**Home Loan Credibility Assessment**

**Data Acquisition:**

The loan application data, which is provided by [Home Credit](http://www.homecredit.net/about-us.aspx), is used to predict whether or not an applicant will be able to repay a loan. The data is available on <https://www.kaggle.com/c/home-credit-default-risk/data>.

The data is acquired from seven different sources. The first dataset, application\_train/ application\_test, is the main training and testing data with information about each loan application. Each row is identified by the feature SK\_ID\_CURR. The TARGET feature in the training data represents load repaid by 0 and not repaid by 1. The second data source bureau provides the client's previous credits from other financial institutions. The third one is bureau\_balance, which provides monthly balances of previous credits in Credit Bureau. POS\_CASH\_balance provides monthly balance snapshots of the previous point of sales and cash loans that the applicant had with Home Credit. The fifth data source credit\_card\_balance presents the monthly balance snapshots of previous credit cards that the applicant has with Home Credit. All previous applications for Home Credit loans of clients who have loans are mentioned in the previous\_application. Repayment history for the previous loans is provided in the seventh data source installments\_payment.

**Missing Values:**

The columns with missing values will be marked and the number of missing values will be counted for each columns using df.isnull().sum(). The missing values can be dropped by df.dropna(axis = 0, how = 'any').

**Data exploration:**

The columns are explored using df.dtypes command and all the columns that have object data types will be examined and the number of unique classes in each object column will be studied using df.select\_dtypes('object').apply(pd.Series.nunique). This code will list the categorical variables and their unique entries. The unique values can be assessed by df.nunique() command and it can be confirmed that SK\_ID\_CURR is unique for application\_train data.

**Anomalies:**

The DAYS\_EMPLOYED column of application\_train data contains anomalies in it, which can be found using the code df['DAYS\_EMPLOYED']/365).describe()). It produces the following result:

count 307511.000000

mean 174.835742

std 387.056895

min -49.073973

25% -7.561644

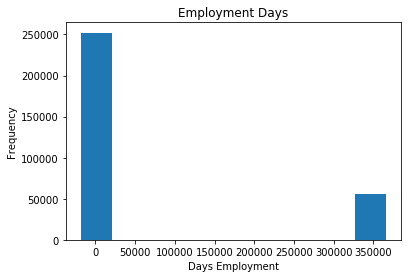
50% -3.323288

75% -0.791781

max 1000.665753

Name: DAYS\_EMPLOYED, dtype: float64

One can observe from the result that the max employed value is 1000 years. The can be viewed using the command train\_data['DAYS\_EMPLOYED'].plot.hist(title = 'Employment Days'), which yields the following plot:



The anomalies can be replaced by nan values using np.nan. The code train\_data['DAYS\_EMPLOYED'].replace({365243: np.nan}, inplace = True). This produces a better distribution of the DAYS\_EMPLOYED column data.

