Data Science: Loan eligibility problem

Contents

SUBMITTED BY -

Abhinav Kumar (abhinav.kr.86@gmail.com)

Shashank Pathak (shank15pathak@gmail.om)

INTRODUCTION

Loan eligibility problem

Summary

XYZ bank deals in all home loans. They have presence across all urban, semi urban, rural areas. Customer first apply for home loan after than company validates the customer eligibility for loan.

XYZ banks wants to understand the usability of the customer details it has collected using online / offline application form. Hence, they need to automate the loan eligibility process (real time) based on customer detail provided to them. XYZ Bank signs a project with ABC Data Services to make their customer details more informative and identify the customer segments, those are eligible for loan amount so that they can specifically target those customers.

So basically, we are trying **"To understand the usability of the customer details and predicting loan approvals using statistical techniques".**

OBJECTIVE

Analyze the submitted data by XYZ Bank. Write an algorithm using the test and train data set and provide a data that should generate a sample_output.csv file.

DATA

File Name	Description	Format
test_data.csv	the test set	.csv
train_data.csv	the training set	.csv
Sample_submission.csv	a submission data set consists of loan status.	.csv

Description

Serial No.	Variable	Defination	Type (as per R)
1	Application_ID	Unique Loan ID	Integer
2	Gender	Male/Female	Factor
3	Married	Applicant Married (Y/N)	Factor
4	Dependents	No of dependents	Factor
5	Education	Applicant Education(Grad/Non Grad)	Factor
6	Self_Employed	Self Employed (Y/N)	Factor
7	ApplicantIncome	Applicant Income	Integer
8	CoapplicantIncome	Coapplicant Income	Integer
9	LoanAmount	Loan amount in thousands	Integer
10	Loan_Amount_Term	Term of loan in months	Integer
11	Credit_History	Credit history meet guidelines	Integer
12	Property_Area	Urban/Semi Urban/Rural	Factor
13	Loan_Status	Loan approved (Y/N)	Factor

TOOL'S and SOFTWARE

Loan eligibility problem

Tools -

- Frequency Table
- Logistic Regression
- Data mining classifiers: Artificial Neural Network, Decision Tree and Naïve Bayesian classifier

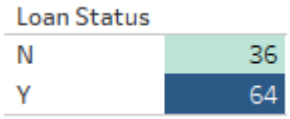
Software –

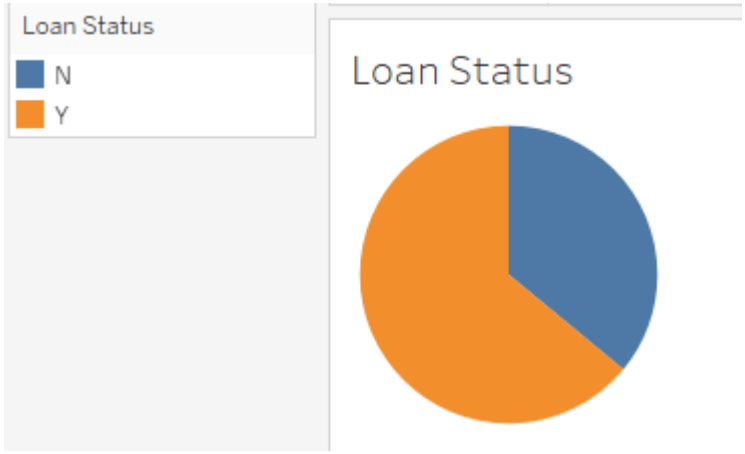
- Excel
- Tableau
- R

UNDERSTANDING VARIABLES

(i) **Loan_Status** - We are going to predict, Loan_Status which is called response or dependent variables. Rest all are predictor, explanatory, or independent variables. Loan_Status is a binay variable and takes Y or N. As per below graph, out of all the 100 applications received in train_data.csv –

- 36 % rejected
- 64 % accepted





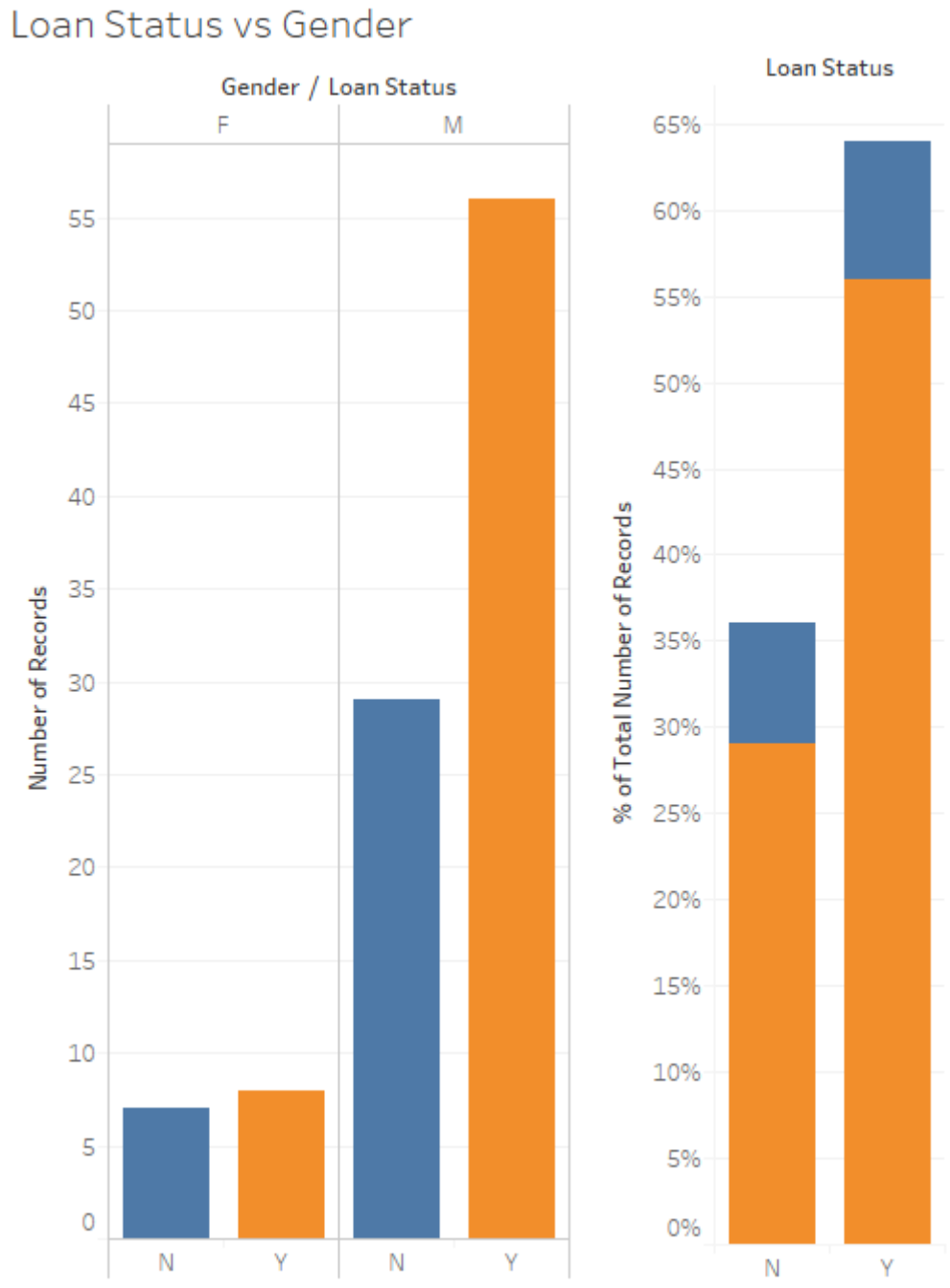
(ii) **Gender**

Loan eligibility problem

Gender	Loan Status N	Loan Status Y
F	7	8
M	29	56

Gender	Loan Status N	Loan Status Y
F	46.67%	53.33%
M	34.12%	65.88%

Gender	Loan Status N	Loan Status Y
F	19.44%	12.50%
M	80.56%	87.50%



We can clearly see that loan approval rate is higher in males as compared to females.

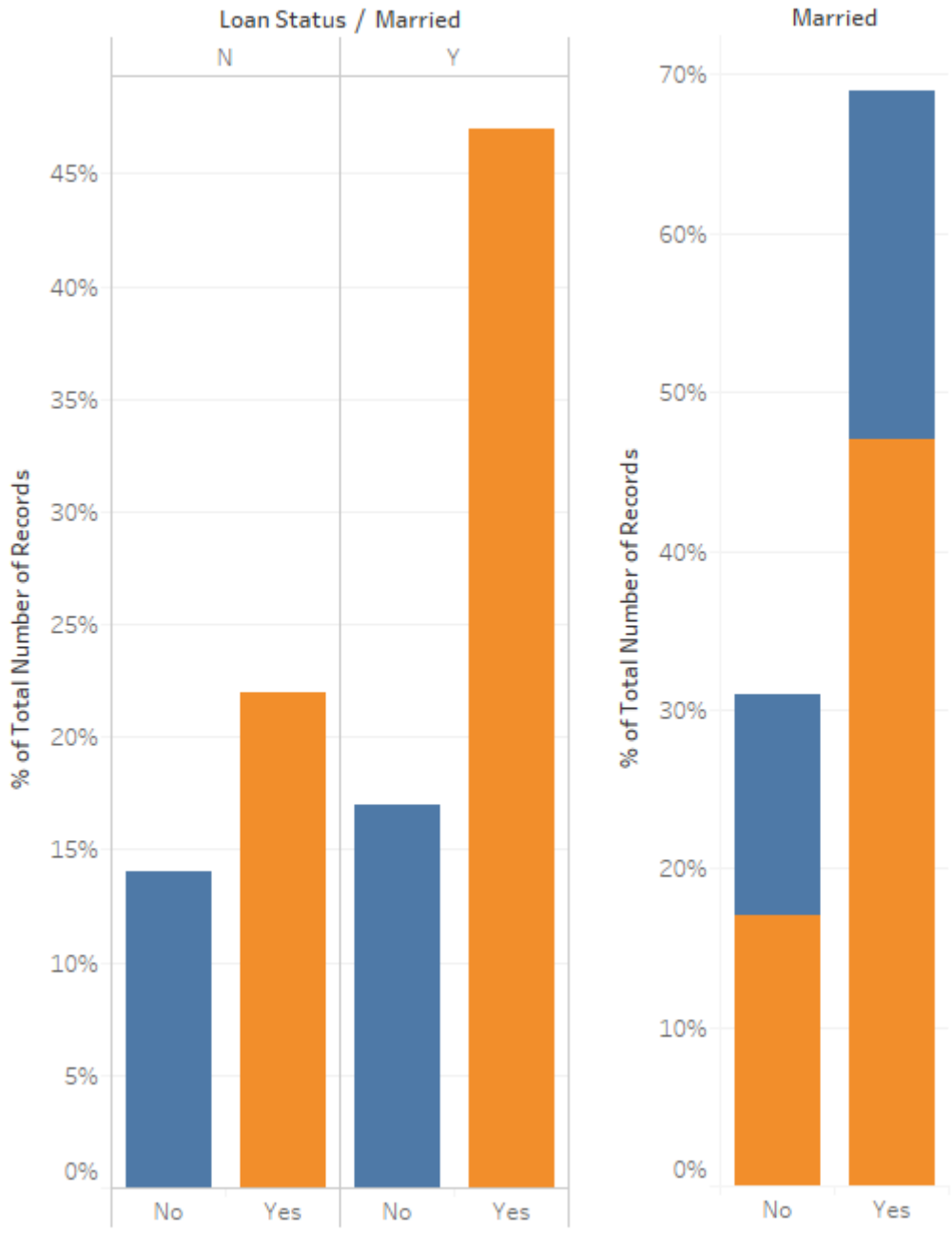
(iii) Married

Loan eligibility problem

Married	Loan Status N	Loan Status Y
No	14	17
Yes	22	47

Married	Loan Status N	Loan Status Y
No	45.16%	54.84%
Yes	31.88%	68.12%

Married	N	Y
No	38.89%	26.56%
Yes	61.11%	73.44%



We can clearly see, loan approval rate is higher for married people.

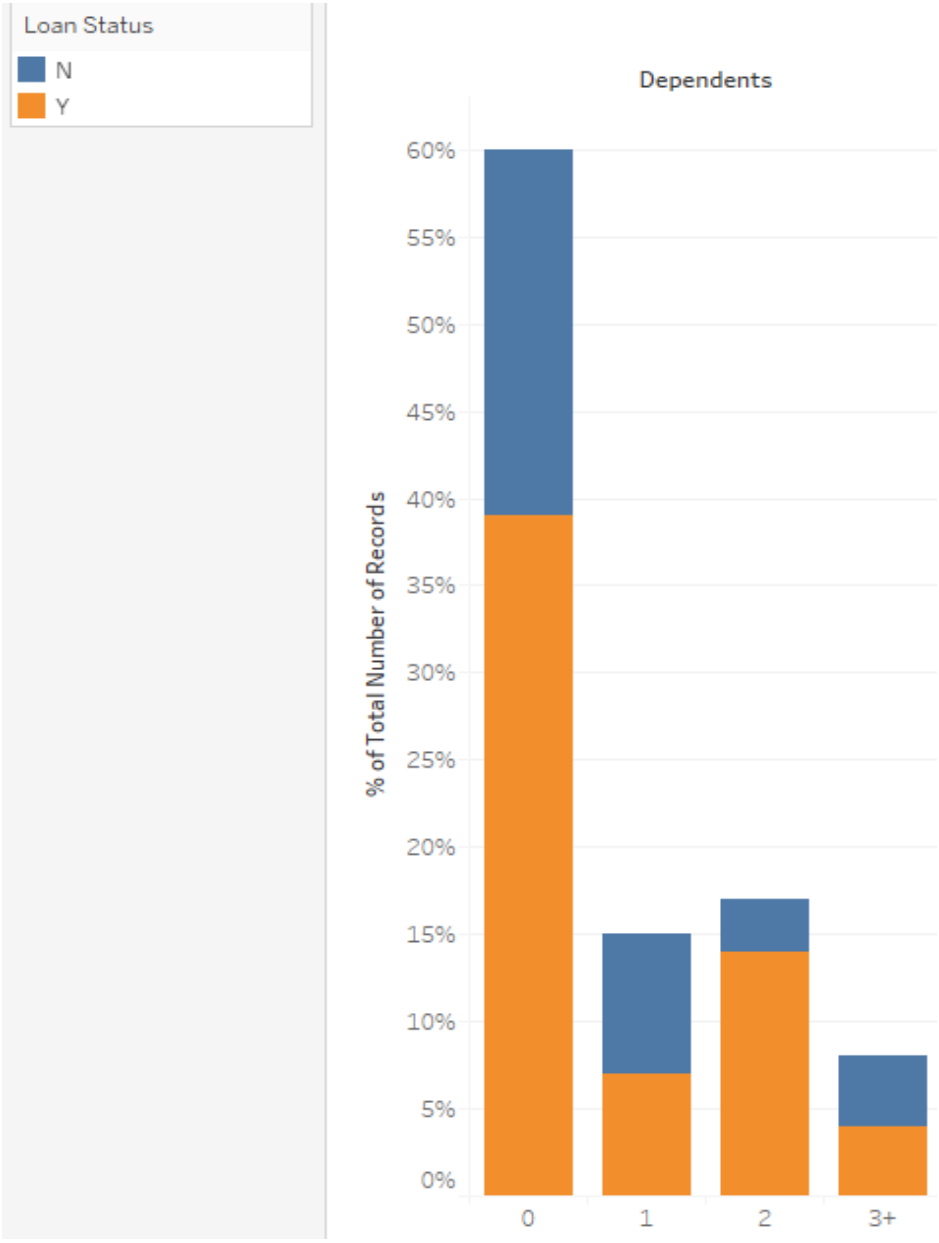
(iv) **Dependents**

Loan eligibility problem

Dependents	Loan Status N	Loan Status Y
0	21	39
1	8	7
2	3	14
3+	4	4

Dependents	Loan Status N	Loan Status Y
0	58.33%	60.94%
1	22.22%	10.94%
2	8.33%	21.88%
3+	11.11%	6.25%

Dependents	Loan Status N	Loan Status Y
0	35.00%	65.00%
1	53.33%	46.67%
2	17.65%	82.35%
3+	50.00%	50.00%



Loan approval as well as rejection is highest for appilcants having 0 dependents.

(v) Education

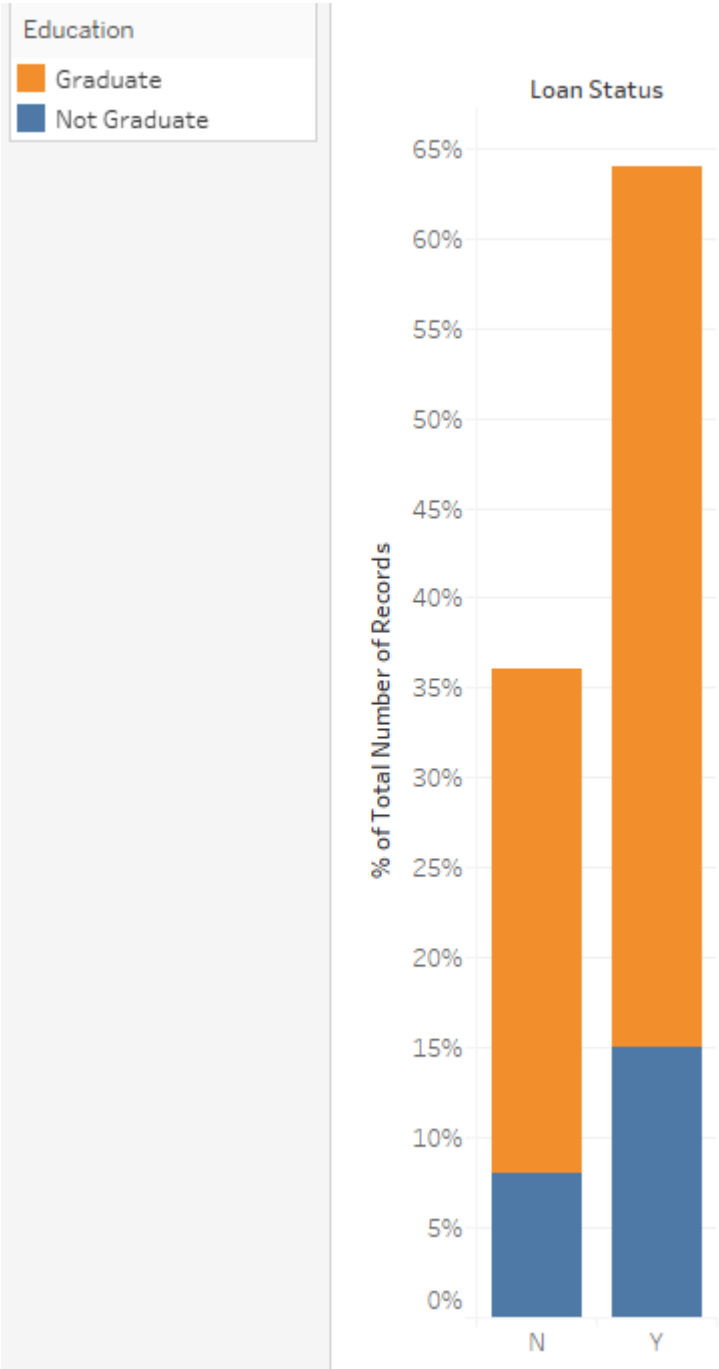
Loan eligibility problem

	Loan Status	
Education	N	Y
Graduate	28	49
Not Graduate	8	15

	Loan Status	
Education	N	Y
Graduate	77.78%	76.56%
Not Graduate	22.22%	23.44%

	Loan Status	
Education	N	Y
Graduate	36.36%	63.64%
Not Graduate	34.78%	65.22%

We can see Graduates have more chances of Loan approval.



(vi) Self_Employed

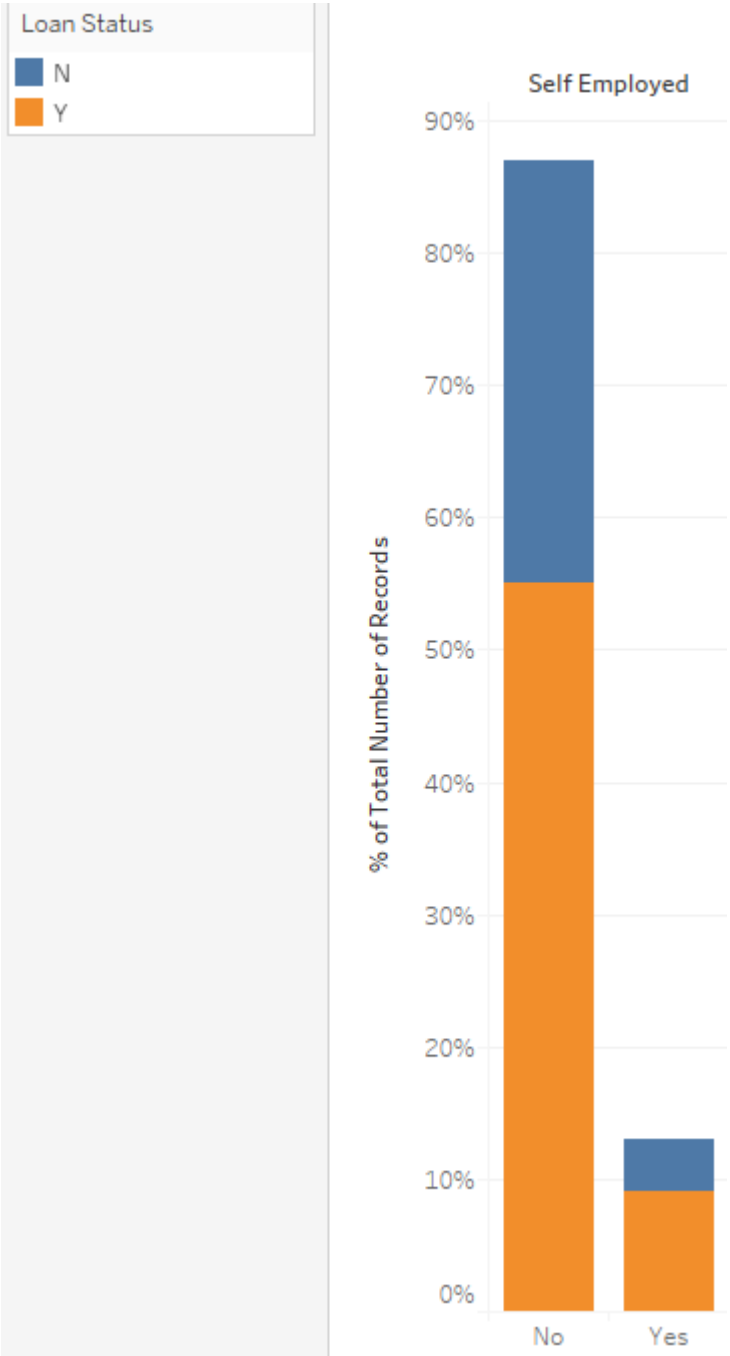
Loan eligibility problem

Self Employ..	Loan Status N	Loan Status Y
No	32	55
Yes	4	9

Self Employ..	Loan Status N	Loan Status Y
No	88.89%	85.94%
Yes	11.11%	14.06%

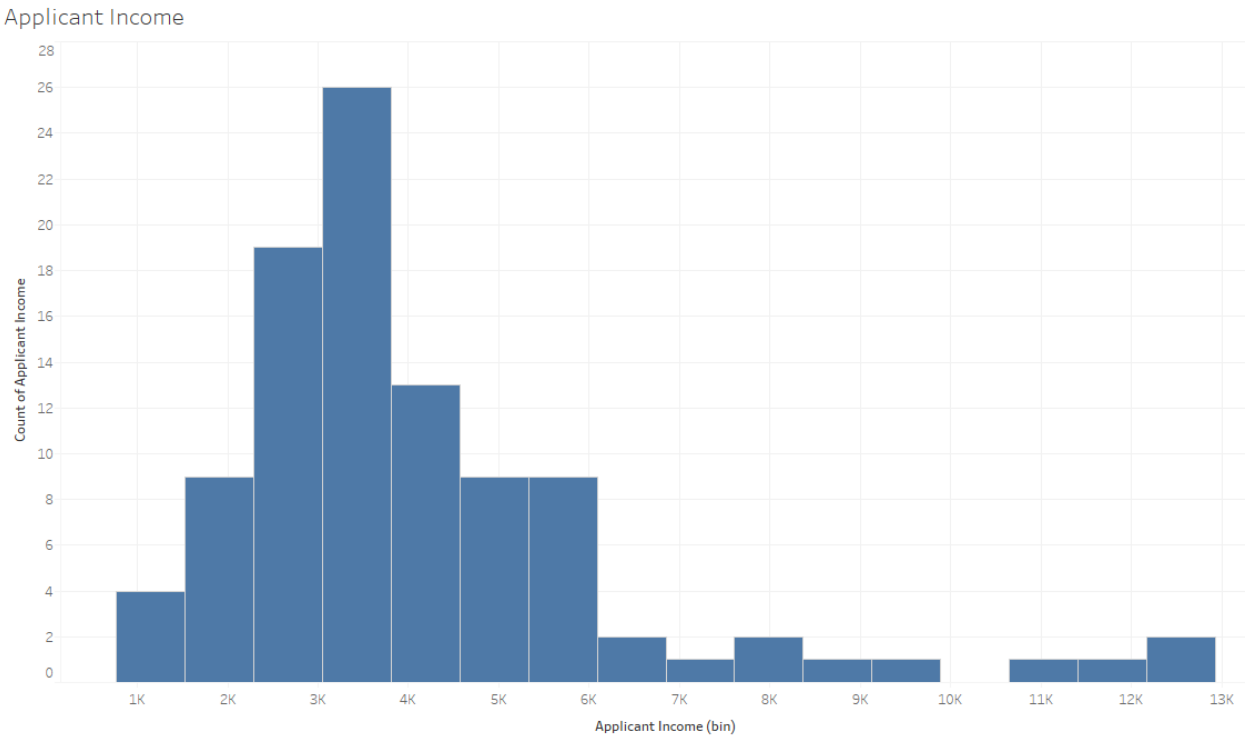
Self Emplo	Loan Status N	Loan Status Y
No	36.78%	63.22%
Yes	30.77%	69.23%

Non self employed person as more chances of loan approval



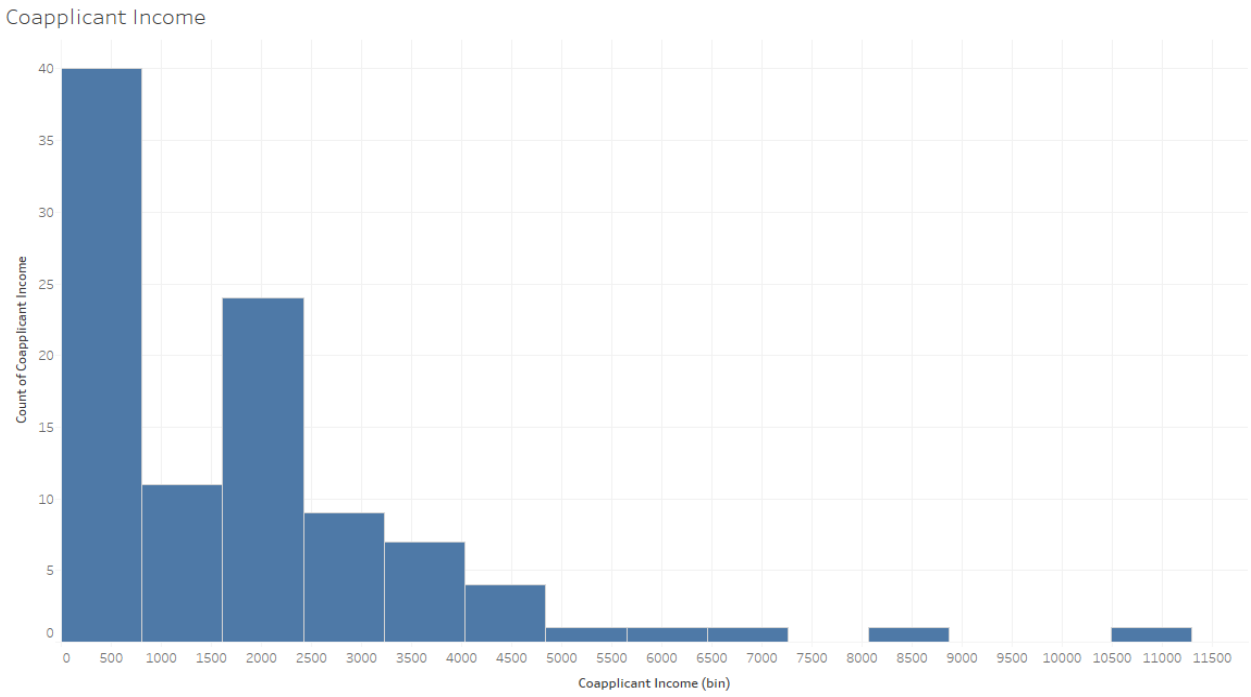
(vii) ApplicantIncome

Loan eligibility problem



We can see Applicant Income distribution is right skewed hence will have to perform normalization of data.

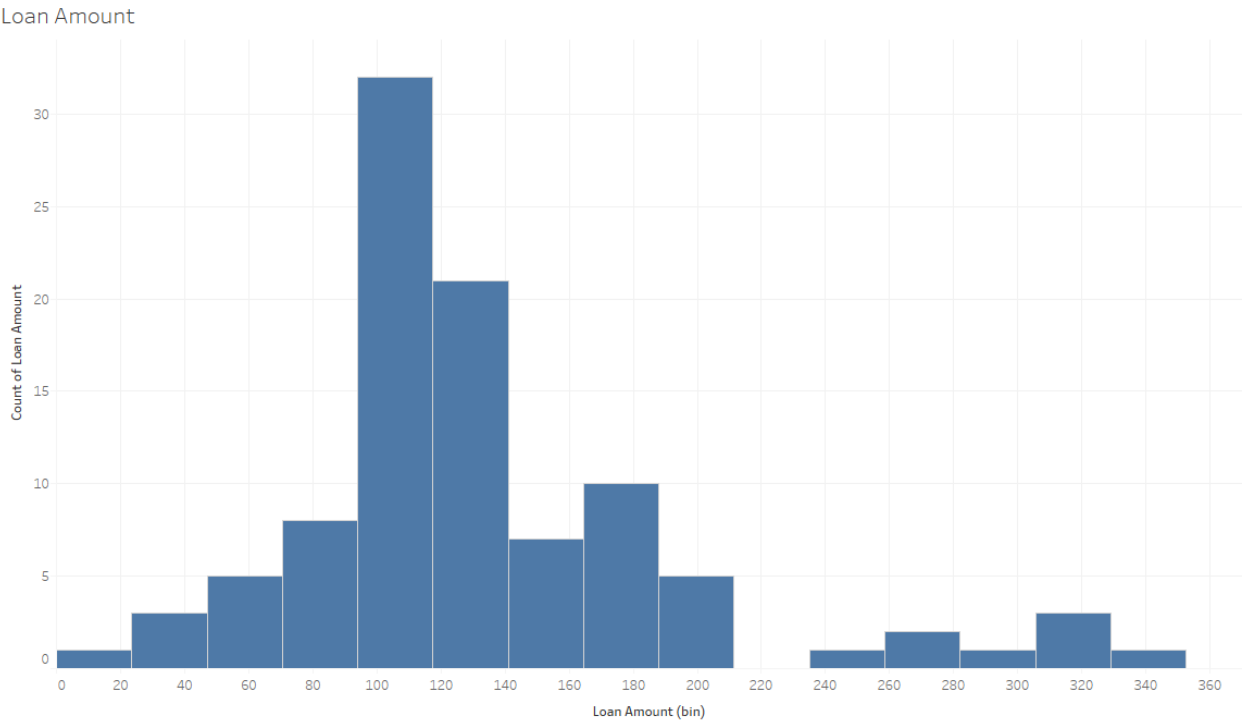
(viii) CoapplicantIncome



Loan eligibility problem

We can see Coapplicant Income distribution is right skewed hence will have to perform normalization of data.

(ix) LoanAmount



Loan Amount almost follows normal distribution with a few outliers.

(x) Loan_Amount_Term

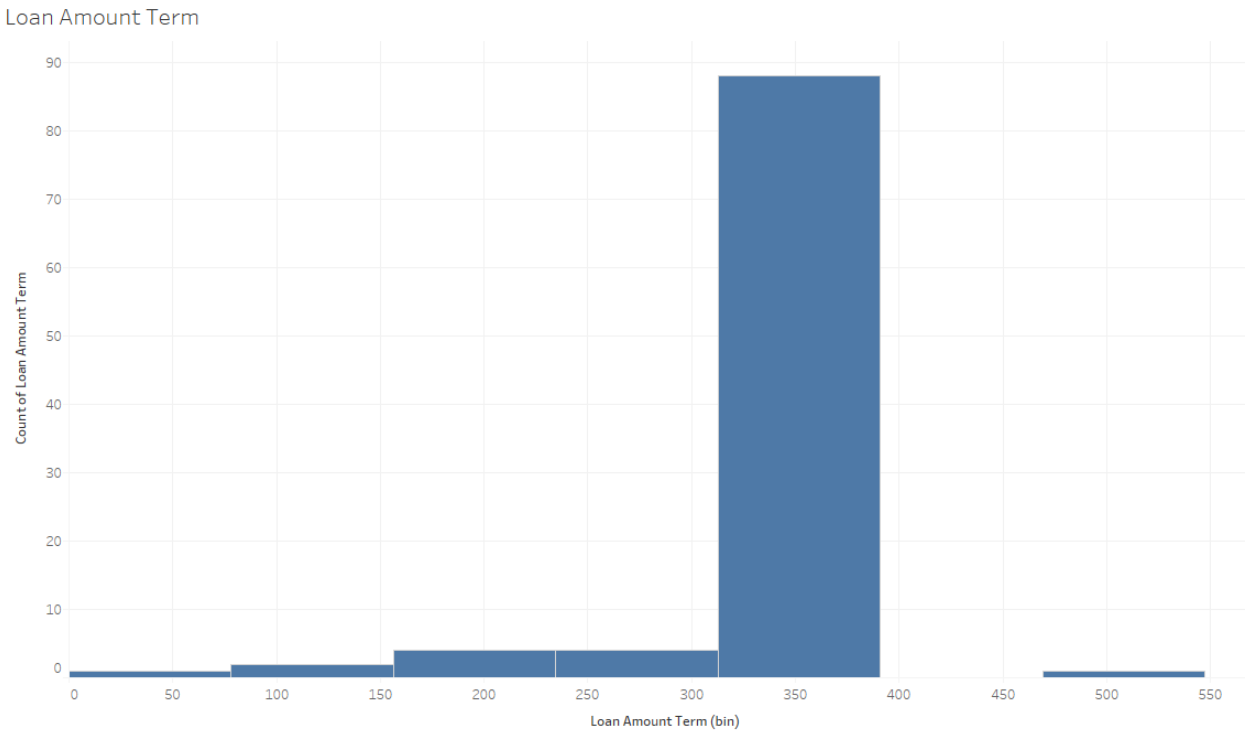
Loan Amou..	Loan Status N	Loan Status Y
60		1
120		2
180	2	2
240		2
300	1	1
360	32	56
480	1	

Loan Amou..	Loan Status N	Loan Status Y
60		1.56%
120		3.13%
180	5.56%	3.13%
240		3.13%
300	2.78%	1.56%
360	88.89%	87.50%
480	2.78%	

Loan Amou..	Loan Status N	Loan Status Y
60		100.00%
120		100.00%
180	50.00%	50.00%
240		100.00%
300	50.00%	50.00%
360	36.36%	63.64%
480	100.00%	

This shows that chances of loan approval are high when the loan amount term is less. However, the majority of loan application is for 360 months.

Loan eligibility problem



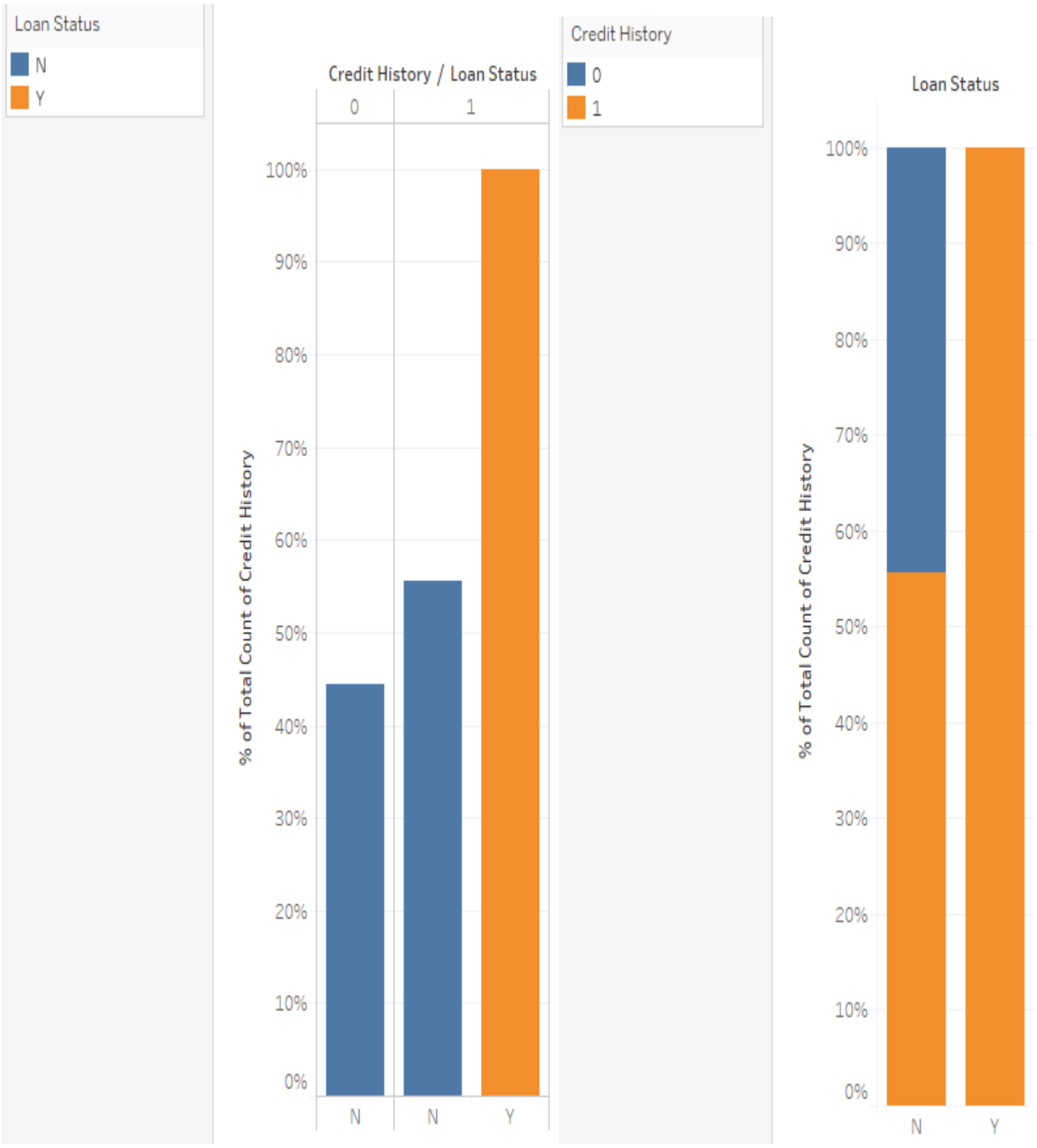
(xi) Credit_History

	Loan Status	
Credit Hist..	N	Y
0	16	
1	20	64

	Loan Status	
Credit Hist..	N	Y
0	100.00%	
1	23.81%	76.19%

	Loan Status	
Credit Hist..	N	Y
0	44.44%	
1	55.56%	100.00%

Loan eligibility problem



(xii) Property_Area

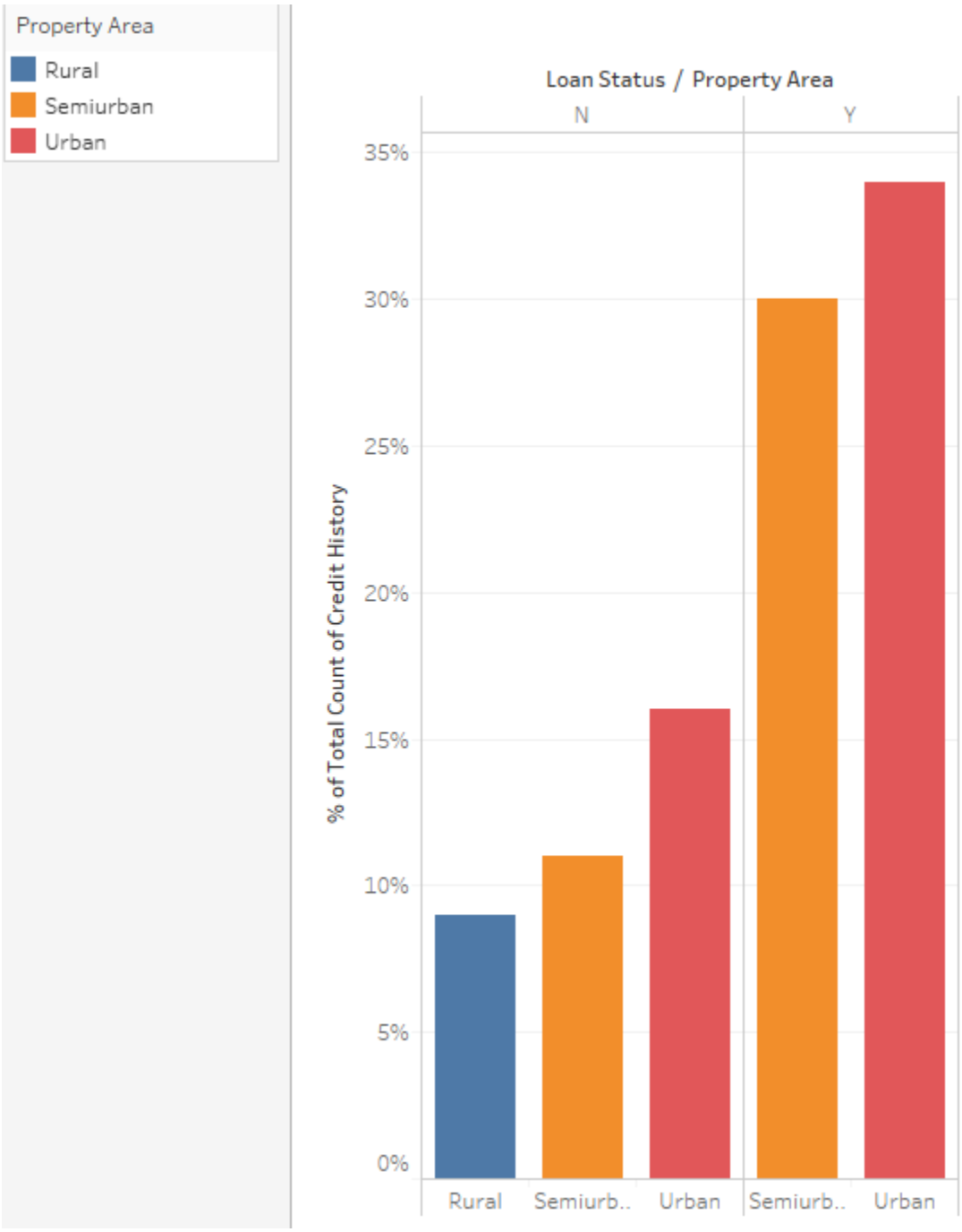
Loan eligibility problem

	Loan Status	
Property Ar..	N	Y
Rural	9	
Semiurban	11	30
Urban	16	34

	Loan Status	
Property Ar..	N	Y
Rural	25.00%	
Semiurban	30.56%	46.88%
Urban	44.44%	53.13%

	Loan Status	
Property Ar..	N	Y
Rural	100.00%	
Semiurban	26.83%	73.17%
Urban	32.00%	68.00%

Loan eligibility problem



Descriptive Analytics –

Variable Name	Mean	SD	Q1	median	Q3
ApplicantIncome	1000	2258.894	2636	4123	4710
CoapplicantIncome	1701	1947.669	0	1558	2394

Loan eligibility problem

LoanAmount	132.98	62.10092	99.75	120.00	145.75

(a) Average Applicant Income is 1000 with standard deviation of 2258.894 . But 75% applicant have a income less than 4710.
(b) Average coapplicant income is 1701 with standard deviation of 1947.669. But 75% coapplicant has income less than 2394.
(c) The std deviation for Loan Amount is 62.10092 and average loan amount is 132.98.

Based on above observation and histograms for applicant Income and Co-applicant income, we see the data is highly skewed to the right. Hence would be performing Normalization for them.

MODEL BUILDING

After careful observation of data, we have decided not delete observations since the data size is less and any deletion of observations will further reduce it.

We have applied kNN imputation on training and testing data to calculate the missing values.

kNN Imputation: Imputation using k-nearest neighbors. For each record, identify missing features. For each missing feature find the k nearest neighbors which have that feature. Impute the missing value using the imputation function on the k-length vector of values found from the neighbors.

It is followed by normalization of values for applicant and coapplicant income as they are highly skewed. We have used z-transformation for normalization.

Finally, we have applied the **Naïve-Bayes** model.

Naïve Bayes: The Naïve Bayes algorithm is a simple probabilistic classifier that calculates a set of probabilities by counting the frequency and combinations of values in a given dataset. The algorithm uses Bayes theorem and assumes all attributes to be independent given the value of class variable. This conditional independence assumption rarely holds true in real world applications, hence the characterization as Naïve yet the algorithm tends to perform well and learn rapidly in various supervised classification problems. Naïve Bayes classifier is based on Bayes theorem and the theorem of total probability.

In this classifier we compute the conditional probability P(Cj/X) and assign X to those class Ci having large probability i.e. Xϵ Cj if P(Cj/X)>P(Ci/X) for all i$\neq$j=1,2,….,m.

FINDINGS

In our case, we get below probabilities after applying Naïve-Bayes -

```
> classifier_NB

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
   N    Y
0.36 0.64

Conditional probabilities:
   Gender
Y           F         M
  N 0.1944444 0.8055556
  Y 0.1250000 0.8750000

   Married
Y          No       Yes
  N 0.3888889 0.6111111
  Y 0.2656250 0.7343750

   Dependents
Y           0          1          2         3+
  N 0.58333333 0.22222222 0.08333333 0.11111111
  Y 0.60937500 0.10937500 0.21875000 0.06250000

   Education
Y    Graduate Not Graduate
  N 0.7777778    0.2222222
  Y 0.7656250    0.2343750

   Self_Employed
Y          No       Yes
  N 0.8888889 0.1111111
  Y 0.8593750 0.1406250

   ApplicantIncome
Y         [,1]      [,2]
  N  0.1587566 1.2339369
  Y -0.0893006 0.8384725

   CoapplicantIncome
Y          [,1]      [,2]
  N  0.04126300 1.1374881
  Y -0.02321044 0.9225436

   LoanAmount
Y       [,1]     [,2]
  N 146.6111 69.74455
  Y 125.3125 56.49747
```

Loan eligibility problem

```
   Loan_Amount_Term
Y       [,1]     [,2]
  N 351.6667 47.89870
  Y 337.5000 65.46537

   Credit_History
Y        [,1]      [,2]
  N 0.5555556 0.5039526
  Y 1.0000000 0.0000000

   Property_Area
Y       Rural Semiurban     Urban
  N 0.2500000 0.3055556 0.4444444
  Y 0.0000000 0.4687500 0.5312500
```

Now the accuracy has been tested for the training data and confusion matrix has been created.

```
> confusionMatrix(newtraindata$Loan_Status, prediction_NB)
Confusion Matrix and Statistics

          Reference
Prediction  N  Y
         N 18 18
         Y  0 64

               Accuracy : 0.82
                 95% CI : (0.7305, 0.8897)
    No Information Rate : 0.82
    P-Value [Acc > NIR] : 0.5626

                  Kappa : 0.5614
 Mcnemar's Test P-Value : 6.151e-05

            Sensitivity : 1.0000
            Specificity : 0.7805
         Pos Pred Value : 0.5000
         Neg Pred Value : 1.0000
             Prevalence : 0.1800
         Detection Rate : 0.1800
   Detection Prevalence : 0.3600
      Balanced Accuracy : 0.8902

       'Positive' Class : N
```

Naïve Bayesian classifier achieves accuracy 82% with sensitivity 1 and specificity .7805

Finally, prediction for test data has been generated using same model.

REFERENCES

Loan eligibility problem

[1] http://www.r-tutor.com/

[2] https://www.rdocumentation.org/packages/imputation/versions/2.0.3/topics/kNNImpute

[3] https://www.rdocumentation.org/packages/e1071/versions/1.6-8