German Credit Risk Analysis

07.03.2019

Rujuta Kelkar
Tejas Choudhary
Divyam Mehta

ACM Karyavarta

Overview

When a bank receives a loan application, based on the applicant's profile the bank has to make a decision regarding whether to go ahead with the loan approval or not. An applicant's demographic and socio-economic profiles are considered by loan managers before a decision is taken regarding his/her loan application.Two types of risks are associated with the bank's decision –

- If the applicant is a good credit risk, i.e. is likely to repay the loan, then not approving the loan to the person results in a loss of business to the bank
- If the applicant is a bad credit risk, i.e. is not likely to repay the loan, then approving the loan to the person results in a financial loss to the bank.

Goals

1. Minimization of risk and maximization of profit on behalf of the bank.
2. Reducing manpower required for evaluation of the loans.
3. Overcoming human bias and errors in evaluation of creditors.
4. Improving efficiency of the process by automation.

Specifications

Financial losses due to bad loans are a problem that majority of banks face. Using ML models in python, we train our model using a very reliable dataset provided by PennState University, Eberly College of Science, in the course Applied Data Mining and Statistical Learning (STAT 508).

The German Credit Data contains data on 20 variables and the classification whether an applicant is considered a Good or a Bad credit risk for 1000 loan applicants. A predictive model developed on this data is expected to provide a bank manager guidance for making a decision whether to approve a loan to a prospective applicant based on his/her profiles.

Exploratory Data Analysis

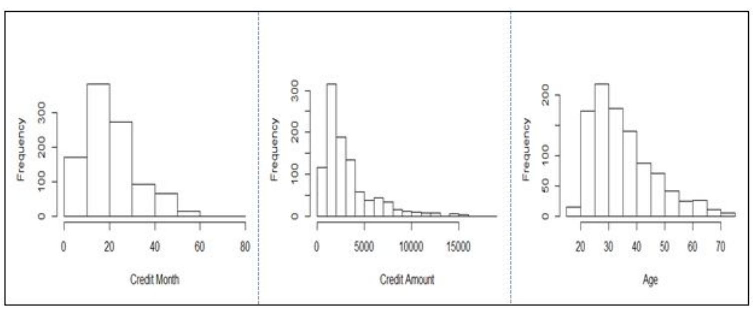
Before getting into any sophisticated analysis, the first step is to do an EDA and data cleaning.

To analyse the dataset, we have used pandas and numpy libraries in python and formulated the data into matrices to understand the patterns and distributions.

Predictor (Categorical)	Levels and Proportions				
Account Balance	No Account	None	Below 200 DM	200 DM or Above	
(%)	27.4%	26.9%	6.3%	39.4%	
Payment Status	Delayed	Other Credits	Paid Up	No Problem with Current Credits	Previous Credits Paid
(%)	4.0%	4.9%	53.0%	8.8%	29.3%
Savings/ Stock Value	None	Below 100 DM	[100, 500)	[500, 1000)	Above 1000
	60.3%	10.3%	6.3%	4.8%	18.3%
Length of Current Employment	Unemployed	<1 Year	[1, 4)	[4, 7)	Above 7
	6.2%	17.2%	33.9%	17.4%	25.3%
Installments %	Above 35%	(25%, 35%)	[20%, 25%)	Below 20%	
	13.6%	23.1%	15.7%	47.6%	
Occupation	Unemployed, unskilled	Unskilled Permanent Resident	Skilled	Executive	
	2.2%	20.0%	63.0%	14.8%	
Sex and Marital Status	Male, Divorced	Male, Single	Male, Married/Widowed	Female	
	5.0%	31.0%	54.8%	9.2%	
Duration in Current Address	<1 Year	[1, 4)	[4, 7)	Above 7	
	13.0%	30.8%	14.9%	41.3%	
Type of Apartment	Free	Rented	Owned		
	17.9%	71.4%	10.7%		
Most Valuable Asset	None	Car	Life Insurance	Real Estate	
	28.2%	23.2%	33.2%	15.4%	
No. of credits at Bank	1	2 or 3	4 or 5	Above 6	
	63.3%	33.3%	2.8%	0.06%	
Guarantor	None	Co-applicant	Guarantor		
	90.7%	4.1%	5.2%		
Concurrent Credits	Other Banks	Dept. Store	None		
	13.9%	4.7%	81.4%		
No. of Departments	3 or More	Less than 3			
	84.5%	15.5%			
Telephone	Yes	No			

]:

	Parameter 1	Parameter 2	Parameter 3	Parameter 4	Parameter 5
1	4.0	4.9	53.0	8.8	29.3
2	60.3	10.3	6.3	4.8	18.3
3	6.2	17.2	33.9	17.4	25.3
0	27.4	26.9	6.3	39.4	NaN
4	13.6	23.1	15.7	47.6	NaN
5	5.0	31.0	54.8	9.2	NaN
7	13.0	30.8	14.9	41.3	NaN
8	28.2	23.2	33.2	15.4	NaN
11	63.3	33.3	2.8	0.6	NaN
12	2.2	20.0	63.0	14.8	NaN
6	90.7	4.1	5.2	NaN	NaN
9	13.9	4.7	81.4	NaN	NaN
10	17.9	71.4	10.7	NaN	NaN
13	84.5	15.5	NaN	NaN	NaN
14	59.6	40.4	NaN	NaN	NaN
15	96.3	3.7	NaN	NaN	NaN

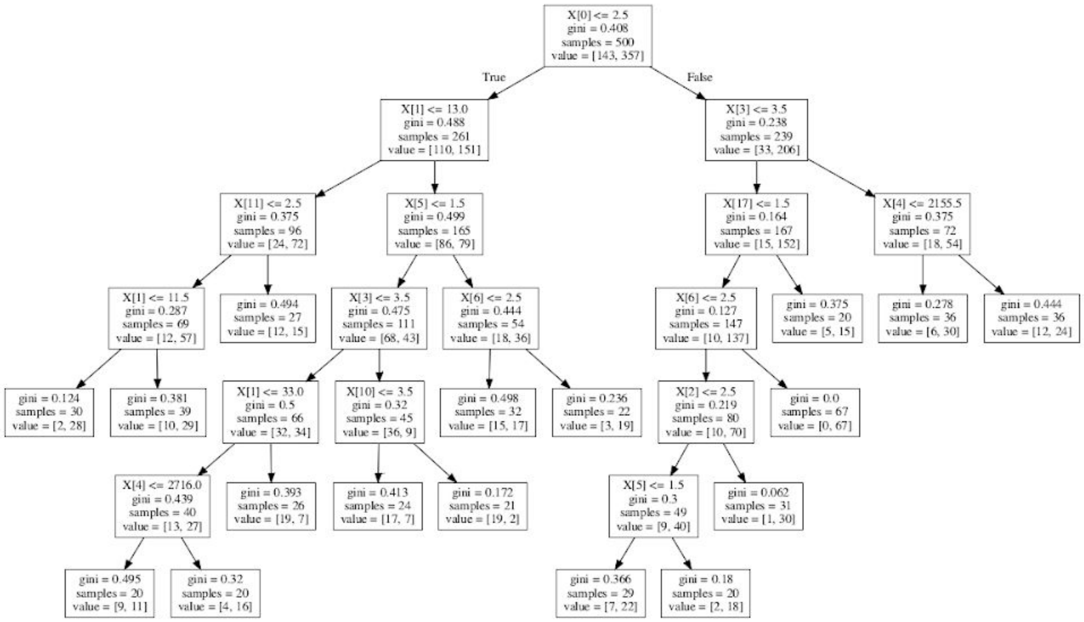


Tree-Based Methods

To choose optimal predictors to divide the data and form a decision tree, the scikit library in python was used and our model was trained and tested using this. The results obtained were as follows:

1. Decision Tree: This works by taking into consideration a set of parameters use to give out loans, and branching out based on whether those conditional parameters apply or not.

Gini coefficient applies to binary classification and requires a classifier that can in some way rank examples according to the likelihood of being in a positive class.




```
Accuracy of Decision Tree:  66.8
                Creditable  Non-Creditable
Creditable             314              43
Non-Creditable         123              20
Profit per applicant = 0.029800000000000001
Total Profit =  14.900000000000006
```

2. Random Forest : This is a collection of different decision trees and outputting the mean result as predicted by majority of the trees.

```
Accuracy of Random Forest =  72.0
                Creditable  Non-Creditable
Creditable             356               1
Non-Creditable         139               4
Profit per applicant =  0.034599999999999999
Profit  17.299999999999997
```

Cost Profit Consideration

Our model is expected to output a decision that will minimise losses, ie, take a correct decision.

A correct decision here means that the bank predicts an application to be good or credit-worthy and it actually turns out to be creditworthy. When the opposite is true, i.e. bank predicts the application to be good but it turns out to be bad credit, then the loss is 100%. If the bank predicts an application to be non-creditworthy, then loan facility is not extended to that applicant and bank does not incur any loss (opportunity loss is not considered here).

```
Profit percentage taken =  0.2
Profit per applicant =  -0.1078
Profit  -53.900000000000006
Profit percentage taken =  0.25
Profit per applicant =  -0.07220000000000001
Profit  -36.10000000000001
Profit percentage taken =  0.3
Profit per applicant =  -0.03660000000000002
Profit  -18.30000000000001
Profit percentage taken =  0.35
Profit per applicant =  -0.0010000000000000284
Profit  -0.5000000000000142
Profit percentage taken =  0.4
Profit per applicant =  0.03459999999999999
Profit  17.299999999999997
Profit percentage taken =  0.45
Profit per applicant =  0.07020000000000001
Profit  35.10000000000001
Profit percentage taken =  0.5
Profit per applicant =  0.10579999999999998
Profit  52.89999999999999
```

Conclusion

We used the data available on the PennState website, used the scikit, panda and numpy libraries in python, and formed a table, by **fragmenting the data based on individual parameters**, into segments. We observed that some of the segments of the applicants had low levels of participation (based on a particular attribute), and hence were disregarded.

We used 2 ML classifiers to implement this, the Decision Tree and the Random forest. These classifiers developed rules based on the training data. The accuracy of the model signifies how many times our model **correctly predicted the credibility of the loan**. The decision tree gave us an accuracy of 66.8%, while the random forest gave us 72.0%.

While formulating the rules for prediction , it was observed that **certain attributes highly influenced the credibility of the loan,** compared to others. These were ,in descending order of importance, Account balance, Duration of credit month, and the purpose of the loan.

The **amount of risk that a bank** can handle depends on how much the **return rate for a good loan** is. Based on this, we assumed a variety of profit return rates in the range 20% to 50% and observed that our model is profitable in a scenario where the rate is greater than 35%. Upto 35% the inaccuracies in the model pose a high risk and might lead to an aggregate loss.

Learning objectives achieved

1. **Understanding of the loan process**

   We learnt how the lending process in a bank works, what information it requires, and what factors are considered when making a decision about a loan.

2. **Risk Management:**

   We learnt how to manage capital while lending and noticing trends amongst the applicants who were considered unworthy of a loan from the bank.

3. **Analysis of raw data:**

   When provided with raw data, we learnt how to pre process and analyse it to gain insight on further classification.

4. **ML Models:**

   We learnt how the algorithms work, and which classifiers work better for a certain condition.