRY_MLM partners', 'NAME_GOODS_CATEGORY_Education', 'NA
ME_SELLER_INDUSTRY_XNA', 'CODE_REJECT_REASON_CLIENT', 'NAME_GOODS_CATEGORY_M
edicine', 'NAME_CASH_LOAN_PURPOSE_Wedding / gift / holiday', 'NAME_GOODS_CAT
EGORY_Office Appliances', 'PRODUCT_COMBINATION_Cash', 'FLAG_LAST_APPL_PER_CO
NTRACT_Y', 'CODE_REJECT_REASON_HC', 'NAME_GOODS_CATEGORY_Audio/Video', 'NAME
_CASH_LOAN_PURPOSE_Business development', 'NAME_PORTFOLIO_Cars', 'WEEKDAY_AP
PR_PROCESS_START_MONDAY', 'NAME_SELLER_INDUSTRY_Tourism', 'NAME_CASH_LOAN_PU
RPOSE_Payments on other loans', 'DAYS_DECISION', 'NAME_CASH_LOAN_PURPOSE_XNA
', 'WEEKDAY_APPR_PROCESS_START_TUESDAY', 'NAME_PRODUCT_TYPE_x-sell', 'NAME_P
AYMENT_TYPE_Non-cash from your account', 'NAME_SELLER_INDUSTRY_Auto technolo
gy', 'NFLAG_LAST_APPL_IN_DAY', 'PRODUCT_COMBINATION_Card Street', 'NFLAG_INS
URED_ON_APPROVAL', 'AMT_DOWN_PAYMENT', 'NAME_CONTRACT_TYPE_Cash loans']][0:
5]:
   PREV_REJ_CNT  CNT_PAYMENT  NAME_GOODS_CATEGORY_Consumer Electronics  \
0             0         12.0                                         0
1             0         36.0                                         0
2             0         12.0                                         0
3             0         12.0                                         0
4             0         24.0                                         0

   PRODUCT_COMBINATION_POS industry with interest  \
0                                               0
1                                               0
2                                               0
3                                               0
4                                               0

   NAME_SELLER_INDUSTRY_Construction  CODE_REJECT_REASON_SCOFR  \
0                                  0                          0
1                                  0                          0
2                                  0                          0
3                                  0                          0
4                                  0                          0

   NAME_GOODS_CATEGORY_Tourism  NAME_GOODS_CATEGORY_Computers  \
0                            0                              0
1                            0                              0
2                            0                              0
3                            0                              0
4                            0                              0

   NAME_GOODS_CATEGORY_Additional Service  NAME_PORTFOLIO_XNA  ...  \
0                                       0                   0  ...
1                                       0                   0  ...
```

```
2                                       0                   0  ...
3                                       0                   0  ...
4                                       0                   0  ...

   NAME_CASH_LOAN_PURPOSE_XNA  WEEKDAY_APPR_PROCESS_START_TUESDAY  \
0                           0                                   0
1                           1                                   0
2                           1                                   1
3                           1                                   0
4                           0                                   0

   NAME_PRODUCT_TYPE_x-sell  NAME_PAYMENT_TYPE_Non-cash from your account  \
0                         0                                            0
1                         1                                            0
2                         1                                            0
3                         1                                            0
4                         0                                            0

   NAME_SELLER_INDUSTRY_Auto technology  NFLAG_LAST_APPL_IN_DAY  \
0                                     0                       1
1                                     0                       1
2                                     0                       1
3                                     0                       1
4                                     0                       1

   PRODUCT_COMBINATION_Card Street  NFLAG_INSURED_ON_APPROVAL  \
0                                0                        0.0
1                                0                        1.0
2                                0                        1.0
3                                0                        1.0
4                                0                        NaN

   AMT_DOWN_PAYMENT  NAME_CONTRACT_TYPE_Cash loans
0               0.0                             0
1               NaN                             1
2               NaN                             1
3               NaN                             1
4               NaN                             1

[5 rows x 166 columns]
--------
Aggregated Features:
AMT_ANNUITY_count
AMT_ANNUITY_max
AMT_ANNUITY_mean
AMT_ANNUITY_median
AMT_ANNUITY_min
AMT_ANNUITY_sum
AMT_ANNUITY_var
AMT_APPLICATION_count
AMT_APPLICATION_max
AMT_APPLICATION_mean
AMT_APPLICATION_median
AMT_APPLICATION_min
AMT_APPLICATION_sum
AMT_APPLICATION_var
AMT_CREDIT_count
AMT_CREDIT_max
AMT_CREDIT_mean
```

AMT_CREDIT_median
AMT_CREDIT_min
AMT_CREDIT_sum
AMT_CREDIT_var
AMT_DOWN_PAYMENT_count
AMT_DOWN_PAYMENT_max
AMT_DOWN_PAYMENT_mean
AMT_DOWN_PAYMENT_median
AMT_DOWN_PAYMENT_min
AMT_DOWN_PAYMENT_sum
AMT_DOWN_PAYMENT_var
AMT_GOODS_PRICE_count
AMT_GOODS_PRICE_max
AMT_GOODS_PRICE_mean
AMT_GOODS_PRICE_median
AMT_GOODS_PRICE_min
AMT_GOODS_PRICE_sum
AMT_GOODS_PRICE_var
CHANNEL_TYPE_AP+ (Cash loan)_mean
CHANNEL_TYPE_AP+ (Cash loan)_median
CHANNEL_TYPE_AP+ (Cash loan)_var
CHANNEL_TYPE_Car dealer_mean
CHANNEL_TYPE_Car dealer_median
CHANNEL_TYPE_Car dealer_var
CHANNEL_TYPE_Channel of corporate sales_mean
CHANNEL_TYPE_Channel of corporate sales_median
CHANNEL_TYPE_Channel of corporate sales_var
CHANNEL_TYPE_Contact center_mean
CHANNEL_TYPE_Contact center_median
CHANNEL_TYPE_Contact center_var
CHANNEL_TYPE_Country-wide_mean
CHANNEL_TYPE_Country-wide_median
CHANNEL_TYPE_Country-wide_var
CHANNEL_TYPE_Credit and cash offices_mean
CHANNEL_TYPE_Credit and cash offices_median
CHANNEL_TYPE_Credit and cash offices_var
CHANNEL_TYPE_Regional / Local_mean
CHANNEL_TYPE_Regional / Local_median
CHANNEL_TYPE_Regional / Local_var
CHANNEL_TYPE_Stone_mean
CHANNEL_TYPE_Stone_median
CHANNEL_TYPE_Stone_var
CNT_PAYMENT_count
CNT_PAYMENT_max
CNT_PAYMENT_mean
CNT_PAYMENT_median
CNT_PAYMENT_min
CNT_PAYMENT_sum
CNT_PAYMENT_var
CODE_REJECT_REASON_CLIENT_mean
CODE_REJECT_REASON_CLIENT_median
CODE_REJECT_REASON_CLIENT_var
CODE_REJECT_REASON_HC_mean
CODE_REJECT_REASON_HC_median
CODE_REJECT_REASON_HC_var
CODE_REJECT_REASON_LIMIT_mean
CODE_REJECT_REASON_LIMIT_median
CODE_REJECT_REASON_LIMIT_var
CODE_REJECT_REASON_SCOFR_mean

CODE_REJECT_REASON_SCOFR_median
CODE_REJECT_REASON_SCOFR_var
CODE_REJECT_REASON_SCO_mean
CODE_REJECT_REASON_SCO_median
CODE_REJECT_REASON_SCO_var
CODE_REJECT_REASON_SYSTEM_mean
CODE_REJECT_REASON_SYSTEM_median
CODE_REJECT_REASON_SYSTEM_var
CODE_REJECT_REASON_VERIF_mean
CODE_REJECT_REASON_VERIF_median
CODE_REJECT_REASON_VERIF_var
CODE_REJECT_REASON_XAP_mean
CODE_REJECT_REASON_XAP_median
CODE_REJECT_REASON_XAP_var
CODE_REJECT_REASON_XNA_mean
CODE_REJECT_REASON_XNA_median
CODE_REJECT_REASON_XNA_var
DAYS_DECISION_count
DAYS_DECISION_max
DAYS_DECISION_mean
DAYS_DECISION_median
DAYS_DECISION_min
DAYS_DECISION_sum
DAYS_DECISION_var
DAYS_FIRST_DUE_count
DAYS_FIRST_DUE_max
DAYS_FIRST_DUE_mean
DAYS_FIRST_DUE_median
DAYS_FIRST_DUE_min
DAYS_FIRST_DUE_sum
DAYS_FIRST_DUE_var
DAYS_LAST_DUE_1ST_VERSION_count
DAYS_LAST_DUE_1ST_VERSION_max
DAYS_LAST_DUE_1ST_VERSION_mean
DAYS_LAST_DUE_1ST_VERSION_median
DAYS_LAST_DUE_1ST_VERSION_min
DAYS_LAST_DUE_1ST_VERSION_sum
DAYS_LAST_DUE_1ST_VERSION_var
DAYS_LAST_DUE_count
DAYS_LAST_DUE_max
DAYS_LAST_DUE_mean
DAYS_LAST_DUE_median
DAYS_LAST_DUE_min
DAYS_LAST_DUE_sum
DAYS_LAST_DUE_var
DAYS_TERMINATION_count
DAYS_TERMINATION_max
DAYS_TERMINATION_mean
DAYS_TERMINATION_median
DAYS_TERMINATION_min
DAYS_TERMINATION_sum
DAYS_TERMINATION_var
FLAG_LAST_APPL_PER_CONTRACT_N_mean
FLAG_LAST_APPL_PER_CONTRACT_N_median
FLAG_LAST_APPL_PER_CONTRACT_N_var
FLAG_LAST_APPL_PER_CONTRACT_Y_mean
FLAG_LAST_APPL_PER_CONTRACT_Y_median
FLAG_LAST_APPL_PER_CONTRACT_Y_var
HOUR_APPR_PROCESS_START_count

HOUR_APPR_PROCESS_START_max
HOUR_APPR_PROCESS_START_mean
HOUR_APPR_PROCESS_START_median
HOUR_APPR_PROCESS_START_min
HOUR_APPR_PROCESS_START_sum
HOUR_APPR_PROCESS_START_var
NAME_CASH_LOAN_PURPOSE_Building a house or an annex_mean
NAME_CASH_LOAN_PURPOSE_Building a house or an annex_median
NAME_CASH_LOAN_PURPOSE_Building a house or an annex_var
NAME_CASH_LOAN_PURPOSE_Business development_mean
NAME_CASH_LOAN_PURPOSE_Business development_median
NAME_CASH_LOAN_PURPOSE_Business development_var
NAME_CASH_LOAN_PURPOSE_Buying a garage_mean
NAME_CASH_LOAN_PURPOSE_Buying a garage_median
NAME_CASH_LOAN_PURPOSE_Buying a garage_var
NAME_CASH_LOAN_PURPOSE_Buying a holiday home / land_mean
NAME_CASH_LOAN_PURPOSE_Buying a holiday home / land_median
NAME_CASH_LOAN_PURPOSE_Buying a holiday home / land_var
NAME_CASH_LOAN_PURPOSE_Buying a home_mean
NAME_CASH_LOAN_PURPOSE_Buying a home_median
NAME_CASH_LOAN_PURPOSE_Buying a home_var
NAME_CASH_LOAN_PURPOSE_Buying a new car_mean
NAME_CASH_LOAN_PURPOSE_Buying a new car_median
NAME_CASH_LOAN_PURPOSE_Buying a new car_var
NAME_CASH_LOAN_PURPOSE_Buying a used car_mean
NAME_CASH_LOAN_PURPOSE_Buying a used car_median
NAME_CASH_LOAN_PURPOSE_Buying a used car_var
NAME_CASH_LOAN_PURPOSE_Car repairs_mean
NAME_CASH_LOAN_PURPOSE_Car repairs_median
NAME_CASH_LOAN_PURPOSE_Car repairs_var
NAME_CASH_LOAN_PURPOSE_Education_mean
NAME_CASH_LOAN_PURPOSE_Education_median
NAME_CASH_LOAN_PURPOSE_Education_var
NAME_CASH_LOAN_PURPOSE_Everyday expenses_mean
NAME_CASH_LOAN_PURPOSE_Everyday expenses_median
NAME_CASH_LOAN_PURPOSE_Everyday expenses_var
NAME_CASH_LOAN_PURPOSE_Furniture_mean
NAME_CASH_LOAN_PURPOSE_Furniture_median
NAME_CASH_LOAN_PURPOSE_Furniture_var
NAME_CASH_LOAN_PURPOSE_Gasification / water supply_mean
NAME_CASH_LOAN_PURPOSE_Gasification / water supply_median
NAME_CASH_LOAN_PURPOSE_Gasification / water supply_var
NAME_CASH_LOAN_PURPOSE_Hobby_mean
NAME_CASH_LOAN_PURPOSE_Hobby_median
NAME_CASH_LOAN_PURPOSE_Hobby_var
NAME_CASH_LOAN_PURPOSE_Journey_mean
NAME_CASH_LOAN_PURPOSE_Journey_median
NAME_CASH_LOAN_PURPOSE_Journey_var
NAME_CASH_LOAN_PURPOSE_Medicine_mean
NAME_CASH_LOAN_PURPOSE_Medicine_median
NAME_CASH_LOAN_PURPOSE_Medicine_var
NAME_CASH_LOAN_PURPOSE_Money for a third person_mean
NAME_CASH_LOAN_PURPOSE_Money for a third person_median
NAME_CASH_LOAN_PURPOSE_Money for a third person_var
NAME_CASH_LOAN_PURPOSE_Other_mean
NAME_CASH_LOAN_PURPOSE_Other_median
NAME_CASH_LOAN_PURPOSE_Other_var
NAME_CASH_LOAN_PURPOSE_Payments on other loans_mean
NAME_CASH_LOAN_PURPOSE_Payments on other loans_median

NAME_CASH_LOAN_PURPOSE_Payments on other loans_var
NAME_CASH_LOAN_PURPOSE_Purchase of electronic equipment_mean
NAME_CASH_LOAN_PURPOSE_Purchase of electronic equipment_median
NAME_CASH_LOAN_PURPOSE_Purchase of electronic equipment_var
NAME_CASH_LOAN_PURPOSE_Refusal to name the goal_mean
NAME_CASH_LOAN_PURPOSE_Refusal to name the goal_median
NAME_CASH_LOAN_PURPOSE_Refusal to name the goal_var
NAME_CASH_LOAN_PURPOSE_Repairs_mean
NAME_CASH_LOAN_PURPOSE_Repairs_median
NAME_CASH_LOAN_PURPOSE_Repairs_var
NAME_CASH_LOAN_PURPOSE_Urgent needs_mean
NAME_CASH_LOAN_PURPOSE_Urgent needs_median
NAME_CASH_LOAN_PURPOSE_Urgent needs_var
NAME_CASH_LOAN_PURPOSE_Wedding / gift / holiday_mean
NAME_CASH_LOAN_PURPOSE_Wedding / gift / holiday_median
NAME_CASH_LOAN_PURPOSE_Wedding / gift / holiday_var
NAME_CASH_LOAN_PURPOSE_XAP_mean
NAME_CASH_LOAN_PURPOSE_XAP_median
NAME_CASH_LOAN_PURPOSE_XAP_var
NAME_CASH_LOAN_PURPOSE_XNA_mean
NAME_CASH_LOAN_PURPOSE_XNA_median
NAME_CASH_LOAN_PURPOSE_XNA_var
NAME_CLIENT_TYPE_New_mean
NAME_CLIENT_TYPE_New_median
NAME_CLIENT_TYPE_New_var
NAME_CLIENT_TYPE_Refreshed_mean
NAME_CLIENT_TYPE_Refreshed_median
NAME_CLIENT_TYPE_Refreshed_var
NAME_CLIENT_TYPE_Repeater_mean
NAME_CLIENT_TYPE_Repeater_median
NAME_CLIENT_TYPE_Repeater_var
NAME_CLIENT_TYPE_XNA_mean
NAME_CLIENT_TYPE_XNA_median
NAME_CLIENT_TYPE_XNA_var
NAME_CONTRACT_STATUS_Approved_mean
NAME_CONTRACT_STATUS_Approved_median
NAME_CONTRACT_STATUS_Approved_var
NAME_CONTRACT_STATUS_Canceled_mean
NAME_CONTRACT_STATUS_Canceled_median
NAME_CONTRACT_STATUS_Canceled_var
NAME_CONTRACT_STATUS_Refused_mean
NAME_CONTRACT_STATUS_Refused_median
NAME_CONTRACT_STATUS_Refused_var
NAME_CONTRACT_STATUS_Unused offer_mean
NAME_CONTRACT_STATUS_Unused offer_median
NAME_CONTRACT_STATUS_Unused offer_var
NAME_CONTRACT_TYPE_Cash loans_mean
NAME_CONTRACT_TYPE_Cash loans_median
NAME_CONTRACT_TYPE_Cash loans_var
NAME_CONTRACT_TYPE_Consumer loans_mean
NAME_CONTRACT_TYPE_Consumer loans_median
NAME_CONTRACT_TYPE_Consumer loans_var
NAME_CONTRACT_TYPE_Revolving loans_mean
NAME_CONTRACT_TYPE_Revolving loans_median
NAME_CONTRACT_TYPE_Revolving loans_var
NAME_CONTRACT_TYPE_XNA_mean
NAME_CONTRACT_TYPE_XNA_median
NAME_CONTRACT_TYPE_XNA_var
NAME_GOODS_CATEGORY_Additional Service_mean

NAME_GOODS_CATEGORY_Additional Service_median
NAME_GOODS_CATEGORY_Additional Service_var
NAME_GOODS_CATEGORY_Animals_mean
NAME_GOODS_CATEGORY_Animals_median
NAME_GOODS_CATEGORY_Animals_var
NAME_GOODS_CATEGORY_Audio/Video_mean
NAME_GOODS_CATEGORY_Audio/Video_median
NAME_GOODS_CATEGORY_Audio/Video_var
NAME_GOODS_CATEGORY_Auto Accessories_mean
NAME_GOODS_CATEGORY_Auto Accessories_median
NAME_GOODS_CATEGORY_Auto Accessories_var
NAME_GOODS_CATEGORY_Clothing and Accessories_mean
NAME_GOODS_CATEGORY_Clothing and Accessories_median
NAME_GOODS_CATEGORY_Clothing and Accessories_var
NAME_GOODS_CATEGORY_Computers_mean
NAME_GOODS_CATEGORY_Computers_median
NAME_GOODS_CATEGORY_Computers_var
NAME_GOODS_CATEGORY_Construction Materials_mean
NAME_GOODS_CATEGORY_Construction Materials_median
NAME_GOODS_CATEGORY_Construction Materials_var
NAME_GOODS_CATEGORY_Consumer Electronics_mean
NAME_GOODS_CATEGORY_Consumer Electronics_median
NAME_GOODS_CATEGORY_Consumer Electronics_var
NAME_GOODS_CATEGORY_Direct Sales_mean
NAME_GOODS_CATEGORY_Direct Sales_median
NAME_GOODS_CATEGORY_Direct Sales_var
NAME_GOODS_CATEGORY_Education_mean
NAME_GOODS_CATEGORY_Education_median
NAME_GOODS_CATEGORY_Education_var
NAME_GOODS_CATEGORY_Fitness_mean
NAME_GOODS_CATEGORY_Fitness_median
NAME_GOODS_CATEGORY_Fitness_var
NAME_GOODS_CATEGORY_Furniture_mean
NAME_GOODS_CATEGORY_Furniture_median
NAME_GOODS_CATEGORY_Furniture_var
NAME_GOODS_CATEGORY_Gardening_mean
NAME_GOODS_CATEGORY_Gardening_median
NAME_GOODS_CATEGORY_Gardening_var
NAME_GOODS_CATEGORY_Homewares_mean
NAME_GOODS_CATEGORY_Homewares_median
NAME_GOODS_CATEGORY_Homewares_var
NAME_GOODS_CATEGORY_House Construction_mean
NAME_GOODS_CATEGORY_House Construction_median
NAME_GOODS_CATEGORY_House Construction_var
NAME_GOODS_CATEGORY_Insurance_mean
NAME_GOODS_CATEGORY_Insurance_median
NAME_GOODS_CATEGORY_Insurance_var
NAME_GOODS_CATEGORY_Jewelry_mean
NAME_GOODS_CATEGORY_Jewelry_median
NAME_GOODS_CATEGORY_Jewelry_var
NAME_GOODS_CATEGORY_Medical Supplies_mean
NAME_GOODS_CATEGORY_Medical Supplies_median
NAME_GOODS_CATEGORY_Medical Supplies_var
NAME_GOODS_CATEGORY_Medicine_mean
NAME_GOODS_CATEGORY_Medicine_median
NAME_GOODS_CATEGORY_Medicine_var
NAME_GOODS_CATEGORY_Mobile_mean
NAME_GOODS_CATEGORY_Mobile_median
NAME_GOODS_CATEGORY_Mobile_var

NAME_GOODS_CATEGORY_Office Appliances_mean
NAME_GOODS_CATEGORY_Office Appliances_median
NAME_GOODS_CATEGORY_Office Appliances_var
NAME_GOODS_CATEGORY_Other_mean
NAME_GOODS_CATEGORY_Other_median
NAME_GOODS_CATEGORY_Other_var
NAME_GOODS_CATEGORY_Photo / Cinema Equipment_mean
NAME_GOODS_CATEGORY_Photo / Cinema Equipment_median
NAME_GOODS_CATEGORY_Photo / Cinema Equipment_var
NAME_GOODS_CATEGORY_Sport and Leisure_mean
NAME_GOODS_CATEGORY_Sport and Leisure_median
NAME_GOODS_CATEGORY_Sport and Leisure_var
NAME_GOODS_CATEGORY_Tourism_mean
NAME_GOODS_CATEGORY_Tourism_median
NAME_GOODS_CATEGORY_Tourism_var
NAME_GOODS_CATEGORY_Vehicles_mean
NAME_GOODS_CATEGORY_Vehicles_median
NAME_GOODS_CATEGORY_Vehicles_var
NAME_GOODS_CATEGORY_Weapon_mean
NAME_GOODS_CATEGORY_Weapon_median
NAME_GOODS_CATEGORY_Weapon_var
NAME_GOODS_CATEGORY_XNA_mean
NAME_GOODS_CATEGORY_XNA_median
NAME_GOODS_CATEGORY_XNA_var
NAME_PAYMENT_TYPE_Cash through the bank_mean
NAME_PAYMENT_TYPE_Cash through the bank_median
NAME_PAYMENT_TYPE_Cash through the bank_var
NAME_PAYMENT_TYPE_Cashless from the account of the employer_mean
NAME_PAYMENT_TYPE_Cashless from the account of the employer_median
NAME_PAYMENT_TYPE_Cashless from the account of the employer_var
NAME_PAYMENT_TYPE_Non-cash from your account_mean
NAME_PAYMENT_TYPE_Non-cash from your account_median
NAME_PAYMENT_TYPE_Non-cash from your account_var
NAME_PAYMENT_TYPE_XNA_mean
NAME_PAYMENT_TYPE_XNA_median
NAME_PAYMENT_TYPE_XNA_var
NAME_PORTFOLIO_Cards_mean
NAME_PORTFOLIO_Cards_median
NAME_PORTFOLIO_Cards_var
NAME_PORTFOLIO_Cars_mean
NAME_PORTFOLIO_Cars_median
NAME_PORTFOLIO_Cars_var
NAME_PORTFOLIO_Cash_mean
NAME_PORTFOLIO_Cash_median
NAME_PORTFOLIO_Cash_var
NAME_PORTFOLIO_POS_mean
NAME_PORTFOLIO_POS_median
NAME_PORTFOLIO_POS_var
NAME_PORTFOLIO_XNA_mean
NAME_PORTFOLIO_XNA_median
NAME_PORTFOLIO_XNA_var
NAME_PRODUCT_TYPE_XNA_mean
NAME_PRODUCT_TYPE_XNA_median
NAME_PRODUCT_TYPE_XNA_var
NAME_PRODUCT_TYPE_walk-in_mean
NAME_PRODUCT_TYPE_walk-in_median
NAME_PRODUCT_TYPE_walk-in_var
NAME_PRODUCT_TYPE_x-sell_mean
NAME_PRODUCT_TYPE_x-sell_median

NAME_PRODUCT_TYPE_x-sell_var
NAME_SELLER_INDUSTRY_Auto technology_mean
NAME_SELLER_INDUSTRY_Auto technology_median
NAME_SELLER_INDUSTRY_Auto technology_var
NAME_SELLER_INDUSTRY_Clothing_mean
NAME_SELLER_INDUSTRY_Clothing_median
NAME_SELLER_INDUSTRY_Clothing_var
NAME_SELLER_INDUSTRY_Connectivity_mean
NAME_SELLER_INDUSTRY_Connectivity_median
NAME_SELLER_INDUSTRY_Connectivity_var
NAME_SELLER_INDUSTRY_Construction_mean
NAME_SELLER_INDUSTRY_Construction_median
NAME_SELLER_INDUSTRY_Construction_var
NAME_SELLER_INDUSTRY_Consumer electronics_mean
NAME_SELLER_INDUSTRY_Consumer electronics_median
NAME_SELLER_INDUSTRY_Consumer electronics_var
NAME_SELLER_INDUSTRY_Furniture_mean
NAME_SELLER_INDUSTRY_Furniture_median
NAME_SELLER_INDUSTRY_Furniture_var
NAME_SELLER_INDUSTRY_Industry_mean
NAME_SELLER_INDUSTRY_Industry_median
NAME_SELLER_INDUSTRY_Industry_var
NAME_SELLER_INDUSTRY_Jewelry_mean
NAME_SELLER_INDUSTRY_Jewelry_median
NAME_SELLER_INDUSTRY_Jewelry_var
NAME_SELLER_INDUSTRY_MLM partners_mean
NAME_SELLER_INDUSTRY_MLM partners_median
NAME_SELLER_INDUSTRY_MLM partners_var
NAME_SELLER_INDUSTRY_Tourism_mean
NAME_SELLER_INDUSTRY_Tourism_median
NAME_SELLER_INDUSTRY_Tourism_var
NAME_SELLER_INDUSTRY_XNA_mean
NAME_SELLER_INDUSTRY_XNA_median
NAME_SELLER_INDUSTRY_XNA_var
NAME_TYPE_SUITE_Children_mean
NAME_TYPE_SUITE_Children_median
NAME_TYPE_SUITE_Children_var
NAME_TYPE_SUITE_Family_mean
NAME_TYPE_SUITE_Family_median
NAME_TYPE_SUITE_Family_var
NAME_TYPE_SUITE_Group of people_mean
NAME_TYPE_SUITE_Group of people_median
NAME_TYPE_SUITE_Group of people_var
NAME_TYPE_SUITE_Other_A_mean
NAME_TYPE_SUITE_Other_A_median
NAME_TYPE_SUITE_Other_A_var
NAME_TYPE_SUITE_Other_B_mean
NAME_TYPE_SUITE_Other_B_median
NAME_TYPE_SUITE_Other_B_var
NAME_TYPE_SUITE_Spouse, partner_mean
NAME_TYPE_SUITE_Spouse, partner_median
NAME_TYPE_SUITE_Spouse, partner_var
NAME_TYPE_SUITE_Unaccompanied_mean
NAME_TYPE_SUITE_Unaccompanied_median
NAME_TYPE_SUITE_Unaccompanied_var
NAME_YIELD_GROUP_XNA_mean
NAME_YIELD_GROUP_XNA_median
NAME_YIELD_GROUP_XNA_var
NAME_YIELD_GROUP_high_mean

NAME_YIELD_GROUP_high_median
NAME_YIELD_GROUP_high_var
NAME_YIELD_GROUP_low_action_mean
NAME_YIELD_GROUP_low_action_median
NAME_YIELD_GROUP_low_action_var
NAME_YIELD_GROUP_low_normal_mean
NAME_YIELD_GROUP_low_normal_median
NAME_YIELD_GROUP_low_normal_var
NAME_YIELD_GROUP_middle_mean
NAME_YIELD_GROUP_middle_median
NAME_YIELD_GROUP_middle_var
NFLAG_INSURED_ON_APPROVAL_count
NFLAG_INSURED_ON_APPROVAL_max
NFLAG_INSURED_ON_APPROVAL_mean
NFLAG_INSURED_ON_APPROVAL_median
NFLAG_INSURED_ON_APPROVAL_min
NFLAG_INSURED_ON_APPROVAL_sum
NFLAG_INSURED_ON_APPROVAL_var
NFLAG_LAST_APPL_IN_DAY_count
NFLAG_LAST_APPL_IN_DAY_max
NFLAG_LAST_APPL_IN_DAY_mean
NFLAG_LAST_APPL_IN_DAY_median
NFLAG_LAST_APPL_IN_DAY_min
NFLAG_LAST_APPL_IN_DAY_sum
NFLAG_LAST_APPL_IN_DAY_var
PREV_APCTN_CRDT_DIFF_count
PREV_APCTN_CRDT_DIFF_max
PREV_APCTN_CRDT_DIFF_mean
PREV_APCTN_CRDT_DIFF_median
PREV_APCTN_CRDT_DIFF_min
PREV_APCTN_CRDT_DIFF_sum
PREV_APCTN_CRDT_DIFF_var
PREV_APCTN_CRDT_RATIO_count
PREV_APCTN_CRDT_RATIO_max
PREV_APCTN_CRDT_RATIO_mean
PREV_APCTN_CRDT_RATIO_median
PREV_APCTN_CRDT_RATIO_min
PREV_APCTN_CRDT_RATIO_sum
PREV_APCTN_CRDT_RATIO_var
PREV_APRV_CNT_count
PREV_APRV_CNT_max
PREV_APRV_CNT_mean
PREV_APRV_CNT_median
PREV_APRV_CNT_min
PREV_APRV_CNT_sum
PREV_APRV_CNT_var
PREV_CRDT_ANNUTY_RATIO_count
PREV_CRDT_ANNUTY_RATIO_max
PREV_CRDT_ANNUTY_RATIO_mean
PREV_CRDT_ANNUTY_RATIO_median
PREV_CRDT_ANNUTY_RATIO_min
PREV_CRDT_ANNUTY_RATIO_sum
PREV_CRDT_ANNUTY_RATIO_var
PREV_DWN_PYMNT_CRDT_RATIO_count
PREV_DWN_PYMNT_CRDT_RATIO_max
PREV_DWN_PYMNT_CRDT_RATIO_mean
PREV_DWN_PYMNT_CRDT_RATIO_median
PREV_DWN_PYMNT_CRDT_RATIO_min
PREV_DWN_PYMNT_CRDT_RATIO_sum

PREV_DWN_PYMNT_CRDT_RATIO_var
PREV_REJ_CNT_count
PREV_REJ_CNT_max
PREV_REJ_CNT_mean
PREV_REJ_CNT_median
PREV_REJ_CNT_min
PREV_REJ_CNT_sum
PREV_REJ_CNT_var
PRODUCT_COMBINATION_Card Street_mean
PRODUCT_COMBINATION_Card Street_median
PRODUCT_COMBINATION_Card Street_var
PRODUCT_COMBINATION_Card X-Sell_mean
PRODUCT_COMBINATION_Card X-Sell_median
PRODUCT_COMBINATION_Card X-Sell_var
PRODUCT_COMBINATION_Cash Street: high_mean
PRODUCT_COMBINATION_Cash Street: high_median
PRODUCT_COMBINATION_Cash Street: high_var
PRODUCT_COMBINATION_Cash Street: low_mean
PRODUCT_COMBINATION_Cash Street: low_median
PRODUCT_COMBINATION_Cash Street: low_var
PRODUCT_COMBINATION_Cash Street: middle_mean
PRODUCT_COMBINATION_Cash Street: middle_median
PRODUCT_COMBINATION_Cash Street: middle_var
PRODUCT_COMBINATION_Cash X-Sell: high_mean
PRODUCT_COMBINATION_Cash X-Sell: high_median
PRODUCT_COMBINATION_Cash X-Sell: high_var
PRODUCT_COMBINATION_Cash X-Sell: low_mean
PRODUCT_COMBINATION_Cash X-Sell: low_median
PRODUCT_COMBINATION_Cash X-Sell: low_var
PRODUCT_COMBINATION_Cash X-Sell: middle_mean
PRODUCT_COMBINATION_Cash X-Sell: middle_median
PRODUCT_COMBINATION_Cash X-Sell: middle_var
PRODUCT_COMBINATION_Cash_mean
PRODUCT_COMBINATION_Cash_median
PRODUCT_COMBINATION_Cash_var
PRODUCT_COMBINATION_POS household with interest_mean
PRODUCT_COMBINATION_POS household with interest_median
PRODUCT_COMBINATION_POS household with interest_var
PRODUCT_COMBINATION_POS household without interest_mean
PRODUCT_COMBINATION_POS household without interest_median
PRODUCT_COMBINATION_POS household without interest_var
PRODUCT_COMBINATION_POS industry with interest_mean
PRODUCT_COMBINATION_POS industry with interest_median
PRODUCT_COMBINATION_POS industry with interest_var
PRODUCT_COMBINATION_POS industry without interest_mean
PRODUCT_COMBINATION_POS industry without interest_median
PRODUCT_COMBINATION_POS industry without interest_var
PRODUCT_COMBINATION_POS mobile with interest_mean
PRODUCT_COMBINATION_POS mobile with interest_median
PRODUCT_COMBINATION_POS mobile with interest_var
PRODUCT_COMBINATION_POS mobile without interest_mean
PRODUCT_COMBINATION_POS mobile without interest_median
PRODUCT_COMBINATION_POS mobile without interest_var
PRODUCT_COMBINATION_POS other with interest_mean
PRODUCT_COMBINATION_POS other with interest_median
PRODUCT_COMBINATION_POS other with interest_var
PRODUCT_COMBINATION_POS others without interest_mean
PRODUCT_COMBINATION_POS others without interest_median
PRODUCT_COMBINATION_POS others without interest_var

```
RATE_DOWN_PAYMENT_count
RATE_DOWN_PAYMENT_max
RATE_DOWN_PAYMENT_mean
RATE_DOWN_PAYMENT_median
RATE_DOWN_PAYMENT_min
RATE_DOWN_PAYMENT_sum
RATE_DOWN_PAYMENT_var
SELLERPLACE_AREA_count
SELLERPLACE_AREA_max
SELLERPLACE_AREA_mean
SELLERPLACE_AREA_median
SELLERPLACE_AREA_min
SELLERPLACE_AREA_sum
SELLERPLACE_AREA_var
SK_ID_PREV_count
WEEKDAY_APPR_PROCESS_START_FRIDAY_mean
WEEKDAY_APPR_PROCESS_START_FRIDAY_median
WEEKDAY_APPR_PROCESS_START_FRIDAY_var
WEEKDAY_APPR_PROCESS_START_MONDAY_mean
WEEKDAY_APPR_PROCESS_START_MONDAY_median
WEEKDAY_APPR_PROCESS_START_MONDAY_var
WEEKDAY_APPR_PROCESS_START_SATURDAY_mean
WEEKDAY_APPR_PROCESS_START_SATURDAY_median
WEEKDAY_APPR_PROCESS_START_SATURDAY_var
WEEKDAY_APPR_PROCESS_START_SUNDAY_mean
WEEKDAY_APPR_PROCESS_START_SUNDAY_median
WEEKDAY_APPR_PROCESS_START_SUNDAY_var
WEEKDAY_APPR_PROCESS_START_THURSDAY_mean
WEEKDAY_APPR_PROCESS_START_THURSDAY_median
WEEKDAY_APPR_PROCESS_START_THURSDAY_var
WEEKDAY_APPR_PROCESS_START_TUESDAY_mean
WEEKDAY_APPR_PROCESS_START_TUESDAY_median
WEEKDAY_APPR_PROCESS_START_TUESDAY_var
WEEKDAY_APPR_PROCESS_START_WEDNESDAY_mean
WEEKDAY_APPR_PROCESS_START_WEDNESDAY_median
WEEKDAY_APPR_PROCESS_START_WEDNESDAY_var

Aggregated Data:

                                                        count        mean  \
HOUR_APPR_PROCESS_START_var                          278399.0    5.977447
PREV_APCTN_CRDT_RATIO_sum                            338856.0    3.744165
NAME_GOODS_CATEGORY_Gardening_median                 338857.0    0.001422
NAME_CONTRACT_STATUS_Unused offer_mean               338857.0    0.014921
NAME_CASH_LOAN_PURPOSE_Money for a third person...  338857.0    0.000014
...                                                       ...         ...
NAME_CASH_LOAN_PURPOSE_Furniture_mean                338857.0    0.000427
NAME_CONTRACT_TYPE_Cash loans_median                 338857.0    0.321591
SELLERPLACE_AREA_median                              338857.0  306.766475
NAME_TYPE_SUITE_Other_A_var                          278399.0    0.005689
NAME_PAYMENT_TYPE_XNA_var                            278399.0    0.205268

                                                            std  min  \
HOUR_APPR_PROCESS_START_var                            6.012819  0.0
PREV_APCTN_CRDT_RATIO_sum                              2.962858  0.0
NAME_GOODS_CATEGORY_Gardening_median                   0.034967  0.0
NAME_CONTRACT_STATUS_Unused offer_mean                 0.068630  0.0
NAME_CASH_LOAN_PURPOSE_Money for a third person...     0.002668  0.0
...                                                         ...  ...
```

```
NAME_CASH_LOAN_PURPOSE_Furniture_mean                  0.013645  0.0
NAME_CONTRACT_TYPE_Cash loans_median                   0.439068  0.0
SELLERPLACE_AREA_median                             9857.957067 -1.0
NAME_TYPE_SUITE_Other_A_var                            0.040032  0.0
NAME_PAYMENT_TYPE_XNA_var                              0.147106  0.0

                                                          25%       50%  \
HOUR_APPR_PROCESS_START_var                          2.000000   4.50000
PREV_APCTN_CRDT_RATIO_sum                            1.827841   2.94966
NAME_GOODS_CATEGORY_Gardening_median                 0.000000   0.00000
NAME_CONTRACT_STATUS_Unused offer_mean               0.000000   0.00000
NAME_CASH_LOAN_PURPOSE_Money for a third person...  0.000000   0.00000
...                                                       ...       ...
NAME_CASH_LOAN_PURPOSE_Furniture_mean                0.000000   0.00000
NAME_CONTRACT_TYPE_Cash loans_median                 0.000000   0.00000
SELLERPLACE_AREA_median                             -1.000000  28.00000
NAME_TYPE_SUITE_Other_A_var                          0.000000   0.00000
NAME_PAYMENT_TYPE_XNA_var                            0.000000   0.25000

                                                          75%           max
HOUR_APPR_PROCESS_START_var                          8.142857  1.805000e+02
PREV_APCTN_CRDT_RATIO_sum                            4.925291  6.650068e+01
NAME_GOODS_CATEGORY_Gardening_median                 0.000000  1.000000e+00
NAME_CONTRACT_STATUS_Unused offer_mean               0.000000  1.000000e+00
NAME_CASH_LOAN_PURPOSE_Money for a third person...   0.000000  1.000000e+00
...                                                       ...           ...
NAME_CASH_LOAN_PURPOSE_Furniture_mean                0.000000  1.000000e+00
NAME_CONTRACT_TYPE_Cash loans_median                 1.000000  1.000000e+00
SELLERPLACE_AREA_median                            103.000000  4.000000e+06
NAME_TYPE_SUITE_Other_A_var                          0.000000  5.000000e-01
NAME_PAYMENT_TYPE_XNA_var                            0.300000  5.000000e-01

[589 rows x 8 columns]
```

CCB - Credit card Balance EDA

1. This function is specifically for Credit Card Balance table. Below are the pre-processing done before passing this table into the pipeline.

    - Absolute value of the Months Balance is used as that allows only use of postive values without effecting the correlation, as it is realtive to the application date.

    - Any column or row with more than 70% of its data as null will be deleted from the dataframe as the threshold is set to .7.

    - Similar to above, once processed, store the transformed csv file. Benefit of this is that we can then pass it directly to model for merging into application train/test table. We do not have to perform expensive EDA/ETL/Transformation everytime we want to process the same data.

In [23]:

```
def ccb_eda(df):
    ccb = df

    #Adding new features
    ccb['MONTHS_BALANCE'] = ccb['MONTHS_BALANCE'].abs()

    #replacing " " with _ for OHE cols.
    ccb['NAME_CONTRACT_STATUS']=ccb['NAME_CONTRACT_STATUS'].apply(lambda x:

    threshold = 0.7

    #Dropping columns with missing value rate higher than threshold
    ccb = ccb[ccb.columns[ccb.isnull().mean() < threshold]]

    #Dropping rows with missing value rate higher than threshold
    ccb = ccb.loc[ccb.isnull().mean(axis=1) < threshold]

    return (eda_transformation(ccb,4))
```

In [24]:

```
ccb = datasets['credit_card_balance']
ccb = ccb_eda(ccb)
datasets_transformed['credit_card_balance'] = ccb

#ccb.to_csv(os.getcwd() + DATA_DIR + 'ccb_agg_data.csv')
```

```
--------
# of ID's: 1
 ID's:
['SK_ID_CURR']

--------
# All features: 22
All features:
['CNT_INSTALMENT_MATURE_CUM', 'AMT_RECEIVABLE_PRINCIPAL', 'AMT_TOTAL_RECEIVA
BLE', 'NAME_CONTRACT_STATUS', 'AMT_BALANCE', 'AMT_RECIVABLE', 'SK_ID_PREV',
'AMT_DRAWINGS_POS_CURRENT', 'AMT_PAYMENT_TOTAL_CURRENT', 'AMT_PAYMENT_CURREN
T', 'AMT_CREDIT_LIMIT_ACTUAL', 'SK_DPD_DEF', 'CNT_DRAWINGS_OTHER_CURRENT', '
AMT_DRAWINGS_OTHER_CURRENT', 'AMT_DRAWINGS_CURRENT', 'AMT_INST_MIN_REGULARIT
Y', 'AMT_DRAWINGS_ATM_CURRENT', 'CNT_DRAWINGS_CURRENT', 'MONTHS_BALANCE', 'S
K_DPD', 'CNT_DRAWINGS_ATM_CURRENT', 'CNT_DRAWINGS_POS_CURRENT']

Missing data:
                             Total    Percent
AMT_PAYMENT_CURRENT         767988  19.998063
CNT_DRAWINGS_POS_CURRENT    749816  19.524872
CNT_DRAWINGS_ATM_CURRENT    749816  19.524872
AMT_DRAWINGS_ATM_CURRENT    749816  19.524872
AMT_DRAWINGS_OTHER_CURRENT  749816  19.524872
CNT_DRAWINGS_OTHER_CURRENT  749816  19.524872
AMT_DRAWINGS_POS_CURRENT    749816  19.524872
AMT_INST_MIN_REGULARITY     305236   7.948208
CNT_INSTALMENT_MATURE_CUM   305236   7.948208
AMT_PAYMENT_TOTAL_CURRENT        0   0.000000
AMT_CREDIT_LIMIT_ACTUAL          0   0.000000
AMT_RECEIVABLE_PRINCIPAL         0   0.000000
SK_ID_PREV                       0   0.000000
AMT_RECIVABLE                    0   0.000000
AMT_DRAWINGS_CURRENT             0   0.000000
AMT_BALANCE                      0   0.000000
NAME_CONTRACT_STATUS             0   0.000000
CNT_DRAWINGS_CURRENT             0   0.000000
MONTHS_BALANCE                   0   0.000000
SK_DPD                           0   0.000000
AMT_TOTAL_RECEIVABLE             0   0.000000
SK_DPD_DEF                       0   0.000000

--------
# of Numerical features: 22
Numerical features:
['SK_ID_PREV', 'SK_ID_CURR', 'MONTHS_BALANCE', 'AMT_BALANCE', 'AMT_CREDIT_LI
MIT_ACTUAL', 'AMT_DRAWINGS_ATM_CURRENT', 'AMT_DRAWINGS_CURRENT', 'AMT_DRAWIN
GS_OTHER_CURRENT', 'AMT_DRAWINGS_POS_CURRENT', 'AMT_INST_MIN_REGULARITY', 'A
MT_PAYMENT_CURRENT', 'AMT_PAYMENT_TOTAL_CURRENT', 'AMT_RECEIVABLE_PRINCIPAL
', 'AMT_RECIVABLE', 'AMT_TOTAL_RECEIVABLE', 'CNT_DRAWINGS_ATM_CURRENT', 'CNT
_DRAWINGS_CURRENT', 'CNT_DRAWINGS_OTHER_CURRENT', 'CNT_DRAWINGS_POS_CURRENT
', 'CNT_INSTALMENT_MATURE_CUM', 'SK_DPD', 'SK_DPD_DEF']

Numerical Statistical Summary:

         SK_ID_PREV    SK_ID_CURR  MONTHS_BALANCE   AMT_BALANCE  \
count  3.840312e+06  3.840312e+06    3.840312e+06  3.840312e+06
mean   1.904504e+06  2.783242e+05    3.452192e+01  5.830016e+04
std    5.364695e+05  1.027045e+05    2.666775e+01  1.063070e+05
min    1.000018e+06  1.000060e+05    1.000000e+00 -4.202502e+05
```

25%	1.434385e+06	1.895170e+05	1.100000e+01	0.000000e+00
50%	1.897122e+06	2.783960e+05	2.800000e+01	0.000000e+00
75%	2.369328e+06	3.675800e+05	5.500000e+01	8.904669e+04
max	2.843496e+06	4.562500e+05	9.600000e+01	1.505902e+06

	AMT_CREDIT_LIMIT_ACTUAL	AMT_DRAWINGS_ATM_CURRENT	\
count	3.840312e+06	3.090496e+06	
mean	1.538080e+05	5.961325e+03	
std	1.651457e+05	2.822569e+04	
min	0.000000e+00	-6.827310e+03	
25%	4.500000e+04	0.000000e+00	
50%	1.125000e+05	0.000000e+00	
75%	1.800000e+05	0.000000e+00	
max	1.350000e+06	2.115000e+06	

	AMT_DRAWINGS_CURRENT	AMT_DRAWINGS_OTHER_CURRENT	\
count	3.840312e+06	3.090496e+06	
mean	7.433388e+03	2.881696e+02	
std	3.384608e+04	8.201989e+03	
min	-6.211620e+03	0.000000e+00	
25%	0.000000e+00	0.000000e+00	
50%	0.000000e+00	0.000000e+00	
75%	0.000000e+00	0.000000e+00	
max	2.287098e+06	1.529847e+06	

	AMT_DRAWINGS_POS_CURRENT	AMT_INST_MIN_REGULARITY	...	\
count	3.090496e+06	3.535076e+06	...	
mean	2.968805e+03	3.540204e+03	...	
std	2.079689e+04	5.600154e+03	...	
min	0.000000e+00	0.000000e+00	...	
25%	0.000000e+00	0.000000e+00	...	
50%	0.000000e+00	0.000000e+00	...	
75%	0.000000e+00	6.633911e+03	...	
max	2.239274e+06	2.028820e+05	...	

	AMT_RECEIVABLE_PRINCIPAL	AMT_RECIVABLE	AMT_TOTAL_RECEIVABLE	\
count	3.840312e+06	3.840312e+06	3.840312e+06	
mean	5.596588e+04	5.808881e+04	5.809829e+04	
std	1.025336e+05	1.059654e+05	1.059718e+05	
min	-4.233058e+05	-4.202502e+05	-4.202502e+05	
25%	0.000000e+00	0.000000e+00	0.000000e+00	
50%	0.000000e+00	0.000000e+00	0.000000e+00	
75%	8.535924e+04	8.889949e+04	8.891451e+04	
max	1.472317e+06	1.493338e+06	1.493338e+06	

	CNT_DRAWINGS_ATM_CURRENT	CNT_DRAWINGS_CURRENT	\
count	3.090496e+06	3.840312e+06	
mean	3.094490e-01	7.031439e-01	
std	1.100401e+00	3.190347e+00	
min	0.000000e+00	0.000000e+00	
25%	0.000000e+00	0.000000e+00	
50%	0.000000e+00	0.000000e+00	
75%	0.000000e+00	0.000000e+00	
max	5.100000e+01	1.650000e+02	

	CNT_DRAWINGS_OTHER_CURRENT	CNT_DRAWINGS_POS_CURRENT	\
count	3.090496e+06	3.090496e+06	
mean	4.812496e-03	5.594791e-01	
std	8.263861e-02	3.240649e+00	

```
min                  0.000000e+00              0.000000e+00
25%                  0.000000e+00              0.000000e+00
50%                  0.000000e+00              0.000000e+00
75%                  0.000000e+00              0.000000e+00
max                  1.200000e+01              1.650000e+02

       CNT_INSTALMENT_MATURE_CUM        SK_DPD    SK_DPD_DEF
count               3.535076e+06  3.840312e+06  3.840312e+06
mean                2.082508e+01  9.283667e+00  3.316220e-01
std                 2.005149e+01  9.751570e+01  2.147923e+01
min                 0.000000e+00  0.000000e+00  0.000000e+00
25%                 4.000000e+00  0.000000e+00  0.000000e+00
50%                 1.500000e+01  0.000000e+00  0.000000e+00
75%                 3.200000e+01  0.000000e+00  0.000000e+00
max                 1.200000e+02  3.260000e+03  3.260000e+03

[8 rows x 22 columns]

--------
# of Categorical features: 1
Categorical features:
['NAME_CONTRACT_STATUS']

Categorical Statistical Summary:

       NAME_CONTRACT_STATUS
count               3840312
unique                    7
top                  Active
freq                3698436

Categories:

  NAME_CONTRACT_STATUS
0               Active
1            Completed
2               Demand
3               Signed
4        Sent_proposal
5              Refused
6             Approved

--------
# of OHE categorical features: 7
OHE Categorical features: ['NAME_CONTRACT_STATUS_Refused', 'NAME_CONTRACT_ST
ATUS_Demand', 'NAME_CONTRACT_STATUS_Approved', 'NAME_CONTRACT_STATUS_Signed
', 'NAME_CONTRACT_STATUS_Sent_proposal', 'NAME_CONTRACT_STATUS_Active', 'NAM
E_CONTRACT_STATUS_Completed']
--------
df.shape: (3840312, 29)

Aggregated Features:
df[['CNT_INSTALMENT_MATURE_CUM', 'AMT_RECEIVABLE_PRINCIPAL', 'NAME_CONTRACT_
STATUS_Refused', 'AMT_TOTAL_RECEIVABLE', 'NAME_CONTRACT_STATUS_Approved', 'A
MT_BALANCE', 'AMT_RECIVABLE', 'NAME_CONTRACT_STATUS_Signed', 'SK_ID_PREV', '
NAME_CONTRACT_STATUS_Sent_proposal', 'AMT_DRAWINGS_POS_CURRENT', 'AMT_PAYMEN
T_TOTAL_CURRENT', 'NAME_CONTRACT_STATUS_Active', 'AMT_CREDIT_LIMIT_ACTUAL',
'AMT_PAYMENT_CURRENT', 'SK_DPD_DEF', 'CNT_DRAWINGS_OTHER_CURRENT', 'AMT_DRAW
INGS_OTHER_CURRENT', 'AMT_DRAWINGS_CURRENT', 'AMT_INST_MIN_REGULARITY', 'NAM
```

```
E_CONTRACT_STATUS_Demand', 'AMT_DRAWINGS_ATM_CURRENT', 'CNT_DRAWINGS_CURRENT
', 'MONTHS_BALANCE', 'NAME_CONTRACT_STATUS_Completed', 'SK_DPD', 'CNT_DRAWIN
GS_ATM_CURRENT', 'CNT_DRAWINGS_POS_CURRENT']][0:5]:
   CNT_INSTALMENT_MATURE_CUM  AMT_RECEIVABLE_PRINCIPAL  \
0                       35.0                      0.000
1                       69.0                  60175.080
2                       30.0                  26926.425
3                       10.0                 224949.285
4                      101.0                 443044.395

   NAME_CONTRACT_STATUS_Refused  AMT_TOTAL_RECEIVABLE  \
0                             0                 0.000
1                             0             64875.555
2                             0             31460.085
3                             0            233048.970
4                             0            453919.455

   NAME_CONTRACT_STATUS_Approved  AMT_BALANCE  AMT_RECIVABLE  \
0                              0       56.970          0.000
1                              0    63975.555      64875.555
2                              0    31815.225      31460.085
3                              0   236572.110     233048.970
4                              0   453919.455     453919.455

   NAME_CONTRACT_STATUS_Signed  SK_ID_PREV  \
0                            0     2562384
1                            0     2582071
2                            0     1740877
3                            0     1389973
4                            0     1891521

   NAME_CONTRACT_STATUS_Sent_proposal  ...  AMT_DRAWINGS_CURRENT  \
0                                   0  ...                 877.5
1                                   0  ...                2250.0
2                                   0  ...                   0.0
3                                   0  ...                2250.0
4                                   0  ...               11547.0

   AMT_INST_MIN_REGULARITY  NAME_CONTRACT_STATUS_Demand  \
0                 1700.325                            0
1                 2250.000                            0
2                 2250.000                            0
3                11795.760                            0
4                22924.890                            0

   AMT_DRAWINGS_ATM_CURRENT  CNT_DRAWINGS_CURRENT  MONTHS_BALANCE  \
0                       0.0                     1               6
1                    2250.0                     1               1
2                       0.0                     0               7
3                    2250.0                     1               4
4                       0.0                     1               1

   NAME_CONTRACT_STATUS_Completed  SK_DPD  CNT_DRAWINGS_ATM_CURRENT  \
0                               0       0                       0.0
1                               0       0                       1.0
2                               0       0                       0.0
3                               0       0                       1.0
4                               0       0                       0.0
```

```
   CNT_DRAWINGS_POS_CURRENT
0                       1.0
1                       0.0
2                       0.0
3                       0.0
4                       1.0

[5 rows x 28 columns]
--------
Aggregated Features:
AMT_BALANCE_count
AMT_BALANCE_max
AMT_BALANCE_mean
AMT_BALANCE_median
AMT_BALANCE_min
AMT_BALANCE_sum
AMT_BALANCE_var
AMT_CREDIT_LIMIT_ACTUAL_count
AMT_CREDIT_LIMIT_ACTUAL_max
AMT_CREDIT_LIMIT_ACTUAL_mean
AMT_CREDIT_LIMIT_ACTUAL_median
AMT_CREDIT_LIMIT_ACTUAL_min
AMT_CREDIT_LIMIT_ACTUAL_sum
AMT_CREDIT_LIMIT_ACTUAL_var
AMT_DRAWINGS_ATM_CURRENT_count
AMT_DRAWINGS_ATM_CURRENT_max
AMT_DRAWINGS_ATM_CURRENT_mean
AMT_DRAWINGS_ATM_CURRENT_median
AMT_DRAWINGS_ATM_CURRENT_min
AMT_DRAWINGS_ATM_CURRENT_sum
AMT_DRAWINGS_ATM_CURRENT_var
AMT_DRAWINGS_CURRENT_count
AMT_DRAWINGS_CURRENT_max
AMT_DRAWINGS_CURRENT_mean
AMT_DRAWINGS_CURRENT_median
AMT_DRAWINGS_CURRENT_min
AMT_DRAWINGS_CURRENT_sum
AMT_DRAWINGS_CURRENT_var
AMT_DRAWINGS_OTHER_CURRENT_count
AMT_DRAWINGS_OTHER_CURRENT_max
AMT_DRAWINGS_OTHER_CURRENT_mean
AMT_DRAWINGS_OTHER_CURRENT_median
AMT_DRAWINGS_OTHER_CURRENT_min
AMT_DRAWINGS_OTHER_CURRENT_sum
AMT_DRAWINGS_OTHER_CURRENT_var
AMT_DRAWINGS_POS_CURRENT_count
AMT_DRAWINGS_POS_CURRENT_max
AMT_DRAWINGS_POS_CURRENT_mean
AMT_DRAWINGS_POS_CURRENT_median
AMT_DRAWINGS_POS_CURRENT_min
AMT_DRAWINGS_POS_CURRENT_sum
AMT_DRAWINGS_POS_CURRENT_var
AMT_INST_MIN_REGULARITY_count
AMT_INST_MIN_REGULARITY_max
AMT_INST_MIN_REGULARITY_mean
AMT_INST_MIN_REGULARITY_median
AMT_INST_MIN_REGULARITY_min
AMT_INST_MIN_REGULARITY_sum
AMT_INST_MIN_REGULARITY_var
```

AMT_PAYMENT_CURRENT_count
AMT_PAYMENT_CURRENT_max
AMT_PAYMENT_CURRENT_mean
AMT_PAYMENT_CURRENT_median
AMT_PAYMENT_CURRENT_min
AMT_PAYMENT_CURRENT_sum
AMT_PAYMENT_CURRENT_var
AMT_PAYMENT_TOTAL_CURRENT_count
AMT_PAYMENT_TOTAL_CURRENT_max
AMT_PAYMENT_TOTAL_CURRENT_mean
AMT_PAYMENT_TOTAL_CURRENT_median
AMT_PAYMENT_TOTAL_CURRENT_min
AMT_PAYMENT_TOTAL_CURRENT_sum
AMT_PAYMENT_TOTAL_CURRENT_var
AMT_RECEIVABLE_PRINCIPAL_count
AMT_RECEIVABLE_PRINCIPAL_max
AMT_RECEIVABLE_PRINCIPAL_mean
AMT_RECEIVABLE_PRINCIPAL_median
AMT_RECEIVABLE_PRINCIPAL_min
AMT_RECEIVABLE_PRINCIPAL_sum
AMT_RECEIVABLE_PRINCIPAL_var
AMT_RECIVABLE_count
AMT_RECIVABLE_max
AMT_RECIVABLE_mean
AMT_RECIVABLE_median
AMT_RECIVABLE_min
AMT_RECIVABLE_sum
AMT_RECIVABLE_var
AMT_TOTAL_RECEIVABLE_count
AMT_TOTAL_RECEIVABLE_max
AMT_TOTAL_RECEIVABLE_mean
AMT_TOTAL_RECEIVABLE_median
AMT_TOTAL_RECEIVABLE_min
AMT_TOTAL_RECEIVABLE_sum
AMT_TOTAL_RECEIVABLE_var
CNT_DRAWINGS_ATM_CURRENT_count
CNT_DRAWINGS_ATM_CURRENT_max
CNT_DRAWINGS_ATM_CURRENT_mean
CNT_DRAWINGS_ATM_CURRENT_median
CNT_DRAWINGS_ATM_CURRENT_min
CNT_DRAWINGS_ATM_CURRENT_sum
CNT_DRAWINGS_ATM_CURRENT_var
CNT_DRAWINGS_CURRENT_count
CNT_DRAWINGS_CURRENT_max
CNT_DRAWINGS_CURRENT_mean
CNT_DRAWINGS_CURRENT_median
CNT_DRAWINGS_CURRENT_min
CNT_DRAWINGS_CURRENT_sum
CNT_DRAWINGS_CURRENT_var
CNT_DRAWINGS_OTHER_CURRENT_count
CNT_DRAWINGS_OTHER_CURRENT_max
CNT_DRAWINGS_OTHER_CURRENT_mean
CNT_DRAWINGS_OTHER_CURRENT_median
CNT_DRAWINGS_OTHER_CURRENT_min
CNT_DRAWINGS_OTHER_CURRENT_sum
CNT_DRAWINGS_OTHER_CURRENT_var
CNT_DRAWINGS_POS_CURRENT_count
CNT_DRAWINGS_POS_CURRENT_max
CNT_DRAWINGS_POS_CURRENT_mean

```
CNT_DRAWINGS_POS_CURRENT_median
CNT_DRAWINGS_POS_CURRENT_min
CNT_DRAWINGS_POS_CURRENT_sum
CNT_DRAWINGS_POS_CURRENT_var
CNT_INSTALMENT_MATURE_CUM_count
CNT_INSTALMENT_MATURE_CUM_max
CNT_INSTALMENT_MATURE_CUM_mean
CNT_INSTALMENT_MATURE_CUM_median
CNT_INSTALMENT_MATURE_CUM_min
CNT_INSTALMENT_MATURE_CUM_sum
CNT_INSTALMENT_MATURE_CUM_var
MONTHS_BALANCE_count
MONTHS_BALANCE_max
MONTHS_BALANCE_mean
MONTHS_BALANCE_median
MONTHS_BALANCE_min
MONTHS_BALANCE_sum
MONTHS_BALANCE_var
NAME_CONTRACT_STATUS_Active_mean
NAME_CONTRACT_STATUS_Active_median
NAME_CONTRACT_STATUS_Active_var
NAME_CONTRACT_STATUS_Approved_mean
NAME_CONTRACT_STATUS_Approved_median
NAME_CONTRACT_STATUS_Approved_var
NAME_CONTRACT_STATUS_Completed_mean
NAME_CONTRACT_STATUS_Completed_median
NAME_CONTRACT_STATUS_Completed_var
NAME_CONTRACT_STATUS_Demand_mean
NAME_CONTRACT_STATUS_Demand_median
NAME_CONTRACT_STATUS_Demand_var
NAME_CONTRACT_STATUS_Refused_mean
NAME_CONTRACT_STATUS_Refused_median
NAME_CONTRACT_STATUS_Refused_var
NAME_CONTRACT_STATUS_Sent_proposal_mean
NAME_CONTRACT_STATUS_Sent_proposal_median
NAME_CONTRACT_STATUS_Sent_proposal_var
NAME_CONTRACT_STATUS_Signed_mean
NAME_CONTRACT_STATUS_Signed_median
NAME_CONTRACT_STATUS_Signed_var
SK_DPD_DEF_count
SK_DPD_DEF_max
SK_DPD_DEF_mean
SK_DPD_DEF_median
SK_DPD_DEF_min
SK_DPD_DEF_sum
SK_DPD_DEF_var
SK_DPD_count
SK_DPD_max
SK_DPD_mean
SK_DPD_median
SK_DPD_min
SK_DPD_sum
SK_DPD_var
SK_ID_PREV_count

Aggregated Data:

                                    count          mean           std  min  \
SK_DPD_DEF_min                   103558.0  4.422642e-03  1.423225e+00  0.0
```

```
AMT_PAYMENT_TOTAL_CURRENT_max   103558.0  8.126568e+04  1.350253e+05  0.0
AMT_DRAWINGS_OTHER_CURRENT_min   72194.0  5.086295e+00  6.446515e+02  0.0
CNT_DRAWINGS_POS_CURRENT_count  103558.0  2.984314e+01  3.542705e+01  0.0
AMT_DRAWINGS_CURRENT_var        102866.0  1.798590e+09  7.646675e+09  0.0
...                                  ...           ...           ...  ...
AMT_INST_MIN_REGULARITY_median  103558.0  3.473576e+03  6.076720e+03  0.0
CNT_DRAWINGS_CURRENT_mean       103558.0  1.513550e+00  3.654926e+00  0.0
SK_DPD_max                      103558.0  1.640187e+01  1.419661e+02  0.0
CNT_DRAWINGS_POS_CURRENT_min     72194.0  1.656370e-01  1.353768e+00  0.0
MONTHS_BALANCE_sum              103558.0  1.280200e+03  1.743939e+03  1.0

                                 25%           50%           75%           m
ax
SK_DPD_DEF_min                   0.0  0.000000e+00  0.000000e+00  4.580000e+
02
AMT_PAYMENT_TOTAL_CURRENT_max    0.0  2.343600e+04  1.075137e+05  4.278316e+
06
AMT_DRAWINGS_OTHER_CURRENT_min   0.0  0.000000e+00  0.000000e+00  1.125000e+
05
CNT_DRAWINGS_POS_CURRENT_count   0.0  1.200000e+01  5.600000e+01  1.920000e+
02
AMT_DRAWINGS_CURRENT_var         0.0  1.918232e+08  1.215640e+09  6.075000e+
11
...                              ...           ...           ...
...
AMT_INST_MIN_REGULARITY_median   0.0  0.000000e+00  4.992508e+03  4.685744e+
04
CNT_DRAWINGS_CURRENT_mean        0.0  2.051282e-01  1.285714e+00  1.182500e+
02
SK_DPD_max                       0.0  0.000000e+00  0.000000e+00  3.260000e+
03
CNT_DRAWINGS_POS_CURRENT_min     0.0  0.000000e+00  0.000000e+00  7.800000e+
01
MONTHS_BALANCE_sum              65.0  2.530000e+02  2.850000e+03  9.312000e+
03

[162 rows x 8 columns]
```

Installment Payments

1. This function is specifically for Installment Payments table. Below are the pre-processing done before passing this table into the pipeline.

    - Absolute value of the Days Installment and Days Entry Payment are used as that allows only use of postive values without effecting the correlation. As more negative means longer time from the intial date.

    - A new column of IP_DIFF_PAYMNT_INSTLMNT was also calculated from difference of AMT_INSTALMENT from AMT_PAYMENT as a part of the feature select/transformation process.

    - Any column or row with more than 70% of its data as null will be deleted from the dataframe as the threshold is set to .7.

    - Similar to above, once processed, store the transfored csv file. Benefit of this is that we can then pass it directly to model for merging into application train/test table. We do not have to perform expensive EDA/ETL/Transformation everytime we want to process the data.

In [11]:

```
def ip_eda(df):
    ip = df
    drop_list_ip = []

    #Adding new features
    ip['DAYS_INSTALMENT'] = ip['DAYS_INSTALMENT'].abs()
    ip['DAYS_ENTRY_PAYMENT'] = ip['DAYS_ENTRY_PAYMENT'].abs()
    ip['IP_DIFF_PAYMNT_INSTLMNT'] = ip['AMT_PAYMENT'] - ip['AMT_INSTALMENT']

    threshold = 0.7

    #Dropping columns with missing value rate higher than threshold
    ip = ip[ip.columns[ip.isnull().mean() < threshold]]

    #Dropping rows with missing value rate higher than threshold
    ip = ip.loc[ip.isnull().mean(axis=1) < threshold]

    return (eda_transformation(ip,4))
```

In [12]:

```
ip = datasets['installments_payments']
ip = ip_eda(ip)
datasets_transformed['installments_payments'] = ip

ip.to_csv(os.getcwd() + DATA_DIR + 'ip_agg_data.csv')
```

```
--------
# of ID's: 1
 ID's:
['SK_ID_CURR']

--------
# All features: 8
All features:
['IP_DIFF_PAYMNT_INSTLMNT', 'SK_ID_PREV', 'AMT_INSTALMENT', 'NUM_INSTALMENT_
VERSION', 'DAYS_INSTALMENT', 'AMT_PAYMENT', 'NUM_INSTALMENT_NUMBER', 'DAYS_E
NTRY_PAYMENT']

Missing data:
                         Total   Percent
IP_DIFF_PAYMNT_INSTLMNT   2905  0.021352
AMT_PAYMENT               2905  0.021352
DAYS_ENTRY_PAYMENT        2905  0.021352
SK_ID_PREV                   0  0.000000
AMT_INSTALMENT               0  0.000000
NUM_INSTALMENT_VERSION       0  0.000000
DAYS_INSTALMENT              0  0.000000
NUM_INSTALMENT_NUMBER        0  0.000000

--------
# of Numerical features: 9
Numerical features:
['SK_ID_PREV', 'SK_ID_CURR', 'NUM_INSTALMENT_VERSION', 'NUM_INSTALMENT_NUMBE
R', 'DAYS_INSTALMENT', 'DAYS_ENTRY_PAYMENT', 'AMT_INSTALMENT', 'AMT_PAYMENT
', 'IP_DIFF_PAYMNT_INSTLMNT']

Numerical Statistical Summary:

         SK_ID_PREV    SK_ID_CURR  NUM_INSTALMENT_VERSION  \
count  1.360540e+07  1.360540e+07            1.360540e+07
mean   1.903365e+06  2.784449e+05            8.566373e-01
std    5.362029e+05  1.027183e+05            1.035216e+00
min    1.000001e+06  1.000010e+05            0.000000e+00
25%    1.434191e+06  1.896390e+05            0.000000e+00
50%    1.896520e+06  2.786850e+05            1.000000e+00
75%    2.369094e+06  3.675300e+05            1.000000e+00
max    2.843499e+06  4.562550e+05            1.780000e+02

       NUM_INSTALMENT_NUMBER  DAYS_INSTALMENT  DAYS_ENTRY_PAYMENT  \
count           1.360540e+07     1.360540e+07        1.360250e+07
mean            1.887090e+01     1.042270e+03        1.051114e+03
std             2.666407e+01     8.009463e+02        8.005859e+02
min             1.000000e+00     1.000000e+00        1.000000e+00
25%             4.000000e+00     3.610000e+02        3.700000e+02
50%             8.000000e+00     8.180000e+02        8.270000e+02
75%             1.900000e+01     1.654000e+03        1.662000e+03
max             2.770000e+02     2.922000e+03        4.921000e+03

       AMT_INSTALMENT   AMT_PAYMENT  IP_DIFF_PAYMNT_INSTLMNT
count    1.360540e+07  1.360250e+07             1.360250e+07
mean     1.705091e+04  1.723822e+04             1.871538e+02
std      5.057025e+04  5.473578e+04             1.910673e+04
min      0.000000e+00  0.000000e+00            -2.424726e+06
25%      4.226085e+03  3.398265e+03             0.000000e+00
50%      8.884080e+03  8.125515e+03             0.000000e+00
```

```
75%      1.671021e+04  1.610842e+04             0.000000e+00
max      3.771488e+06  3.771488e+06             2.630909e+06

--------
# of Categorical features: 0
Categorical features:
[]

Categorical Statistical Summary:


Categories:

Series([], dtype: float64)

--------
# of OHE categorical features: 0
OHE Categorical features: []
--------
df.shape: (13605401, 9)

Aggregated Features:
df[['IP_DIFF_PAYMNT_INSTLMNT', 'SK_ID_PREV', 'AMT_INSTALMENT', 'NUM_INSTALME
NT_VERSION', 'DAYS_INSTALMENT', 'AMT_PAYMENT', 'NUM_INSTALMENT_NUMBER', 'DAY
S_ENTRY_PAYMENT']][0:5]:
   IP_DIFF_PAYMNT_INSTLMNT  SK_ID_PREV  AMT_INSTALMENT  \
0                    0.000     1054186        6948.360
1                    0.000     1330831        1716.525
2                    0.000     2085231       25425.000
3                    0.000     2452527       24350.130
4                   -4.455     2714724        2165.040

   NUM_INSTALMENT_VERSION  DAYS_INSTALMENT  AMT_PAYMENT  \
0                     1.0           1180.0     6948.360
1                     0.0           2156.0     1716.525
2                     2.0             63.0    25425.000
3                     1.0           2418.0    24350.130
4                     1.0           1383.0     2160.585

   NUM_INSTALMENT_NUMBER  DAYS_ENTRY_PAYMENT
0                      6              1187.0
1                     34              2156.0
2                      1                63.0
3                      3              2426.0
4                      2              1366.0
--------
Aggregated Features:
AMT_INSTALMENT_count
AMT_INSTALMENT_max
AMT_INSTALMENT_mean
AMT_INSTALMENT_median
AMT_INSTALMENT_min
AMT_INSTALMENT_sum
AMT_INSTALMENT_var
AMT_PAYMENT_count
AMT_PAYMENT_max
AMT_PAYMENT_mean
AMT_PAYMENT_median
AMT_PAYMENT_min
```

```
AMT_PAYMENT_sum
AMT_PAYMENT_var
DAYS_ENTRY_PAYMENT_count
DAYS_ENTRY_PAYMENT_max
DAYS_ENTRY_PAYMENT_mean
DAYS_ENTRY_PAYMENT_median
DAYS_ENTRY_PAYMENT_min
DAYS_ENTRY_PAYMENT_sum
DAYS_ENTRY_PAYMENT_var
DAYS_INSTALMENT_count
DAYS_INSTALMENT_max
DAYS_INSTALMENT_mean
DAYS_INSTALMENT_median
DAYS_INSTALMENT_min
DAYS_INSTALMENT_sum
DAYS_INSTALMENT_var
IP_DIFF_PAYMNT_INSTLMNT_count
IP_DIFF_PAYMNT_INSTLMNT_max
IP_DIFF_PAYMNT_INSTLMNT_mean
IP_DIFF_PAYMNT_INSTLMNT_median
IP_DIFF_PAYMNT_INSTLMNT_min
IP_DIFF_PAYMNT_INSTLMNT_sum
IP_DIFF_PAYMNT_INSTLMNT_var
NUM_INSTALMENT_NUMBER_count
NUM_INSTALMENT_NUMBER_max
NUM_INSTALMENT_NUMBER_mean
NUM_INSTALMENT_NUMBER_median
NUM_INSTALMENT_NUMBER_min
NUM_INSTALMENT_NUMBER_sum
NUM_INSTALMENT_NUMBER_var
NUM_INSTALMENT_VERSION_count
NUM_INSTALMENT_VERSION_max
NUM_INSTALMENT_VERSION_mean
NUM_INSTALMENT_VERSION_median
NUM_INSTALMENT_VERSION_min
NUM_INSTALMENT_VERSION_sum
NUM_INSTALMENT_VERSION_var
SK_ID_PREV_count

Aggregated Data:

                                   count          mean           std  \
AMT_PAYMENT_max                 339578.0  1.400438e+05  2.496554e+05
AMT_INSTALMENT_sum              339587.0  6.831369e+05  8.933805e+05
IP_DIFF_PAYMNT_INSTLMNT_var     338610.0  3.465416e+08  3.435981e+09
DAYS_ENTRY_PAYMENT_mean         339578.0  9.220955e+02  5.971820e+02
NUM_INSTALMENT_VERSION_min      339587.0  7.899095e-01  4.198890e-01
DAYS_INSTALMENT_median          339587.0  8.813710e+02  6.624234e+02
NUM_INSTALMENT_NUMBER_median    339587.0  8.712772e+00  1.041643e+01
NUM_INSTALMENT_VERSION_median   339587.0  9.250030e-01  6.009429e-01
AMT_PAYMENT_count               339587.0  4.005600e+01  4.104650e+01
NUM_INSTALMENT_NUMBER_sum       339587.0  7.560540e+02  1.892398e+03
DAYS_ENTRY_PAYMENT_median       339578.0  8.927967e+02  6.612506e+02
AMT_INSTALMENT_count            339587.0  4.006455e+01  4.105334e+01
AMT_PAYMENT_median              339578.0  1.270843e+04  1.809615e+04
AMT_INSTALMENT_median           339587.0  1.295001e+04  1.745949e+04
AMT_INSTALMENT_var              338615.0  2.675638e+09  1.738837e+10
DAYS_ENTRY_PAYMENT_var          338610.0  2.738659e+05  3.331144e+05
IP_DIFF_PAYMNT_INSTLMNT_mean    339578.0  3.907863e+02  5.242748e+03
```

AMT_PAYMENT_min	339578.0	5.282753e+03	1.449867e+04
IP_DIFF_PAYMNT_INSTLMNT_median	339578.0	4.782494e+01	1.362075e+03
NUM_INSTALMENT_NUMBER_var	338615.0	1.245410e+02	3.491713e+02
AMT_INSTALMENT_min	339587.0	7.048657e+03	1.445801e+04
NUM_INSTALMENT_NUMBER_min	339587.0	1.043962e+00	7.479743e-01
DAYS_ENTRY_PAYMENT_min	339578.0	3.284334e+02	5.373678e+02
DAYS_INSTALMENT_var	338615.0	2.736165e+05	3.339608e+05
DAYS_INSTALMENT_sum	339587.0	4.175808e+04	5.666080e+04
DAYS_ENTRY_PAYMENT_count	339587.0	4.005600e+01	4.104650e+01
NUM_INSTALMENT_VERSION_max	339587.0	2.130435e+00	1.912109e+00
DAYS_INSTALMENT_min	339587.0	3.202087e+02	5.399782e+02
AMT_PAYMENT_sum	339587.0	6.904942e+05	9.308977e+05
NUM_INSTALMENT_NUMBER_count	339587.0	4.006455e+01	4.105334e+01
IP_DIFF_PAYMNT_INSTLMNT_max	339578.0	2.385596e+04	1.094190e+05
DAYS_ENTRY_PAYMENT_sum	339587.0	4.210341e+04	5.686314e+04
NUM_INSTALMENT_NUMBER_mean	339587.0	9.830926e+00	1.140517e+01
AMT_PAYMENT_mean	339578.0	1.900381e+04	2.523135e+04
NUM_INSTALMENT_VERSION_mean	339587.0	1.047498e+00	6.207509e-01
DAYS_INSTALMENT_mean	339587.0	9.107428e+02	5.985713e+02
DAYS_ENTRY_PAYMENT_max	339578.0	1.600876e+03	9.134565e+02
NUM_INSTALMENT_VERSION_var	338615.0	4.831801e-01	3.783123e+00
DAYS_INSTALMENT_max	339587.0	1.588918e+03	9.150291e+02
NUM_INSTALMENT_NUMBER_max	339587.0	2.451555e+01	3.016271e+01
IP_DIFF_PAYMNT_INSTLMNT_min	339578.0	-6.767578e+03	1.523606e+04
IP_DIFF_PAYMNT_INSTLMNT_count	339587.0	4.005600e+01	4.104650e+01
NUM_INSTALMENT_VERSION_sum	339587.0	3.432079e+01	3.825223e+01
DAYS_INSTALMENT_count	339587.0	4.006455e+01	4.105334e+01
AMT_INSTALMENT_mean	339587.0	1.861577e+04	2.346509e+04
NUM_INSTALMENT_VERSION_count	339587.0	4.006455e+01	4.105334e+01
IP_DIFF_PAYMNT_INSTLMNT_sum	339587.0	7.496631e+03	1.777886e+05
AMT_PAYMENT_var	338610.0	3.038643e+09	1.829612e+10
SK_ID_PREV_count	339587.0	4.006455e+01	4.105334e+01
AMT_INSTALMENT_max	339587.0	1.381566e+05	2.473304e+05

	min	25%	50% \
AMT_PAYMENT_max	0.225	1.439653e+04	3.703561e+04
AMT_INSTALMENT_sum	0.000	1.366834e+05	3.343952e+05
IP_DIFF_PAYMNT_INSTLMNT_var	0.000	0.000000e+00	0.000000e+00
DAYS_ENTRY_PAYMENT_mean	3.000	4.338571e+02	8.065764e+02
NUM_INSTALMENT_VERSION_min	0.000	1.000000e+00	1.000000e+00
DAYS_INSTALMENT_median	3.000	3.550000e+02	6.675000e+02
NUM_INSTALMENT_NUMBER_median	1.000	4.000000e+00	6.000000e+00
NUM_INSTALMENT_VERSION_median	0.000	1.000000e+00	1.000000e+00
AMT_PAYMENT_count	0.000	1.200000e+01	2.500000e+01
NUM_INSTALMENT_NUMBER_sum	1.000	6.200000e+01	1.560000e+02
DAYS_ENTRY_PAYMENT_median	3.000	3.680000e+02	6.790000e+02
AMT_INSTALMENT_count	1.000	1.200000e+01	2.500000e+01
AMT_PAYMENT_median	0.045	5.625000e+03	9.108090e+03
AMT_INSTALMENT_median	0.000	5.840415e+03	9.440640e+03
AMT_INSTALMENT_var	0.000	5.010991e+06	6.298936e+07
DAYS_ENTRY_PAYMENT_var	0.000	1.273954e+04	1.152055e+05
IP_DIFF_PAYMNT_INSTLMNT_mean	-146145.900	-4.358552e+02	0.000000e+00
AMT_PAYMENT_min	0.000	4.153500e+01	2.323575e+03
IP_DIFF_PAYMNT_INSTLMNT_median	-201907.350	0.000000e+00	0.000000e+00
NUM_INSTALMENT_NUMBER_var	0.000	7.228571e+00	1.229568e+01
AMT_INSTALMENT_min	0.000	1.872765e+03	4.868145e+03
NUM_INSTALMENT_NUMBER_min	1.000	1.000000e+00	1.000000e+00
DAYS_ENTRY_PAYMENT_min	1.000	2.600000e+01	6.500000e+01
DAYS_INSTALMENT_var	0.000	1.210818e+04	1.142342e+05

DAYS_INSTALMENT_sum	3.000	6.630000e+03	2.094900e+04
DAYS_ENTRY_PAYMENT_count	0.000	1.200000e+01	2.500000e+01
NUM_INSTALMENT_VERSION_max	0.000	1.000000e+00	2.000000e+00
DAYS_INSTALMENT_min	1.000	1.800000e+01	4.800000e+01
AMT_PAYMENT_sum	0.000	1.332007e+05	3.248035e+05
NUM_INSTALMENT_NUMBER_count	1.000	1.200000e+01	2.500000e+01
IP_DIFF_PAYMNT_INSTLMNT_max	-20512.350	0.000000e+00	0.000000e+00
DAYS_ENTRY_PAYMENT_sum	0.000	6.832000e+03	2.123300e+04
NUM_INSTALMENT_NUMBER_mean	1.000	4.560000e+00	6.045455e+00
AMT_PAYMENT_mean	0.189	7.582365e+03	1.240398e+04
NUM_INSTALMENT_VERSION_mean	0.000	1.000000e+00	1.018519e+00
DAYS_INSTALMENT_mean	3.000	4.210000e+02	7.949677e+02
DAYS_ENTRY_PAYMENT_max	3.000	7.290000e+02	1.557000e+03
NUM_INSTALMENT_VERSION_var	0.000	0.000000e+00	7.692308e-02
DAYS_INSTALMENT_max	3.000	7.160000e+02	1.544000e+03
NUM_INSTALMENT_NUMBER_max	1.000	1.000000e+01	1.200000e+01
IP_DIFF_PAYMNT_INSTLMNT_min	-2424726.405	-9.122062e+03	0.000000e+00
IP_DIFF_PAYMNT_INSTLMNT_count	0.000	1.200000e+01	2.500000e+01
NUM_INSTALMENT_VERSION_sum	0.000	1.200000e+01	2.300000e+01
DAYS_INSTALMENT_count	1.000	1.200000e+01	2.500000e+01
AMT_INSTALMENT_mean	0.000	7.897470e+03	1.272616e+04
NUM_INSTALMENT_VERSION_count	1.000	1.200000e+01	2.500000e+01
IP_DIFF_PAYMNT_INSTLMNT_sum	-3037735.575	-1.570462e+04	0.000000e+00
AMT_PAYMENT_var	0.000	8.979742e+06	7.302708e+07
SK_ID_PREV_count	1.000	1.200000e+01	2.500000e+01
AMT_INSTALMENT_max	0.000	1.432051e+04	3.629011e+04

	75%	max
AMT_PAYMENT_max	1.350000e+05	3.771488e+06
AMT_INSTALMENT_sum	8.577727e+05	3.247978e+07
IP_DIFF_PAYMNT_INSTLMNT_var	1.161385e+07	4.459936e+11
DAYS_ENTRY_PAYMENT_mean	1.312043e+03	3.071000e+03
NUM_INSTALMENT_VERSION_min	1.000000e+00	3.900000e+01
DAYS_INSTALMENT_median	1.294000e+03	2.922000e+03
NUM_INSTALMENT_NUMBER_median	8.000000e+00	1.450000e+02
NUM_INSTALMENT_VERSION_median	1.000000e+00	3.900000e+01
AMT_PAYMENT_count	5.100000e+01	3.720000e+02
NUM_INSTALMENT_NUMBER_sum	4.405000e+02	4.391900e+04
DAYS_ENTRY_PAYMENT_median	1.304000e+03	3.071000e+03
AMT_INSTALMENT_count	5.100000e+01	3.720000e+02
AMT_PAYMENT_median	1.494228e+04	2.504590e+06
AMT_INSTALMENT_median	1.545175e+04	2.504590e+06
AMT_INSTALMENT_var	5.929413e+08	2.436698e+12
DAYS_ENTRY_PAYMENT_var	4.748150e+05	4.161612e+06
IP_DIFF_PAYMNT_INSTLMNT_mean	0.000000e+00	3.374968e+05
AMT_PAYMENT_min	7.158724e+03	2.504590e+06
IP_DIFF_PAYMNT_INSTLMNT_median	0.000000e+00	1.125000e+05
NUM_INSTALMENT_NUMBER_var	4.134611e+01	7.549267e+03
AMT_INSTALMENT_min	8.995095e+03	2.504590e+06
NUM_INSTALMENT_NUMBER_min	1.000000e+00	6.700000e+01
DAYS_ENTRY_PAYMENT_min	3.870000e+02	3.071000e+03
DAYS_INSTALMENT_var	4.740715e+05	4.219512e+06
DAYS_INSTALMENT_sum	4.940550e+04	5.989430e+05
DAYS_ENTRY_PAYMENT_count	5.100000e+01	3.720000e+02
NUM_INSTALMENT_VERSION_max	2.000000e+00	1.780000e+02
DAYS_INSTALMENT_min	3.760000e+02	2.922000e+03
AMT_PAYMENT_sum	8.497309e+05	3.268928e+07
NUM_INSTALMENT_NUMBER_count	5.100000e+01	3.720000e+02
IP_DIFF_PAYMNT_INSTLMNT_max	0.000000e+00	2.630909e+06

DAYS_ENTRY_PAYMENT_sum	4.985300e+04	6.024990e+05
NUM_INSTALMENT_NUMBER_mean	9.500000e+00	1.442081e+02
AMT_PAYMENT_mean	2.164505e+04	2.504590e+06
NUM_INSTALMENT_VERSION_mean	1.111111e+00	3.900000e+01
DAYS_INSTALMENT_mean	1.302113e+03	2.922000e+03
DAYS_ENTRY_PAYMENT_max	2.493000e+03	4.921000e+03
NUM_INSTALMENT_VERSION_var	2.500000e-01	6.806145e+02
DAYS_INSTALMENT_max	2.481000e+03	2.922000e+03
NUM_INSTALMENT_NUMBER_max	2.400000e+01	2.770000e+02
IP_DIFF_PAYMNT_INSTLMNT_min	0.000000e+00	5.086116e+04
IP_DIFF_PAYMNT_INSTLMNT_count	5.100000e+01	3.720000e+02
NUM_INSTALMENT_VERSION_sum	4.400000e+01	1.826000e+03
DAYS_INSTALMENT_count	5.100000e+01	3.720000e+02
AMT_INSTALMENT_mean	2.164516e+04	2.504590e+06
NUM_INSTALMENT_VERSION_count	5.100000e+01	3.720000e+02
IP_DIFF_PAYMNT_INSTLMNT_sum	0.000000e+00	4.417384e+06
AMT_PAYMENT_var	6.600174e+08	2.436698e+12
SK_ID_PREV_count	5.100000e+01	3.720000e+02
AMT_INSTALMENT_max	1.329430e+05	3.771488e+06

Visual EDA

EDA Visualization functions

Following are the functions created to do visualization. These are rudimentary level plots which provide insights into a dataframe. The input to these functions will either be a dataframe or a combination of dataframe and type of attribute (numerical, categorical) depending upon the type of function.

- **Attr_Type**: This function plots a bar graph which shows the count of type of attribute, i.e. Number, Categorical, Date etc. We are using similar logic in summary statistics where depending upon the type of variable, we derive the numerical and categorical features to further do the processing.
- **Unique Values**: This function plots the bar chart for the unique value for each attribute. This process will give us some insights about the number of binary , ordinal and continuous (more than 10 unique values) features in the dataset. This info can also be read from info and summary python functions. Another thing to notice here is that y-axis is on log scale. This is because on such a large data set, few elements have more unique values(like CURR ID, SK ID PREV) which makes distribution highly skewed. By taking log, we will have fair idea about other features with lesser unique values.
- **Percent Missing**: This bar plot will show to percentage of nulls in the data. This plot will be helpful for us to see how much nulls a attribute has. In phase I, we created pipeline with attributes with more than 50% nulls. This information will also help while finding the correlation and cross-verify if some values are less or highly correlated. 1 thing to note here is that this plot also has y-axis log scaled for the same reason as provided above.
- **Categorical_count**: This function will take a dataframe and categorical features as input and will produce a bar plot for each category. This will enable us to understand if certain set of values are more correlated to the target. For example, we saw that less educated people are more likely to default on their loan, as per the data provided. Now that we have seen this info as a plot, we can analyze a little more. This will also help us group certain values which are negligible, For example, if 2 categories are related to 90 % of the data, we can group certain low occurring values to make them as 1 category. This can also be seen by using the F score to see how statistically important it can be.
- **Dendo**: This unique plot will groups together columns that have strong correlations in nullity. If a number of columns are grouped together at level zero, then the presence of nulls in one of those columns is directly related to the presence or absence of nulls in the others columns. The more separated the columns in the tree, the less likely the null values can be correlated between the columns.
- **Numerical_features**: This function will take only the numerical attributes from a dataframe and produce a dot plot. Each dot in this plot is a sample in our dataset and each subplot represents a different feature. The y-axis shows the feature value, while the

each subplot represents a different feature. The y-axis shows the feature value, while the x-axis is the sample index. These kind of plots provide a lot of ideas for data cleaning and EDA.

- **Num Hist**: This functions takes dataframe and numerical attributes. This will plot histogram for all the features supplied. A very useful visualization to see the data distribution in 1 view. We can take note of attributes which needs transformation, as well as outliers. For a better input data, we should remove the outliers so that it does not influence the target outcome.
- **all_missing_values_plot**: This function will plot the missing in all the features of the dataframe supplied. For attributes which have no nulls will show as 1, if an attribute has half of the records null, then it will show 0.5.

In [77]:

```
def attr_type(df):
    plt.figure(figsize=(20,20))
    pd.value_counts(df.dtypes).sort_values().plot(kind="bar", figsize=(15, 8
                                                  title="Type of features- Numer
                                                  ylabel="Number of features");
    plt.show()

def unique_values(df):
    plt.figure(figsize=(20,20))
    unique_values = df.select_dtypes(include="number").nunique().sort_values
    unique_values.plot.bar(logy=True, figsize=(15, 4), title="Unique values
    plt.show()

def percent_missing(df):
    plt.figure(figsize=(20,20))
    df.isna().mean().sort_values().plot(kind="bar", figsize=(15, 8),logy=Tru
                                        title="Percentage of missing valu
                                        ylabel="Ratio of missing values p
    plt.show()

def categorical_count(df,categorical):
    plt.figure(figsize=(20,20))
    for i, col in enumerate(feat_cat):
        ax = plt.subplot(5, 4, i+1)
        sns.countplot(data=df[categorical], y=col, ax=ax,)
    plt.suptitle('Category counts for all categorical variables')
    plt.tight_layout()
    plt.show()

def dendo(df):
    plt.figure(figsize=(20,20))
    msno.dendrogram(df)
    plt.show()

def numerical_features(df,num_cat):
    plt.figure(figsize=(20,20))
    df[num_cat].plot(lw=0, marker=".", subplots=True, layout=(-1, 4),
              figsize=(15, 12), markersize=1);

    plt.show()

def num_hist(df):
    plt.figure(figsize=(20,20))
```

```
    plt.figure(figsize=(20,20))
    df[np.isfinite(df)].hist(bins=25, figsize=(15, 25), layout=(-1, 5), edg
    plt.tight_layout();
    plt.show();


def continous_features(df,feat_num):
    plt.figure(figsize=(20,20))
    cols_continuous = df.select_dtypes(include="number").nunique() >= 25
```

In [78]:

```
for i,ds_name in enumerate(datasets.keys()):

    #n = True
    #if n:

    if ds_name.lower() not in (#"application_train",
                               "application_test", "bureau_balance"):
        print("Table under consideration:",ds_name.upper())
        print("-------------------------------------------------------------
        id_cols, feat_num, feat_cat, features =  id_num_cat_feature(dataset
        only_num_cat = list(set(feat_num)-set(['SK_ID_CURR','SK_ID_PREV','S
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------Type of Features--
        attr_type(datasets[ds_name])
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------UNIQUE VALUES-----
        unique_values(datasets[ds_name])
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------MISSING PERCENTAGE
        percent_missing(datasets[ds_name])
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------CATEGORICAL COUNT-
        categorical_count(datasets[ds_name],feat_cat)
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------NUM FEATURES-DOT P
        numerical_features(datasets[ds_name],only_num_cat)
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------NUM FEATURES - HIS
        num_hist(datasets[ds_name][only_num_cat])
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------Continous Features
        #continous_features(df_x,only_num_cat)
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------All Missing Featur
        all_missing_values_plot(datasets[ds_name])
        print("-------------------------------------------------------------
        print(ds_name.upper(),":-------------------------DendoGram for Null
        dendo(datasets[ds_name])
        print("-------------------------------------------------------------
```

```
Table under consideration: APPLICATION_TRAIN
-----------------------------------------------------------------------
-----------------------------------------------------------------------
APPLICATION_TRAIN :-------------------------Type of Features
--------------------------------
```

APPLICATION_TRAIN :----------UNIQUE VALUES

APPLICATION_TRAIN :----------MISSING PERCENTAGE

APPLICATION_TRAIN :----------CATEGORICAL COUNT

```
--------------------------------------------------------------------------
APPLICATION_TRAIN :-------------------------NUM FEATURES-DOT PLOT
----------------------
<Figure size 1440x1440 with 0 Axes>
```

```
----------------------------------------------------------------------------
APPLICATION_TRAIN :-------------------------NUM FEATURES - HISTOGRAM
----------------------
<Figure size 1440x1440 with 0 Axes>
```

FLAG_PHONE
EXT_SOURCE_2
REG_REGION_NOT_WORK_REGION
FLOORSMIN_MODE
COMMONAREA_MEDI
BASEMENTAREA_MEDI
DAYS_EMPLOYED
NONLIVINGAREA_MEDI
FLAG_DOCUMENT_21
AMT_REQ_CREDIT_BUREAU_DAY
LIVINGAPARTMENTS_MODE
FLAG_DOCUMENT_3
FLOORSMIN_AVG
LANDAREA_MEDI
REG_CITY_NOT_LIVE_CITY
FLAG_EMP_PHONE
FLOORSMAX_MODE
AMT_REQ_CREDIT_BUREAU_WEEK
LIVE_REGION_NOT_WORK_REGION
COMMONAREA_AVG
FLAG_EMAIL
FLAG_CONT_MOBILE
REGION_POPULATION_RELATIVE
TARGET
REG_REGION_NOT_LIVE_REGION
ENTRANCES_AVG
FLAG_DOCUMENT_16
CNT_FAM_MEMBERS
EXT_SOURCE_1
APARTMENTS_MODE
FLAG_DOCUMENT_6
HOUR_APPR_PROCESS_START
LIVINGAREA_MEDI
AMT_REQ_CREDIT_BUREAU_MON
ENTRANCES_MODE
APARTMENTS_MEDI
BASEMENTAREA_MODE
LANDAREA_AVG
NONLIVINGAREA_MODE
LIVINGAREA_MODE
DAYS_LAST_PHONE_CHANGE
FLAG_DOCUMENT_20
YEARS_BUILD_AVG
NONLIVINGAREA_AVG
YEARS_BEGINEXPLUATATION_MODE
OBS_60_CNT_SOCIAL_CIRCLE
YEARS_BEGINEXPLUATATION_MEDI
EXT_SOURCE_3
OWN_CAR_AGE
DAYS_REGISTRATION
FLAG_DOCUMENT_5
FLAG_DOCUMENT_12
REG_CITY_NOT_WORK_CITY
AMT_REQ_CREDIT_BUREAU_HOUR
ELEVATORS_AVG
ELEVATORS_MODE
OBS_30_CNT_SOCIAL_CIRCLE
LIVE_CITY_NOT_WORK_CITY
YEARS_BEGINEXPLUATATION_AVG
DEF_60_CNT_SOCIAL_CIRCLE
FLAG_DOCUMENT_17
BASEMENTAREA_AVG
FLOORSMIN_MEDI
FLAG_MOBIL
FLOORSMAX_MEDI
FLAG_DOCUMENT_2
FLAG_DOCUMENT_10
LANDAREA_MODE
COMMONAREA_MODE
DAYS_ID_PUBLISH
LIVINGAPARTMENTS_AVG
YEARS_BUILD_MODE
DEF_30_CNT_SOCIAL_CIRCLE
FLOORSMAX_AVG
AMT_CREDIT
AMT_ANNUITY
FLAG_DOCUMENT_15
FLAG_DOCUMENT_19
FLAG_DOCUMENT_7
FLAG_DOCUMENT_13
NONLIVINGAPARTMENTS_MEDI
AMT_INCOME_TOTAL
LIVINGAPARTMENTS_MEDI
FLAG_WORK_PHONE
YEARS_BUILD_MEDI
AMT_REQ_CREDIT_BUREAU_YEAR
FLAG_DOCUMENT_14
ENTRANCES_MEDI
LIVINGAREA_AVG
REGION_RATING_CLIENT
NONLIVINGAPARTMENTS_AVG
APARTMENTS_AVG
AMT_GOODS_PRICE
FLAG_DOCUMENT_4
FLAG_DOCUMENT_9
TOTALAREA_MODE
AMT_REQ_CREDIT_BUREAU_QRT
REGION_RATING_CLIENT_W_CITY
FLAG_DOCUMENT_18
ELEVATORS_MEDI
DAYS_BIRTH
CNT_CHILDREN
FLAG_DOCUMENT_11
FLAG_DOCUMENT_8
NONLIVINGAPARTMENTS_MODE

```
-----------------------------------------------------------------------------
APPLICATION_TRAIN :-------------------------Continous Features
----------------------------
-----------------------------------------------------------------------------
APPLICATION_TRAIN :-------------------------All Missing Features
----------------------------
```

```
-----------------------------------------------------------------------------
APPLICATION_TRAIN :-------------------------DendoGram for Nulls
----------------------------
<Figure size 1440x1440 with 0 Axes>
```

OCCUPATION_TYPE
NAME_TYPE_SUITE
DEF_60_CNT_SOCIAL_CIRCLE
OBS_60_CNT_SOCIAL_CIRCLE
OBS_30_CNT_SOCIAL_CIRCLE
DEF_30_CNT_SOCIAL_CIRCLE
EXT_SOURCE_2
AMT_GOODS_PRICE
AMT_ANNUITY
CNT_FAM_MEMBERS
DAYS_LAST_PHONE_CHANGE
FLAG_DOCUMENT_21
FLAG_DOCUMENT_20
FLAG_DOCUMENT_19
FLAG_DOCUMENT_18
FLAG_DOCUMENT_17
FLAG_DOCUMENT_16
FLAG_DOCUMENT_15
FLAG_DOCUMENT_14
FLAG_DOCUMENT_13
FLAG_DOCUMENT_12
FLAG_DOCUMENT_11
FLAG_DOCUMENT_10
FLAG_DOCUMENT_9
FLAG_DOCUMENT_8
FLAG_DOCUMENT_7
FLAG_DOCUMENT_6
FLAG_DOCUMENT_5
FLAG_DOCUMENT_4
FLAG_DOCUMENT_3
FLAG_DOCUMENT_2
ORGANIZATION_TYPE
LIVE_CITY_NOT_WORK_CITY
REG_CITY_NOT_WORK_CITY
REG_CITY_NOT_LIVE_CITY
LIVE_REGION_NOT_WORK_REGION
REG_REGION_NOT_WORK_REGION
REG_REGION_NOT_LIVE_REGION
HOUR_APPR_PROCESS_START
WEEKDAY_APPR_PROCESS_START
REGION_RATING_CLIENT_W_CITY
REGION_RATING_CLIENT
FLAG_EMAIL
FLAG_PHONE
FLAG_CONT_MOBILE
FLAG_WORK_PHONE
FLAG_EMP_PHONE
FLAG_MOBIL
DAYS_ID_PUBLISH
DAYS_REGISTRATION
DAYS_EMPLOYED
DAYS_BIRTH
REGION_POPULATION_RELATIVE
NAME_HOUSING_TYPE
NAME_FAMILY_STATUS
NAME_EDUCATION_TYPE
NAME_INCOME_TYPE
AMT_CREDIT
AMT_INCOME_TOTAL
CNT_CHILDREN
FLAG_OWN_REALTY
FLAG_OWN_CAR
CODE_GENDER
NAME_CONTRACT_TYPE
SK_ID_CURR
TARGET
EXT_SOURCE_3
AMT_REQ_CREDIT_BUREAU_YEAR
AMT_REQ_CREDIT_BUREAU_QRT
AMT_REQ_CREDIT_BUREAU_MON
AMT_REQ_CREDIT_BUREAU_WEEK
AMT_REQ_CREDIT_BUREAU_HOUR
AMT_REQ_CREDIT_BUREAU_DAY
COMMONAREA_MEDI
COMMONAREA_AVG
COMMONAREA_MODE
FONDKAPREMONT_MODE
LIVINGAPARTMENTS_MEDI
LIVINGAPARTMENTS_AVG
LIVINGAPARTMENTS_MODE
NONLIVINGAPARTMENTS_MEDI
NONLIVINGAPARTMENTS_AVG
NONLIVINGAPARTMENTS_MODE
FLOORSMIN_MEDI
FLOORSMIN_AVG
FLOORSMIN_MODE
YEARS_BUILD_MEDI
YEARS_BUILD_AVG
YEARS_BUILD_MODE
NONLIVINGAREA_MEDI
NONLIVINGAREA_AVG
NONLIVINGAREA_MODE
ELEVATORS_MEDI
ELEVATORS_AVG
ELEVATORS_MODE
APARTMENTS_MEDI
APARTMENTS_AVG
APARTMENTS_MODE
HOUSETYPE_MODE
WALLSMATERIAL_MODE
ENTRANCES_MEDI
ENTRANCES_AVG
ENTRANCES_MODE
FLOORSMAX_MEDI
FLOORSMAX_AVG
FLOORSMAX_MODE
LIVINGAREA_MEDI
LIVINGAREA_AVG
LIVINGAREA_MODE
EMERGENCYSTATE_MODE
TOTALAREA_MODE
YEARS_BEGINEXPLUATATION_MEDI
YEARS_BEGINEXPLUATATION_AVG
YEARS_BEGINEXPLUATATION_MODE
BASEMENTAREA_MEDI
BASEMENTAREA_AVG
BASEMENTAREA_MODE
LANDAREA_MEDI
LANDAREA_AVG
LANDAREA_MODE
OWN_CAR_AGE
EXT_SOURCE_1
400
350
300
250
200
150
100
50
0

```
----------------------------------------------------------------------
Table under consideration: BUREAU
----------------------------------------------------------------------
----------------------------------------------------------------------
BUREAU :-------------------------Type of Features
--------------------------------
```

```
----------------------------------------------------------------------
BUREAU :-------------------------UNIQUE VALUES
-----------------------------------
```

```
----------------------------------------------------------------------
BUREAU :-------------------------MISSING PERCENTAGE
------------------------------
```

```
--------------------------------------------------------------------------
BUREAU :-------------------------CATEGORICAL COUNT
-------------------------------
```

```
--------------------------------------------------------------------------
BUREAU :------------------------NUM FEATURES-DOT PLOT----------------------
<Figure size 1440x1440 with 0 Axes>
```

```
------------------------------------------------------------------------
BUREAU :-------------------------NUM FEATURES - HISTOGRAM
----------------------
<Figure size 1440x1440 with 0 Axes>
```

DAYS_CREDIT_ENDDATE
DAYS_CREDIT_UPDATE
CNT_CREDIT_PROLONG
AMT_CREDIT_SUM_OVERDUE
CREDIT_DAY_OVERDUE
AMT_CREDIT_SUM_LIMIT
AMT_ANNUITY
DAYS_CREDIT
AMT_CREDIT_SUM
AMT_CREDIT_SUM_DEBT
DAYS_ENDDATE_FACT
AMT_CREDIT_MAX_OVERDUE

```
-----------------------------------------------------------------------
BUREAU :-------------------------Continous Features
----------------------------
-----------------------------------------------------------------------
BUREAU :-------------------------All Missing Features
---------------------------
```

```
-----------------------------------------------------------------------
BUREAU :-------------------------DendoGram for Nulls
----------------------------
<Figure size 1440x1440 with 0 Axes>
```

```
-----------------------------------------------------------------------
Table under consideration: CREDIT_CARD_BALANCE
-----------------------------------------------------------------------
-----------------------------------------------------------------------
CREDIT_CARD_BALANCE :-------------------------Type of Features
-------------------------------
```

CREDIT_CARD_BALANCE :----------UNIQUE VALUES

CREDIT_CARD_BALANCE :----------MISSING PERCENTAGE

```
--------------------------------------------------------------------------------
CREDIT_CARD_BALANCE :-------------------------CATEGORICAL COUNT
-------------------------------
```

```
--------------------------------------------------------------------------------
CREDIT_CARD_BALANCE :-------------------------NUM FEATURES-DOT PLOT
----------------------
<Figure size 1440x1440 with 0 Axes>
```

```
----------------------------------------------------------------------
CREDIT_CARD_BALANCE :-------------------------NUM FEATURES - HISTOGRAM
----------------------
<Figure size 1440x1440 with 0 Axes>
```

AMT_RECIVABLE
AMT_DRAWINGS_POS_CURRENT
AMT_RECEIVABLE_PRINCIPAL
AMT_PAYMENT_TOTAL_CURRENT
SK_DPD_DEF
AMT_TOTAL_RECEIVABLE
AMT_CREDIT_LIMIT_ACTUAL
AMT_PAYMENT_CURRENT
MONTHS_BALANCE
CNT_DRAWINGS_CURRENT
AMT_DRAWINGS_CURRENT
SK_DPD
AMT_BALANCE
CNT_DRAWINGS_OTHER_CURRENT
AMT_DRAWINGS_ATM_CURRENT
CNT_DRAWINGS_POS_CURRENT
AMT_DRAWINGS_OTHER_CURRENT
AMT_INST_MIN_REGULARITY
CNT_INSTALMENT_MATURE_CUM
CNT_DRAWINGS_ATM_CURRENT

```
--------------------------------------------------------------------------
CREDIT_CARD_BALANCE :-------------------------Continous Features
----------------------------
--------------------------------------------------------------------------
CREDIT_CARD_BALANCE :-------------------------All Missing Features
----------------------------
```

```
--------------------------------------------------------------------------
CREDIT_CARD_BALANCE :-------------------------DendoGram for Nulls
----------------------------
<Figure size 1440x1440 with 0 Axes>
```

```
--------------------------------------------------------------------------
Table under consideration: INSTALLMENTS_PAYMENTS
--------------------------------------------------------------------------
--------------------------------------------------------------------------
INSTALLMENTS_PAYMENTS :-------------------------Type of Features
-------------------------------
```

INSTALLMENTS_PAYMENTS :----------UNIQUE VALUES

INSTALLMENTS_PAYMENTS :----------MISSING PERCENTAGE

```
----------------------------------------------------------------------
INSTALLMENTS_PAYMENTS :-------------------------CATEGORICAL COUNT
------------------------------
<Figure size 1440x1440 with 0 Axes>
----------------------------------------------------------------------
INSTALLMENTS_PAYMENTS :-------------------------NUM FEATURES-DOT PLOT
-----------------------
<Figure size 1440x1440 with 0 Axes>
```

```
-----------------------------------------------------------------------
INSTALLMENTS_PAYMENTS :-------------------------NUM FEATURES - HISTOGRAM
----------------------
<Figure size 1440x1440 with 0 Axes>
```

NUM_INSTALMENT_VERSION
AMT_PAYMENT
DAYS_INSTALMENT
DAYS_ENTRY_PAYMENT
AMT_INSTALMENT

NUM_INSTALMENT_NUMBER
DIFF_PAYMNT_INSTLMNT

```
----------------------------------------------------------------------
INSTALLMENTS_PAYMENTS :-------------------------Continous Features
----------------------------
----------------------------------------------------------------------
INSTALLMENTS_PAYMENTS :-------------------------All Missing Features
----------------------------
```

```
----------------------------------------------------------------------
INSTALLMENTS_PAYMENTS :-------------------------DendoGram for Nulls
----------------------------
<Figure size 1440x1440 with 0 Axes>
```

```
----------------------------------------------------------------------
Table under consideration: PREVIOUS_APPLICATION
----------------------------------------------------------------------
----------------------------------------------------------------------
PREVIOUS_APPLICATION :-------------------------Type of Features
--------------------------------
```

PREVIOUS_APPLICATION :----------UNIQUE VALUES

PREVIOUS_APPLICATION :----------MISSING PERCENTAGE

```
------------------------------------------------------------------------------
PREVIOUS_APPLICATION :-------------------------CATEGORICAL COUNT
------------------------------
```

```
--------------------------------------------------------------------------------
PREVIOUS_APPLICATION :-------------------------NUM FEATURES-DOT PLOT
----------------------
<Figure size 1440x1440 with 0 Axes>
```

```
----------------------------------------------------------------------------
PREVIOUS_APPLICATION :-------------------------NUM FEATURES - HISTOGRAM
-----------------------
<Figure size 1440x1440 with 0 Axes>
```

AMT_DOWN_PAYMENT
HOUR_APPR_PROCESS_START
NFLAG_INSURED_ON_APPROVAL
PREV_DWN_PYMNT_CRDT_RATIO
RATE_INTEREST_PRIMARY
RATE_DOWN_PAYMENT
DAYS_TERMINATION
NFLAG_LAST_APPL_IN_DAY
PREV_REJ_CNT
PREV_APCTN_CRDT_RATIO
PREV_APRV_CNT
PREV_CRDT_ANNUTY_RATIO
AMT_GOODS_PRICE
DAYS_FIRST_DUE
RATE_INTEREST_PRIVILEGED
DAYS_DECISION
PREV_APCTN_CRDT_DIFF
DAYS_LAST_DUE
AMT_CREDIT
AMT_ANNUITY
AMT_APPLICATION
CNT_PAYMENT
SELLERPLACE_AREA
DAYS_FIRST_DRAWING
DAYS_LAST_DUE_1ST_VERSION

PREVIOUS_APPLICATION :-------------------------Continous Features
---

PREVIOUS_APPLICATION :-------------------------All Missing Features
---

PREVIOUS_APPLICATION :-------------------------DendoGram for Nulls
---
<Figure size 1440x1440 with 0 Axes>

Table under consideration: POS_CASH_BALANCE

POS_CASH_BALANCE :-------------------------Type of Features

POS_CASH_BALANCE :----------UNIQUE VALUES

POS_CASH_BALANCE :----------MISSING PERCENTAGE

```
-----------------------------------------------------------------------------
POS_CASH_BALANCE :-------------------------CATEGORICAL COUNT
-------------------------------
```

```
-----------------------------------------------------------------------------
POS_CASH_BALANCE :-------------------------NUM FEATURES-DOT PLOT
----------------------
<Figure size 1440x1440 with 0 Axes>
```

```
-----------------------------------------------------------------------------
POS_CASH_BALANCE :-------------------------NUM FEATURES - HISTOGRAM
----------------------
<Figure size 1440x1440 with 0 Axes>
```

CNT_INSTALMENT
1e6
2.5
2.0
1.5
1.0
CNT_INSTALMENT_FUTURE
1e6
2.5
2.0
1.5
1.0
POS_PERC_INSTL_PNDNG
1e6
3.0
2.5
2.0
1.5
1.0
POS_CNT_INSTAL_PNDNG
1e6
3.5
3.0
2.5
2.0
1.5
1.0

```
--------------------------------------------------------------------------
POS_CASH_BALANCE :-------------------------Continous Features
----------------------------
--------------------------------------------------------------------------
POS_CASH_BALANCE :-------------------------All Missing Features
----------------------------
```

```
--------------------------------------------------------------------------
POS_CASH_BALANCE :-------------------------DendoGram for Nulls
----------------------------
<Figure size 1440x1440 with 0 Axes>
```

```
----------------------------------------------------------------------
```

Correlation Plots Maps

A correlation matrix is a table showing correlation coefficients between variables. Each cell in the table shows the correlation between two variables. This correlation is a value put on the relationship between two attributes. A correlation matrix is used to summarize data, then with that information do more advanced analysis.

In [80]:

```
def corr_plot(data, remove=["Id"], corr_coef = "pearson", figsize=(20, 20))
    if len(remove) > 0:
        num_cols2 = [x for x in data.columns if (x not in remove)]

    sns.set(font_scale=1.1)
    c = data[num_cols2].corr(method = corr_coef)
    mask = np.triu(c.corr(method = corr_coef))
    plt.figure(figsize=figsize)
    sns.heatmap(c,
                annot=True,
                fmt='.1f',
                cmap='coolwarm',
                square=True,
                mask=mask,
                linewidths=1,
                cbar=False)
    plt.show()
```

In [81]:

```
for i,ds_name in enumerate(datasets.keys()):
    if(ds_name != "bureau_balance"):
        print("------------------------------------------------------------
        print("Table under consideration FOR CORRELATION PLOT:",ds_name.upp
        corr_plot(datasets[ds_name], remove=['SK_ID_CURR','SK_ID_BUREAU'], 
        print("------------------------------------------------------------
        print("------------------------------------------------------------
```

```
----------------------------------------------------------------------
Table under consideration FOR CORRELATION PLOT: APPLICATION_TRAIN
```

Table under consideration FOR CORRELATION PLOT: APPLICATION_TEST

Table under consideration FOR CORRELATION PLOT: BUREAU

Table under consideration FOR CORRELATION PLOT: CREDIT_CARD_BALANCE

Table under consideration FOR CORRELATION PLOT: INSTALLMENTS_PAYMENTS

Table under consideration FOR CORRELATION PLOT: PREVIOUS_APPLICATION

Table under consideration FOR CORRELATION PLOT: POS_CASH_BALANCE

As we have seen in Application train and application test dataframes, most of the **FLAG_DOCUMENT_XX** are null, we will remove these columns at later point of time

Numerical Plots and Visualizations

These visualizations consist of the paired plots for each column in the each table.

The first is a histogram, a diagram consisting of rectangles whose area is proportional to the frequency of a variable and whose width is equal to the class interval, which helps to visualize distribution.

The second is a boxplot, which is a standardized way of displaying the distribution of data based on a five number summary (“minimum”, first quartile Q1, median, third quartile Q3 and “maximum”). It can tell you about your outliers and what their values are.

https://builtin.com/data-science/boxplot#:~:text=What%20Is%20a%20Boxplot%3F,and%20what%20their%20values%20are.

The third is a Kernel Density Plot which also helps to visualize distriubtion over a time peroid or some continuous value.

In [ ]:

```
def grab_col_names(dataframe, cat_th=10, car_th=20, show_date=False):
    date_cols = [col for col in dataframe.columns if dataframe[col].dtypes =
    cat_cols = dataframe.select_dtypes(["object", "category"]).columns.toli

    num_but_cat = [col for col in dataframe.select_dtypes(["float", "intege
    cat_but_car = [col for col in dataframe.select_dtypes(["object", "categ

    cat_cols = cat_cols + num_but_cat
    cat_cols = [col for col in cat_cols if col not in cat_but_car]

    num_cols = dataframe.select_dtypes(["float", "integer"]).columns
    num_cols = [col for col in num_cols if col not in num_but_cat]

    if show_date == True:
        return date_cols, cat_cols, cat_but_car, num_cols, num_but_cat
    else:
        return cat_cols, cat_but_car, num_cols, num_but_cat
```

In [82]:

```
def num_plot(data, num_cols, remove=["Id"], hist_bins=10, figsize=(20, 4)):

    if len(remove) > 0:
        num_cols2 = [x for x in num_cols if (x not in remove)]

    for i in num_cols2:
        fig, axes = plt.subplots(1, 3, figsize=figsize)
        data.hist(str(i), bins=hist_bins, ax=axes[0])
        data.boxplot(str(i), ax=axes[1], vert=False);
        try:
            sns.kdeplot(np.array(data[str(i)]))
        except:
            ValueError

        axes[1].set_yticklabels([])
        axes[1].set_yticks([])
        axes[0].set_title(i + " | Histogram")
        axes[1].set_title(i + " | Boxplot")
        axes[2].set_title(i + " | Density")
        plt.show()
```

In [83]:

```
for i,ds_name in enumerate(datasets.keys()):

    if ds_name.lower() not in ("application_train",
                               "application_test", "bureau_balance"):
        print("-----------------------------------------------------------------
        #_,_, num_cols, _ = grab_col_names(datasets[ds_name], car_th=10)
        _,num_cols,_,_ = id_num_cat_feature(datasets[ds_name], text = False
        datasets[ds_name].replace([np.inf, -np.inf], 0, inplace=True)
        print("Table under consideration FOR NUMERICAL PLOTS:",ds_name.uppe
        num_plot(datasets[ds_name], num_cols, remove=['SK_ID_CURR','SK_ID_BU
        print("-----------------------------------------------------------------
        print("-----------------------------------------------------------------
```

```
-----------------------------------------------------------------------
Table under consideration FOR NUMERICAL PLOTS: BUREAU
```

DAYS_CREDIT_ENDDATE | Histogram
DAYS_CREDIT_ENDDATE | Boxplot
DAYS_CREDIT_ENDDATE | Density
DAYS_ENDDATE_FACT | Histogram
DAYS_ENDDATE_FACT | Boxplot
DAYS_ENDDATE_FACT | Density
AMT_CREDIT_MAX_OVERDUE | Histogram
AMT_CREDIT_MAX_OVERDUE | Boxplot
AMT_CREDIT_MAX_OVERDUE | Density
CNT_CREDIT_PROLONG | Histogram
CNT_CREDIT_PROLONG | Boxplot
CNT_CREDIT_PROLONG | Density
AMT_CREDIT_SUM | Histogram
AMT_CREDIT_SUM | Boxplot
AMT_CREDIT_SUM | Density
AMT_CREDIT_SUM_DEBT | Histogram
AMT_CREDIT_SUM_DEBT | Boxplot
AMT_CREDIT_SUM_DEBT | Density
Density
1e6
1e8

Table under consideration FOR NUMERICAL PLOTS: CREDIT_CARD_BALANCE

AMT_BALANCE | Histogram
AMT_BALANCE | Boxplot
AMT_BALANCE | Density
AMT_CREDIT_LIMIT_ACTUAL | Histogram
AMT_CREDIT_LIMIT_ACTUAL | Boxplot
AMT_CREDIT_LIMIT_ACTUAL | Density
AMT_DRAWINGS_ATM_CURRENT | Histogram
AMT_DRAWINGS_ATM_CURRENT | Boxplot
AMT_DRAWINGS_ATM_CURRENT | Density
AMT_DRAWINGS_CURRENT | Histogram
AMT_DRAWINGS_CURRENT | Boxplot
AMT_DRAWINGS_CURRENT | Density
AMT_DRAWINGS_OTHER_CURRENT | Histogram
AMT_DRAWINGS_OTHER_CURRENT | Boxplot
AMT_DRAWINGS_OTHER_CURRENT | Density
Density
1e6
1e−5

AMT_DRAWINGS_POS_CURRENT | Histogram
AMT_DRAWINGS_POS_CURRENT | Boxplot
AMT_DRAWINGS_POS_CURRENT | Density
AMT_INST_MIN_REGULARITY | Histogram
AMT_INST_MIN_REGULARITY | Boxplot
AMT_INST_MIN_REGULARITY | Density
AMT_PAYMENT_CURRENT | Histogram
AMT_PAYMENT_CURRENT | Boxplot
AMT_PAYMENT_CURRENT | Density
AMT_PAYMENT_TOTAL_CURRENT | Histogram
AMT_PAYMENT_TOTAL_CURRENT | Boxplot
AMT_PAYMENT_TOTAL_CURRENT | Density
AMT_RECEIVABLE_PRINCIPAL | Histogram
AMT_RECEIVABLE_PRINCIPAL | Boxplot
AMT_RECEIVABLE_PRINCIPAL | Density
AMT_RECIVABLE | Histogram
AMT_RECIVABLE | Boxplot
AMT_RECIVABLE | Density
Density
1e6

AMT_TOTAL_RECEIVABLE | Histogram
AMT_TOTAL_RECEIVABLE | Boxplot
AMT_TOTAL_RECEIVABLE | Density
CNT_DRAWINGS_ATM_CURRENT | Histogram
CNT_DRAWINGS_ATM_CURRENT | Boxplot
CNT_DRAWINGS_ATM_CURRENT | Density
CNT_DRAWINGS_CURRENT | Histogram
CNT_DRAWINGS_CURRENT | Boxplot
CNT_DRAWINGS_CURRENT | Density
CNT_DRAWINGS_OTHER_CURRENT | Histogram
CNT_DRAWINGS_OTHER_CURRENT | Boxplot
CNT_DRAWINGS_OTHER_CURRENT | Density
CNT_DRAWINGS_POS_CURRENT | Histogram
CNT_DRAWINGS_POS_CURRENT | Boxplot
CNT_DRAWINGS_POS_CURRENT | Density
CNT_INSTALMENT_MATURE_CUM | Histogram
CNT_INSTALMENT_MATURE_CUM | Boxplot
CNT_INSTALMENT_MATURE_CUM | Density
Density

Table under consideration FOR NUMERICAL PLOTS: INSTALLMENTS_PAYMENTS

Table under consideration FOR NUMERICAL PLOTS: PREVIOUS_APPLICATION

AMT_APPLICATION | Histogram
AMT_APPLICATION | Boxplot
AMT_APPLICATION | Density
AMT_CREDIT | Histogram
AMT_CREDIT | Boxplot
AMT_CREDIT | Density
AMT_DOWN_PAYMENT | Histogram
AMT_DOWN_PAYMENT | Boxplot
AMT_DOWN_PAYMENT | Density
AMT_GOODS_PRICE | Histogram
AMT_GOODS_PRICE | Boxplot
AMT_GOODS_PRICE | Density
HOUR_APPR_PROCESS_START | Histogram
HOUR_APPR_PROCESS_START | Boxplot
HOUR_APPR_PROCESS_START | Density
NFLAG_LAST_APPL_IN_DAY | Histogram
NFLAG_LAST_APPL_IN_DAY | Boxplot
NFLAG_LAST_APPL_IN_DAY | Density
Density
1e6
1e−6

RATE_DOWN_PAYMENT | Histogram
RATE_DOWN_PAYMENT | Boxplot
RATE_DOWN_PAYMENT | Density
Density
RATE_INTEREST_PRIMARY | Histogram
RATE_INTEREST_PRIMARY | Boxplot
RATE_INTEREST_PRIMARY | Density
Density
RATE_INTEREST_PRIVILEGED | Histogram
RATE_INTEREST_PRIVILEGED | Boxplot
RATE_INTEREST_PRIVILEGED | Density
Density
DAYS_DECISION | Histogram
DAYS_DECISION | Boxplot
DAYS_DECISION | Density
Density
SELLERPLACE_AREA | Histogram
1e6
SELLERPLACE_AREA | Boxplot
1e6
SELLERPLACE_AREA | Density
1e−6
Density
1e6
CNT_PAYMENT | Histogram
CNT_PAYMENT | Boxplot
CNT_PAYMENT | Density
Density

DAYS_FIRST_DRAWING | Histogram
DAYS_FIRST_DRAWING | Boxplot
DAYS_FIRST_DRAWING | Density
DAYS_FIRST_DUE | Histogram
DAYS_FIRST_DUE | Boxplot
DAYS_FIRST_DUE | Density
DAYS_LAST_DUE_1ST_VERSION | Histogram
DAYS_LAST_DUE_1ST_VERSION | Boxplot
DAYS_LAST_DUE_1ST_VERSION | Density
DAYS_LAST_DUE | Histogram
DAYS_LAST_DUE | Boxplot
DAYS_LAST_DUE | Density
DAYS_TERMINATION | Histogram
DAYS_TERMINATION | Boxplot
DAYS_TERMINATION | Density
NFLAG_INSURED_ON_APPROVAL | Histogram
NFLAG_INSURED_ON_APPROVAL | Boxplot
NFLAG_INSURED_ON_APPROVAL | Density
Density

PREV_APRV_CNT | Histogram
PREV_APRV_CNT | Boxplot
PREV_APRV_CNT | Density
PREV_REJ_CNT | Histogram
PREV_REJ_CNT | Boxplot
PREV_REJ_CNT | Density
PREV_APCTN_CRDT_DIFF | Histogram
PREV_APCTN_CRDT_DIFF | Boxplot
PREV_APCTN_CRDT_DIFF | Density
PREV_APCTN_CRDT_RATIO | Histogram
PREV_APCTN_CRDT_RATIO | Boxplot
PREV_APCTN_CRDT_RATIO | Density
PREV_CRDT_ANNUTY_RATIO | Histogram
PREV_CRDT_ANNUTY_RATIO | Boxplot
PREV_CRDT_ANNUTY_RATIO | Density
PREV_DWN_PYMNT_CRDT_RATIO | Histogram
PREV_DWN_PYMNT_CRDT_RATIO | Boxplot
PREV_DWN_PYMNT_CRDT_RATIO | Density
Density

```
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
Table under consideration FOR NUMERICAL PLOTS: POS_CASH_BALANCE
```

```
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
```

Check Skewness/Distribution : Numerical data

Look for skewed column in numerical data but ignore dates, Days,Flags, status, ID's.

Skewness in : AMT_INCOME_TOTAL,AMT_CREDIT,AMT_ANNUITY

In [84]:

```
application_train = datasets["application_train"]
numerical_ix = application_train.select_dtypes(include=['int64', 'float64']
categorical_ix = application_train.select_dtypes(include=['object', 'bool']
num_features = list(numerical_ix)
cat_features = list(categorical_ix)
print(f"# of numerical   features: {len(numerical_ix)}")
print(f"Numerical   features: {numerical_ix}")
print('--------')
print(f"# of categorical features: {len(categorical_ix)}")
print(f"Categorical features: {categorical_ix}")
```

```
# of numerical   features: 106
Numerical   features: Index(['SK_ID_CURR', 'TARGET', 'CNT_CHILDREN', 'AMT_IN
COME_TOTAL',
       'AMT_CREDIT', 'AMT_ANNUITY', 'AMT_GOODS_PRICE',
       'REGION_POPULATION_RELATIVE', 'DAYS_BIRTH', 'DAYS_EMPLOYED',
       ...
       'FLAG_DOCUMENT_18', 'FLAG_DOCUMENT_19', 'FLAG_DOCUMENT_20',
       'FLAG_DOCUMENT_21', 'AMT_REQ_CREDIT_BUREAU_HOUR',
       'AMT_REQ_CREDIT_BUREAU_DAY', 'AMT_REQ_CREDIT_BUREAU_WEEK',
       'AMT_REQ_CREDIT_BUREAU_MON', 'AMT_REQ_CREDIT_BUREAU_QRT',
       'AMT_REQ_CREDIT_BUREAU_YEAR'],
      dtype='object', length=106)
--------
# of categorical features: 16
Categorical features: Index(['NAME_CONTRACT_TYPE', 'CODE_GENDER', 'FLAG_OWN_
CAR', 'FLAG_OWN_REALTY',
       'NAME_TYPE_SUITE', 'NAME_INCOME_TYPE', 'NAME_EDUCATION_TYPE',
       'NAME_FAMILY_STATUS', 'NAME_HOUSING_TYPE', 'OCCUPATION_TYPE',
       'WEEKDAY_APPR_PROCESS_START', 'ORGANIZATION_TYPE', 'FONDKAPREMONT_MOD
E',
       'HOUSETYPE_MODE', 'WALLSMATERIAL_MODE', 'EMERGENCYSTATE_MODE'],
      dtype='object')
```

In [85]:

```
datasets["application_train"][numerical_ix]
```

Out[85]:

	SK_ID_CURR	TARGET	CNT_CHILDREN	AMT_INCOME_TOTAL	AMT_CREDIT	AMT_ANN
0	100002	1	0	202500.0	406597.5	247
1	100003	0	0	270000.0	1293502.5	356
2	100004	0	0	67500.0	135000.0	67
3	100006	0	0	135000.0	312682.5	296
4	100007	0	0	121500.0	513000.0	218
...	...	...	...	...	...	
307506	456251	0	0	157500.0	254700.0	275
307507	456252	0	0	72000.0	269550.0	120
307508	456253	0	0	153000.0	677664.0	299
307509	456254	1	0	171000.0	370107.0	202
307510	456255	0	0	157500.0	675000.0	491

307511 rows × 106 columns

AMT_CREDIT and AMT_ANNUITY looks skewed. we wil do log transformation on these attributes and make them more mormalized.

AMT_CREDIT

In [86]:

```
plt.hist(datasets["application_train"]['AMT_CREDIT'], bins=25);
plt.xlabel('AMT_CREDIT')
plt.ylabel('Count')
plt.title("Distribution - AMT_CREDIT ")
```

Out[86]:

```
Text(0.5, 1.0, 'Distribution - AMT_CREDIT ')
```

In [87]:

```
plt.hist(np.log(datasets["application_train"]['AMT_CREDIT']), bins=30);
plt.xlabel('Log(AMT_CREDIT)')
plt.ylabel('Count')
plt.title("Distribution - Log(AMT_CREDIT) ")
```

Out[87]:

```
Text(0.5, 1.0, 'Distribution - Log(AMT_CREDIT) ')
```

AMT_ANNUITY

In [88]:

```
plt.hist(datasets["application_train"]['AMT_ANNUITY'], bins=25);
plt.xlabel('AMT_ANNUITY')
plt.ylabel('Count')
plt.title("Distribution - AMT_ANNUITY ")
```

Out[88]:

```
Text(0.5, 1.0, 'Distribution - AMT_ANNUITY ')
```

In [89]:

```
plt.hist(np.log(datasets["application_train"]['AMT_ANNUITY']), bins=30);
plt.xlabel('Log(AMT_ANNUITY)')
plt.ylabel('Count')
plt.title("Distribution - Log(AMT_ANNUITY) ")
```

Out[89]:

```
Text(0.5, 1.0, 'Distribution - Log(AMT_ANNUITY) ')
```

Missing data for application train

In [90]:

```
percent = (datasets["application_train"].isnull().sum()/datasets["applicati
sum_missing = datasets["application_train"].isna().sum().sort_values(ascend
missing_application_train_data  = pd.concat([percent, sum_missing], axis=1,
missing_application_train_data.head(20)
```

Out[90]:

	Percent	Train Missing Count
COMMONAREA_MEDI	69.87	214865
COMMONAREA_AVG	69.87	214865
COMMONAREA_MODE	69.87	214865
NONLIVINGAPARTMENTS_MODE	69.43	213514
NONLIVINGAPARTMENTS_AVG	69.43	213514
NONLIVINGAPARTMENTS_MEDI	69.43	213514
FONDKAPREMONT_MODE	68.39	210295
LIVINGAPARTMENTS_MODE	68.35	210199
LIVINGAPARTMENTS_AVG	68.35	210199
LIVINGAPARTMENTS_MEDI	68.35	210199
FLOORSMIN_AVG	67.85	208642
FLOORSMIN_MODE	67.85	208642
FLOORSMIN_MEDI	67.85	208642
YEARS_BUILD_MEDI	66.50	204488
YEARS_BUILD_MODE	66.50	204488
YEARS_BUILD_AVG	66.50	204488
OWN_CAR_AGE	65.99	202929
LANDAREA_MEDI	59.38	182590
LANDAREA_MODE	59.38	182590
LANDAREA_AVG	59.38	182590

Determine more than 50% null

Determine attributes which have more than 50% NULLS. Once done, these will be used as part of feature engineering.

In [93]:

```
nulls_50 = missing_application_train_data[round(missing_application_train_da
#nulls_50.index

remove_num_nulls = list(set(nulls_50.index).intersection(set(numerical_ix))
remove_cat_nulls = list(set(nulls_50.index).intersection(set(categorical_ix
```

In [94]:

```
percent = (datasets["application_test"].isnull().sum()/datasets["applicatio
sum_missing = datasets["application_test"].isna().sum().sort_values(ascendi
missing_application_train_data  = pd.concat([percent, sum_missing], axis=1,
missing_application_train_data.head(20)
```

Out[94]:

	Percent	Test Missing Count
COMMONAREA_AVG	68.72	33495
COMMONAREA_MODE	68.72	33495
COMMONAREA_MEDI	68.72	33495
NONLIVINGAPARTMENTS_AVG	68.41	33347
NONLIVINGAPARTMENTS_MODE	68.41	33347
NONLIVINGAPARTMENTS_MEDI	68.41	33347
FONDKAPREMONT_MODE	67.28	32797
LIVINGAPARTMENTS_AVG	67.25	32780
LIVINGAPARTMENTS_MODE	67.25	32780
LIVINGAPARTMENTS_MEDI	67.25	32780
FLOORSMIN_MEDI	66.61	32466
FLOORSMIN_AVG	66.61	32466
FLOORSMIN_MODE	66.61	32466
OWN_CAR_AGE	66.29	32312
YEARS_BUILD_AVG	65.28	31818
YEARS_BUILD_MEDI	65.28	31818
YEARS_BUILD_MODE	65.28	31818
LANDAREA_MEDI	57.96	28254
LANDAREA_AVG	57.96	28254
LANDAREA_MODE	57.96	28254

Distribution of the target column

In [95]:

```
datasets["application_train"]['TARGET'].astype(int).plot.hist();
```

This shows that around 8% of people are not able to repay the loans back.

In [96]:
```
datasets["application_train"]['TARGET'].value_counts()/datasets["applicatio
```

Out[96]:
```
0    0.919271
1    0.080729
Name: TARGET, dtype: float64
```

Correlation with the target column

In [97]:
```
def correlation_against_target(df):
    df_joined = {}
    df_joined =  datasets['application_train'][["SK_ID_CURR", "TARGET"]].me
    cols = df.columns
    keys = list(df.columns)
    keys.append("TARGET")
    return df_joined[keys].corr()['TARGET'].sort_values(ascending=False)
```

Correlations against the target on the Original Tables

In [98]:
```
original_corr = {}
#barplot
for i,ds_name in enumerate(datasets.keys()):
    if(ds_name.upper() != "APPLICATION_TRAIN"):
        if (ds_name.upper() != "APPLICATION_TEST"):
            if (ds_name.upper() != "BUREAU_BALANCE"):
                if (ds_name.upper() == "BUREAU"):
                    if 'AMT_ANNUITY' in datasets["bureau"].columns:
                        datasets["bureau"] = datasets["bureau"].drop(["AMT_
                print("-------------------------------------------------
                print("Correlation for Orignal Table:", ds_name.upper())
                print("-------------------------------------------------
                ds = correlation_against_target(datasets[ds_name])
                print("-------------------------------------------------
                original_corr[ds_name] = ds
                print(ds)
                print("-------------------------------------------------
                print("-------------------------------------------------
```

```
----------------------------------------------------------------------
Correlation for Orignal Table: BUREAU
----------------------------------------------------------------------
----------------------------------------------------------------------
TARGET                     1.000000
DAYS_CREDIT                0.061556
DAYS_CREDIT_UPDATE         0.041076
DAYS_ENDDATE_FACT          0.039057
DAYS_CREDIT_ENDDATE        0.026497
AMT_CREDIT_SUM_OVERDUE     0.006253
CREDIT_DAY_OVERDUE         0.002652
AMT_CREDIT_SUM_DEBT        0.002539
AMT_CREDIT_MAX_OVERDUE     0.001587
CNT_CREDIT_PROLONG         0.001523
SK_ID_CURR                -0.003024
AMT_CREDIT_SUM_LIMIT      -0.005990
SK_ID_BUREAU              -0.009018
AMT_CREDIT_SUM            -0.010606
Name: TARGET, dtype: float64
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
Correlation for Orignal Table: CREDIT_CARD_BALANCE
----------------------------------------------------------------------
----------------------------------------------------------------------
TARGET                         1.000000
AMT_BALANCE                    0.050098
AMT_TOTAL_RECEIVABLE           0.049839
AMT_RECIVABLE                  0.049803
AMT_RECEIVABLE_PRINCIPAL       0.049692
AMT_INST_MIN_REGULARITY        0.039798
CNT_DRAWINGS_ATM_CURRENT       0.038437
CNT_DRAWINGS_CURRENT           0.037793
CNT_DRAWINGS_POS_CURRENT       0.029536
AMT_DRAWINGS_ATM_CURRENT       0.024700
AMT_DRAWINGS_CURRENT           0.022378
AMT_CREDIT_LIMIT_ACTUAL        0.013823
AMT_PAYMENT_CURRENT            0.012929
AMT_PAYMENT_TOTAL_CURRENT      0.012302
SK_DPD_DEF                     0.010538
AMT_DRAWINGS_POS_CURRENT       0.005084
AMT_DRAWINGS_OTHER_CURRENT     0.003843
CNT_DRAWINGS_OTHER_CURRENT     0.003044
SK_ID_PREV                     0.002571
SK_DPD                         0.001684
SK_ID_CURR                    -0.004617
CNT_INSTALMENT_MATURE_CUM     -0.023684
MONTHS_BALANCE                -0.035695
Name: TARGET, dtype: float64
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
Correlation for Orignal Table: INSTALLMENTS_PAYMENTS
----------------------------------------------------------------------
----------------------------------------------------------------------
TARGET                      1.000000
SK_ID_PREV                 -0.000212
AMT_INSTALMENT             -0.001498
SK_ID_CURR                 -0.002540
```

```
AMT_PAYMENT               -0.003623
IP_DIFF_PAYMNT_INSTLMNT   -0.006376
NUM_INSTALMENT_VERSION    -0.009896
NUM_INSTALMENT_NUMBER     -0.016190
DAYS_INSTALMENT           -0.034974
DAYS_ENTRY_PAYMENT        -0.035122
Name: TARGET, dtype: float64
--------------------------------------------------------------------------
--------------------------------------------------------------------------
--------------------------------------------------------------------------
Correlation for Orignal Table: PREVIOUS_APPLICATION
--------------------------------------------------------------------------
--------------------------------------------------------------------------
TARGET                        1.000000
CNT_PAYMENT                   0.030480
PREV_CRDT_ANNUTY_RATIO        0.029673
RATE_INTEREST_PRIVILEGED      0.028640
SK_ID_PREV                    0.002009
NFLAG_INSURED_ON_APPROVAL     0.000653
AMT_GOODS_PRICE               0.000254
SK_ID_CURR                   -0.001246
RATE_INTEREST_PRIMARY        -0.001470
AMT_CREDIT                   -0.002350
SELLERPLACE_AREA             -0.002539
NFLAG_LAST_APPL_IN_DAY       -0.002887
AMT_APPLICATION              -0.005583
PREV_APCTN_CRDT_DIFF         -0.012277
AMT_ANNUITY                  -0.014922
AMT_DOWN_PAYMENT             -0.016918
DAYS_LAST_DUE_1ST_VERSION    -0.017128
DAYS_TERMINATION             -0.022160
PREV_DWN_PYMNT_CRDT_RATIO    -0.022275
DAYS_LAST_DUE                -0.022598
PREV_APCTN_CRDT_RATIO        -0.022678
RATE_DOWN_PAYMENT            -0.026111
HOUR_APPR_PROCESS_START      -0.027809
DAYS_FIRST_DUE               -0.029025
DAYS_DECISION                -0.039901
PREV_APRV_CNT                -0.049161
DAYS_FIRST_DRAWING           -0.095723
PREV_REJ_CNT                       NaN
Name: TARGET, dtype: float64
--------------------------------------------------------------------------
--------------------------------------------------------------------------
--------------------------------------------------------------------------
Correlation for Orignal Table: POS_CASH_BALANCE
--------------------------------------------------------------------------
--------------------------------------------------------------------------
TARGET                    1.000000
CNT_INSTALMENT_FUTURE     0.021972
CNT_INSTALMENT            0.018506
SK_DPD                    0.009866
SK_DPD_DEF                0.008594
POS_PERC_INSTL_PNDNG      0.008377
POS_DAYS_WTHT_TOLRNC      0.008126
SK_ID_PREV               -0.000056
SK_ID_CURR               -0.002245
POS_CNT_INSTAL_PNDNG     -0.003865
MONTHS_BALANCE           -0.020147
```

```
Name: TARGET, dtype: float64
-----------------------------------------------------------------------
-----------------------------------------------------------------------
```

Correlations on the Transformed Tables done with a right merge against the table

In [99]:

```
transformed_corr = {}
for i,ds_name in enumerate(datasets_transformed.keys()):
    datasets_transformed[ds_name].reset_index()
    #if(ds_name.upper() not in ( "APPLICATION_TRAIN","APPLICATION_TEST","BU
    if(ds_name.upper() != "APPLICATION_TRAIN"):
        if (ds_name.upper() != "APPLICATION_TEST"):
            #if (ds_name.upper() != "POS_CASH_BALANCE"):
                if (ds_name.upper() != "BUREAU_BALANCE"):
                    print("-----------------------------------------------
                    print("Correlation for Transformed Table:", ds_name.upp
                    print("-----------------------------------------------
                    ds = correlation_against_target(datasets_transformed[ds_
                    print("-----------------------------------------------
                    transformed_corr[ds_name] = ds
                    print(ds)
                    print("-----------------------------------------------
                    print("-----------------------------------------------
```

```
----------------------------------------------------------------------
Correlation for Transformed Table: BUREAU
----------------------------------------------------------------------
----------------------------------------------------------------------
TARGE                                                          1.000000
CREDIT_ACTIVE_Active_mean                                      0.072625
CREDIT_ACTIVE_Active_median                                    0.070590
STATUS_1_var_max                                               0.070020
STATUS_1_var_mean                                              0.069149
                                                                  ...
CREDIT_TYPE_Interbank_credit_median                               NaN
CREDIT_TYPE_Cash_loan_non_earmarked_median                        NaN
CREDIT_ACTIVE_Bad_debt_median                                     NaN
CREDIT_TYPE_Mobile_operator_loan_median                           NaN
CREDIT_TYPE_Loan_for_purchase_of_shares_margin_lending_median     NaN
Name: TARGET, Length: 297, dtype: float64
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
Correlation for Transformed Table: POS_CASH_BALANCE
----------------------------------------------------------------------
----------------------------------------------------------------------
TARGET                                                 1.000000
POS_PERC_INSTL_PNDNG_mean                              0.027904
CNT_INSTALMENT_FUTURE_mean                             0.027827
POS_PERC_INSTL_PNDNG_median                            0.027011
CNT_INSTALMENT_FUTURE_median                           0.026968
POS_PERC_INSTL_PNDNG_min                               0.021933
CNT_INSTALMENT_min                                     0.019840
CNT_INSTALMENT_FUTURE_min                              0.019010
CNT_INSTALMENT_mean                                    0.018066
CNT_INSTALMENT_FUTURE_var                              0.017262
CNT_INSTALMENT_FUTURE_max                              0.013324
CNT_INSTALMENT_max                                     0.013296
NAME_CONTRACT_STATUS_Returned_to_the_store_mean        0.012036
CNT_INSTALMENT_var                                     0.011916
NAME_CONTRACT_STATUS_Returned_to_the_store_var         0.011821
CNT_INSTALMENT_median                                  0.011622
NAME_CONTRACT_STATUS_Signed_mean                       0.010492
NAME_CONTRACT_STATUS_Signed_var                        0.009890
NAME_CONTRACT_STATUS_Demand_var                        0.008903
NAME_CONTRACT_STATUS_Active_var                        0.008285
NAME_CONTRACT_STATUS_Amortized_debt_var                0.007904
NAME_CONTRACT_STATUS_Demand_mean                       0.007201
NAME_CONTRACT_STATUS_Amortized_debt_mean               0.006389
POS_CNT_INSTAL_PNDNG_min                               0.006016
NAME_CONTRACT_STATUS_Signed_median                     0.005164
NAME_CONTRACT_STATUS_Returned_to_the_store_median      0.005135
NAME_CONTRACT_STATUS_Demand_median                     0.003160
NAME_CONTRACT_STATUS_Completed_var                     0.002736
NAME_CONTRACT_STATUS_Approved_median                   0.001096
NAME_CONTRACT_STATUS_Approved_var                      0.000944
NAME_CONTRACT_STATUS_Completed_median                  0.000757
NAME_CONTRACT_STATUS_Approved_mean                     0.000699
NAME_CONTRACT_STATUS_Completed_mean                    0.000434
NAME_CONTRACT_STATUS_Canceled_var                     -0.000248
NAME_CONTRACT_STATUS_Canceled_mean                    -0.000248
NAME_CONTRACT_STATUS_XNA_mean                         -0.000784
NAME_CONTRACT_STATUS_Amortized_debt_median            -0.000784
```

```
NAME_CONTRACT_STATUS_XNA_var                        -0.000784
SK_ID_CURR                                          -0.002137
POS_PERC_INSTL_PNDNG_var                            -0.003301
POS_PERC_INSTL_PNDNG_max                            -0.005398
NAME_CONTRACT_STATUS_Active_median                  -0.005739
CNT_INSTALMENT_FUTURE_sum                           -0.005881
NAME_CONTRACT_STATUS_Active_mean                    -0.008031
POS_CNT_INSTAL_PNDNG_median                         -0.008950
POS_CNT_INSTAL_PNDNG_mean                           -0.014498
CNT_INSTALMENT_sum                                  -0.014670
POS_CNT_INSTAL_PNDNG_var                            -0.017377
POS_CNT_INSTAL_PNDNG_max                            -0.024394
POS_CNT_INSTAL_PNDNG_sum                            -0.027543
POS_PERC_INSTL_PNDNG_sum                            -0.030009
SK_ID_PREV_count                                    -0.035632
CNT_INSTALMENT_count                                -0.035802
POS_PERC_INSTL_PNDNG_count                          -0.035805
POS_CNT_INSTAL_PNDNG_count                          -0.035805
CNT_INSTALMENT_FUTURE_count                         -0.035807
NAME_CONTRACT_STATUS_Canceled_median                      NaN
NAME_CONTRACT_STATUS_XNA_median                           NaN
Name: TARGET, dtype: float64
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
Correlation for Transformed Table: PREVIOUS_APPLICATION
----------------------------------------------------------------------
----------------------------------------------------------------------
TARGET                                            1.000000
NAME_CONTRACT_STATUS_Refused_mean                 0.077671
NAME_CONTRACT_STATUS_Refused_var                  0.075867
CODE_REJECT_REASON_XAP_var                        0.071560
NAME_CONTRACT_STATUS_Refused_median               0.064930
                                                     ...
PREV_REJ_CNT_var                                     NaN
NAME_GOODS_CATEGORY_Animals_median                   NaN
NAME_GOODS_CATEGORY_House Construction_median        NaN
NAME_GOODS_CATEGORY_House Construction_mean          NaN
NAME_GOODS_CATEGORY_House Construction_var           NaN
Name: TARGET, Length: 601, dtype: float64
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
Correlation for Transformed Table: CREDIT_CARD_BALANCE
----------------------------------------------------------------------
----------------------------------------------------------------------
TARGET                                        1.000000
CNT_DRAWINGS_ATM_CURRENT_mean                 0.107692
CNT_DRAWINGS_CURRENT_max                      0.101389
AMT_BALANCE_mean                              0.087177
AMT_TOTAL_RECEIVABLE_mean                     0.086490
                                                 ...
SK_DPD_DEF_min                                   NaN
SK_DPD_min                                       NaN
NAME_CONTRACT_STATUS_Refused_median              NaN
NAME_CONTRACT_STATUS_Sent_proposal_median        NaN
NAME_CONTRACT_STATUS_Approved_median             NaN
Name: TARGET, Length: 164, dtype: float64
----------------------------------------------------------------------
```

```
---------------------------------------------------------------------------
---------------------------------------------------------------------------
Correlation for Transformed Table: INSTALLMENTS_PAYMENTS
---------------------------------------------------------------------------
---------------------------------------------------------------------------
TARGET                             1.000000
NUM_INSTALMENT_NUMBER_max          0.006304
DAYS_INSTALMENT_min                0.003231
AMT_INSTALMENT_max                 0.002324
DAYS_ENTRY_PAYMENT_min             0.002298
AMT_PAYMENT_max                    0.001554
NUM_INSTALMENT_NUMBER_var          0.001040
AMT_INSTALMENT_var                -0.002151
NUM_INSTALMENT_NUMBER_min         -0.002334
SK_ID_CURR                        -0.002363
AMT_PAYMENT_var                   -0.003841
NUM_INSTALMENT_NUMBER_mean        -0.009537
IP_DIFF_PAYMNT_INSTLMNT_var       -0.009684
NUM_INSTALMENT_VERSION_var        -0.011427
IP_DIFF_PAYMNT_INSTLMNT_max       -0.014018
IP_DIFF_PAYMNT_INSTLMNT_median    -0.014302
NUM_INSTALMENT_NUMBER_median      -0.014897
NUM_INSTALMENT_NUMBER_sum         -0.017441
AMT_INSTALMENT_mean               -0.018409
NUM_INSTALMENT_VERSION_max        -0.018611
IP_DIFF_PAYMNT_INSTLMNT_min       -0.019144
AMT_INSTALMENT_sum                -0.019811
AMT_INSTALMENT_min                -0.020257
NUM_INSTALMENT_VERSION_median     -0.020722
DAYS_INSTALMENT_count             -0.021096
NUM_INSTALMENT_NUMBER_count       -0.021096
NUM_INSTALMENT_VERSION_count      -0.021096
SK_ID_PREV_count                  -0.021096
AMT_INSTALMENT_count              -0.021096
IP_DIFF_PAYMNT_INSTLMNT_count     -0.021217
AMT_PAYMENT_count                 -0.021217
DAYS_ENTRY_PAYMENT_count          -0.021217
AMT_PAYMENT_mean                  -0.023169
AMT_PAYMENT_sum                   -0.024375
AMT_INSTALMENT_median             -0.024873
AMT_PAYMENT_min                   -0.025724
NUM_INSTALMENT_VERSION_mean       -0.027323
IP_DIFF_PAYMNT_INSTLMNT_sum       -0.027326
IP_DIFF_PAYMNT_INSTLMNT_mean      -0.029339
AMT_PAYMENT_median                -0.029548
NUM_INSTALMENT_VERSION_sum        -0.030063
NUM_INSTALMENT_VERSION_min        -0.032039
DAYS_INSTALMENT_sum               -0.035064
DAYS_ENTRY_PAYMENT_sum            -0.035227
DAYS_INSTALMENT_median            -0.039252
DAYS_ENTRY_PAYMENT_median         -0.039654
DAYS_INSTALMENT_mean              -0.043509
DAYS_ENTRY_PAYMENT_mean           -0.043992
DAYS_ENTRY_PAYMENT_var            -0.052071
DAYS_INSTALMENT_var               -0.052273
DAYS_INSTALMENT_max               -0.058648
DAYS_ENTRY_PAYMENT_max            -0.058794
Name: TARGET, dtype: float64
---------------------------------------------------------------------------
```

Highest Correlations on Original Tables

The below gives you the highest correlated attributes from each table with a threshold of .7. This is correlation to other attributes in the same table.

In [100…

```
# Get high correlated variables
def high_correlation(data, remove=['SK_ID_CURR', 'SK_ID_BUREAU'], corr_coef=
    if len(remove) > 0:
        cols = [x for x in data.columns if (x not in remove)]
        c = data[cols].corr(method=corr_coef)
    else:
        c = data.corr(method=corr_coef)

    for i in c.columns:
        cr = c.loc[i].loc[(c.loc[i] >= corr_value) | (c.loc[i] <= -corr_valu
        if len(cr) > 0:
            print(i)
            print("-------------------------------")
            print(cr.sort_values(ascending=False))
            print("\n")
```

In [101…

```
datasets["application_train"]
```

Out[101]:

	SK_ID_CURR	TARGET	NAME_CONTRACT_TYPE	CODE_GENDER	FLAG_OWN_CAR	FI
0	100002	1	Cash loans	M	N	
1	100003	0	Cash loans	F	N	
2	100004	0	Revolving loans	M	Y	
3	100006	0	Cash loans	F	N	
4	100007	0	Cash loans	M	N	
...	...	...	...	...	...	
307506	456251	0	Cash loans	M	N	
307507	456252	0	Cash loans	F	N	
307508	456253	0	Cash loans	F	N	
307509	456254	1	Cash loans	F	N	
307510	456255	0	Cash loans	F	N	

307511 rows × 122 columns

Below we are determining the highest correlations for each table.

In [102…

```
for i,ds_name in enumerate(datasets.keys()):
    print("-----------------------------------------------------------------
    print("Table under consideration FOR HIGHEST CORRELATIONS:",ds_name.uppe
    high_correlation(datasets[ds_name], remove=['SK_ID_CURR','SK_ID_BUREAU'
                                                 'FLAG_DOCUMENT_12','FLAG_DO
                                                'FLAG_DOCUMENT_15','FLAG_DOCU
                                                'FLAG_DOCUMENT_18','FLAG_DOCU
                                                'FLAG_DOCUMENT_21','PREV_REJ_
    print("-----------------------------------------------------------------
    print("-----------------------------------------------------------------
```

```
--------------------------------------------------------------------------
Table under consideration FOR HIGHEST CORRELATIONS: APPLICATION_TRAIN
CNT_CHILDREN
-------------------------------
CNT_FAM_MEMBERS    0.811559
Name: CNT_CHILDREN, dtype: float64


AMT_CREDIT
-------------------------------
AMT_GOODS_PRICE    0.984898
AMT_ANNUITY        0.830225
Name: AMT_CREDIT, dtype: float64


AMT_ANNUITY
-------------------------------
AMT_CREDIT         0.830225
AMT_GOODS_PRICE    0.828034
Name: AMT_ANNUITY, dtype: float64


AMT_GOODS_PRICE
-------------------------------
AMT_CREDIT     0.984898
AMT_ANNUITY    0.828034
Name: AMT_GOODS_PRICE, dtype: float64


CNT_FAM_MEMBERS
-------------------------------
CNT_CHILDREN    0.811559
Name: CNT_FAM_MEMBERS, dtype: float64


REGION_RATING_CLIENT
-------------------------------
REGION_RATING_CLIENT_W_CITY    0.950041
Name: REGION_RATING_CLIENT, dtype: float64


REGION_RATING_CLIENT_W_CITY
-------------------------------
REGION_RATING_CLIENT    0.950041
Name: REGION_RATING_CLIENT_W_CITY, dtype: float64


REG_REGION_NOT_WORK_REGION
-------------------------------
LIVE_REGION_NOT_WORK_REGION    0.860627
Name: REG_REGION_NOT_WORK_REGION, dtype: float64


LIVE_REGION_NOT_WORK_REGION
-------------------------------
REG_REGION_NOT_WORK_REGION    0.860627
Name: LIVE_REGION_NOT_WORK_REGION, dtype: float64
```

```
REG_CITY_NOT_WORK_CITY
-------------------------------
LIVE_CITY_NOT_WORK_CITY    0.825575
Name: REG_CITY_NOT_WORK_CITY, dtype: float64


LIVE_CITY_NOT_WORK_CITY
-------------------------------
REG_CITY_NOT_WORK_CITY    0.825575
Name: LIVE_CITY_NOT_WORK_CITY, dtype: float64


APARTMENTS_AVG
-------------------------------
APARTMENTS_MEDI          0.995231
LIVINGAPARTMENTS_AVG     0.973052
LIVINGAPARTMENTS_MEDI    0.969330
APARTMENTS_MODE          0.964837
LIVINGAPARTMENTS_MODE    0.943403
LIVINGAREA_AVG           0.906021
LIVINGAREA_MEDI          0.902152
TOTALAREA_MODE           0.899922
LIVINGAREA_MODE          0.873567
FLOORSMAX_AVG            0.756743
FLOORSMAX_MEDI           0.755532
FLOORSMAX_MODE           0.751695
Name: APARTMENTS_AVG, dtype: float64


BASEMENTAREA_AVG
-------------------------------
BASEMENTAREA_MEDI    0.994929
BASEMENTAREA_MODE    0.959931
LIVINGAREA_AVG       0.719254
LIVINGAREA_MEDI      0.716333
TOTALAREA_MODE       0.711465
Name: BASEMENTAREA_AVG, dtype: float64


YEARS_BEGINEXPLUATATION_AVG
-------------------------------
YEARS_BEGINEXPLUATATION_MEDI    0.997432
YEARS_BUILD_AVG                 0.987100
YEARS_BUILD_MEDI                0.985658
YEARS_BEGINEXPLUATATION_MODE    0.985231
YEARS_BUILD_MODE                0.975302
Name: YEARS_BEGINEXPLUATATION_AVG, dtype: float64


YEARS_BUILD_AVG
-------------------------------
YEARS_BUILD_MEDI                0.998411
YEARS_BUILD_MODE                0.987791
YEARS_BEGINEXPLUATATION_AVG     0.987100
YEARS_BEGINEXPLUATATION_MEDI    0.984721
YEARS_BEGINEXPLUATATION_MODE    0.969280
Name: YEARS_BUILD_AVG, dtype: float64
```

```
COMMONAREA_AVG
-------------------------------
COMMONAREA_MEDI    0.995734
COMMONAREA_MODE    0.965539
Name: COMMONAREA_AVG, dtype: float64


ELEVATORS_AVG
-------------------------------
ELEVATORS_MEDI     0.991028
ELEVATORS_MODE     0.968827
FLOORSMAX_AVG      0.849527
FLOORSMAX_MEDI     0.842046
FLOORSMAX_MODE     0.822010
LIVINGAREA_AVG     0.728138
LIVINGAREA_MEDI    0.718854
TOTALAREA_MODE     0.717516
Name: ELEVATORS_AVG, dtype: float64


ENTRANCES_AVG
-------------------------------
ENTRANCES_MEDI    0.993427
ENTRANCES_MODE    0.957861
Name: ENTRANCES_AVG, dtype: float64


FLOORSMAX_AVG
-------------------------------
FLOORSMAX_MEDI           0.994876
FLOORSMAX_MODE           0.981639
ELEVATORS_AVG            0.849527
ELEVATORS_MEDI           0.844572
ELEVATORS_MODE           0.829461
LIVINGAREA_AVG           0.783331
TOTALAREA_MODE           0.778106
LIVINGAREA_MEDI          0.777966
APARTMENTS_AVG           0.756743
APARTMENTS_MEDI          0.750783
LIVINGAREA_MODE          0.744586
LIVINGAPARTMENTS_AVG     0.725439
LIVINGAPARTMENTS_MEDI    0.720643
APARTMENTS_MODE          0.717633
Name: FLOORSMAX_AVG, dtype: float64


FLOORSMIN_AVG
-------------------------------
FLOORSMIN_MEDI    0.996223
FLOORSMIN_MODE    0.982649
Name: FLOORSMIN_AVG, dtype: float64


LANDAREA_AVG
-------------------------------
LANDAREA_MEDI    0.996279
LANDAREA_MODE    0.967293
Name: LANDAREA_AVG, dtype: float64
```

```
LIVINGAPARTMENTS_AVG
-------------------------------
LIVINGAPARTMENTS_MEDI    0.996199
APARTMENTS_AVG           0.973052
LIVINGAPARTMENTS_MODE    0.969746
APARTMENTS_MEDI          0.967137
APARTMENTS_MODE          0.928390
LIVINGAREA_AVG           0.896052
LIVINGAREA_MEDI          0.890798
TOTALAREA_MODE           0.875265
LIVINGAREA_MODE          0.854260
FLOORSMAX_AVG            0.725439
FLOORSMAX_MEDI           0.723179
FLOORSMAX_MODE           0.717750
Name: LIVINGAPARTMENTS_AVG, dtype: float64


LIVINGAREA_AVG
-------------------------------
LIVINGAREA_MEDI          0.995506
LIVINGAREA_MODE          0.964163
TOTALAREA_MODE           0.939516
APARTMENTS_AVG           0.906021
APARTMENTS_MEDI          0.901134
LIVINGAPARTMENTS_AVG     0.896052
LIVINGAPARTMENTS_MEDI    0.892313
APARTMENTS_MODE          0.870116
LIVINGAPARTMENTS_MODE    0.866025
FLOORSMAX_AVG            0.783331
FLOORSMAX_MEDI           0.782036
FLOORSMAX_MODE           0.776620
ELEVATORS_AVG            0.728138
ELEVATORS_MEDI           0.723821
BASEMENTAREA_AVG         0.719254
BASEMENTAREA_MEDI        0.713141
ELEVATORS_MODE           0.709480
Name: LIVINGAREA_AVG, dtype: float64


NONLIVINGAPARTMENTS_AVG
-------------------------------
NONLIVINGAPARTMENTS_MEDI    0.981857
NONLIVINGAPARTMENTS_MODE    0.932255
Name: NONLIVINGAPARTMENTS_AVG, dtype: float64


NONLIVINGAREA_AVG
-------------------------------
NONLIVINGAREA_MEDI    0.980959
NONLIVINGAREA_MODE    0.918531
Name: NONLIVINGAREA_AVG, dtype: float64


APARTMENTS_MODE
-------------------------------
APARTMENTS_MEDI          0.970022
APARTMENTS_AVG           0.964837
LIVINGAPARTMENTS_MODE    0.954728
```

```
LIVINGAPARTMENTS_MEDI    0.932233
LIVINGAPARTMENTS_AVG     0.928390
LIVINGAREA_MODE          0.893166
TOTALAREA_MODE           0.876214
LIVINGAREA_MEDI          0.874638
LIVINGAREA_AVG           0.870116
FLOORSMAX_MODE           0.735464
FLOORSMAX_MEDI           0.720331
FLOORSMAX_AVG            0.717633
Name: APARTMENTS_MODE, dtype: float64


BASEMENTAREA_MODE
-------------------------------
BASEMENTAREA_MEDI    0.963427
BASEMENTAREA_AVG     0.959931
LIVINGAREA_MODE      0.703070
Name: BASEMENTAREA_MODE, dtype: float64


YEARS_BEGINEXPLUATATION_MODE
-------------------------------
YEARS_BEGINEXPLUATATION_MEDI    0.985670
YEARS_BEGINEXPLUATATION_AVG     0.985231
YEARS_BUILD_MODE                0.980396
YEARS_BUILD_MEDI                0.969592
YEARS_BUILD_AVG                 0.969280
Name: YEARS_BEGINEXPLUATATION_MODE, dtype: float64


YEARS_BUILD_MODE
-------------------------------
YEARS_BUILD_MEDI                0.988070
YEARS_BUILD_AVG                 0.987791
YEARS_BEGINEXPLUATATION_MODE    0.980396
YEARS_BEGINEXPLUATATION_AVG     0.975302
YEARS_BEGINEXPLUATATION_MEDI    0.975063
Name: YEARS_BUILD_MODE, dtype: float64


COMMONAREA_MODE
-------------------------------
COMMONAREA_MEDI    0.970918
COMMONAREA_AVG     0.965539
Name: COMMONAREA_MODE, dtype: float64


ELEVATORS_MODE
-------------------------------
ELEVATORS_MEDI     0.977774
ELEVATORS_AVG      0.968827
FLOORSMAX_MODE     0.833087
FLOORSMAX_MEDI     0.830596
FLOORSMAX_AVG      0.829461
LIVINGAREA_AVG     0.709480
LIVINGAREA_MEDI    0.708099
TOTALAREA_MODE     0.706147
Name: ELEVATORS_MODE, dtype: float64
```

```
ENTRANCES_MODE
-------------------------------
ENTRANCES_MEDI    0.965184
ENTRANCES_AVG     0.957861
Name: ENTRANCES_MODE, dtype: float64


FLOORSMAX_MODE
-------------------------------
FLOORSMAX_MEDI           0.986487
FLOORSMAX_AVG            0.981639
ELEVATORS_MODE           0.833087
ELEVATORS_MEDI           0.826030
ELEVATORS_AVG            0.822010
LIVINGAREA_AVG           0.776620
TOTALAREA_MODE           0.775891
LIVINGAREA_MEDI          0.775173
LIVINGAREA_MODE          0.760952
APARTMENTS_AVG           0.751695
APARTMENTS_MEDI          0.749337
APARTMENTS_MODE          0.735464
LIVINGAPARTMENTS_AVG     0.717750
LIVINGAPARTMENTS_MEDI    0.716091
LIVINGAPARTMENTS_MODE    0.701954
Name: FLOORSMAX_MODE, dtype: float64


FLOORSMIN_MODE
-------------------------------
FLOORSMIN_MEDI    0.986356
FLOORSMIN_AVG     0.982649
Name: FLOORSMIN_MODE, dtype: float64


LANDAREA_MODE
-------------------------------
LANDAREA_MEDI    0.970821
LANDAREA_AVG     0.967293
Name: LANDAREA_MODE, dtype: float64


LIVINGAPARTMENTS_MODE
-------------------------------
LIVINGAPARTMENTS_MEDI    0.973672
LIVINGAPARTMENTS_AVG     0.969746
APARTMENTS_MODE          0.954728
APARTMENTS_MEDI          0.946574
APARTMENTS_AVG           0.943403
LIVINGAREA_MODE          0.873984
LIVINGAREA_MEDI          0.868653
LIVINGAREA_AVG           0.866025
TOTALAREA_MODE           0.852984
FLOORSMAX_MODE           0.701954
Name: LIVINGAPARTMENTS_MODE, dtype: float64


LIVINGAREA_MODE
-------------------------------
```

```
LIVINGAREA_MEDI          0.969152
LIVINGAREA_AVG           0.964163
TOTALAREA_MODE           0.917478
APARTMENTS_MODE          0.893166
APARTMENTS_MEDI          0.877614
LIVINGAPARTMENTS_MODE    0.873984
APARTMENTS_AVG           0.873567
LIVINGAPARTMENTS_MEDI    0.857547
LIVINGAPARTMENTS_AVG     0.854260
FLOORSMAX_MODE           0.760952
FLOORSMAX_MEDI           0.747649
FLOORSMAX_AVG            0.744586
BASEMENTAREA_MODE        0.703070
Name: LIVINGAREA_MODE, dtype: float64


NONLIVINGAPARTMENTS_MODE
-------------------------------
NONLIVINGAPARTMENTS_MEDI    0.951783
NONLIVINGAPARTMENTS_AVG     0.932255
Name: NONLIVINGAPARTMENTS_MODE, dtype: float64


NONLIVINGAREA_MODE
-------------------------------
NONLIVINGAREA_MEDI    0.941964
NONLIVINGAREA_AVG     0.918531
Name: NONLIVINGAREA_MODE, dtype: float64


APARTMENTS_MEDI
-------------------------------
APARTMENTS_AVG           0.995231
LIVINGAPARTMENTS_MEDI    0.970286
APARTMENTS_MODE          0.970022
LIVINGAPARTMENTS_AVG     0.967137
LIVINGAPARTMENTS_MODE    0.946574
LIVINGAREA_MEDI          0.904186
LIVINGAREA_AVG           0.901134
TOTALAREA_MODE           0.894649
LIVINGAREA_MODE          0.877614
FLOORSMAX_MEDI           0.752624
FLOORSMAX_AVG            0.750783
FLOORSMAX_MODE           0.749337
Name: APARTMENTS_MEDI, dtype: float64


BASEMENTAREA_MEDI
-------------------------------
BASEMENTAREA_AVG     0.994929
BASEMENTAREA_MODE    0.963427
LIVINGAREA_MEDI      0.714951
LIVINGAREA_AVG       0.713141
TOTALAREA_MODE       0.704938
Name: BASEMENTAREA_MEDI, dtype: float64


YEARS_BEGINEXPLUATATION_MEDI
-------------------------------
```

```
YEARS_BEGINEXPLUATATION_AVG     0.997432
YEARS_BUILD_MEDI                0.986140
YEARS_BEGINEXPLUATATION_MODE    0.985670
YEARS_BUILD_AVG                 0.984721
YEARS_BUILD_MODE                0.975063
Name: YEARS_BEGINEXPLUATATION_MEDI, dtype: float64


YEARS_BUILD_MEDI
-------------------------------
YEARS_BUILD_AVG                 0.998411
YEARS_BUILD_MODE                0.988070
YEARS_BEGINEXPLUATATION_MEDI    0.986140
YEARS_BEGINEXPLUATATION_AVG     0.985658
YEARS_BEGINEXPLUATATION_MODE    0.969592
Name: YEARS_BUILD_MEDI, dtype: float64


COMMONAREA_MEDI
-------------------------------
COMMONAREA_AVG     0.995734
COMMONAREA_MODE    0.970918
Name: COMMONAREA_MEDI, dtype: float64


ELEVATORS_MEDI
-------------------------------
ELEVATORS_AVG      0.991028
ELEVATORS_MODE     0.977774
FLOORSMAX_MEDI     0.845726
FLOORSMAX_AVG      0.844572
FLOORSMAX_MODE     0.826030
LIVINGAREA_AVG     0.723821
LIVINGAREA_MEDI    0.720717
TOTALAREA_MODE     0.714608
Name: ELEVATORS_MEDI, dtype: float64


ENTRANCES_MEDI
-------------------------------
ENTRANCES_AVG     0.993427
ENTRANCES_MODE    0.965184
Name: ENTRANCES_MEDI, dtype: float64


FLOORSMAX_MEDI
-------------------------------
FLOORSMAX_AVG            0.994876
FLOORSMAX_MODE           0.986487
ELEVATORS_MEDI           0.845726
ELEVATORS_AVG            0.842046
ELEVATORS_MODE           0.830596
LIVINGAREA_AVG           0.782036
LIVINGAREA_MEDI          0.779970
TOTALAREA_MODE           0.777272
APARTMENTS_AVG           0.755532
APARTMENTS_MEDI          0.752624
LIVINGAREA_MODE          0.747649
LIVINGAPARTMENTS_AVG     0.723179
```

```
LIVINGAPARTMENTS_MEDI    0.721035
APARTMENTS_MODE          0.720331
Name: FLOORSMAX_MEDI, dtype: float64


FLOORSMIN_MEDI
-------------------------------
FLOORSMIN_AVG     0.996223
FLOORSMIN_MODE    0.986356
Name: FLOORSMIN_MEDI, dtype: float64


LANDAREA_MEDI
-------------------------------
LANDAREA_AVG     0.996279
LANDAREA_MODE    0.970821
Name: LANDAREA_MEDI, dtype: float64


LIVINGAPARTMENTS_MEDI
-------------------------------
LIVINGAPARTMENTS_AVG     0.996199
LIVINGAPARTMENTS_MODE    0.973672
APARTMENTS_MEDI          0.970286
APARTMENTS_AVG           0.969330
APARTMENTS_MODE          0.932233
LIVINGAREA_MEDI          0.892645
LIVINGAREA_AVG           0.892313
TOTALAREA_MODE           0.871263
LIVINGAREA_MODE          0.857547
FLOORSMAX_MEDI           0.721035
FLOORSMAX_AVG            0.720643
FLOORSMAX_MODE           0.716091
Name: LIVINGAPARTMENTS_MEDI, dtype: float64


LIVINGAREA_MEDI
-------------------------------
LIVINGAREA_AVG           0.995506
LIVINGAREA_MODE          0.969152
TOTALAREA_MODE           0.934818
APARTMENTS_MEDI          0.904186
APARTMENTS_AVG           0.902152
LIVINGAPARTMENTS_MEDI    0.892645
LIVINGAPARTMENTS_AVG     0.890798
APARTMENTS_MODE          0.874638
LIVINGAPARTMENTS_MODE    0.868653
FLOORSMAX_MEDI           0.779970
FLOORSMAX_AVG            0.777966
FLOORSMAX_MODE           0.775173
ELEVATORS_MEDI           0.720717
ELEVATORS_AVG            0.718854
BASEMENTAREA_AVG         0.716333
BASEMENTAREA_MEDI        0.714951
ELEVATORS_MODE           0.708099
Name: LIVINGAREA_MEDI, dtype: float64


NONLIVINGAPARTMENTS_MEDI
```

```
--------------------------------
NONLIVINGAPARTMENTS_AVG     0.981857
NONLIVINGAPARTMENTS_MODE    0.951783
Name: NONLIVINGAPARTMENTS_MEDI, dtype: float64


NONLIVINGAREA_MEDI
--------------------------------
NONLIVINGAREA_AVG     0.980959
NONLIVINGAREA_MODE    0.941964
Name: NONLIVINGAREA_MEDI, dtype: float64


TOTALAREA_MODE
--------------------------------
LIVINGAREA_AVG           0.939516
LIVINGAREA_MEDI          0.934818
LIVINGAREA_MODE          0.917478
APARTMENTS_AVG           0.899922
APARTMENTS_MEDI          0.894649
APARTMENTS_MODE          0.876214
LIVINGAPARTMENTS_AVG     0.875265
LIVINGAPARTMENTS_MEDI    0.871263
LIVINGAPARTMENTS_MODE    0.852984
FLOORSMAX_AVG            0.778106
FLOORSMAX_MEDI           0.777272
FLOORSMAX_MODE           0.775891
ELEVATORS_AVG            0.717516
ELEVATORS_MEDI           0.714608
BASEMENTAREA_AVG         0.711465
ELEVATORS_MODE           0.706147
BASEMENTAREA_MEDI        0.704938
Name: TOTALAREA_MODE, dtype: float64


OBS_30_CNT_SOCIAL_CIRCLE
--------------------------------
OBS_60_CNT_SOCIAL_CIRCLE    0.997336
Name: OBS_30_CNT_SOCIAL_CIRCLE, dtype: float64


DEF_30_CNT_SOCIAL_CIRCLE
--------------------------------
DEF_60_CNT_SOCIAL_CIRCLE    0.845118
Name: DEF_30_CNT_SOCIAL_CIRCLE, dtype: float64


OBS_60_CNT_SOCIAL_CIRCLE
--------------------------------
OBS_30_CNT_SOCIAL_CIRCLE    0.997336
Name: OBS_60_CNT_SOCIAL_CIRCLE, dtype: float64


DEF_60_CNT_SOCIAL_CIRCLE
--------------------------------
DEF_30_CNT_SOCIAL_CIRCLE    0.845118
Name: DEF_60_CNT_SOCIAL_CIRCLE, dtype: float64
```

```
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
Table under consideration FOR HIGHEST CORRELATIONS: APPLICATION_TEST
CNT_CHILDREN
-------------------------------
CNT_FAM_MEMBERS    0.807375
Name: CNT_CHILDREN, dtype: float64


AMT_CREDIT
-------------------------------
AMT_GOODS_PRICE    0.984098
AMT_ANNUITY        0.835129
Name: AMT_CREDIT, dtype: float64


AMT_ANNUITY
-------------------------------
AMT_GOODS_PRICE    0.842940
AMT_CREDIT         0.835129
Name: AMT_ANNUITY, dtype: float64


AMT_GOODS_PRICE
-------------------------------
AMT_CREDIT     0.984098
AMT_ANNUITY    0.842940
Name: AMT_GOODS_PRICE, dtype: float64


CNT_FAM_MEMBERS
-------------------------------
CNT_CHILDREN    0.807375
Name: CNT_FAM_MEMBERS, dtype: float64


REGION_RATING_CLIENT
-------------------------------
REGION_RATING_CLIENT_W_CITY    0.941908
Name: REGION_RATING_CLIENT, dtype: float64


REGION_RATING_CLIENT_W_CITY
-------------------------------
REGION_RATING_CLIENT    0.941908
Name: REGION_RATING_CLIENT_W_CITY, dtype: float64


REG_REGION_NOT_WORK_REGION
-------------------------------
LIVE_REGION_NOT_WORK_REGION    0.839604
Name: REG_REGION_NOT_WORK_REGION, dtype: float64


LIVE_REGION_NOT_WORK_REGION
-------------------------------
REG_REGION_NOT_WORK_REGION    0.839604
Name: LIVE_REGION_NOT_WORK_REGION, dtype: float64
```

```
REG_CITY_NOT_WORK_CITY
--------------------------------
LIVE_CITY_NOT_WORK_CITY    0.826969
Name: REG_CITY_NOT_WORK_CITY, dtype: float64


LIVE_CITY_NOT_WORK_CITY
--------------------------------
REG_CITY_NOT_WORK_CITY    0.826969
Name: LIVE_CITY_NOT_WORK_CITY, dtype: float64


APARTMENTS_AVG
--------------------------------
APARTMENTS_MEDI          0.994632
LIVINGAPARTMENTS_AVG     0.973463
LIVINGAPARTMENTS_MEDI    0.968980
APARTMENTS_MODE          0.961944
LIVINGAPARTMENTS_MODE    0.940507
LIVINGAREA_AVG           0.904505
LIVINGAREA_MEDI          0.900305
TOTALAREA_MODE           0.894386
LIVINGAREA_MODE          0.869058
FLOORSMAX_AVG            0.748481
FLOORSMAX_MEDI           0.747302
FLOORSMAX_MODE           0.742223
ELEVATORS_AVG            0.714303
ELEVATORS_MEDI           0.709995
Name: APARTMENTS_AVG, dtype: float64


BASEMENTAREA_AVG
--------------------------------
BASEMENTAREA_MEDI    0.994877
BASEMENTAREA_MODE    0.959769
LIVINGAREA_AVG       0.710668
LIVINGAREA_MEDI      0.706965
TOTALAREA_MODE       0.705886
Name: BASEMENTAREA_AVG, dtype: float64


YEARS_BEGINEXPLUATATION_AVG
--------------------------------
YEARS_BEGINEXPLUATATION_MEDI    0.997355
YEARS_BUILD_AVG                 0.987155
YEARS_BEGINEXPLUATATION_MODE    0.985585
YEARS_BUILD_MEDI                0.985565
YEARS_BUILD_MODE                0.976124
Name: YEARS_BEGINEXPLUATATION_AVG, dtype: float64


YEARS_BUILD_AVG
--------------------------------
YEARS_BUILD_MEDI                0.998322
YEARS_BUILD_MODE                0.988460
YEARS_BEGINEXPLUATATION_AVG     0.987155
YEARS_BEGINEXPLUATATION_MEDI    0.984721
```

```
YEARS_BEGINEXPLUATATION_MODE    0.968456
Name: YEARS_BUILD_AVG, dtype: float64


COMMONAREA_AVG
-------------------------------
COMMONAREA_MEDI    0.994019
COMMONAREA_MODE    0.962452
Name: COMMONAREA_AVG, dtype: float64


ELEVATORS_AVG
-------------------------------
ELEVATORS_MEDI           0.989661
ELEVATORS_MODE           0.967564
FLOORSMAX_AVG            0.855026
FLOORSMAX_MEDI           0.846966
FLOORSMAX_MODE           0.826127
LIVINGAREA_AVG           0.744731
LIVINGAREA_MEDI          0.735083
TOTALAREA_MODE           0.730455
APARTMENTS_AVG           0.714303
LIVINGAPARTMENTS_AVG     0.712193
APARTMENTS_MEDI          0.703754
LIVINGAPARTMENTS_MEDI    0.703036
Name: ELEVATORS_AVG, dtype: float64


ENTRANCES_AVG
-------------------------------
ENTRANCES_MEDI    0.992810
ENTRANCES_MODE    0.957458
Name: ENTRANCES_AVG, dtype: float64


FLOORSMAX_AVG
-------------------------------
FLOORSMAX_MEDI           0.994176
FLOORSMAX_MODE           0.979453
ELEVATORS_AVG            0.855026
ELEVATORS_MEDI           0.849530
ELEVATORS_MODE           0.835080
LIVINGAREA_AVG           0.775683
TOTALAREA_MODE           0.771127
LIVINGAREA_MEDI          0.770812
APARTMENTS_AVG           0.748481
APARTMENTS_MEDI          0.742081
LIVINGAREA_MODE          0.733726
LIVINGAPARTMENTS_AVG     0.718340
LIVINGAPARTMENTS_MEDI    0.712238
APARTMENTS_MODE          0.708192
Name: FLOORSMAX_AVG, dtype: float64


FLOORSMIN_AVG
-------------------------------
FLOORSMIN_MEDI    0.995686
FLOORSMIN_MODE    0.979987
Name: FLOORSMIN_AVG, dtype: float64
```

```
LANDAREA_AVG
-------------------------------
LANDAREA_MEDI    0.996372
LANDAREA_MODE    0.968833
Name: LANDAREA_AVG, dtype: float64


LIVINGAPARTMENTS_AVG
-------------------------------
LIVINGAPARTMENTS_MEDI    0.995205
APARTMENTS_AVG           0.973463
LIVINGAPARTMENTS_MODE    0.965974
APARTMENTS_MEDI          0.965672
APARTMENTS_MODE          0.923313
LIVINGAREA_AVG           0.895941
LIVINGAREA_MEDI          0.889038
TOTALAREA_MODE           0.872361
LIVINGAREA_MODE          0.849039
FLOORSMAX_AVG            0.718340
FLOORSMAX_MEDI           0.716529
ELEVATORS_AVG            0.712193
FLOORSMAX_MODE           0.710573
ELEVATORS_MEDI           0.706874
Name: LIVINGAPARTMENTS_AVG, dtype: float64


LIVINGAREA_AVG
-------------------------------
LIVINGAREA_MEDI          0.994913
LIVINGAREA_MODE          0.961424
TOTALAREA_MODE           0.937703
APARTMENTS_AVG           0.904505
APARTMENTS_MEDI          0.898458
LIVINGAPARTMENTS_AVG     0.895941
LIVINGAPARTMENTS_MEDI    0.890591
APARTMENTS_MODE          0.867221
LIVINGAPARTMENTS_MODE    0.862619
FLOORSMAX_AVG            0.775683
FLOORSMAX_MEDI           0.773934
FLOORSMAX_MODE           0.768277
ELEVATORS_AVG            0.744731
ELEVATORS_MEDI           0.739427
ELEVATORS_MODE           0.725688
BASEMENTAREA_AVG         0.710668
BASEMENTAREA_MEDI        0.704951
Name: LIVINGAREA_AVG, dtype: float64


NONLIVINGAPARTMENTS_AVG
-------------------------------
NONLIVINGAPARTMENTS_MEDI    0.978331
NONLIVINGAPARTMENTS_MODE    0.928955
Name: NONLIVINGAPARTMENTS_AVG, dtype: float64


NONLIVINGAREA_AVG
-------------------------------
```

```
NONLIVINGAREA_MEDI    0.979568
NONLIVINGAREA_MODE    0.916613
Name: NONLIVINGAREA_AVG, dtype: float64


APARTMENTS_MODE
-------------------------------
APARTMENTS_MEDI          0.968119
APARTMENTS_AVG           0.961944
LIVINGAPARTMENTS_MODE    0.950547
LIVINGAPARTMENTS_MEDI    0.929219
LIVINGAPARTMENTS_AVG     0.923313
LIVINGAREA_MODE          0.891958
LIVINGAREA_MEDI          0.873242
TOTALAREA_MODE           0.867664
LIVINGAREA_AVG           0.867221
FLOORSMAX_MODE           0.727099
FLOORSMAX_MEDI           0.711602
FLOORSMAX_AVG            0.708192
Name: APARTMENTS_MODE, dtype: float64


BASEMENTAREA_MODE
--------------------------------
BASEMENTAREA_MEDI    0.963024
BASEMENTAREA_AVG     0.959769
Name: BASEMENTAREA_MODE, dtype: float64


YEARS_BEGINEXPLUATATION_MODE
--------------------------------
YEARS_BEGINEXPLUATATION_MEDI    0.986807
YEARS_BEGINEXPLUATATION_AVG     0.985585
YEARS_BUILD_MODE                0.978744
YEARS_BUILD_MEDI                0.968909
YEARS_BUILD_AVG                 0.968456
Name: YEARS_BEGINEXPLUATATION_MODE, dtype: float64


YEARS_BUILD_MODE
--------------------------------
YEARS_BUILD_MEDI                0.988910
YEARS_BUILD_AVG                 0.988460
YEARS_BEGINEXPLUATATION_MODE    0.978744
YEARS_BEGINEXPLUATATION_MEDI    0.976287
YEARS_BEGINEXPLUATATION_AVG     0.976124
Name: YEARS_BUILD_MODE, dtype: float64


COMMONAREA_MODE
--------------------------------
COMMONAREA_MEDI    0.970991
COMMONAREA_AVG     0.962452
Name: COMMONAREA_MODE, dtype: float64


ELEVATORS_MODE
--------------------------------
ELEVATORS_MEDI     0.977952
```

```
ELEVATORS_AVG      0.967564
FLOORSMAX_MODE     0.838788
FLOORSMAX_MEDI     0.836448
FLOORSMAX_AVG      0.835080
LIVINGAREA_MEDI    0.725975
LIVINGAREA_AVG     0.725688
TOTALAREA_MODE     0.716136
LIVINGAREA_MODE    0.704161
Name: ELEVATORS_MODE, dtype: float64


ENTRANCES_MODE
-------------------------------
ENTRANCES_MEDI    0.965316
ENTRANCES_AVG     0.957458
Name: ENTRANCES_MODE, dtype: float64


FLOORSMAX_MODE
-------------------------------
FLOORSMAX_MEDI           0.984829
FLOORSMAX_AVG            0.979453
ELEVATORS_MODE           0.838788
ELEVATORS_MEDI           0.831342
ELEVATORS_AVG            0.826127
TOTALAREA_MODE           0.768698
LIVINGAREA_AVG           0.768277
LIVINGAREA_MEDI          0.768071
LIVINGAREA_MODE          0.752385
APARTMENTS_AVG           0.742223
APARTMENTS_MEDI          0.739938
APARTMENTS_MODE          0.727099
LIVINGAPARTMENTS_AVG     0.710573
LIVINGAPARTMENTS_MEDI    0.708846
Name: FLOORSMAX_MODE, dtype: float64


FLOORSMIN_MODE
-------------------------------
FLOORSMIN_MEDI    0.983774
FLOORSMIN_AVG     0.979987
Name: FLOORSMIN_MODE, dtype: float64


LANDAREA_MODE
-------------------------------
LANDAREA_MEDI    0.972139
LANDAREA_AVG     0.968833
Name: LANDAREA_MODE, dtype: float64


LIVINGAPARTMENTS_MODE
-------------------------------
LIVINGAPARTMENTS_MEDI    0.972144
LIVINGAPARTMENTS_AVG     0.965974
APARTMENTS_MODE          0.950547
APARTMENTS_MEDI          0.944644
APARTMENTS_AVG           0.940507
LIVINGAREA_MODE          0.869175
```

```
LIVINGAREA_MEDI          0.866720
LIVINGAREA_AVG           0.862619
TOTALAREA_MODE           0.845449
Name: LIVINGAPARTMENTS_MODE, dtype: float64


LIVINGAREA_MODE
-------------------------------
LIVINGAREA_MEDI          0.967546
LIVINGAREA_AVG           0.961424
TOTALAREA_MODE           0.914948
APARTMENTS_MODE          0.891958
APARTMENTS_MEDI          0.873796
LIVINGAPARTMENTS_MODE    0.869175
APARTMENTS_AVG           0.869058
LIVINGAPARTMENTS_MEDI    0.853986
LIVINGAPARTMENTS_AVG     0.849039
FLOORSMAX_MODE           0.752385
FLOORSMAX_MEDI           0.737918
FLOORSMAX_AVG            0.733726
ELEVATORS_MODE           0.704161
Name: LIVINGAREA_MODE, dtype: float64


NONLIVINGAPARTMENTS_MODE
-------------------------------
NONLIVINGAPARTMENTS_MEDI    0.952847
NONLIVINGAPARTMENTS_AVG     0.928955
Name: NONLIVINGAPARTMENTS_MODE, dtype: float64


NONLIVINGAREA_MODE
-------------------------------
NONLIVINGAREA_MEDI    0.942506
NONLIVINGAREA_AVG     0.916613
Name: NONLIVINGAREA_MODE, dtype: float64


APARTMENTS_MEDI
-------------------------------
APARTMENTS_AVG           0.994632
LIVINGAPARTMENTS_MEDI    0.969877
APARTMENTS_MODE          0.968119
LIVINGAPARTMENTS_AVG     0.965672
LIVINGAPARTMENTS_MODE    0.944644
LIVINGAREA_MEDI          0.902426
LIVINGAREA_AVG           0.898458
TOTALAREA_MODE           0.888217
LIVINGAREA_MODE          0.873796
FLOORSMAX_MEDI           0.744244
FLOORSMAX_AVG            0.742081
FLOORSMAX_MODE           0.739938
ELEVATORS_MEDI           0.707036
ELEVATORS_AVG            0.703754
Name: APARTMENTS_MEDI, dtype: float64


BASEMENTAREA_MEDI
-------------------------------
```

```
BASEMENTAREA_AVG     0.994877
BASEMENTAREA_MODE    0.963024
LIVINGAREA_MEDI      0.706826
LIVINGAREA_AVG       0.704951
Name: BASEMENTAREA_MEDI, dtype: float64


YEARS_BEGINEXPLUATATION_MEDI
-------------------------------
YEARS_BEGINEXPLUATATION_AVG     0.997355
YEARS_BEGINEXPLUATATION_MODE    0.986807
YEARS_BUILD_MEDI                0.986278
YEARS_BUILD_AVG                 0.984721
YEARS_BUILD_MODE                0.976287
Name: YEARS_BEGINEXPLUATATION_MEDI, dtype: float64


YEARS_BUILD_MEDI
-------------------------------
YEARS_BUILD_AVG                 0.998322
YEARS_BUILD_MODE                0.988910
YEARS_BEGINEXPLUATATION_MEDI    0.986278
YEARS_BEGINEXPLUATATION_AVG     0.985565
YEARS_BEGINEXPLUATATION_MODE    0.968909
Name: YEARS_BUILD_MEDI, dtype: float64


COMMONAREA_MEDI
-------------------------------
COMMONAREA_AVG     0.994019
COMMONAREA_MODE    0.970991
Name: COMMONAREA_MEDI, dtype: float64


ELEVATORS_MEDI
-------------------------------
ELEVATORS_AVG            0.989661
ELEVATORS_MODE           0.977952
FLOORSMAX_MEDI           0.851706
FLOORSMAX_AVG            0.849530
FLOORSMAX_MODE           0.831342
LIVINGAREA_AVG           0.739427
LIVINGAREA_MEDI          0.737704
TOTALAREA_MODE           0.726485
APARTMENTS_AVG           0.709995
APARTMENTS_MEDI          0.707036
LIVINGAPARTMENTS_AVG     0.706874
LIVINGAPARTMENTS_MEDI    0.705205
Name: ELEVATORS_MEDI, dtype: float64


ENTRANCES_MEDI
-------------------------------
ENTRANCES_AVG     0.992810
ENTRANCES_MODE    0.965316
Name: ENTRANCES_MEDI, dtype: float64


FLOORSMAX_MEDI
```

```
-------------------------------
FLOORSMAX_AVG            0.994176
FLOORSMAX_MODE           0.984829
ELEVATORS_MEDI           0.851706
ELEVATORS_AVG            0.846966
ELEVATORS_MODE           0.836448
LIVINGAREA_AVG           0.773934
LIVINGAREA_MEDI          0.772938
TOTALAREA_MODE           0.770572
APARTMENTS_AVG           0.747302
APARTMENTS_MEDI          0.744244
LIVINGAREA_MODE          0.737918
LIVINGAPARTMENTS_AVG     0.716529
LIVINGAPARTMENTS_MEDI    0.713618
APARTMENTS_MODE          0.711602
Name: FLOORSMAX_MEDI, dtype: float64


FLOORSMIN_MEDI
-------------------------------
FLOORSMIN_AVG     0.995686
FLOORSMIN_MODE    0.983774
Name: FLOORSMIN_MEDI, dtype: float64


LANDAREA_MEDI
-------------------------------
LANDAREA_AVG     0.996372
LANDAREA_MODE    0.972139
Name: LANDAREA_MEDI, dtype: float64


LIVINGAPARTMENTS_MEDI
-------------------------------
LIVINGAPARTMENTS_AVG     0.995205
LIVINGAPARTMENTS_MODE    0.972144
APARTMENTS_MEDI          0.969877
APARTMENTS_AVG           0.968980
APARTMENTS_MODE          0.929219
LIVINGAREA_MEDI          0.890945
LIVINGAREA_AVG           0.890591
TOTALAREA_MODE           0.866845
LIVINGAREA_MODE          0.853986
FLOORSMAX_MEDI           0.713618
FLOORSMAX_AVG            0.712238
FLOORSMAX_MODE           0.708846
ELEVATORS_MEDI           0.705205
ELEVATORS_AVG            0.703036
Name: LIVINGAPARTMENTS_MEDI, dtype: float64


LIVINGAREA_MEDI
-------------------------------
LIVINGAREA_AVG           0.994913
LIVINGAREA_MODE          0.967546
TOTALAREA_MODE           0.933154
APARTMENTS_MEDI          0.902426
APARTMENTS_AVG           0.900305
LIVINGAPARTMENTS_MEDI    0.890945
```

```
LIVINGAPARTMENTS_AVG     0.889038
APARTMENTS_MODE          0.873242
LIVINGAPARTMENTS_MODE    0.866720
FLOORSMAX_MEDI           0.772938
FLOORSMAX_AVG            0.770812
FLOORSMAX_MODE           0.768071
ELEVATORS_MEDI           0.737704
ELEVATORS_AVG            0.735083
ELEVATORS_MODE           0.725975
BASEMENTAREA_AVG         0.706965
BASEMENTAREA_MEDI        0.706826
Name: LIVINGAREA_MEDI, dtype: float64


NONLIVINGAPARTMENTS_MEDI
--------------------------------
NONLIVINGAPARTMENTS_AVG     0.978331
NONLIVINGAPARTMENTS_MODE    0.952847
Name: NONLIVINGAPARTMENTS_MEDI, dtype: float64


NONLIVINGAREA_MEDI
--------------------------------
NONLIVINGAREA_AVG     0.979568
NONLIVINGAREA_MODE    0.942506
Name: NONLIVINGAREA_MEDI, dtype: float64


TOTALAREA_MODE
--------------------------------
LIVINGAREA_AVG           0.937703
LIVINGAREA_MEDI          0.933154
LIVINGAREA_MODE          0.914948
APARTMENTS_AVG           0.894386
APARTMENTS_MEDI          0.888217
LIVINGAPARTMENTS_AVG     0.872361
APARTMENTS_MODE          0.867664
LIVINGAPARTMENTS_MEDI    0.866845
LIVINGAPARTMENTS_MODE    0.845449
FLOORSMAX_AVG            0.771127
FLOORSMAX_MEDI           0.770572
FLOORSMAX_MODE           0.768698
ELEVATORS_AVG            0.730455
ELEVATORS_MEDI           0.726485
ELEVATORS_MODE           0.716136
BASEMENTAREA_AVG         0.705886
Name: TOTALAREA_MODE, dtype: float64


OBS_30_CNT_SOCIAL_CIRCLE
--------------------------------
OBS_60_CNT_SOCIAL_CIRCLE    0.997634
Name: OBS_30_CNT_SOCIAL_CIRCLE, dtype: float64


DEF_30_CNT_SOCIAL_CIRCLE
--------------------------------
DEF_60_CNT_SOCIAL_CIRCLE    0.851242
Name: DEF_30_CNT_SOCIAL_CIRCLE, dtype: float64
```

```
OBS_60_CNT_SOCIAL_CIRCLE
--------------------------------
OBS_30_CNT_SOCIAL_CIRCLE    0.997634
Name: OBS_60_CNT_SOCIAL_CIRCLE, dtype: float64


DEF_60_CNT_SOCIAL_CIRCLE
--------------------------------
DEF_30_CNT_SOCIAL_CIRCLE    0.851242
Name: DEF_60_CNT_SOCIAL_CIRCLE, dtype: float64


--------------------------------------------------------------------------
--------------------------------------------------------------------------
--------------------------------------------------------------------------
Table under consideration FOR HIGHEST CORRELATIONS: BUREAU
DAYS_CREDIT
--------------------------------
DAYS_ENDDATE_FACT      0.873758
DAYS_CREDIT_UPDATE     0.743959
DAYS_CREDIT_ENDDATE    0.741778
Name: DAYS_CREDIT, dtype: float64


CREDIT_DAY_OVERDUE
--------------------------------
AMT_CREDIT_SUM_OVERDUE    0.933708
Name: CREDIT_DAY_OVERDUE, dtype: float64


DAYS_CREDIT_ENDDATE
--------------------------------
DAYS_ENDDATE_FACT     0.880629
DAYS_CREDIT_UPDATE    0.809760
DAYS_CREDIT           0.741778
Name: DAYS_CREDIT_ENDDATE, dtype: float64


DAYS_ENDDATE_FACT
--------------------------------
DAYS_CREDIT_ENDDATE    0.880629
DAYS_CREDIT            0.873758
DAYS_CREDIT_UPDATE     0.871854
Name: DAYS_ENDDATE_FACT, dtype: float64


AMT_CREDIT_SUM_OVERDUE
--------------------------------
CREDIT_DAY_OVERDUE    0.933708
Name: AMT_CREDIT_SUM_OVERDUE, dtype: float64


DAYS_CREDIT_UPDATE
--------------------------------
DAYS_ENDDATE_FACT      0.871854
DAYS_CREDIT_ENDDATE    0.809760
DAYS_CREDIT            0.743959
```

```
Name: DAYS_CREDIT_UPDATE, dtype: float64


--------------------------------------------------------------------------
--------------------------------------------------------------------------
--------------------------------------------------------------------------
Table under consideration FOR HIGHEST CORRELATIONS: BUREAU_BALANCE
--------------------------------------------------------------------------
--------------------------------------------------------------------------
--------------------------------------------------------------------------
Table under consideration FOR HIGHEST CORRELATIONS: CREDIT_CARD_BALANCE
AMT_BALANCE
--------------------------------
AMT_RECEIVABLE_PRINCIPAL     0.977144
AMT_TOTAL_RECEIVABLE         0.971200
AMT_RECIVABLE                0.971191
AMT_INST_MIN_REGULARITY      0.918747
AMT_PAYMENT_TOTAL_CURRENT    0.799350
Name: AMT_BALANCE, dtype: float64


AMT_DRAWINGS_ATM_CURRENT
--------------------------------
CNT_DRAWINGS_ATM_CURRENT    0.997019
AMT_DRAWINGS_CURRENT        0.808052
CNT_DRAWINGS_CURRENT        0.786588
Name: AMT_DRAWINGS_ATM_CURRENT, dtype: float64


AMT_DRAWINGS_CURRENT
--------------------------------
CNT_DRAWINGS_CURRENT        0.989757
AMT_DRAWINGS_ATM_CURRENT    0.808052
CNT_DRAWINGS_ATM_CURRENT    0.804339
Name: AMT_DRAWINGS_CURRENT, dtype: float64


AMT_DRAWINGS_OTHER_CURRENT
--------------------------------
CNT_DRAWINGS_OTHER_CURRENT    0.981215
Name: AMT_DRAWINGS_OTHER_CURRENT, dtype: float64


AMT_DRAWINGS_POS_CURRENT
--------------------------------
CNT_DRAWINGS_POS_CURRENT    0.998779
Name: AMT_DRAWINGS_POS_CURRENT, dtype: float64


AMT_INST_MIN_REGULARITY
--------------------------------
AMT_BALANCE                  0.918747
AMT_RECEIVABLE_PRINCIPAL     0.902537
AMT_TOTAL_RECEIVABLE         0.886655
AMT_RECIVABLE                0.886644
AMT_PAYMENT_TOTAL_CURRENT    0.852021
AMT_PAYMENT_CURRENT          0.756538
Name: AMT_INST_MIN_REGULARITY, dtype: float64
```

```
AMT_PAYMENT_CURRENT
-------------------------------
AMT_PAYMENT_TOTAL_CURRENT    0.882329
AMT_INST_MIN_REGULARITY      0.756538
Name: AMT_PAYMENT_CURRENT, dtype: float64


AMT_PAYMENT_TOTAL_CURRENT
-------------------------------
AMT_PAYMENT_CURRENT         0.882329
AMT_INST_MIN_REGULARITY     0.852021
AMT_BALANCE                 0.799350
AMT_RECEIVABLE_PRINCIPAL    0.777337
AMT_TOTAL_RECEIVABLE        0.764223
AMT_RECIVABLE               0.764178
Name: AMT_PAYMENT_TOTAL_CURRENT, dtype: float64


AMT_RECEIVABLE_PRINCIPAL
-------------------------------
AMT_BALANCE                  0.977144
AMT_RECIVABLE                0.970944
AMT_TOTAL_RECEIVABLE         0.970912
AMT_INST_MIN_REGULARITY      0.902537
AMT_PAYMENT_TOTAL_CURRENT    0.777337
Name: AMT_RECEIVABLE_PRINCIPAL, dtype: float64


AMT_RECIVABLE
-------------------------------
AMT_TOTAL_RECEIVABLE         0.999965
AMT_BALANCE                  0.971191
AMT_RECEIVABLE_PRINCIPAL     0.970944
AMT_INST_MIN_REGULARITY      0.886644
AMT_PAYMENT_TOTAL_CURRENT    0.764178
Name: AMT_RECIVABLE, dtype: float64


AMT_TOTAL_RECEIVABLE
-------------------------------
AMT_RECIVABLE                0.999965
AMT_BALANCE                  0.971200
AMT_RECEIVABLE_PRINCIPAL     0.970912
AMT_INST_MIN_REGULARITY      0.886655
AMT_PAYMENT_TOTAL_CURRENT    0.764223
Name: AMT_TOTAL_RECEIVABLE, dtype: float64


CNT_DRAWINGS_ATM_CURRENT
-------------------------------
AMT_DRAWINGS_ATM_CURRENT    0.997019
AMT_DRAWINGS_CURRENT        0.804339
CNT_DRAWINGS_CURRENT        0.789030
Name: CNT_DRAWINGS_ATM_CURRENT, dtype: float64


CNT_DRAWINGS_CURRENT
-------------------------------
```

```
AMT_DRAWINGS_CURRENT        0.989757
CNT_DRAWINGS_ATM_CURRENT    0.789030
AMT_DRAWINGS_ATM_CURRENT    0.786588
Name: CNT_DRAWINGS_CURRENT, dtype: float64


CNT_DRAWINGS_OTHER_CURRENT
-------------------------------
AMT_DRAWINGS_OTHER_CURRENT    0.981215
Name: CNT_DRAWINGS_OTHER_CURRENT, dtype: float64


CNT_DRAWINGS_POS_CURRENT
-------------------------------
AMT_DRAWINGS_POS_CURRENT    0.998779
Name: CNT_DRAWINGS_POS_CURRENT, dtype: float64


SK_DPD
-------------------------------
SK_DPD_DEF    0.745372
Name: SK_DPD, dtype: float64


SK_DPD_DEF
-------------------------------
SK_DPD    0.745372
Name: SK_DPD_DEF, dtype: float64


-------------------------------------------------------------------------
-------------------------------------------------------------------------
-------------------------------------------------------------------------
Table under consideration FOR HIGHEST CORRELATIONS: INSTALLMENTS_PAYMENTS
DAYS_INSTALMENT
-------------------------------
DAYS_ENTRY_PAYMENT    0.999402
Name: DAYS_INSTALMENT, dtype: float64


DAYS_ENTRY_PAYMENT
-------------------------------
DAYS_INSTALMENT    0.999402
Name: DAYS_ENTRY_PAYMENT, dtype: float64


AMT_INSTALMENT
-------------------------------
AMT_PAYMENT    0.916463
Name: AMT_INSTALMENT, dtype: float64


AMT_PAYMENT
-------------------------------
AMT_INSTALMENT    0.916463
Name: AMT_PAYMENT, dtype: float64


-------------------------------------------------------------------------
```

```
------------------------------------------------------------------------
------------------------------------------------------------------------
Table under consideration FOR HIGHEST CORRELATIONS: PREVIOUS_APPLICATION
AMT_ANNUITY
-------------------------------
AMT_GOODS_PRICE    0.889194
AMT_CREDIT         0.880038
AMT_APPLICATION    0.829717
Name: AMT_ANNUITY, dtype: float64


AMT_APPLICATION
-------------------------------
AMT_GOODS_PRICE    0.999891
AMT_CREDIT         0.916730
AMT_ANNUITY        0.829717
Name: AMT_APPLICATION, dtype: float64


AMT_CREDIT
-------------------------------
AMT_GOODS_PRICE           0.984859
AMT_APPLICATION           0.916730
AMT_ANNUITY               0.880038
PREV_CRDT_ANNUTY_RATIO    0.706005
Name: AMT_CREDIT, dtype: float64


AMT_DOWN_PAYMENT
-------------------------------
RATE_DOWN_PAYMENT            0.918136
PREV_DWN_PYMNT_CRDT_RATIO    0.918135
PREV_APCTN_CRDT_DIFF         0.812550
DAYS_FIRST_DRAWING           0.785737
PREV_APCTN_CRDT_RATIO        0.771830
Name: AMT_DOWN_PAYMENT, dtype: float64


AMT_GOODS_PRICE
-------------------------------
AMT_APPLICATION    0.999891
AMT_CREDIT         0.984859
AMT_ANNUITY        0.889194
Name: AMT_GOODS_PRICE, dtype: float64


RATE_DOWN_PAYMENT
-------------------------------
PREV_DWN_PYMNT_CRDT_RATIO    1.000000
AMT_DOWN_PAYMENT             0.918136
PREV_APCTN_CRDT_RATIO        0.866425
PREV_APCTN_CRDT_DIFF         0.846313
DAYS_FIRST_DRAWING           0.785269
Name: RATE_DOWN_PAYMENT, dtype: float64


RATE_INTEREST_PRIMARY
-------------------------------
RATE_INTEREST_PRIVILEGED    0.781797
```

```
Name: RATE_INTEREST_PRIMARY, dtype: float64


RATE_INTEREST_PRIVILEGED
-------------------------------
RATE_INTEREST_PRIMARY    0.781797
Name: RATE_INTEREST_PRIVILEGED, dtype: float64


DAYS_DECISION
-------------------------------
DAYS_FIRST_DUE               0.998726
DAYS_FIRST_DRAWING           0.977977
DAYS_LAST_DUE                0.946444
DAYS_TERMINATION             0.939959
DAYS_LAST_DUE_1ST_VERSION    0.832155
Name: DAYS_DECISION, dtype: float64


DAYS_FIRST_DRAWING
-------------------------------
DAYS_FIRST_DUE               0.998407
DAYS_DECISION                0.977977
AMT_DOWN_PAYMENT             0.785737
RATE_DOWN_PAYMENT            0.785269
PREV_DWN_PYMNT_CRDT_RATIO    0.785269
DAYS_LAST_DUE                0.760570
PREV_APCTN_CRDT_RATIO       -0.758784
Name: DAYS_FIRST_DRAWING, dtype: float64


DAYS_FIRST_DUE
-------------------------------
DAYS_DECISION                0.998726
DAYS_FIRST_DRAWING           0.998407
DAYS_LAST_DUE                0.946939
DAYS_TERMINATION             0.941859
DAYS_LAST_DUE_1ST_VERSION    0.836426
Name: DAYS_FIRST_DUE, dtype: float64


DAYS_LAST_DUE_1ST_VERSION
-------------------------------
DAYS_LAST_DUE       0.935636
DAYS_TERMINATION    0.933585
DAYS_FIRST_DUE      0.836426
DAYS_DECISION       0.832155
Name: DAYS_LAST_DUE_1ST_VERSION, dtype: float64


DAYS_LAST_DUE
-------------------------------
DAYS_TERMINATION             0.991670
DAYS_FIRST_DUE               0.946939
DAYS_DECISION                0.946444
DAYS_LAST_DUE_1ST_VERSION    0.935636
DAYS_FIRST_DRAWING           0.760570
Name: DAYS_LAST_DUE, dtype: float64
```

```
DAYS_TERMINATION
-------------------------------
DAYS_LAST_DUE                0.991670
DAYS_FIRST_DUE               0.941859
DAYS_DECISION                0.939959
DAYS_LAST_DUE_1ST_VERSION    0.933585
Name: DAYS_TERMINATION, dtype: float64


PREV_APCTN_CRDT_DIFF
-------------------------------
PREV_APCTN_CRDT_RATIO        0.955565
RATE_DOWN_PAYMENT            0.846313
PREV_DWN_PYMNT_CRDT_RATIO    0.846313
AMT_DOWN_PAYMENT             0.812550
Name: PREV_APCTN_CRDT_DIFF, dtype: float64


PREV_APCTN_CRDT_RATIO
-------------------------------
PREV_APCTN_CRDT_DIFF         0.955565
RATE_DOWN_PAYMENT            0.866425
PREV_DWN_PYMNT_CRDT_RATIO    0.866425
AMT_DOWN_PAYMENT             0.771830
DAYS_FIRST_DRAWING          -0.758784
Name: PREV_APCTN_CRDT_RATIO, dtype: float64


PREV_CRDT_ANNUTY_RATIO
-------------------------------
AMT_CREDIT    0.706005
Name: PREV_CRDT_ANNUTY_RATIO, dtype: float64


PREV_DWN_PYMNT_CRDT_RATIO
-------------------------------
RATE_DOWN_PAYMENT        1.000000
AMT_DOWN_PAYMENT         0.918135
PREV_APCTN_CRDT_RATIO    0.866425
PREV_APCTN_CRDT_DIFF     0.846313
DAYS_FIRST_DRAWING       0.785269
Name: PREV_DWN_PYMNT_CRDT_RATIO, dtype: float64


--------------------------------------------------------------------------
--------------------------------------------------------------------------
--------------------------------------------------------------------------
Table under consideration FOR HIGHEST CORRELATIONS: POS_CASH_BALANCE
CNT_INSTALMENT
-------------------------------
CNT_INSTALMENT_FUTURE    0.740763
Name: CNT_INSTALMENT, dtype: float64


CNT_INSTALMENT_FUTURE
-------------------------------
POS_PERC_INSTL_PNDNG    0.790487
CNT_INSTALMENT          0.740763
```

```
Name: CNT_INSTALMENT_FUTURE, dtype: float64


SK_DPD
-------------------------------
POS_DAYS_WTHT_TOLRNC    0.791953
Name: SK_DPD, dtype: float64


POS_PERC_INSTL_PNDNG
-------------------------------
CNT_INSTALMENT_FUTURE    0.790487
POS_CNT_INSTAL_PNDNG    -0.737500
Name: POS_PERC_INSTL_PNDNG, dtype: float64


POS_CNT_INSTAL_PNDNG
-------------------------------
POS_PERC_INSTL_PNDNG   -0.7375
Name: POS_CNT_INSTAL_PNDNG, dtype: float64


POS_DAYS_WTHT_TOLRNC
-------------------------------
SK_DPD    0.791953
Name: POS_DAYS_WTHT_TOLRNC, dtype: float64


------------------------------------------------------------------------
------------------------------------------------------------------------
```

Now, we find the attributes which are highly correlated, either positive or negative. Looks like EXT_SOURCE columns are highly correlated.

In [103…

```
correlations = datasets["application_train"].corr()['TARGET'].sort_values()
print('Most Positive Correlations:\n', correlations.tail(10))
print('\nMost Negative Correlations:\n', correlations.head(10))
```

```
Most Positive Correlations:
 FLAG_DOCUMENT_3                0.044346
REG_CITY_NOT_LIVE_CITY         0.044395
FLAG_EMP_PHONE                 0.045982
REG_CITY_NOT_WORK_CITY         0.050994
DAYS_ID_PUBLISH                0.051457
DAYS_LAST_PHONE_CHANGE         0.055218
REGION_RATING_CLIENT           0.058899
REGION_RATING_CLIENT_W_CITY    0.060893
DAYS_BIRTH                     0.078239
TARGET                         1.000000
Name: TARGET, dtype: float64

Most Negative Correlations:
 EXT_SOURCE_3                 -0.178919
EXT_SOURCE_2                 -0.160472
EXT_SOURCE_1                 -0.155317
DAYS_EMPLOYED                -0.044932
FLOORSMAX_AVG                -0.044003
FLOORSMAX_MEDI               -0.043768
FLOORSMAX_MODE               -0.043226
AMT_GOODS_PRICE              -0.039645
REGION_POPULATION_RELATIVE   -0.037227
ELEVATORS_AVG                -0.034199
Name: TARGET, dtype: float64
```

Top 10 highly correlated columns:

In [104…

```
top_10 = correlations.abs().sort_values().tail(11)
top_10
```

Out[104]:

```
FLAG_EMP_PHONE                 0.045982
REG_CITY_NOT_WORK_CITY         0.050994
DAYS_ID_PUBLISH                0.051457
DAYS_LAST_PHONE_CHANGE         0.055218
REGION_RATING_CLIENT           0.058899
REGION_RATING_CLIENT_W_CITY    0.060893
DAYS_BIRTH                     0.078239
EXT_SOURCE_1                   0.155317
EXT_SOURCE_2                   0.160472
EXT_SOURCE_3                   0.178919
TARGET                         1.000000
Name: TARGET, dtype: float64
```

Applicants Age

In [105…

```
plt.hist(datasets["application_train"]['DAYS_BIRTH'] / -365, edgecolor = 'k
plt.title('Age of Client'); plt.xlabel('Age (years)'); plt.ylabel('Count');
```

Applicants occupations

In [106…

```
sns.countplot(x='OCCUPATION_TYPE', data=datasets["application_train"]);
plt.title('Applicants Occupation');
plt.xticks(rotation=90);
```

17% of our applicants are labourers and around 10% are from the Sales. This seems like folks which are from the lower income range which apply for the loan.

Other Dataset questions

previous applications - Phase I Analysis

The persons in the kaggle submission file have had previous applications in the `previous_application.csv` . 47,800 out 48,744 people have had previous appications.

In [42]:
```
appsDF = datasets["previous_application"]
```

In [43]:
```
len(np.intersect1d(datasets["previous_application"]["SK_ID_CURR"], datasets
```

Out[43]:
```
47800
```

In [44]:
```
print(f"There are  {appsDF.shape[0]:,} previous applications")
```

```
There are  1,670,214 previous applications
```

In [45]:
```
# How many entries are there for each month?
prevAppCounts = appsDF['SK_ID_CURR'].value_counts(dropna=False)
```

In [46]:
```
len(prevAppCounts[prevAppCounts >40])  #more that 40 previous applications
```

Out[46]:
```
101
```

In [47]:
```
prevAppCounts[prevAppCounts >50].plot(kind='bar')
plt.xticks(rotation=25)
plt.show()
```

Histogram of Number of previous applications for an ID

In [48]:

```
sum(appsDF['SK_ID_CURR'].value_counts()==1)
```

Out[48]:

```
60458
```

In [49]:

```
plt.hist(appsDF['SK_ID_CURR'].value_counts(), cumulative =True, bins = 100)
plt.grid()
plt.ylabel('cumulative number of IDs')
plt.xlabel('Number of previous applications per ID')
plt.title('Histogram of Number of previous applications for an ID')
```

Out[49]:

```
Text(0.5, 1.0, 'Histogram of Number of previous applications for an ID')
```

Can we differentiate applications by low, medium and high previous apps?

```
* Low = <5 claims (22%)
* Medium = 10 to 39 claims (58%)
* High = 40 or more claims (20%)
```

In [50]:

```
apps_all = appsDF['SK_ID_CURR'].nunique()
apps_5plus = appsDF['SK_ID_CURR'].value_counts()>=5
#print(apps_5plus)
apps_40plus = appsDF['SK_ID_CURR'].value_counts()>=40
apps_med_plus = 100 - apps_5plus- apps_40plus
print('Percentage with 10 or more previous apps:', np.round(100.*(sum(apps_
print('Percentage with 11 to 39 no of apps:', np.round(100-(100.*(sum(apps_
print('Percentage with 40 or more previous apps:', np.round(100.*(sum(apps_
```

```
Percentage with 10 or more previous apps: 41.76895
Percentage with 11 to 39 no of apps: 58.19653
Percentage with 40 or more previous apps: 0.03453
```

In [51]:

```
df1 = pd.DataFrame(prevAppCounts)
```

Baseline Machine Learning Pipelines

Pre-Processing

OHE when previously unseen unique values in the test/validation set

Train, validation and Test sets (and the leakage problem we have mentioned previously):

Let's look at a small usecase to tell us how to deal with this:

- The OneHotEncoder is fitted to the training set, which means that for each unique value present in the training set, for each feature, a new column is created. Let's say we have 39 columns after the encoding up from 30 (before preprocessing).
- The output is a numpy array (when the option sparse=False is used), which has the disadvantage of losing all the information about the original column names and values.
- When we try to transform the test set, after having fitted the encoder to the training set, we obtain a `ValueError`. This is because the there are new, previously unseen unique values in the test set and the encoder doesn't know how to handle these values. In order to use both the transformed training and test sets in machine learning algorithms, we need them to have the same number of columns.

This last problem can be solved by using the option handle_unknown='ignore'of the OneHotEncoder, which, as the name suggests, will ignore previously unseen values when transforming the test set.

Here is a example that in action:

```
# Identify the categorical features we wish to consider.
cat_attribs = ['CODE_GENDER',
'FLAG_OWN_REALTY','FLAG_OWN_CAR','NAME_CONTRACT_TYPE',

'NAME_EDUCATION_TYPE','OCCUPATION_TYPE','NAME_INCOME_TYPE']

# Notice handle_unknown="ignore" in OHE which ignore values from
the validation/test that
# do NOT occur in the training set
cat_pipeline = Pipeline([
        ('selector', DataFrameSelector(cat_attribs)),
        ('imputer', SimpleImputer(strategy='most_frequent')),
        ('ohe', OneHotEncoder(sparse=False,
handle_unknown="ignore"))
    ])
```

Please this blog for more details of OHE when the validation/test have previously unseen unique values.

DataFrameSelector Class

In [10]:

```
# Create a class to select numerical or categorical columns
# since Scikit-Learn doesn't handle DataFrames yet
class DataFrameSelector(BaseEstimator, TransformerMixin):
    def __init__(self, attribute_names):
        self.attribute_names = attribute_names
    def fit(self, X, y=None):
        return self
    def transform(self, X):
        return X[self.attribute_names].values
```

Phase 1 Baseline Model

Here are some model runs using only the `application_train` data, with various feature selections.

Baseline with 14 features

In [13]:

```
# Identify the numeric features we wish to consider.
from sklearn.pipeline import Pipeline, FeatureUnion, make_pipeline
from sklearn.compose import ColumnTransformer
from sklearn.datasets import load_iris
import numpy as np
import pandas as pd
from time import time
from sklearn.model_selection import train_test_split  # sklearn.cross_valid
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.impute import SimpleImputer
from sklearn.metrics import accuracy_score
from sklearn.metrics import roc_auc_score

import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split

data = datasets["application_train"]
y = data['TARGET']
X = data.drop(['SK_ID_CURR', 'TARGET'], axis = 1) #drop some features with

# Split the provided training data into training and validationa and test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15, ra
X_train, X_valid, y_train, y_valid = train_test_split(X_train, y_train, tes
print(f"X train           shape: {X_train.shape}")
print(f"X validation      shape: {X_valid.shape}")
print(f"X test            shape: {X_test.shape}")

num_attribs = [
    'AMT_INCOME_TOTAL',  'AMT_CREDIT','DAYS_EMPLOYED','DAYS_BIRTH','EXT_SOU
    'EXT_SOURCE_2','EXT_SOURCE_3']

num_pipeline = Pipeline([
        ('selector', DataFrameSelector(num_attribs)),
        ('imputer', SimpleImputer(strategy='mean')),
        ('std_scaler', StandardScaler()),
    ])
# Identify the categorical features we wish to consider.
cat_attribs = ['CODE_GENDER', 'FLAG_OWN_REALTY','FLAG_OWN_CAR','NAME_CONTRA
               'NAME_EDUCATION_TYPE','OCCUPATION_TYPE','NAME_INCOME_TYPE']

selected_features = num_attribs + cat_attribs
# Notice handle_unknown="ignore" in OHE which ignore values from the valida
# do NOT occur in the training set
cat_pipeline = Pipeline([
        ('selector', DataFrameSelector(cat_attribs)),
        #('imputer', SimpleImputer(strategy='most_frequent')),
        ('imputer', SimpleImputer(strategy='constant', fill_value='missing'
        ('ohe', OneHotEncoder(sparse=False, handle_unknown="ignore"))
    ])

data_pipeline = ColumnTransformer(transformers=[
        ("num_pipeline", num_pipeline, num_attribs),
        ("cat_pipeline", cat_pipeline, cat_attribs)],
        remainder='drop',
        n_jobs=-1
    )
```

```python
full_pipeline_with_predictor = Pipeline([
        ("preparation", data_pipeline),
        ("linear", LogisticRegression())
    ])

param_grid = {'linear__penalty':[#'l1', 'l2', 'elasticnet',
                                 'none']
              #,'linear__C':[1.0#, 10.0, 100.0 ]
             }

gd2 = GridSearchCV(full_pipeline_with_predictor, param_grid= param_grid, cv

model = gd2.fit(X_train, y_train)


try:
    expLog
except NameError:
    expLog = pd.DataFrame(columns=["exp_name",
                                   "Train Acc",
                                   "Valid Acc",
                                   "Test  Acc",
                                   "Train AUC",
                                   "Valid AUC",
                                   "Test  AUC"
                                  ])

exp_name = f"Baseline_{len(selected_features)}_features"
expLog.loc[len(expLog)] = [f"{exp_name}"] + list(np.round(
               [accuracy_score(y_train, model.predict(X_train)),
                accuracy_score(y_valid, model.predict(X_valid)),
                accuracy_score(y_test, model.predict(X_test)),
                roc_auc_score(y_train, model.predict_proba(X_train)[:, 1]),
                roc_auc_score(y_valid, model.predict_proba(X_valid)[:, 1]),
                roc_auc_score(y_test, model.predict_proba(X_test)[:, 1])],
    4))
expLog
```

```
X train           shape: (209107, 120)
X validation      shape: (52277, 120)
X test            shape: (46127, 120)
```

Out[13]:

	exp_name	Train Acc	Valid Acc	Test Acc	Train AUC	Valid AUC	Test AUC
0	Baseline_14_features	0.9199	0.9163	0.9193	0.7358	0.7357	0.7359

Split train, validation and test sets

In [14]:

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split

data = datasets["application_train"]
y = data['TARGET']
X = data.drop(['SK_ID_CURR', 'TARGET'], axis = 1) #drop some features with

# Split the provided training data into training and validationa and test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15, ra
X_train, X_valid, y_train, y_valid = train_test_split(X_train, y_train, tes
print(f"X train           shape: {X_train.shape}")
print(f"X validation      shape: {X_valid.shape}")
print(f"X test            shape: {X_test.shape}")
```

```
X train           shape: (209107, 120)
X validation      shape: (52277, 120)
X test            shape: (46127, 120)
```

Baseline 2: All features

In [15]:

```
from sklearn.pipeline import Pipeline, FeatureUnion, make_pipeline
from sklearn.compose import ColumnTransformer
from sklearn.datasets import load_iris
import numpy as np
import pandas as pd
from time import time
from sklearn.model_selection import train_test_split  # sklearn.cross_valid
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.impute import SimpleImputer
from sklearn.metrics import accuracy_score
from sklearn.metrics import roc_auc_score


data = datasets["application_train"]
y = data['TARGET']
X = data.drop(['SK_ID_CURR', 'TARGET'], axis = 1) #drop some features with 

# Split the provided training data into training and validationa and test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15, ra
X_train, X_valid, y_train, y_valid = train_test_split(X_train, y_train, tes
print(f"X train           shape: {X_train.shape}")
print(f"X validation      shape: {X_valid.shape}")
print(f"X test            shape: {X_test.shape}")

numerical_features = list(numerical_ix[2:])

num_pipeline =Pipeline([
    ('imputer',SimpleImputer(strategy="median")),
    ('std_scaler', StandardScaler())
])
categorical_features = list(categorical_ix)

selected_features = (numerical_features) + (categorical_features)

cat_pipeline = Pipeline([
        ('imputer', SimpleImputer(strategy='most_frequent')),
        ('ohe', OneHotEncoder(sparse=False, handle_unknown="ignore"))
    ])


data_pipeline = ColumnTransformer(transformers=[
        ("num_pipeline", num_pipeline, numerical_features),
        ("cat_pipeline", cat_pipeline, categorical_features)],
        remainder='drop',
        n_jobs=-1
    )


full_pipeline_with_predictor = Pipeline([
        ("preparation", data_pipeline),
        ("linear", LogisticRegression())
    ])

param_grid = {'linear__penalty':[#'l1', 'l2', 'elasticnet',
                                 'none']
              #,'linear__C':[1.0#, 10.0, 100.0]
             }
```

```
gd1 = GridSearchCV(full_pipeline_with_predictor, param_grid= param_grid, cv

model = gd1.fit(X_train, y_train)


try:
    expLog
except NameError:
    expLog = pd.DataFrame(columns=["exp_name",
                                   "Train Acc",
                                   "Valid Acc",
                                   "Test  Acc",
                                   "Train AUC",
                                   "Valid AUC",
                                   "Test  AUC"
                                  ])

exp_name = f"Baseline_{len(selected_features)}_features"
expLog.loc[len(expLog)] = [f"{exp_name}"] + list(np.round(
               [accuracy_score(y_train, model.predict(X_train)),
                accuracy_score(y_valid, model.predict(X_valid)),
                accuracy_score(y_test, model.predict(X_test)),
                roc_auc_score(y_train, model.predict_proba(X_train)[:, 1]),
                roc_auc_score(y_valid, model.predict_proba(X_valid)[:, 1]),
                roc_auc_score(y_test, model.predict_proba(X_test)[:, 1])],
    4))
expLog
```

```
X train           shape: (209107, 120)
X validation      shape: (52277, 120)
X test            shape: (46127, 120)
```

Out[15]:

	exp_name	Train Acc	Valid Acc	Test Acc	Train AUC	Valid AUC	Test AUC
0	Baseline_14_features	0.9199	0.9163	0.9193	0.7358	0.7357	0.7359
1	Baseline_120_features	0.9200	0.9163	0.9193	0.7478	0.7472	0.7434

Baseline 3: 79 Features

Selected Features

Remove elements with more than 50% nulls

In [20]:

```
from sklearn.pipeline import Pipeline, FeatureUnion, make_pipeline
from sklearn.compose import ColumnTransformer
from sklearn.datasets import load_iris
import numpy as np
import pandas as pd
from time import time
from sklearn.model_selection import train_test_split  # sklearn.cross_valid
from sklearn.preprocessing import StandardScaler, OneHotEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.impute import SimpleImputer
from sklearn.metrics import accuracy_score
from sklearn.metrics import roc_auc_score


data = datasets["application_train"]
y = data['TARGET']
X = data.drop(['SK_ID_CURR', 'TARGET'], axis = 1) #drop some features with 

# Split the provided training data into training and validationa and test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15, ra
X_train, X_valid, y_train, y_valid = train_test_split(X_train, y_train, tes
print(f"X train           shape: {X_train.shape}")
print(f"X validation      shape: {X_valid.shape}")
print(f"X test            shape: {X_test.shape}")

numerical_features = list(numerical_ix[2:].drop(remove_num_nulls))

num_pipeline =Pipeline([
    ('imputer',SimpleImputer(strategy="median")),
    ('std_scaler', StandardScaler())
])
categorical_features = list(categorical_ix.drop(remove_cat_nulls))

selected_features = (numerical_features) + (categorical_features)

cat_pipeline = Pipeline([
        ('imputer', SimpleImputer(strategy='most_frequent')),
        ('ohe', OneHotEncoder(sparse=False, handle_unknown="ignore"))
    ])


data_pipeline = ColumnTransformer(transformers=[
        ("num_pipeline", num_pipeline, numerical_features),
        ("cat_pipeline", cat_pipeline, categorical_features)],
        remainder='drop',
        n_jobs=-1
    )


full_pipeline_with_predictor = Pipeline([
        ("preparation", data_pipeline),
        ("linear", LogisticRegression())
    ])

param_grid = {'linear__penalty':[#'l1', 'l2', 'elasticnet',
                                 'none']
              #,'linear__C':[1.0#, 10.0, 100.0]
                               }
```

```
gd3 = GridSearchCV(full_pipeline_with_predictor, param_grid= param_grid, cv

model = gd3.fit(X_train, y_train)


try:
    expLog
except NameError:
    expLog = pd.DataFrame(columns=["exp_name",
                                   "Train Acc",
                                   "Valid Acc",
                                   "Test  Acc",
                                   "Train AUC",
                                   "Valid AUC",
                                   "Test  AUC"
                                  ])

exp_name = f"Baseline_{len(selected_features)}_features"
expLog.loc[len(expLog)] = [f"{exp_name}"] + list(np.round(
               [accuracy_score(y_train, model.predict(X_train)),
                accuracy_score(y_valid, model.predict(X_valid)),
                accuracy_score(y_test, model.predict(X_test)),
                roc_auc_score(y_train, model.predict_proba(X_train)[:, 1]),
                roc_auc_score(y_valid, model.predict_proba(X_valid)[:, 1]),
                roc_auc_score(y_test, model.predict_proba(X_test)[:, 1])],
    4))
expLog
```

```
X train           shape: (209107, 120)
X validation      shape: (52277, 120)
X test            shape: (46127, 120)
```

Out[20]:

	exp_name	Train Acc	Valid Acc	Test Acc	Train AUC	Valid AUC	Test AUC
0	Baseline_14_features	0.9199	0.9163	0.9193	0.7358	0.7357	0.7359
1	Baseline_120_features	0.9200	0.9163	0.9193	0.7478	0.7472	0.7434
2	Baseline_79_features	0.9200	0.9164	0.9195	0.7441	0.7442	0.7406

Baseline 4 : 79 features; 2 log features

Remove elements with more than 50% nulls with log AMT_ANNUITY and AMT_CREDIT

In [21]:

```
data = datasets["application_train"]
y = data['TARGET']
X = data.drop(['SK_ID_CURR', 'TARGET'], axis = 1) #drop some features with
X['LOG_AMT_ANNUITY'] = np.log(X['AMT_ANNUITY']) #add LOG_AMT_ANNUITY colunm
X = X.drop(['AMT_ANNUITY'], axis = 1) # drop AMT_ANNUITY colunm
X['LOG_AMT_CREDIT'] = np.log(X['AMT_CREDIT']) #add LOG_AMT_ANNUITY colunm
X = X.drop(['AMT_CREDIT'], axis = 1) # drop AMT_ANNUITY colunm

# Split the provided training data into training and validationa and test
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15, r
X_train, X_valid, y_train, y_valid = train_test_split(X_train, y_train, tes
print(f"X train           shape: {X_train.shape}")
print(f"X validation      shape: {X_valid.shape}")
print(f"X test            shape: {X_test.shape}")

numerical_features = list(numerical_ix[2:].drop(remove_num_nulls))
numerical_features.append('LOG_AMT_ANNUITY')
numerical_features.append('LOG_AMT_CREDIT')
numerical_features.remove('AMT_CREDIT')
numerical_features.remove('AMT_ANNUITY')


num_pipeline =Pipeline([
    ('imputer',SimpleImputer(strategy="median")),
    ('std_scaler', StandardScaler())
])

categorical_features =  list(categorical_ix.drop(remove_cat_nulls))

selected_features = (numerical_features) + (categorical_features)

cat_pipeline = Pipeline([
        ('imputer', SimpleImputer(strategy='most_frequent')),
        ('ohe', OneHotEncoder(sparse=False, handle_unknown="ignore"))
    ])


data_pipeline = ColumnTransformer(transformers=[
        ("num_pipeline", num_pipeline, numerical_features),
        ("cat_pipeline", cat_pipeline, categorical_features)],
        remainder='drop',
        n_jobs=-1
    )


full_pipeline_with_predictor = Pipeline([
        ("preparation", data_pipeline),
        ("linear", LogisticRegression())
    ])

param_grid = {'linear__penalty':[#'l1', 'l2', 'elasticnet',
                                 'none']
               #,'linear__C':[1.0#, 10.0, 100.0]
              }

gd4 = GridSearchCV(full_pipeline_with_predictor, param_grid= param_grid, cv

model = gd4.fit(X_train, y_train)
```

```
try:
    expLog
except NameError:
    expLog = pd.DataFrame(columns=["exp_name",
                                   "Train Acc",
                                   "Valid Acc",
                                   "Test  Acc",
                                   "Train AUC",
                                   "Valid AUC",
                                   "Test  AUC"
                                  ])

exp_name = f"Baseline_{len(selected_features)}_features with log attributes
expLog.loc[len(expLog)] = [f"{exp_name}"] + list(np.round(
                [accuracy_score(y_train, model.predict(X_train)),
                 accuracy_score(y_valid, model.predict(X_valid)),
                 accuracy_score(y_test, model.predict(X_test)),
                 roc_auc_score(y_train, model.predict_proba(X_train)[:, 1]),
                 roc_auc_score(y_valid, model.predict_proba(X_valid)[:, 1]),
                 roc_auc_score(y_test, model.predict_proba(X_test)[:, 1])],
    4))
expLog
```

```
X train           shape: (209107, 120)
X validation      shape: (52277, 120)
X test            shape: (46127, 120)
```

Out[21]:

	exp_name	Train Acc	Valid Acc	Test Acc	Train AUC	Valid AUC	Test AUC
0	Baseline_14_features	0.9199	0.9163	0.9193	0.7358	0.7357	0.7359
1	Baseline_120_features	0.9200	0.9163	0.9193	0.7478	0.7472	0.7434
2	Baseline_79_features	0.9200	0.9164	0.9195	0.7441	0.7442	0.7406
3	Baseline_79_features with log attributes	0.9200	0.9164	0.9195	0.7443	0.7447	0.7405

Phase 2 Baseline Model on Aggregated Features

Data Denormalizaation

This first section takes all of the aggregated tables and merges them into the single denormalized table, called `bureau_ip_ccb_prev_pos_merged` .

In [41]:

```
import pandas as pd
import os
from sklearn.model_selection import train_test_split

DATA_DIR =  "/../Data/"

ds_names = ("application_train", "application_test", "ccb_agg_data", "ip_agg
            "pos_agg_data","bureau_agg_data")

datasets_agg = {}

for ds_name in ds_names:
    datasets_agg[ds_name] = pd.read_csv(os.getcwd() + DATA_DIR + f'{ds_name
```

In [42]:

```
pos_merged = datasets_agg["application_train"].merge(datasets_agg["pos_agg_
               .replace(to_replace='\s+', value='_', regex=True) \
               .replace(to_replace='\-', value='_', regex=True) \
               .replace(to_replace='\(', value='', regex=True) \
               .replace(to_replace='\)', value='', regex=True)
```

In [43]:

```
pos_merged_test = datasets_agg["application_test"].merge(datasets_agg["pos_
               .replace(to_replace='\s+', value='_', regex=True) \
               .replace(to_replace='\-', value='_', regex=True) \
               .replace(to_replace='\(', value='', regex=True) \
               .replace(to_replace='\)', value='', regex=True)
```

In [44]:

```
prev_pos_merged = pos_merged.merge(datasets_agg["prevapp_agg_data"], on='SK_
               .replace(to_replace='\s+', value='_', regex=True) \
               .replace(to_replace='\-', value='_', regex=True) \
               .replace(to_replace='\(', value='', regex=True) \
               .replace(to_replace='\)', value='', regex=True)
```

In [45]:

```
prev_pos_merged_test = pos_merged_test.merge(datasets_agg["prevapp_agg_data
               .replace(to_replace='\s+', value='_', regex=True) \
               .replace(to_replace='\-', value='_', regex=True) \
               .replace(to_replace='\(', value='', regex=True) \
               .replace(to_replace='\)', value='', regex=True)
```

In [46]:

```
ccb_prev_pos_merged = prev_pos_merged.merge(datasets_agg["ccb_agg_data"], o
               .replace(to_replace='\s+', value='_', regex=True) \
               .replace(to_replace='\-', value='_', regex=True) \
               .replace(to_replace='\(', value='', regex=True) \
               .replace(to_replace='\)', value='', regex=True)
```

In [47]:

```
ccb_prev_pos_merged_test = prev_pos_merged_test.merge(datasets_agg["ccb_agg_
                .replace(to_replace='\s+', value='_', regex=True) \
                .replace(to_replace='\-', value='_', regex=True) \
                .replace(to_replace='\(', value='', regex=True) \
                .replace(to_replace='\)', value='', regex=True)
```

In [48]:

```
ip_ccb_prev_pos_merged = ccb_prev_pos_merged.merge(datasets_agg["ip_agg_data
                .replace(to_replace='\s+', value='_', regex=True) \
                .replace(to_replace='\-', value='_', regex=True) \
                .replace(to_replace='\(', value='', regex=True) \
                .replace(to_replace='\)', value='', regex=True)
```

In [49]:

```
ip_ccb_prev_pos_merged_test = ccb_prev_pos_merged_test.merge(datasets_agg["
                .replace(to_replace='\s+', value='_', regex=True) \
                .replace(to_replace='\-', value='_', regex=True) \
                .replace(to_replace='\(', value='', regex=True) \
                .replace(to_replace='\)', value='', regex=True)
```

In [50]:

```
bureau_ip_ccb_prev_pos_merged = ip_ccb_prev_pos_merged.merge(datasets_agg["
                .replace(to_replace='\s+', value='_', regex=True) \
                .replace(to_replace='\-', value='_', regex=True) \
                .replace(to_replace='\(', value='', regex=True) \
                .replace(to_replace='\)', value='', regex=True)
```

In [51]:

```
bureau_ip_ccb_prev_pos_merged_test = ip_ccb_prev_pos_merged_test.merge(data
                .replace(to_replace='\s+', value='_', regex=True) \
                .replace(to_replace='\-', value='_', regex=True) \
                .replace(to_replace='\(', value='', regex=True) \
                .replace(to_replace='\)', value='', regex=True)
```

In [52]:

```
bureau_ip_ccb_prev_pos_merged.to_csv(os.getcwd() + DATA_DIR + 'bureau_ip_cc
```

In [53]:

```
bureau_ip_ccb_prev_pos_merged_test.to_csv(os.getcwd() + DATA_DIR + 'bureau_
```

Model Pipeline with All Features

This section runs the baseline ML pipeline using all the integrated data features as outlined by the block diagram below:

In [54]:

```
DATA_DIR =  "/../Data/"

datasets_agg = {}
datasets_agg["bureau_ip_ccb_prev_pos_merged"] = pd.read_csv(os.getcwd() + DA
bureau_ip_ccb_prev_pos_merged = datasets_agg["bureau_ip_ccb_prev_pos_merged
```

In [56]:

```
y = bureau_ip_ccb_prev_pos_merged['TARGET']
X = bureau_ip_ccb_prev_pos_merged.drop(['SK_ID_CURR', 'TARGET', 'Unnamed: 0

#X['LOG_AMT_ANNUITY'] = np.log(X['AMT_ANNUITY']) #add LOG_AMT_ANNUITY colun
#X = X.drop(['AMT_ANNUITY'], axis = 1) # drop AMT_ANNUITY colunm
#X['LOG_AMT_CREDIT'] = np.log(X['AMT_CREDIT']) #add LOG_AMT_ANNUITY colunm
#X = X.drop(['AMT_CREDIT'], axis = 1) # drop AMT_ANNUITY colunm

# Split the provided training data into training and validationa and test
_, X, _, y = train_test_split(X, y, test_size=0.8, random_state=42)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, ra
X_train, X_valid, y_train, y_valid = train_test_split(X_train, y_train, tes

X_train.replace([np.inf, -np.inf], np.nan, inplace=True)
```

In [57]:

```
print(f"X train           shape: {X_train.shape}")
print(f"X validation      shape: {X_valid.shape}")
print(f"X test            shape: {X_test.shape}")
```

```
X train           shape: (157445, 1277)
X validation      shape: (39362, 1277)
X test            shape: (49202, 1277)
```

In [58]:

```
id_cols, feat_num, feat_cat, features =  id_num_cat_feature(X, text = False
```

In [62]:

```
num_pipeline = Pipeline([
    ('imputer',SimpleImputer(strategy="median")),
    ('std_scaler', StandardScaler())
])

#categorical_features =  list(categorical_ix.drop(remove_cat_nulls))
categorical_features =  feat_cat
numerical_features = feat_num

selected_features = (numerical_features) + (categorical_features)

cat_pipeline = Pipeline([
        ('imputer', SimpleImputer(strategy='most_frequent')),
        ('ohe', OneHotEncoder(sparse=False, handle_unknown="ignore"))
    ])


data_pipeline = ColumnTransformer(transformers=[
        ("num_pipeline", num_pipeline, numerical_features),
        ("cat_pipeline", cat_pipeline, categorical_features)],
        remainder='drop',
        n_jobs=-1
    )


full_pipeline_with_predictor = Pipeline([
        ("preparation", data_pipeline),
       # ('pca', decomposition.PCA()),
        ("logistic_Reg", LogisticRegression(solver="liblinear"))
    ])

#n_components = list(range(1,X_train.shape[1]+1,1))
#C = np.logspace(-4, 4, 50)
penalty = [
           #,'l1', 'l2'
          ]

parameters = dict(#pca__n_components=n_components,
                      #logistic_Reg__C=C,
                  #    logistic_Reg__penalty=penalty
                 )
```

In [63]:

```
gd4 = GridSearchCV(full_pipeline_with_predictor, param_grid= parameters, cv
```

In [64]:

```
gd4.fit(X_train, y_train)
```

Out[64]:
```
GridSearchCV(cv=5,
             estimator=Pipeline(steps=[('preparation',
                                        ColumnTransformer(n_jobs=-1,
                                                          transformers=[('nu
m_pipeline',
                                                                         Pip
eline(steps=[('imputer',
                                                                               
SimpleImputer(strategy='median')),
                                                                               
('std_scaler',
                                                                               
StandardScaler())]),
                                                                         ['C
NT_CHILDREN',
                                                                          'A
MT_INCOME_TOTAL',
                                                                          'A
MT_CREDIT',
                                                                          'A
MT_ANNUITY',
                                                                          'A
MT_GOODS_PRICE',
                                                                          'R
EGION_POPULATION_RELATIVE',
                                                                          'D
AYS_BIRTH',
                                                                          'D
AYS_EMPLOYED',...
                                                                          'N
AME_TYPE_SUITE',
                                                                          'N
AME_INCOME_TYPE',
                                                                          'N
AME_EDUCATION_TYPE',
                                                                          'N
AME_FAMILY_STATUS',
                                                                          'N
AME_HOUSING_TYPE',
                                                                          'O
CCUPATION_TYPE',
                                                                          'W
EEKDAY_APPR_PROCESS_START',
                                                                          'O
RGANIZATION_TYPE',
                                                                          'F
ONDKAPREMONT_MODE',
                                                                          'H
OUSETYPE_MODE',
                                                                          'W
ALLSMATERIAL_MODE',
                                                                          'E
MERGENCYSTATE_MODE'])])),
                                        ('logistic_Reg',
                                         LogisticRegression(solver='liblinear
'))]),
             n_jobs=-10, param_grid={}, scoring='roc_auc')
```

In [ ]:

```
# print('Best Penalty:', gd4.best_estimator_.get_params()['logistic_Reg__pe
# print('Best C:', gd4.best_estimator_.get_params()['logistic_Reg__C'])
# #print('Best Number Of Components:', gd4.best_estimator_.get_params()['pc
# print(); print(gd4.best_estimator_.get_params()['logistic_Reg'])
```

In [67]:

```
from sklearn.metrics import accuracy_score
from sklearn.metrics import roc_auc_score

X_valid.replace([np.inf, -np.inf], np.nan, inplace=True)
X_test.replace([np.inf, -np.inf], np.nan, inplace=True)

try:
    expLog
except NameError:
    expLog = pd.DataFrame(columns=["exp_name",
                                   "Train Acc",
                                   "Valid Acc",
                                   "Test  Acc",
                                   "Train AUC",
                                   "Valid AUC",
                                   "Test  AUC"
                                  ])

exp_name = f"Baseline_{len(selected_features)}_features with log attributes
expLog.loc[len(expLog)] = [f"{exp_name}"] + list(np.round(
               [accuracy_score(y_train, gd4.predict(X_train)),
                accuracy_score(y_valid, gd4.predict(X_valid)),
                accuracy_score(y_test, gd4.predict(X_test)),
                roc_auc_score(y_train, gd4.predict_proba(X_train)[:, 1]),
                roc_auc_score(y_valid, gd4.predict_proba(X_valid)[:, 1]),
                roc_auc_score(y_test, gd4.predict_proba(X_test)[:, 1])],
    4))
expLog
```

Out[67]:

	exp_name	Train Acc	Valid Acc	Test Acc	Train AUC	Valid AUC	Test AUC
0	Baseline_1277_features with log attributes	0.9198	0.9209	0.9186	0.7871	0.7663	0.7709

Results and Discussion

Phase 1

Below are the results from our Phase 1 Experiment Log.

Experimental results

Please present the results of the various experiments that you conducted. The results should be shown in a table or image. Try to include the different details for each experiment.

Please include code sections when necessary as well as images or any relevant material

Out[87]:

	exp_name	Train Acc	Valid Acc	Test Acc	Train AUC	Valid AUC	Test AUC
0	Baseline_14_features	0.9198	0.9192	0.9158	0.7358	0.7358	0.7357
1	Baseline_120_features	0.9200	0.9163	0.9193	0.7478	0.7472	0.7434
2	Baseline_79_features	0.9200	0.9164	0.9195	0.7441	0.7442	0.7406
3	Baseline_79_features with log attributes	0.9200	0.9164	0.9195	0.7443	0.7447	0.7405

In Phase 1, the results of our baseline model were a good starting point. With only a modicum of complexity, we were able to establish an ROC-AUC score of 74.34%. Below are the details on phase I results:

1. Initial run was considered with 14 features, 7 numerical and 7 categorical. We got training accuracy of 91.98%, test accuracy of 91.58% and AUC score as 73.57% for this very first model.
    - Num cols : ‘AMT_INCOME_TOTAL', 'AMT_CREDIT','DAYS_EMPLOYED','DAYS_BIRTH','EXT_SOURCE_1', 'EXT_SOURCE_2','EXT_SOURCE_3’,
    - Categorical Cols : 'CODE_GENDER','FLAG_OWN_REALTY','FLAG_OWN_CAR','NAME_CONTRACT_TYF 'OCCUPATION_TYPE', 'NAME_INCOME_TYPE'
2. Next we considered all 120 elements form Application train table. Here we got a marginal increase to 92% train accuracy, and 91.93% test accuracy. But there was improvement in AUC score, which was 74.34 % in this iteration.
3. Next, we used a function to remove features which were more than 50% NULL. This led to a list of 79 features. There was very minute change in the test and train score. AUC score reduced from 74.34% to 74.06%. Although reducing the number of features was definitely a boost but we expect same or higher AUC score.
4. Next, we log transformed 2 attributes which were highly skewed and used the features from run 3 from above step. This did not help either, we got same score as in last run up to first decimal point. 1 reason for this is because the attribute we log transformed had a big null percentage which didn’t really helped the score.

Our Kaggle score for submission was 73.3 for this phase.

Phase 2

Phase 2

Below is the result from the Phase 2 Experiment Log.

Out[67]:

	exp_name	Train Acc	Valid Acc	Test Acc	Train AUC	Valid AUC	Test AUC
0	Baseline_1277_features with log attributes	0.9198	0.9209	0.9186	0.7871	0.7663	0.7709

In Phase 2, with new feature engineering and selection, hyper-parameter tuning, our model performed significantly better with ROC-AUC scores averaging around 77%, with the highest being 77.09%. The inclusion of newly engineered features did help increase our scores, which was anyway our main goal for this phase of the project.

Here we have taken 1277 features for the pipeline run. As explained in the EDA part, we aggregated all features with "min", "max", "count", "sum", "median", "mean", and "var" and subsequently rolled up all those to the main train and test tables.

We also created more features as part of our analysis on features:

From POS:

- Percentage of installments pending.
- Number of installments pending.
- Days with Tolerance.

From Previous Application:

- Count of approved previous application.
- Count of Rejected previous applications.
- Difference: Amount requested in application - Actual credit amount.
- Ratio - Ratio of application amount to amount credited.
- Ratio - Ratio of amount credited to amount annuity
- Ratio - Ratio of down payment to amount credited.

From Installments payments:

- Difference of payment amount from installment amount.

Although we did not see much difference in test score but there was an increase in the test score by almost 1%. Scores were .9198 and .9209 respectively. But to expectations, there was a substantial increment in the test AUC score which now has the score of 77.09 %. This 3.6% increase in AUC score is deemed a good outcome. This was also evident in our Kaggle submission score of 76.17 %.

In both the phases, by using pipelines and parameters, we kept data leakage at bay.

Submission File Prep

For each SK_ID_CURR in the test set, you must predict a probability for the TARGET variable. The file should contain a header and have the following format:

```
SK_ID_CURR,TARGET
100001,0.1
100005,0.9
100013,0.2
etc.
```

In [68]:

```
DATA_DIR =  "/../Data/"

datasets_agg = {}
datasets_agg["bureau_ip_ccb_prev_pos_merged_test"] = pd.read_csv(os.getcwd(
bureau_ip_ccb_prev_pos_merged_test = datasets_agg["bureau_ip_ccb_prev_pos_m
```

In [69]:

```
data_test = bureau_ip_ccb_prev_pos_merged_test
X_Kaggle_test = data_test.drop(['SK_ID_CURR','Unnamed: 0'], axis = 1)
X_Kaggle_test.replace([np.inf, -np.inf], np.nan, inplace=True)

print(X_Kaggle_test.shape)
```

```
(48744, 1277)
```

In [ ]:

```
X_Kaggle_test.replace([np.inf, -np.inf], np.nan, inplace=True)
#X_Kaggle_test.head()
```

In [ ]:

```
test_class_scores = gd4.predict_proba(X_Kaggle_test)[:, 1]
```

In [ ]:

```
submit_df = datasets["application_test"][['SK_ID_CURR']]
submit_df['TARGET'] = test_class_scores
submit_df.head()
```

Out[ ]:

	SK_ID_CURR	TARGET
0	100001	0.047134
1	100005	0.182908
2	100013	0.042883
3	100028	0.026232
4	100038	0.133339

In [ ]:

```
submit_df.to_csv("submission.csv",index=False)
```

Kaggle Submission

Click on this link

Submissions

You selected 0 of 2 submissions to be evaluated for your final leaderboard score. Since you selected less than 2 submission, Kaggle auto-selected up to 2 submissions from among your public best-scoring unselected submissions for evaluation. The evaluated submission with the best Private Score is used for your final score.

0/2

Submissions evaluated for final score

All | Successful | Selected | Errors

Recent

Submission and Description	Private Score	Public Score	Selected
submission3.csv Complete (after deadline) · 3m ago · Second Submission for Phase 2	0.76011	0.7617	☐
submission2.csv Complete (after deadline) · 8h ago · The second submission	0.74958	0.75199	☐
submission.csv Complete (after deadline) · 19h ago · not 0 or 1	0.73653	0.73569	☐

Conclusion

The goal of this project is to predict an aplicant's default risk using data from the loan application, as well as past credit accounts. This is important as it will provide a way to more equitably provide loans to people with poor credit history. With the conclusion of Phase 2, Group 10 has accomplished the following tasks:

1. Analyzed the data structures and relationships through visual and numerical exploratory data analysis.
2. Transformed the normalized data tables into a denormalized input variable and integrated with the machine learning pipeline.
3. Laid the foundation for future hyperparameter tuning and model engineering.

One challenge was building a universal transformation method to apply to the tables. Group 10 solved this issue by initially analyzing the tables separately from one another, and then collaborating intensely once their table domain was understood to find the common transformation methods - also to identify where the table transformations must be different.

Another challenge overcome was working with the large (2.7 gb) high-dimensional dataset. This was overcome by simplifying code, running code chunks in batches, and using GPU-enhanced computing methods like Google Collab. Also special thanks to Pragat for training the model on his machine for *hours* overnight.

With a AUC-ROC score of 0.76 on the test data using the fully integrated dataset, this baseline pipeline shows logistic regression is a decent tool that can successfully use the massive amount of data to make predictions on applicant default risk. Furthermore, it shows an improvement when using *all* of the data as opposed to just data from the applications.

In the next phase, we will focus on hyperparameter tuning, additional feature engineering and feature selection, and refining the overall machine-learning model.

Bibliography

Data Dictionary

	Table	Row	Description	Special
1	application_{train|test}.csv	SK_ID_CURR	ID of loan in our sample	
2	application_{train|test}.csv	TARGET	"Target variable (1 - client with payment difficulties: he/she had late payment more than X days on at least one of the first Y installments of the loan in our sample, 0 - all other cases)"	
5	application_{train|test}.csv	NAME_CONTRACT_TYPE	Identification if loan is cash or revolving	
6	application_{train|test}.csv	CODE_GENDER	Gender of the client	
7	application_{train|test}.csv	FLAG_OWN_CAR	Flag if the client owns a car	
8	application_{train|test}.csv	FLAG_OWN_REALTY	Flag if client owns a house or flat	
9	application_{train|test}.csv	CNT_CHILDREN	Number of children the client has	
10	application_{train|test}.csv	AMT_INCOME_TOTAL	Income of the client	
11	application_{train|test}.csv	AMT_CREDIT	Credit amount of the loan	
12	application_{train|test}.csv	AMT_ANNUITY	Loan annuity	
13	application_{train|test}.csv	AMT_GOODS_PRICE	For consumer loans it is the price of the goods for which the loan is given	
14	application_{train|test}.csv	NAME_TYPE_SUITE	Who was accompanying client when he was applying for the loan	
15	application_{train|test}.csv	NAME_INCOME_TYPE	"Clients income type (businessman, working, maternity leave,)"	
16	application_{train|test}.csv	NAME_EDUCATION_TYPE	Level of highest education the client achieved	
17	application_{train|test}.csv	NAME_FAMILY_STATUS	Family status of the client	
18	application_{train|test}.csv	NAME_HOUSING_TYPE	"What is the housing situation of the client (renting, living with parents,	

			...)"	
19	application_{train|test}.csv	REGION_POPULATION_RELATIVE	Normalized population of region where client lives (higher number means the client lives in more populated region)	normalized
20	application_{train|test}.csv	DAYS_BIRTH	Client's age in days at the time of application	time only relative to the application
21	application_{train|test}.csv	DAYS_EMPLOYED	How many days before the application the person started current employment	time only relative to the application
22	application_{train|test}.csv	DAYS_REGISTRATION	How many days before the application did client change his registration	time only relative to the application
23	application_{train|test}.csv	DAYS_ID_PUBLISH	How many days before the application did client change the identity document with which he applied for the loan	time only relative to the application
24	application_{train|test}.csv	OWN_CAR_AGE	Age of client's car	
25	application_{train|test}.csv	FLAG_MOBIL	"Did client provide mobile phone (1=YES, 0=NO)"	
26	application_{train|test}.csv	FLAG_EMP_PHONE	"Did client provide work phone (1=YES, 0=NO)"	
27	application_{train|test}.csv	FLAG_WORK_PHONE	"Did client provide home phone (1=YES, 0=NO)"	
28	application_{train|test}.csv	FLAG_CONT_MOBILE	"Was mobile phone reachable (1=YES, 0=NO)"	
29	application_{train|test}.csv	FLAG_PHONE	"Did client provide home phone (1=YES, 0=NO)"	
30	application_{train|test}.csv	FLAG_EMAIL	"Did client provide email (1=YES, 0=NO)"	
31	application_{train|test}.csv	OCCUPATION_TYPE	What kind of occupation does the client have	
32	application_{train|test}.csv	CNT_FAM_MEMBERS	How many family members does client have	

33	application_{train|test}.csv	REGION_RATING_CLIENT	"Our rating of the region where client lives (1,2,3)"	
34	application_{train|test}.csv	REGION_RATING_CLIENT_W_CITY	"Our rating of the region where client lives with taking city into account (1,2,3)"	
35	application_{train|test}.csv	WEEKDAY_APPR_PROCESS_START	On which day of the week did the client apply for the loan	
36	application_{train|test}.csv	HOUR_APPR_PROCESS_START	Approximately at what hour did the client apply for the loan	rounded
37	application_{train|test}.csv	REG_REGION_NOT_LIVE_REGION	"Flag if client's permanent address does not match contact address (1=different, 0=same, at region level)"	
38	application_{train|test}.csv	REG_REGION_NOT_WORK_REGION	"Flag if client's permanent address does not match work address (1=different, 0=same, at region level)"	
39	application_{train|test}.csv	LIVE_REGION_NOT_WORK_REGION	"Flag if client's contact address does not match work address (1=different, 0=same, at region level)"	
40	application_{train|test}.csv	REG_CITY_NOT_LIVE_CITY	"Flag if client's permanent address does not match contact address (1=different, 0=same, at city level)"	
41	application_{train|test}.csv	REG_CITY_NOT_WORK_CITY	"Flag if client's permanent address does not match work address (1=different, 0=same, at city level)"	
42	application_{train|test}.csv	LIVE_CITY_NOT_WORK_CITY	"Flag if client's contact address does not match work address (1=different, 0=same, at city level)"	
[illegible]	[illegible]	[illegible]	Type of organization	

43	application_{train|test}.csv	ORGANIZATION_TYPE	Type of organization where client works	
44	application_{train|test}.csv	EXT_SOURCE_1	Normalized score from external data source	normalized
45	application_{train|test}.csv	EXT_SOURCE_2	Normalized score from external data source	normalized
46	application_{train|test}.csv	EXT_SOURCE_3	Normalized score from external data source	normalized
47	application_{train|test}.csv	APARTMENTS_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
48	application_{train|test}.csv	BASEMENTAREA_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
49	application_{train|test}.csv	YEARS_BEGINEXPLUATATION_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
			"Normalized information about	

50	application_{train|test}.csv	YEARS_BUILD_AVG	information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
51	application_{train|test}.csv	COMMONAREA_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
52	application_{train|test}.csv	ELEVATORS_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
53	application_{train|test}.csv	ENTRANCES_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized

54	application_{train|test}.csv	FLOORSMAX_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
55	application_{train|test}.csv	FLOORSMIN_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
56	application_{train|test}.csv	LANDAREA_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
57	application_{train|test}.csv	LIVINGAPARTMENTS_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number	normalized

			the building, number of floor"	
58	application_{train|test}.csv	LIVINGAREA_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
59	application_{train|test}.csv	NONLIVINGAPARTMENTS_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
60	application_{train|test}.csv	NONLIVINGAREA_AVG	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
61	application_{train|test}.csv	APARTMENTS_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of	normalized

			entrances, state of the building, number of floor"	
62	application_{train|test}.csv	BASEMENTAREA_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
63	application_{train|test}.csv	YEARS_BEGINEXPLUATATION_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
64	application_{train|test}.csv	YEARS_BUILD_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
65	application_{train|test}.csv	COMMONAREA_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of	normalized

			building, number of elevators, number of entrances, state of the building, number of floor"	
66	application_{train|test}.csv	ELEVATORS_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
67	application_{train|test}.csv	ENTRANCES_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
68	application_{train|test}.csv	FLOORSMAX_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment size, common area, living area, age of building, number of elevators, number of entrances, state of the building, number of floor"	normalized
69	application_{train|test}.csv	FLOORSMIN_MODE	"Normalized information about building where the client lives, What is average (_AVG suffix), modus (_MODE suffix), median (_MEDI suffix) apartment	normalized