Table of Contents

Problem Statement:

Predict the interest rate on the loan given pertaining parameters related to loan. Build machine learning/statistical models in python to predict the interest rate assigned to a loan.

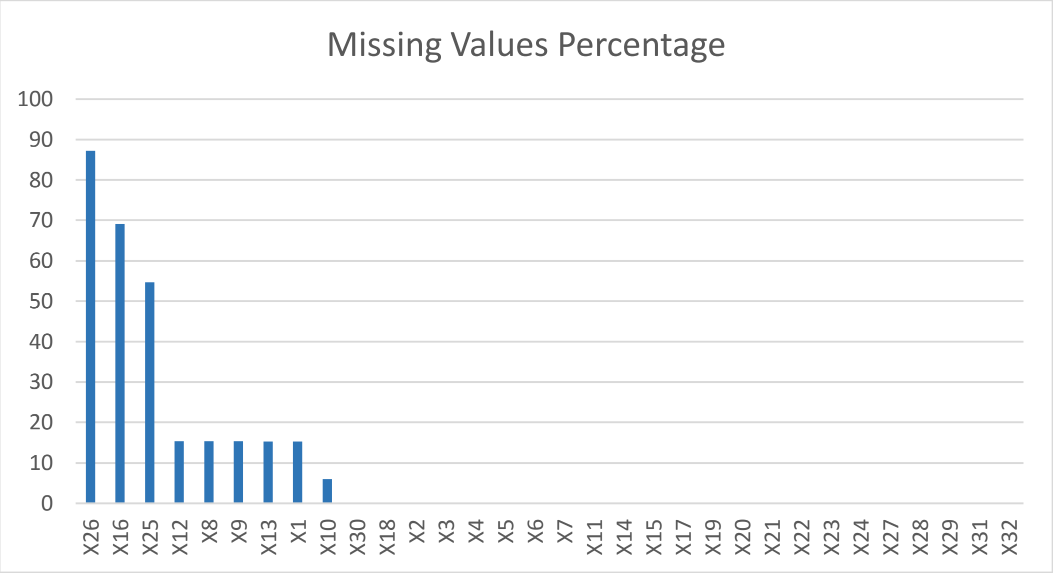
Data Preprocessing:

Data for cleaning & Modelling.csv consists 0.4 Million records and 32 variables. Performed the following preprocessing steps on required variables:

- Variables X1, X4, X5, X6, X7, X21 and X30 are converted from string to float using relevant regular expressions.
- Variable X11 – Number of years employed, levels n/a and < 1 year are merged as 0 years of experience and the rest from 1 to 10 years
- Variables X2, X3 are dropped because they represent ID of the loan
- Employer Title X10 is dropped because of too many categories
- Reason for loan provided by borrower, loan title X16 and X18 are dropped as they are representing free flow text.
    - Can extract features from the text using Latent Semantic Indexing
- Zip code - X19 is dropped because interest rate does not have impact on outcome
- Variable X26 months since last public record is dropped due to 80% missing values

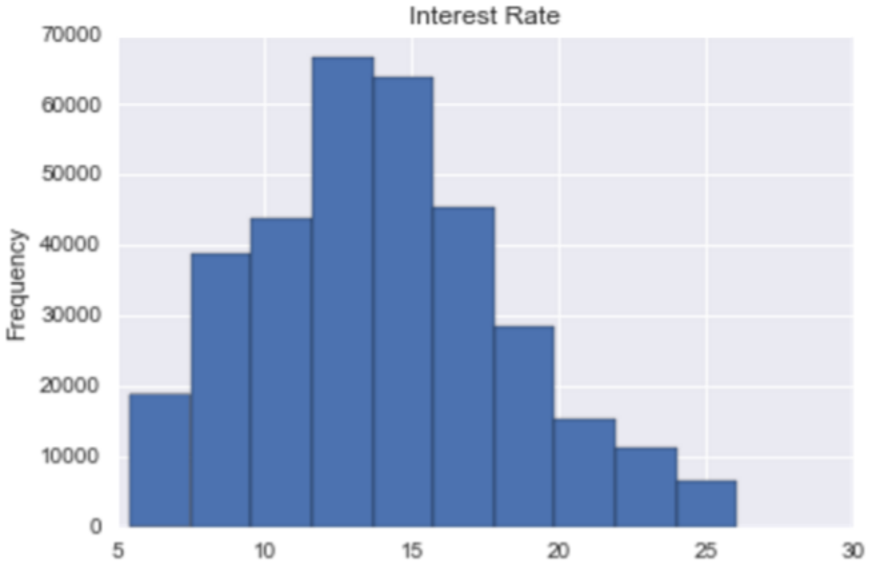
EDA:

Missing Value Treatment:



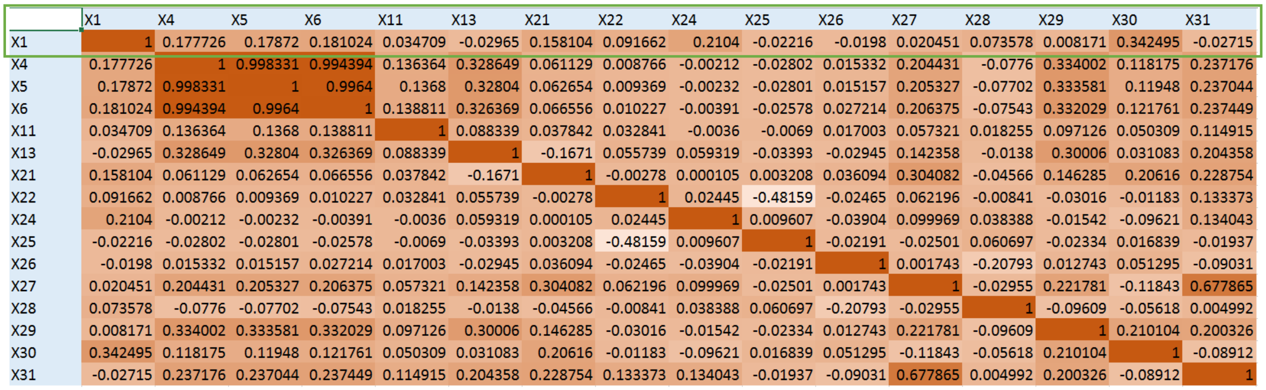
- Missing values in the Target variable Interest rate on the loan are dropped because imputing it may cause bias in the prediction
- Missing values are Imputed using central statistics - replaced with mode for categorical variable and median value for numeric variable.
  - Missing values can also be imputed using machine learning models like KNN – Imputation which imputes the missing values comparing the nearest neighbors but the required package was not available in python
  - Can apply statistical models like linear regression if the relationship between the variables is known but it may overfit in some cases

Outlier Analysis



- The distribution of interest rate has a near normal distribution hence need not transform the variable. There are no values above 1.5 * Quartile3 from the boxplot hence there are no outliers

Correlation Matrix



	X1	X4	X5	X6	X11	X13	X21	X22	X24	X25	X26	X27	X28	X29	X30	X31
X1	1	0.177726	0.17872	0.181024	0.034709	-0.02965	0.158104	0.091662	0.2104	-0.02216	-0.0198	0.020451	0.073578	0.008171	0.342495	-0.02715
X4	0.177726	1	0.998331	0.994394	0.136364	0.328649	0.061129	0.008766	-0.00212	-0.02802	0.015332	0.204431	-0.0776	0.334002	0.118175	0.237176
X5	0.17872	0.998331	1	0.9964	0.1368	0.32804	0.062654	0.009369	-0.00232	-0.02801	0.015157	0.205327	-0.07702	0.333581	0.11948	0.237044
X6	0.181024	0.994394	0.9964	1	0.138811	0.326369	0.066556	0.010227	-0.00391	-0.02578	0.027214	0.206375	-0.07543	0.332029	0.121761	0.237449
X11	0.034709	0.136364	0.1368	0.138811	1	0.088339	0.037842	0.032841	-0.0036	-0.0069	0.017003	0.057321	0.018255	0.097126	0.050309	0.114915
X13	-0.02965	0.328649	0.32804	0.326369	0.088339	1	-0.1671	0.055739	0.059319	-0.03393	-0.02945	0.142358	-0.0138	0.30006	0.031083	0.204358
X21	0.158104	0.061129	0.062654	0.066556	0.037842	-0.1671	1	-0.00278	0.000105	0.003208	0.036094	0.304082	-0.04566	0.146285	0.20616	0.228754
X22	0.091662	0.008766	0.009369	0.010227	0.032841	0.055739	-0.00278	1	0.02445	-0.48159	-0.02465	0.062196	-0.00841	-0.03016	-0.01183	0.133373
X24	0.2104	-0.00212	-0.00232	-0.00391	-0.0036	0.059319	0.000105	0.02445	1	0.009607	-0.03904	0.099969	0.038388	-0.01542	-0.09621	0.134043
X25	-0.02216	-0.02802	-0.02801	-0.02578	-0.0069	-0.03393	0.003208	-0.48159	0.009607	1	-0.02191	-0.02501	0.060697	-0.02334	0.016839	-0.01937
X26	-0.0198	0.015332	0.015157	0.027214	0.017003	-0.02945	0.036094	-0.02465	-0.03904	-0.02191	1	0.001743	-0.20793	0.012743	0.051295	-0.09031
X27	0.020451	0.204431	0.205327	0.206375	0.057321	0.142358	0.304082	0.062196	0.099969	-0.02501	0.001743	1	-0.02955	0.221781	-0.11843	0.677865
X28	0.073578	-0.0776	-0.07702	-0.07543	0.018255	-0.0138	-0.04566	-0.00841	0.038388	0.060697	-0.20793	-0.02955	1	-0.09609	-0.05618	0.004992
X29	0.008171	0.334002	0.333581	0.332029	0.097126	0.30006	0.146285	-0.03016	-0.01542	-0.02334	0.012743	0.221781	-0.09609	1	0.210104	0.200326
X30	0.342495	0.118175	0.11948	0.121761	0.050309	0.031083	0.20616	-0.01183	-0.09621	0.016839	0.051295	-0.11843	-0.05618	0.210104	1	-0.08912
X31	-0.02715	0.237176	0.237044	0.237449	0.114915	0.204358	0.228754	0.133373	0.134043	-0.01937	-0.09031	0.677865	0.004992	0.200326	-0.08912	1

- Most of the variables does not have high correlation with Interest rate on Loan hence linear models like linear regression would not perform better
- There is a high correlation between X4, X5 and X6 ensemble models like random forest would pick the best variable that would explain the target attribute.

Feature engineering:

- Extracted month and year from the date loan was issued and date credit line was opened as year may have significance on interest rate
- Created a variable Income percentage – (AmountFunded) / (Income of Borrower) because that ratio is directly proportional to interest rate
- Created a variable Funding percentage – (InvestorFunding)/ (Loan Amount Funded) because investor funding has high intuitive impact on Interest rate
- Replaced variable X25 missing values as -999 since number since the borrower's last delinquency has a high impact on the interest rate

Modeling:

Pros and cons – Random Forest

Random Forest:

It is an ensemble of decision trees using the bagging technique with random feature selection to add additional diversity to the decision tree models. After the ensemble of trees is generated, the model uses a vote/average to combine the trees' predictions depending on whether it is a classification or regression problem.

Pros:

1. Tree based models easy to interpret as rules. Can handle noisy or missing data as well as categorical or continuous features
2. Automatically determines the feature importance based on Gini Index and does feature selection
3. Can handle large number of features or data because bagging handles very well high dimensional spaces as well as large number of training examples.
4. No need to input linear features or even features that interact linearly.

Cons:

1. It is not explicable model like decision tree. However, you can generate the if-then rules and make it interpretable but its inherently not explicable.
2. Randomization of features will increase the bias. However, it will decrease the variance because it is averaging the outputs from different trees. Hence, the need to tune the parameters is essential which will take time.

Pros and Cons -- Gradient Boosting Machine(GBM)

GBM builds multiple trees with each tree minimizing the errors made by the previous tree.
It combines a set of weak learners and improves the accuracy. At any instant t, the model outcomes are weighed based on the outcomes of previous instant t-1. The outcomes which are close to the actual value will be given less weight compared to the ones where the error is more.

Pros:

1. Boosting deals with bias variance trade-off. Unlike bagging algorithms, which only controls for high variance in a model, boosting controls both bias and variance.
2. Can specify different loss function such as least square regression. It is fairly robust to overfitting based on the parameters selection
3. Tree based model, possible to extract if-then rules similar to random forest. Model can be parallelized

Cons:

1. Harder to tune compared to random forest, because of many hyper parameters
2. Prone to over fitting. However, there are strategies to overcome same and build more generalized trees using a combination of parameters like learning rate and depth of tree. Generally, the two parameters are kept on the lower side to allow for slow learning and better generalization.
3. We can get the variable importance similar to random forest, but it is not possible to determine how the variables interact and contribute to the final prediction.

Results:

RMSE on the train and test splits for the model:

- Random Forest (RMSE – 1.2516)
- Gradient Boosting Machine (RMSE – 1.5427)

Top Five Important Features

- Loan Subgrade
- Date Loan was issued
- Number of Payments
- Revolving Line Utilization Rate
- Number of inquiries by creditors during past six months

Future Work:

1. More features can be created using advanced models like deep neural networks
2. Build regression models by converting the relationship into linear by performing various transformation techniques
3. Extract features from Text data using natural language processing
4. Ensembles of many more models: stacking or blending.
5. Tune the parameters using grid search technique.