DATA ANALYTICS CASE REPORT

BANKING INDUSTRY

Submitted by Group 6 Sec A

Deepak Namdev

Sourabh Mahajan

Manas Bhageria

Preeti

Rajat Agrawal

GUIDED BY PROF. MAHIMA GUPTA

TABLE OF CONTENTS

Case Study on Likelihood Of An Individual's Success In Paying Back The Loan

PROBLEM STATEMENT

Non-performing assets (NPAs) at commercial banks amounted to ₹10.3 trillion, or 11.2% of advances, in March 2018. Public sector banks (PSBs) accounted for ₹8.9 trillion, or 86%, of the total NPAs. The ratio of gross NPA to advances in PSBs was 14.6%. These are levels typically associated with a banking crisis. In 2007-08, NPAs totaled ₹566 billion (a little over half a trillion), or 2.26% of gross advances. The increase in NPAs since then has been staggering.

We are devising a model using R software to find the likelihood of a person's success in paying the loan. This would greatly benefit the banking industry as it would give them immense understanding on the likelihood of the presumed assets turning into NPA.

BUSINESS UNDERSTANDING AND BACKGROUND

The decision of approving a loan is majorly dependent on the personal and financial background of the applicant. Precisely, age, gender, income, employment status, credit history and other attributes contributes to the approval decision. Credit Analysis involves the statistical – quantitative and qualitative measure to investigate the probability of a third party to pay back the loan to the bank on time and predict its default characteristic. Analysis focus on recognizing, assessing and reducing the financial/other risks involved which may otherwise result in the losses incurred by the company while lending. The risk can be business loss by not approving the good candidate or can be financial loss by approving the candidate who is at bad risk. It is very important to manage credit risk and handle challenges efficiently for credit decision as it can have adverse effects on credit management. Therefore, evaluation of credit approval is significant before jumping to any granting decision.

DATA UNDERSTANDING PHASE

Our analysis and model will be based on 19 variables which are shown in Table 2. These variables are grouped into different categories: application category (variables that contain data which each borrower provides during application process such as income, home ownership, etc) and behavioral category (variables that describe borrower's behavior such as grade, open credit lines, number of derogatory public records, account balance, etc).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Loan ID	Customer ID	Loan Status	Current Loan Amount (INR)	Term	Credit Score	Annual Income (INR)	Years in current job	Home Ownership	Monthly Debt	Months since last delinquent	Years of Credit History	Number of Open Accounts	Number of Credit Problems	Current Credit Balance	Maximum Open Credit	Bankruptcies	Tax Liens
2	77598f7b-	e777faab-98ae-	Fully Paid	347666	Long Term	721	806949	3	Own Home	8741.9	NA	12	9	0	256329	386958	0	0
3	11653c24-	d377d2ea-5cf8-	Fully Paid	163966	Short Term	678	719910	9	Home	12778.26	8	6.4	9	1	66025	138248	0	1
4	5bc5f041-	ac460fac-928b-	Fully Paid	392282	Long Term	688	974662	8	Home	10396.42	29	12	11	0	35663	242946	0	0
5	e-461c-bf95-	36-487a-8e4f-8a	Fully Paid	453464	Short Term	712	895147	3	Rent	17007.85	NA	14.2	12	1	137845	222926	1	0
6	29cf8611-	5df79973-ce71-	Fully Paid	132792	Short Term	751	668990	4	Rent	6132.25	NA	14.7	5	0	61199	214742	0	0
7	e67fc85b-	fc5cff9c-c6b5-	Charged Off	119504	Short Term	745	938315	2	Home	11807.17	NA	13	11	0	32300	104170	0	0
8	9cca9017-	f0d6b352-bcf0-	Fully Paid	280588	Short Term	717	671080	3	Rent	17447.89	10	10	10	1	168169	470360	1	0
9	76fa89b9-	9d42ab3f-ccf7-	Fully Paid	340604	Long Term	618	928701	10	Home	21205.52	8	14.4	5	0	291137	368808	0	0
10	856c1ab9-	3edb6bb1-2045-	Charged Off	109802	Short Term	745	474069	0	Rent	1497.39	33	11	2	0	91048	186604	0	0
11	59d28b2c-	7a826762-3889-	Fully Paid	218988	Short Term	740	775409	4	Home	8141.88	NA	14.9	5	0	100206	186230	0	0
12	-4188-b468-	0df-4a68-bec9-00	Fully Paid	133078	Short Term	709	804460	0	Rent	9117.34	76	12.5	10	0	111568	243760	0	0
13	8cc6d0be-	36096b3d-97e7-	Fully Paid	54076	Short Term	744	485697	1	Rent	2655.06	NA	9	6	0	19988	282260	0	0
14	01b645c9-	5e02406a-3cd6-	Fully Paid	348832	Long Term	704	497306	0	Rent	3257.36	25	13	4	0	90022	167860	0	0
15	e8283bff-	6ecfe8c1-8b24-	Fully Paid	214786	Short Term	723	883329	9	Home	11924.97	NA	14.3	5	0	154755	193314	0	0
16	5c2cc96f-	01246538-e5a4-	Fully Paid	109538	Short Term	697	567606	7	Home	5770.68	46	14.3	10	0	86716	151206	0	0
17	932aa46f-	12be2338-32c8-	Fully Paid	190498	Short Term	706	892164	1	Rent	8996.12	NA	13.2	6	0	88160	117744	0	0
18	580b6472-	68b77d5b-94b9-	Fully Paid	151954	Short Term	707	562419	7	Rent	14341.77	NA	12.5	9	1	107692	219142	0	1
19	32200a7e-	64dcd3aa-3c82-	Fully Paid	234806	Long Term	689	866799	9	Home	3676.69	34	14.1	4	2	86051	167750	2	0
20	0eaa9af5-	51fcba2d-c634-	Charged Off	25806	Short Term	685	742976	8	Rent	6377.16	NA	7.1	5	0	8189	47432	0	0
21	191a92bd-	0e0fa488-b6cb-	Fully Paid	341352	Long Term	712	751108	0	Home	10327.83	NA	13.3	11	0	81377	110858	0	0
22	4ada45a7-	117496d1-3c6e-	Fully Paid	301114	Long Term	645	825246	0	Rent	5948.71	53	9	10	1	59888	372746	1	0
23	f657d195-	0ce26174-19c8-	Fully Paid	152790	Short Term	743	678661	10	Rent	4450.94	NA	14	5	1	119510	229086	0	0
24	af030a3c-	529f45cf-801d-	Charged Off	292292	Short Term	741	666805	2	Rent	6223.45	NA	15	7	0	81016	198352	0	0
25	7c328801-	248d929d-28d2-	Fully Paid	86724	Short Term	716	580469	4	Home	7352.62	NA	4.9	6	0	109687	182226	0	0

```
> str(LoanData)
'data.frame':   8774 obs. of  18 variables:
 $ Loan.ID                    : chr  "77598f7b-32e7-4e3b-a6e5-06ba0d98fe8a" "11653c24-aa0a-4435-9ec7-4720d2bf4da8"
 "5bc5f041-9d56-4ed0-af6a-37d12daa5c49" "10da47ac-d3ce-461c-bf95-1b9d0cac176f" ...
 $ Customer.ID                : chr  "e777faab-98ae-45af-9a86-7ce5b33b1011" "d377d2ea-5cf8-4ee2-b7ba-f5be4dbb1b11"
 "ac460fac-928b-4149-b919-69ea4eb9750f" "d110ff2c-c936-487a-8e4f-8a192bad9cd8" ...
 $ Loan.Status                : chr  "Fully Paid" "Fully Paid" "Fully Paid" "Fully Paid" ...
 $ Current.Loan.Amount        : int  347666 163966 392282 453464 132792 119504 280588 340604 109802 218988 ...
 $ Term                       : chr  "Long Term" "Short Term" "Long Term" "Short Term" ...
 $ Credit.Score               : int  721 678 688 712 751 745 717 618 745 740 ...
 $ Annual.Income              : int  806949 719910 974662 895147 668990 938315 671080 928701 474069 775409 ...
 $ Years.in.current.job       : int  3 9 8 3 4 2 3 10 0 4 ...
 $ Home.Ownership             : chr  "Own Home" "Home Mortgage" "Home Mortgage" "Rent" ...
 $ Monthly.Debt               : num  8742 12778 10396 17008 6132 ...
 $ Months.since.last.delinquent: int  NA 8 29 NA NA NA 10 8 33 NA ...
 $ Years.of.Credit.History    : num  12 6.4 12 14.2 14.7 13 10 14.4 11 14.9 ...
 $ Number.of.Open.Accounts    : int  9 9 11 12 5 11 10 5 2 5 ...
 $ Number.of.Credit.Problems  : int  0 1 0 1 0 0 1 0 0 0 ...
 $ Current.Credit.Balance     : int  256329 66025 35663 137845 61199 32300 168169 291137 91048 100206 ...
 $ Maximum.Open.Credit        : int  386958 138248 242946 222926 214742 104170 470360 368808 186604 186230 ...
 $ Bankruptcies               : int  0 0 0 1 0 0 1 0 0 0 ...
 $ Tax.Liens                  : int  0 1 0 0 0 0 0 0 0 0 ...
```

DATA PREPARATION PHASE

This consisted nine steps:

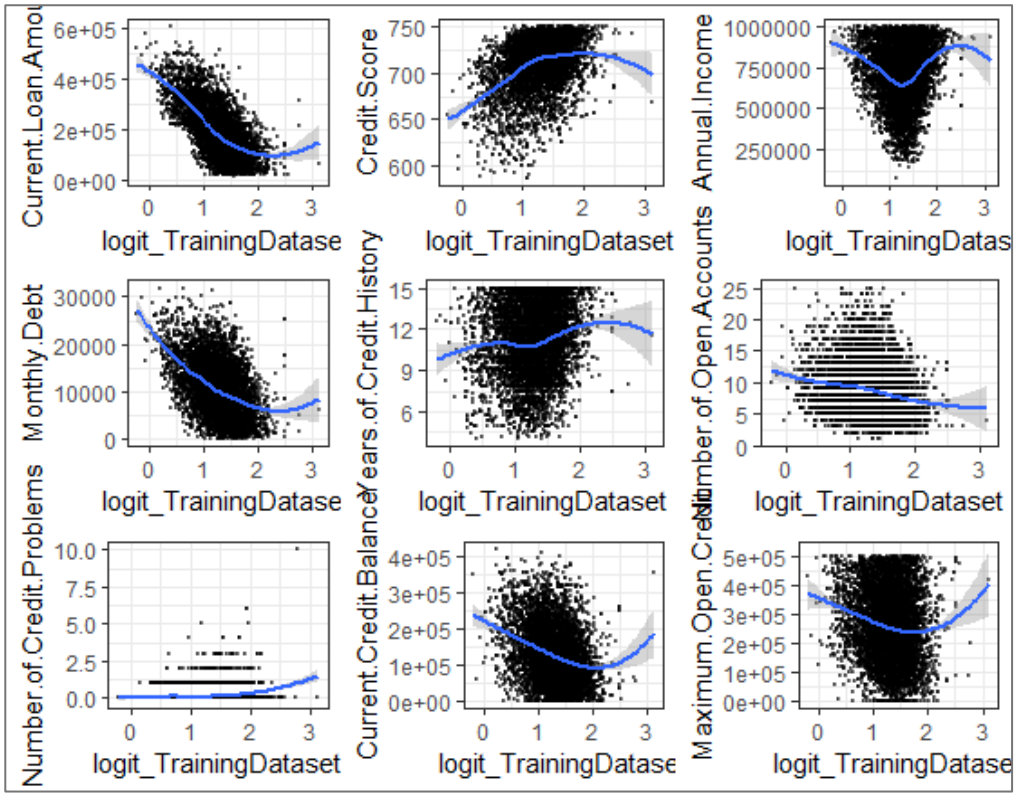
- Checking for the NA values.
- Removing the customer I'd and loan I'd variables.
- Removing the independent variable with more than 50% NA values.
- Omitting the cells with NA values.
- Converting all the categorical variable to dummy variables.
- Partioning the data into test data and train data.
- Checking the proportion of y variable in test and training data.

MODEL 1: LOGISTIC REGRESSION

Regression models generally deal with continuous values but in our case the target values in Approved are binary. Liner Regression would not work well because of its tendency to produce the result outside our binary range. Logistic Regression works well with categorical values. As it deals with probabilities, the predicted values are based on likelihood of events using logit function. LR are special part of generalized linear models and there exists linear relation between link function and predictors.

```
> lr1 = glm(TrainingDataset$Loan.Status ~.,data = TrainingDataset, family = binomial)
> summary(lr1)

Call:
glm(formula = TrainingDataset$Loan.Status ~ ., family = binomial,
    data = TrainingDataset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1765   0.5348   0.6503   0.7313   1.1310

Coefficients:
                              Estimate Std. Error z value Pr(>|z|)
(Intercept)                  9.155e-01  8.572e-01   1.068 0.285521
Current.Loan.Amount         -2.135e-06  3.845e-07  -5.551 2.83e-08 ***
Term                        -3.335e-01  8.910e-02  -3.743 0.000182 ***
Credit.Score                 6.060e-05  1.206e-03   0.050 0.959932
Annual.Income                1.129e-06  1.977e-07   5.714 1.10e-08 ***
Years.in.current.job        -2.820e-02  9.350e-03  -3.016 0.002558 **
Monthly.Debt                -2.575e-05  6.460e-06  -3.986 6.72e-05 ***
Years.of.Credit.History      2.409e-02  1.246e-02   1.934 0.053162 .
Number.of.Open.Accounts     -6.031e-03  8.588e-03  -0.702 0.482492
Number.of.Credit.Problems    3.000e-02  1.852e-01   0.162 0.871311
Current.Credit.Balance      -6.382e-07  5.780e-07  -1.104 0.269493
Maximum.Open.Credit          6.753e-07  3.710e-07   1.820 0.068721 .
Bankruptcies                 3.584e-02  2.112e-01   0.170 0.865233
Tax.Liens                    1.270e-01  2.509e-01   0.506 0.612881
Home.Ownership_HaveMortgage  1.225e+01  1.881e+02   0.065 0.948102
Home.Ownership_HomeMortgage  2.096e-01  7.492e-02   2.798 0.005138 **
Home.Ownership_OwnHome       1.063e-01  9.996e-02   1.063 0.287744
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 7405.4  on 7006  degrees of freedom
Residual deviance: 7262.3  on 6990  degrees of freedom
AIC: 7296.3

Number of Fisher Scoring iterations: 12
```

In the next step we looked for the assumptions and multicollinearity.



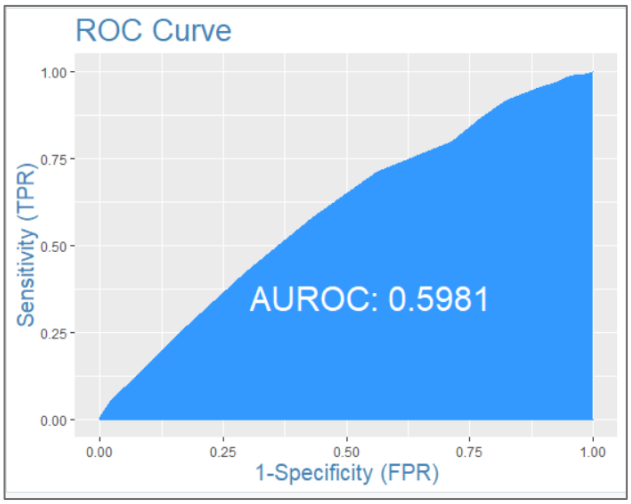
```
> vif(lr1)
       Current.Loan.Amount                       Term               Credit.Score              Annual.Income
                  1.794455                   1.510882                   1.303556                   1.566282
      Years.in.current.job                Monthly.Debt     Years.of.Credit.History     Number.of.Open.Accounts
                  1.129664                   1.588943                   1.146781                   1.314886
 Number.of.Credit.Problems      Current.Credit.Balance          Maximum.Open.Credit                Bankruptcies
                  6.205241                   2.341029                   2.362831                   4.767605
                 Tax.Liens Home.Ownership_HaveMortgage Home.Ownership_HomeMortgage      Home.Ownership_OwnHome
                  2.208132                   1.000000                   1.096294                   1.034171
```

We noticed that the number of credit problem variable showed some multicollinearity. To correct this, we did a step wise regression.

```
> summary(SignificantModel)

Call:
glm(formula = TrainingDataset$Loan.Status ~ Current.Loan.Amount +
    Annual.Income + Monthly.Debt + Term + Home.Ownership_HomeMortgage +
    Years.in.current.job, family = binomial, data = TrainingDataset)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1646   0.5425   0.6539   0.7308   1.1063

Coefficients:
                            Estimate Std. Error z value Pr(>|z|)
(Intercept)                1.225e+00  1.166e-01  10.501  < 2e-16 ***
Current.Loan.Amount       -2.021e-06  3.597e-07  -5.619 1.92e-08 ***
Annual.Income              1.159e-06  1.954e-07   5.931 3.02e-09 ***
Monthly.Debt              -2.665e-05  5.901e-06  -4.516 6.29e-06 ***
Term                      -3.405e-01  8.132e-02  -4.188 2.82e-05 ***
Home.Ownership_HomeMortgage  2.107e-01  7.343e-02   2.870  0.00411 **
Years.in.current.job      -2.155e-02  9.025e-03  -2.388  0.01693 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 7405.4  on 7006  degrees of freedom
Residual deviance: 7276.5  on 7000  degrees of freedom
AIC: 7290.5

Number of Fisher Scoring iterations: 4
```

We checked the accuracy of the model by creating the contingency table and plotting the ROC curve. The accuracy in the contingency table and ROC curve were 0.77798 and 0.5981.

```
> p_SM
             predictedValue
ActualValue     0    1
           0    0  388
           1    1 1363
> accuracy_SM
[1] 0.777968
>
```



MODEL 2: DECISION TREE

We created the decision tree by building the model on the training dataset, with control conditions: minsplit = 1 and complexity parameter = 0. Then the prediction of the classification on the testing dataset from the model was done and the contingency table (predicted and actuals values) to check the accuracy of the model was built.

```
> print(conf.matrix)
   Loan.predict
       0    1
  0  117  271
  1  211 1153
>
> #Accuracy
> (conf.matrix[1,1]+conf.matrix[2,2])/sum(conf.matrix)
[1] 0.7248858
```

To further increase the accuracy, we did the pruning by setting the cp value as the first dip we got.

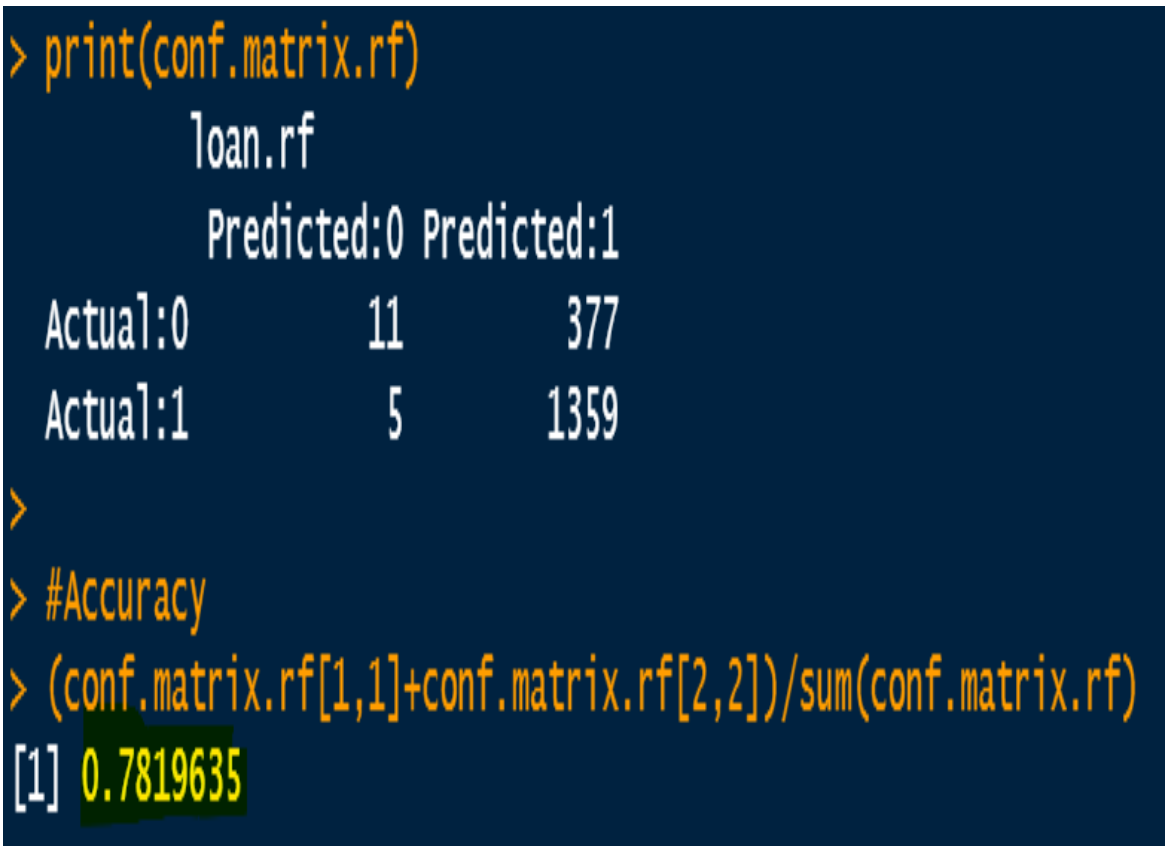
```
          CP nsplit rel error  xerror     xstd
1 0.00225806      0 1.0000000 1.00000 0.022415
2 0.00215054      8 0.9774194 0.99871 0.022405
3 0.00193548     13 0.9638710 1.00065 0.022421
4 0.00177419     19 0.9522581 1.00387 0.022446
5 0.00161290     27 0.9341935 1.02258 0.022594
```

```
> print(conf.matrix)
   prune.Loan.predict
       0    1
  0   19  369
  1   22 1342
> #Accuracy
> (conf.matrix[1,1]+conf.matrix[2,2])/sum(conf.matrix)
[1] 0.7768265
```

With this we could increase the accuracy from 0.724 to 0.7768.

MODEL 3: RANDOM FOREST

We conducted the random forest over other ensembling methods to eradicate the issue of multicollinearity. Since the number of variables were 19 and mtry would be the root of the number of the predictors, so we took 4 as our mtry and conducted the random forest. Prediction of the classification on the testing dataset from the model was performed and the contingency table (predicted and actuals values) was built to check the accuracy of the model.



The accuracy of this model was 0.7819.

THANK YOU