Project Question:

To predict the clients' repayment abilities. Also determine the factors or features that influence prediction by analyzing variety of alternative data--including telco and transactional information. The problem that the client wants to solve by throwing open data challenge in Kaggle is as follows:

- Predict customer ability to repay loan as indicated by TARGET variable in the dataset where TARGET = 0 implies customer is able to repay loan while TARGET = 1 customer's difficulty in repaying loan
- Find correlation among independent feature set that includes exhaustive list of 122 independent features
- Which independent features are influencing the TARGET prediction
- Whether influence is in direct relation or inverse relation to TARGET prediction
- Impact of normalized score of External Source data
- Is there pattern in data distribution among correlated feature set that may help in prediction of TARGET

These data are readily and publicly available at https://www.kaggle.com/c/home-credit-default-risk/data and appear as below:

SK_ID_CURR	TARGET	NAME_CONT	CODE_GEND	FLAG_OWN_	FLAG_OWN_	CNT_CHILDR	AMT_INCOM	AMT_CREDIT	AMT_ANNUI	AMT_GOODS	NAME_TYPE_	NAME_INCO	NAME_EDUC	NAME_FAMI	NAME_HOUS	REGION_POI	DAYS_BIRTH	DAYS_EM
100002	1	Cash loans	M	N	Y	0	202500	406597.5	24700.5	351000	Unaccompar	Working	Secondary /	Single / not r	House / apar	0.018801	-9461	-6
100003	0	Cash loans	F	N	N	0	270000	1293502.5	35698.5	1129500	Family	State servant	Higher educa	Married	House / apar	0.003541	-16765	-11
100004	0	Revolving loa	M	Y	Y	0	67500	135000	6750	135000	Unaccompar	Working	Secondary /	Single / not r	House / apar	0.010032	-19046	-2
100006	0	Cash loans	F	N	Y	0	135000	312682.5	29686.5	297000	Unaccompar	Working	Secondary /	Civil marriag	House / apar	0.008019	-19005	-30
100007	0	Cash loans	M	N	Y	0	121500	513000	21865.5	513000	Unaccompar	Working	Secondary /	Single / not r	House / apar	0.028663	-19932	-30
100008	0	Cash loans	M	N	Y	0	99000	490495.5	27517.5	454500	Spouse, part	State servant	Secondary /	Married	House / apar	0.035792	-16941	-15
100009	0	Cash loans	F	Y	Y	1	171000	1560726	41301	1395000	Unaccompar	Commercial	Higher educa	Married	House / apar	0.035792	-13778	-31
100010	0	Cash loans	M	Y	Y	0	360000	1530000	42075	1530000	Unaccompar	State servant	Higher educa	Married	House / apar	0.003122	-18850	-4
100011	0	Cash loans	F	N	Y	0	112500	1019610	33826.5	913500	Children	Pensioner	Secondary /	Married	House / apar	0.018634	-20099	3652

Dataset given has 122 features with 307K +rows

Of the 122 features, following features showed **correlation** with the target variable 'TARGET' i.e. Loan Repayment Prediction:

- Identification if loan is cash or revolving
- Gender of the client
- Normalized score from external data source 1
- Normalized score from external data source 2

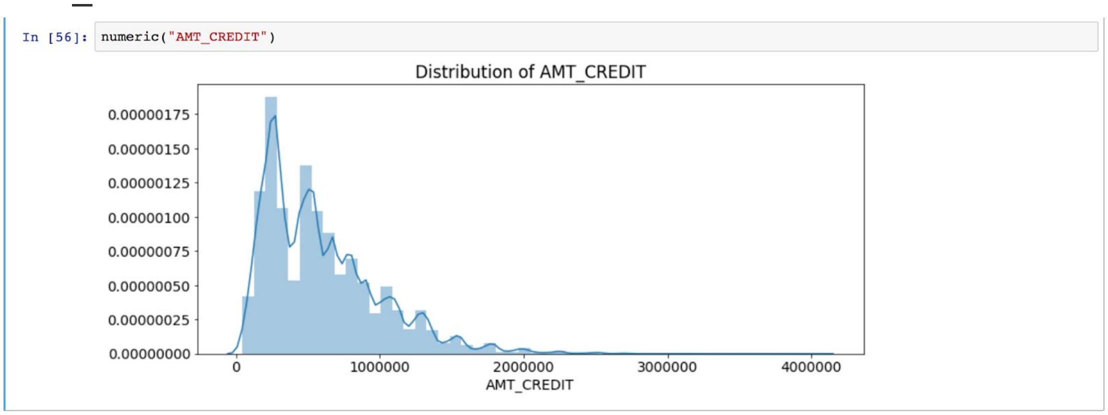
- Normalized score from external data source 3
- Flag if the client owns a car
- Flag if client owns a house or flat
- Number of children the client has
- Income of the client
- Credit amount of the loan
- Loan annuity
- For consumer loans it is the price of the goods for which the loan is given
- Who was accompanying client when he was applying for the loan
- Clients income type (businessman, working, maternity leave,…)
- Level of highest education the client achieved
- Family status of the client
- What is the housing situation of the client (renting, living with parents, ...)

Above dependencies can be verified with following visualization graph plots after performing Data Wrangling steps as explained in google doc link given here: https://docs.google.com/document/d/1MesT5Q-V7EVpQ_Io5EbFUpeElJjFzgqaXq6oCLUjC3A/edit?usp=sharing

1. Exploratory Data Analysis

Visualized distribution of numeric independent features

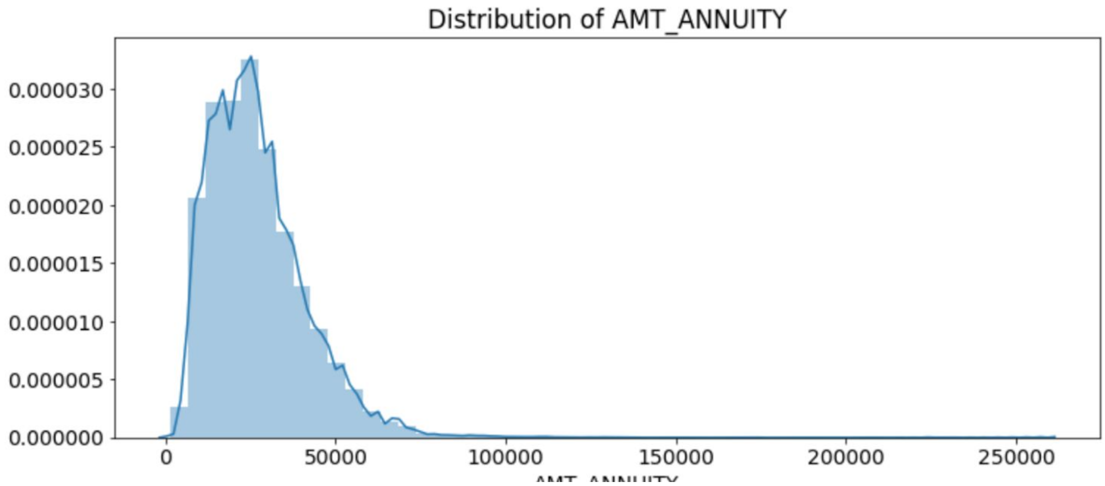
- AMT_CREDIT



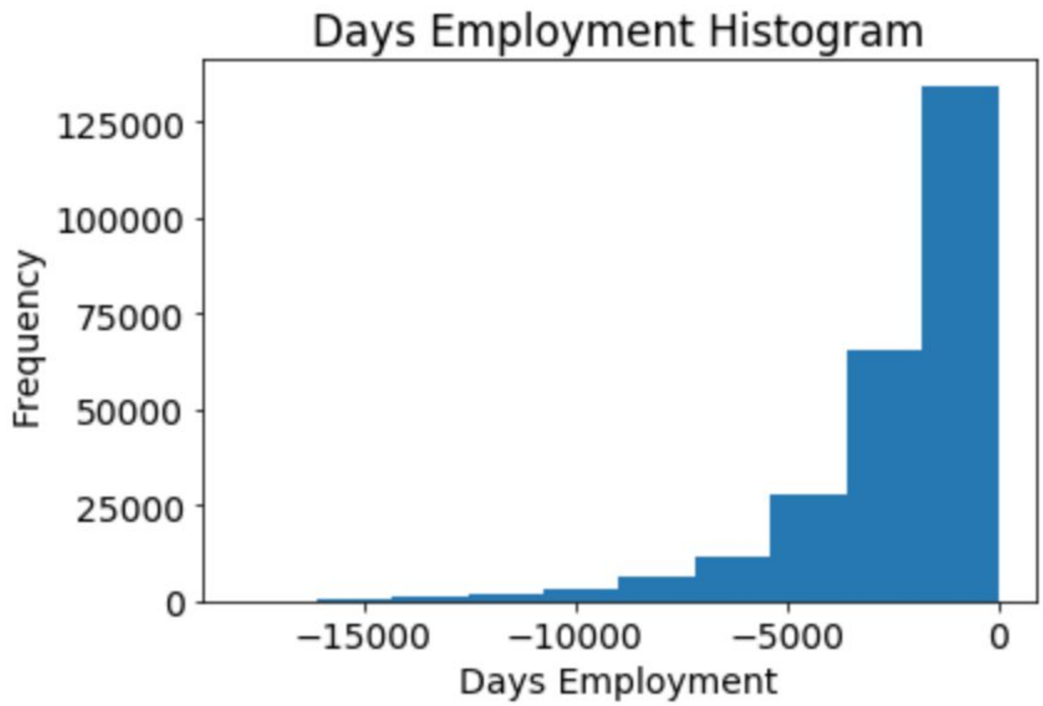
-

- AMT_ANNUITY

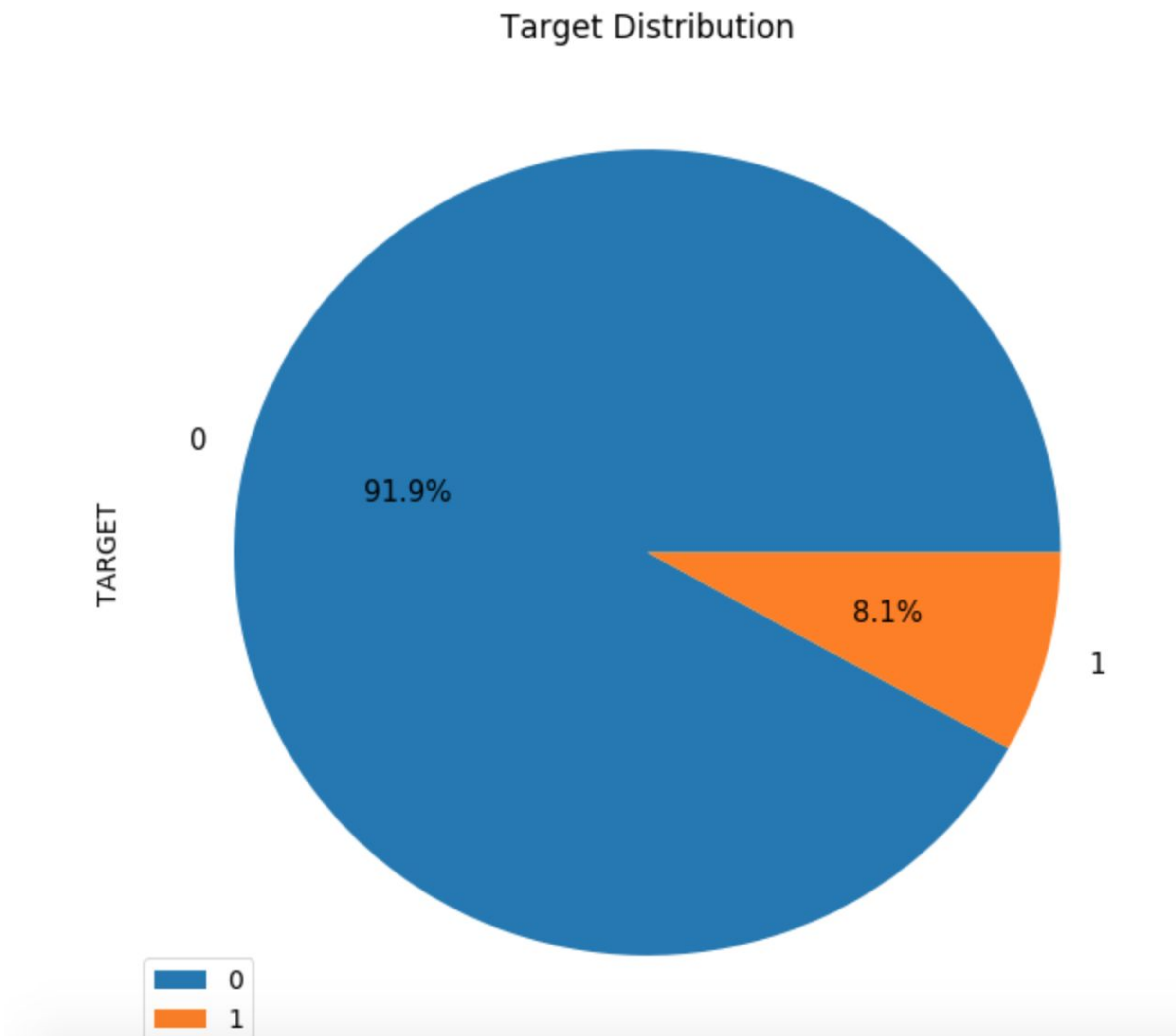
```
numeric("AMT_ANNUITY")
```



2. The ‘Days Employment’ distribution looks to be much more in line with what we would expect, and we also have created a new column to tell the model that these values were originally anomalous (because we will have to fill in the nans with some value, probably the median of the column). The other columns with DAYS in the dataframe look to be about what we expect with no obvious outliers. As an extremely important note, anything we do to the training data we also have to do to the testing data. Let's make sure to create the new column and fill in the existing column with np.nan in the testing data. The distribution looks to be much more in line with what we would expect, and we also have created a new column to tell the model that these values were originally anomalous (because we will have to fill in the nans with some value, probably the median of the column). The other columns with DAYS in the dataframe look to be about what we expect with no obvious outliers. As an extremely important note, anything we do to the training data we also have to do to the testing data. Let's make sure to create the new column and fill in the existing column with np.nan in the testing data.



3. Imbalance of Data



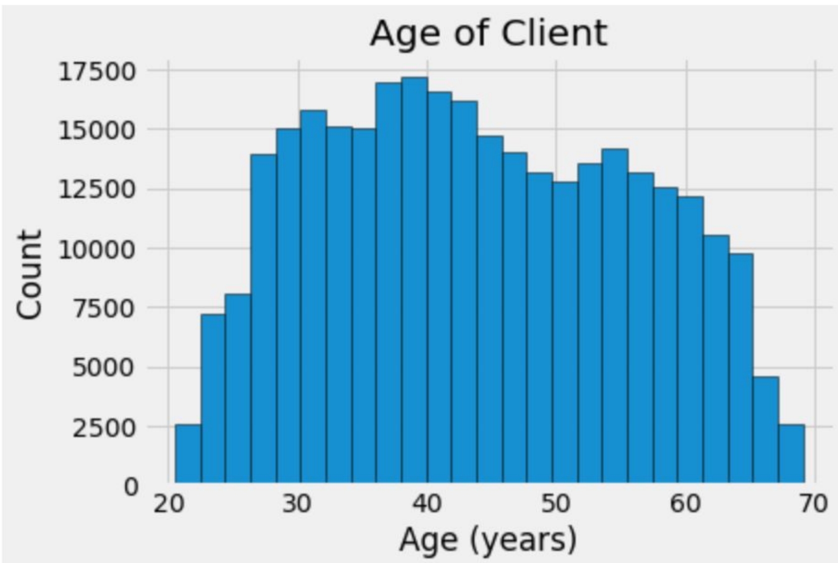
From this pie chart, we see this is an imbalanced class problem(http://www.chioka.in/class-imbalance-problem/). There are far more loans that were repaid on time than loans that were not repaid. Once we get into more sophisticated machine learning models, we can weight the classes by their representation in the data to reflect this imbalance.

4. **Effect of Age on Repayment**

By itself, the distribution of age does not tell us much other than that there are no outliers as all the ages are reasonable. To visualize the effect of the age on the target, we will next make a kernel density estimation plot (KDE) colored by the value of the target. A kernel density estimate plot shows the distribution of a single variable and can be thought of as a smoothed histogram (it is created by computing a kernel, usually a Gaussian, at each data point and then averaging all the individual kernels to develop a single smooth curve). We will use the seaborn kdeplot for this graph.

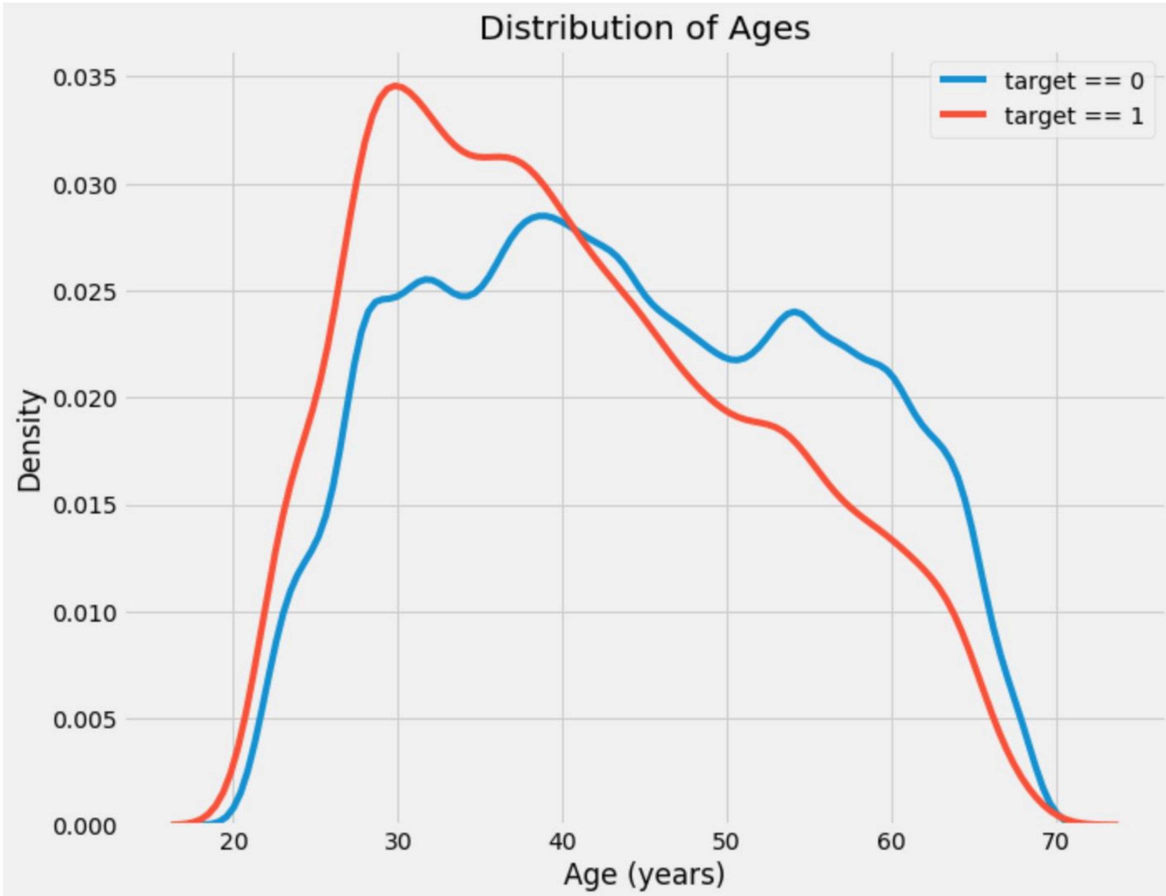
```
# Set the style of plots
plt.style.use('fivethirtyeight')

# Plot the distribution of ages in years
plt.hist(df_train['DAYS_BIRTH'] / 365, edgecolor = 'k', bins = 25)
plt.title('Age of Client'); plt.xlabel('Age (years)'); plt.ylabel('Count');
```



5. The target == 1 curve skews towards the younger end of the range. Although this is not a significant correlation (-0.07 correlation coefficient), this variable is likely going to be useful in a machine learning model because it does affect the target. Let's look at this relationship in another way: average failure to repay loans by age bracket.

To make this graph, first we cut the age category into bins of 5 years each. Then, for each bin, we calculate the average value of the target, which tells us the ratio of loans that were not repaid in each age category.

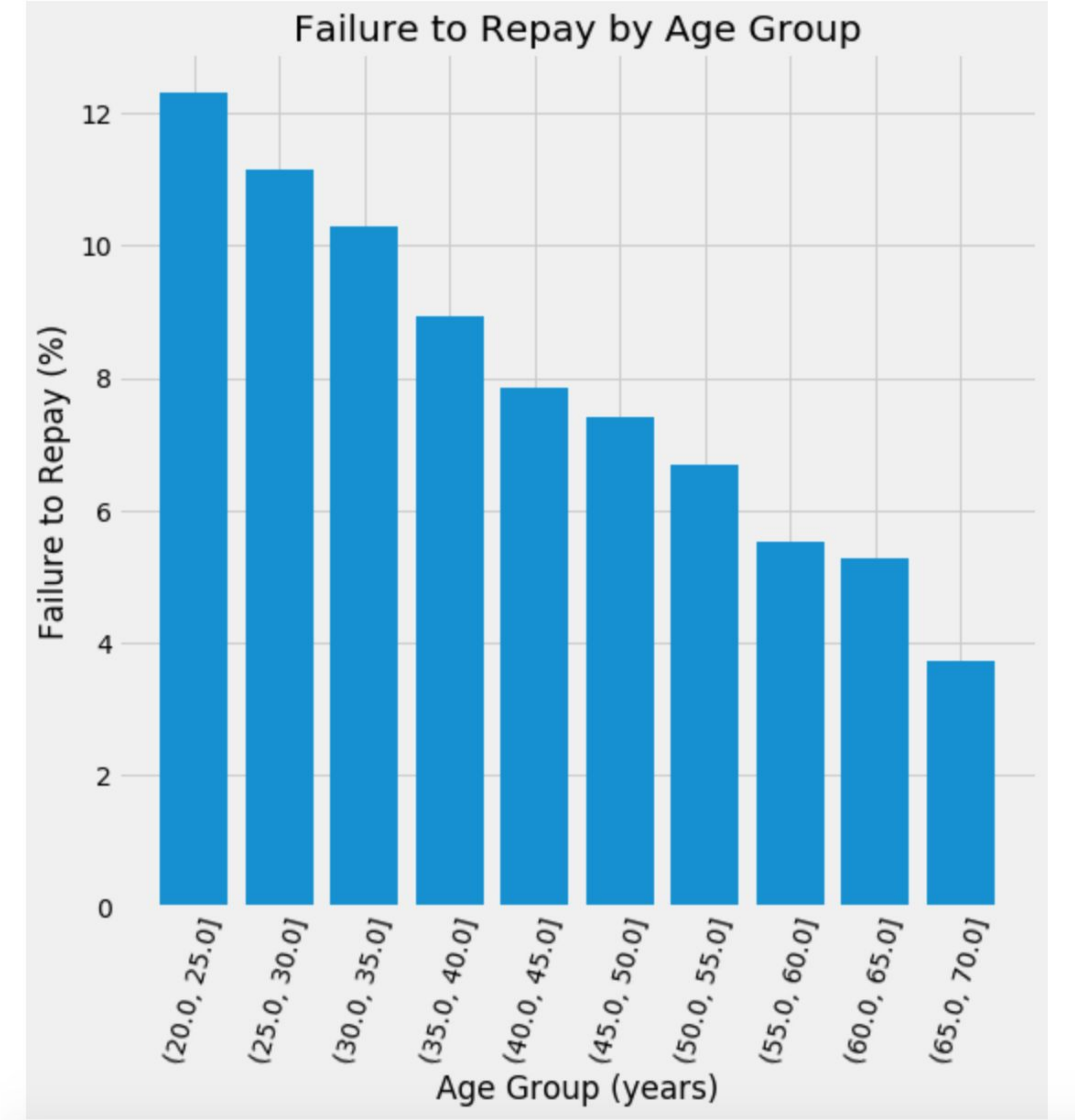


6. Age Group by the Bin & Mean Probability of the TARGET

YEARS_BINNED	TARGET	DAYS_BIRTH	YEARS_BIRTH
(20.0, 25.0]	0.123036	8532.795625	23.377522
(25.0, 30.0]	0.111436	10155.219250	27.822518
(30.0, 35.0]	0.102814	11854.848377	32.479037
(35.0, 40.0]	0.089414	13707.908253	37.555913
(40.0, 45.0]	0.078491	15497.661233	42.459346
(45.0, 50.0]	0.074171	17323.900441	47.462741
(50.0, 55.0]	0.066968	19196.494791	52.593136
(55.0, 60.0]	0.055314	20984.262742	57.491131
(60.0, 65.0]	0.052737	22780.547460	62.412459
(65.0, 70.0]	0.037270	24292.614340	66.555108

7. Failure to Repay by Age Group

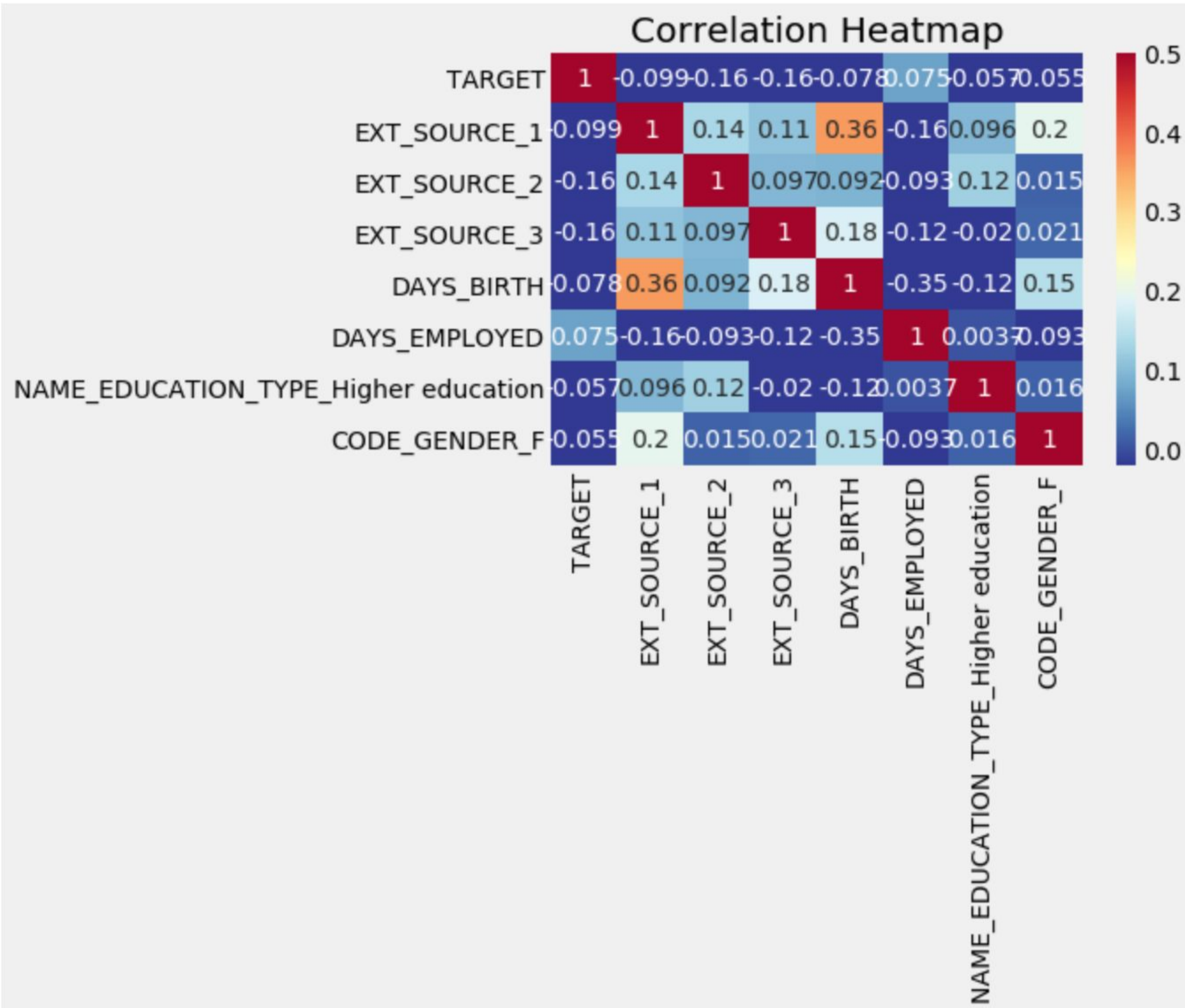
There is a clear trend: younger applicants are more likely to not repay the loan! The rate of failure to repay is above 10% for the youngest three age groups and beolow 5% for the oldest age group.
This is information that could be directly used by the bank: because younger clients are less likely to repay the loan, maybe they should be provided with more guidance or financial planning tips. This does not mean the bank should discriminate against younger clients, but it would be smart to take precautionary measures to help younger clients pay on time.



8. **Exterior Sources**

Exterior Sources The 3 variables with the strongest negative correlations with the target are EXT_SOURCE_1, EXT_SOURCE_2, and EXT_SOURCE_3. According to the documentation, these features represent a "normalized score from external data source". I'm not sure what this exactly means, but it may be a cumulative sort of credit rating made using numerous sources of data.

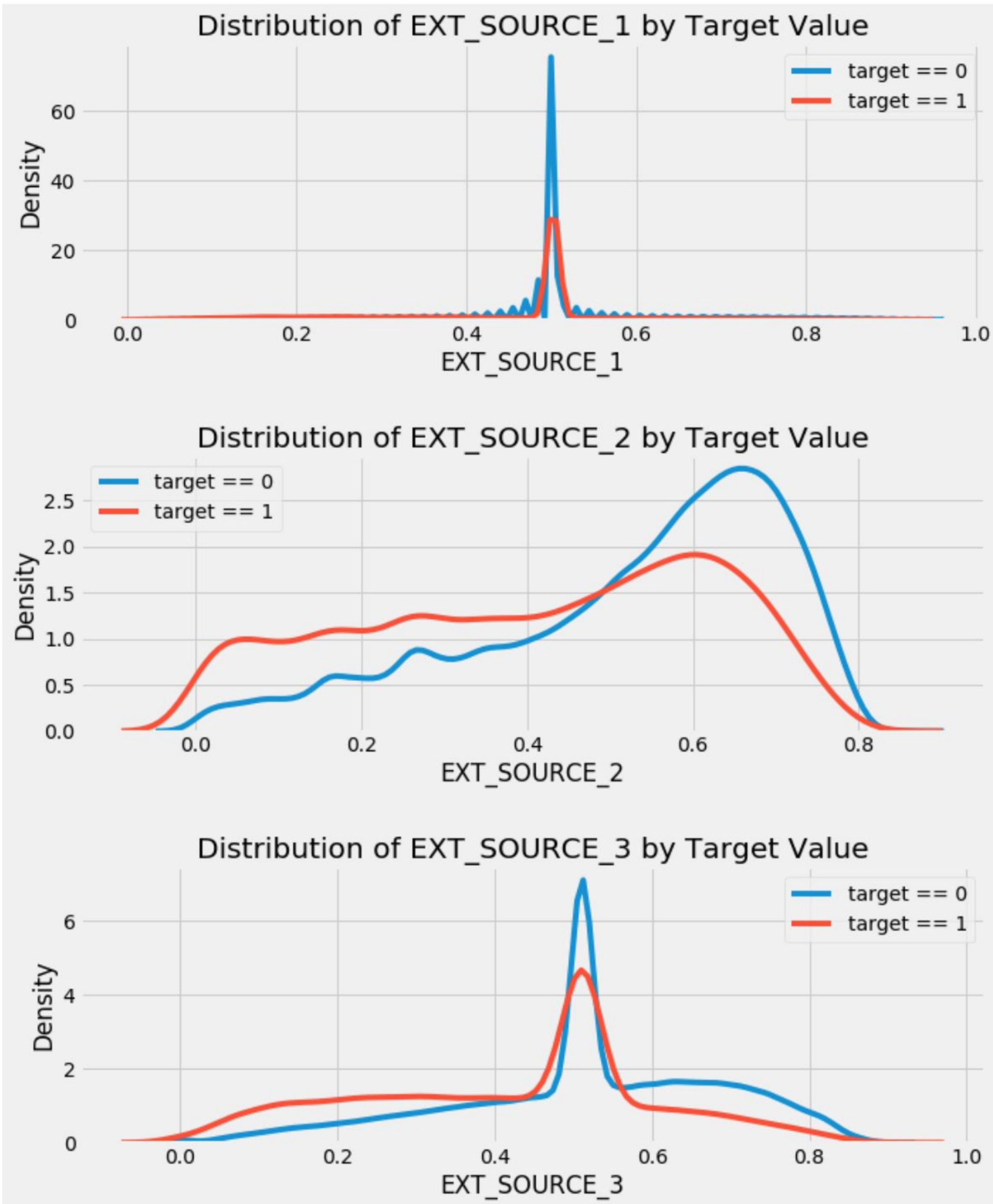
Let's take a look at these variables.
First, we can show the correlations of the EXT_SOURCE features with the target and with each other.



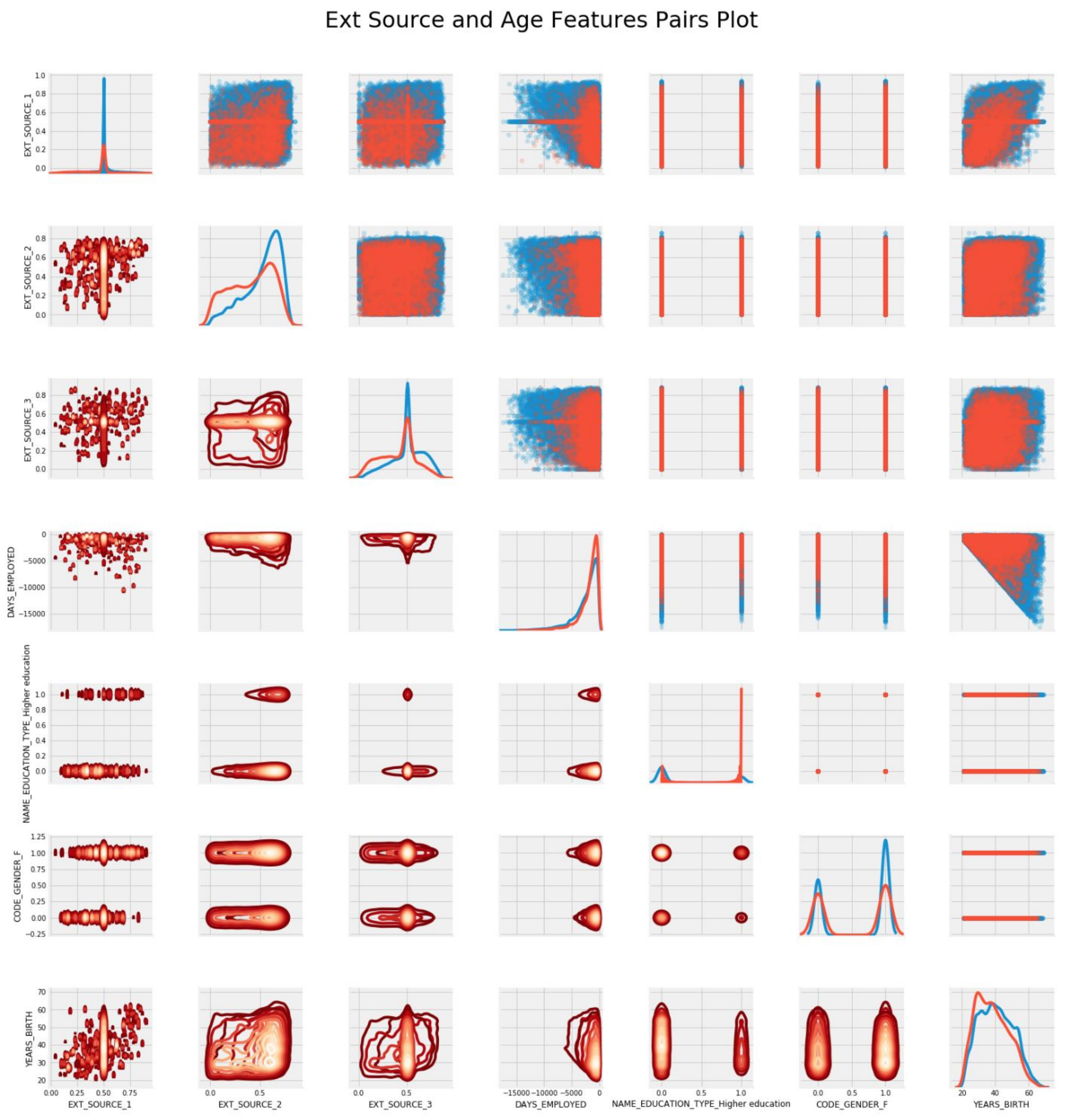
Based on statistical analysis exploring the strength of relationships between TARGET and independent variables such as age, gender, employment duration, external data source, we uncovered the following insights:

- Age distribution indicates by increasing age TARGET prediction to repay loan increases
- External Data Sources influences TARGET inversely
- Gender correlates with Target prediction

- Employment Duration

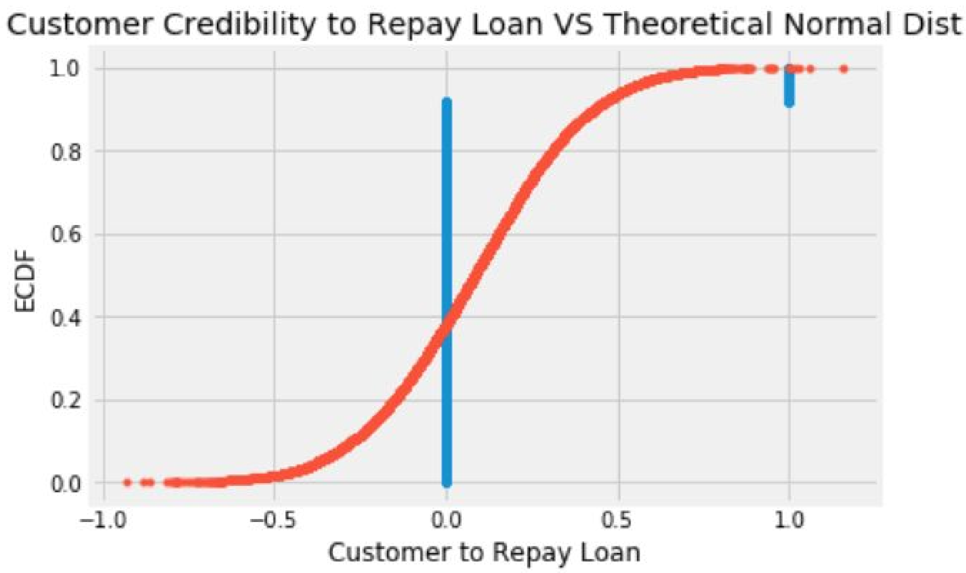


1. Pairs Plot
   As a final exploratory plot, we can make a pairs plot of the EXT_SOURCE variables and the DAYS_BIRTH variable. The Pairs Plot is a great exploration tool because it lets us see relationships between multiple pairs of variables as well as distributions of single variables. Here we are using the seaborn visualization library and the PairGrid function to create a Pairs Plot with scatterplots on the upper triangle, histograms on the diagonal, and 2D kernel density plots and correlation coefficients on the lower triangle.



In this plot, the red indicates loans that were not repaid and the blue are loans that are paid. We can see the different relationships within the data. There does appear to be a moderate positive linear relationship between the EXT_SOURCE_1 and the DAYS_BIRTH (or equivalently YEARS_BIRTH), indicating that this feature may take into account the age of the client.

2. Empirical Continuous Distribution ECDF plot for Credit Loan Default Risk



Compare the distribution of the data to the theoretical distribution of the data. This is done by comparing the ecdf First define a function for computing the ecdf from a data set. Next use np.random.normal to sample the theoretical normal distribution and overlay the ecdf of both data sets to compare distribution. Since theoretical ECDF is continuous curve while real data set is contiguous bar for 0 & 1 since it's classification problem but we may consider any data points closer to value '0' indicates 'will repay loan on time', 1 (will have difficulty repaying loan)

3. Checked Variance, Covariance, Standard Deviation, Pearson Correlation Coefficent

Variance

```
np.var(df_train['TARGET'])
```

```
0.0742116771655796
```

Standard Deviation

```
np.std(df_train['TARGET'])
```

```
0.27241820270602257
```

Covariance

```
np.cov(df_train['TARGET'], df_train['EXT_SOURCE_1'])
```

```
array([[ 0.07421192, -0.0037652 ],
       [-0.0037652 ,  0.01943096]])
```

```
np.cov(df_train['TARGET'], df_train['DAYS_BIRTH'])
```

```
array([[ 7.42119185e-02, -9.30133834e+01],
       [-9.30133834e+01,  1.90443968e+07]])
```

Pearson Correlation Coeffient

```
np.corrcoef(df_train['TARGET'], df_train['DAYS_BIRTH'])
```

```
array([[ 1.        , -0.07823931],
       [-0.07823931,  1.        ]])
```

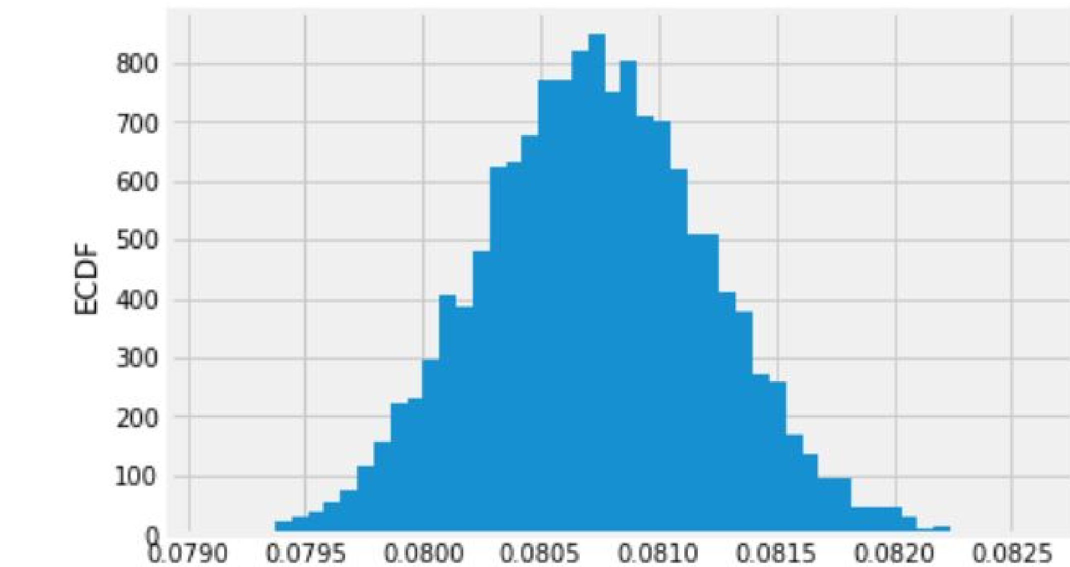
4. Confidence Interval

Assuming 95% Confidence interval i.e. give the 2.5th and 97.5th percentile of bootstrap replicates is stored as bs_replicates

np.percentile(bs_replicates, [2.5, 97.5])
O/p: array([0.07979544, 0.08170765])

Verifying it with histogram for bootstrap replicates

```
0.0004912536560492155
0.0004897216997795728
```



This is bootstrap estimate of the probability distribution function of the mean of 'Credit Loan Default Risk' at the Home Credit Group. Remember, we are estimating the mean 'Credit Loan Default Risk' we would get if the Home Credit Group could repeat all of the measurements over and over again. This is a probabilistic estimate of the mean. I plot the PDF as a histogram, and I see that it is not Normal as it has slightly longer right tail.

In fact, it can be shown theoretically that under not-too-restrictive conditions, the value of the mean will always be Normally distributed. (This does not hold in general, just for the mean and a few other statistics.) The standard deviation of this distribution, called the standard error of the mean, or SEM, is given by the standard deviation of the data divided by the square root of the number of data points. I.e., for a data set. Notice that the SEM we got from the known expression and the bootstrap replicates is the same and the distribution of the bootstrap replicates of the mean is Normal.

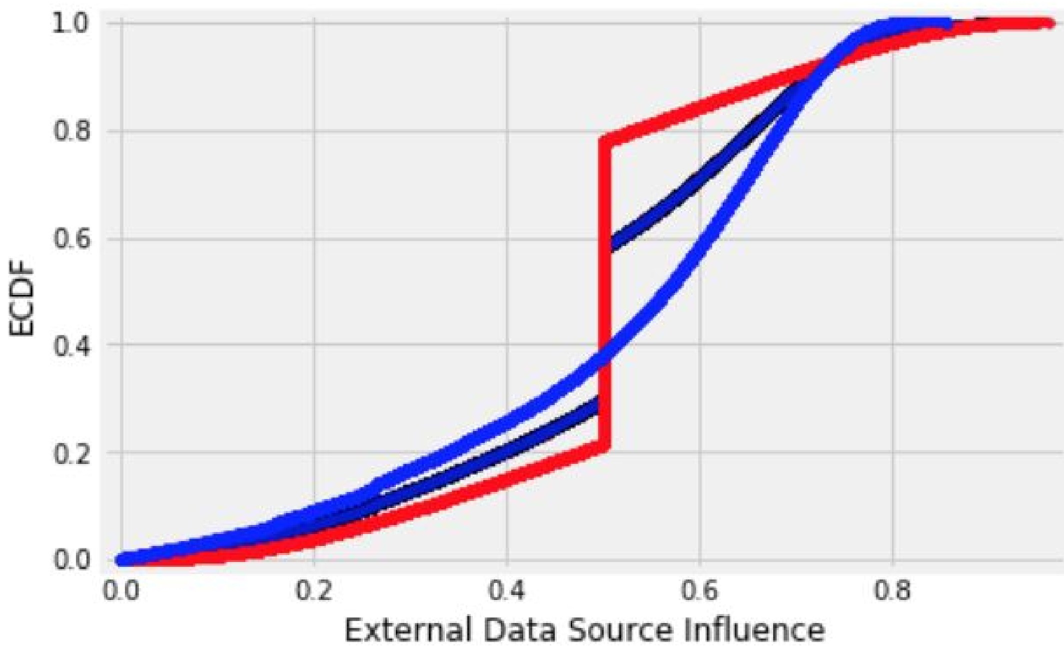
5. Hypothesis Testing

Hypothesis Testing

Null Hypothesis- There is no significant difference between EXT_SOURCE_1 and EXT_SOURCE_2 mean on 'Ability to Repay Loan'

H0: $\mu EXT\_SOURCE\_1 - \mu EXT\_SOURCE\_2 = 0$ Significance Level: 95% Confidence $\alpha=0.05$
Alternate Hypothesis - There is significant difference between EXT_SOURCE_1 and EXT_SOURCE_2 mean on 'Ability to Repay Loan'

HA : $\mu EXT\_SOURCE\_1 - \mu EXT\_SOURCE\_2 \neq 0$



Permutation samples ECDFs overlap and give a purple haze. Few of the ECDFs from the permutation samples overlap with the observed External Source Data1 data towards right of the graph & even fewer overlap towards left, suggesting that the hypothesis is not commensurate with the data. External Source Data1 & External Source Data2 are not identically distributed and do not influence data in similar way. So Null Hypothesis is rejected.

Here is the link to iPython Notebook:
https://github.com/rashi-n/Machine-Learning-Projects/blob/master/Capstone%20Projects/CS%20I%20-%20EDA%20%26%20Inferential%20Statistics.ipynb