CREDIT RISK ANALYZER

- Vineet Jha
BTech (CSE)
USICT, GGSIPU

ABSTRACT:

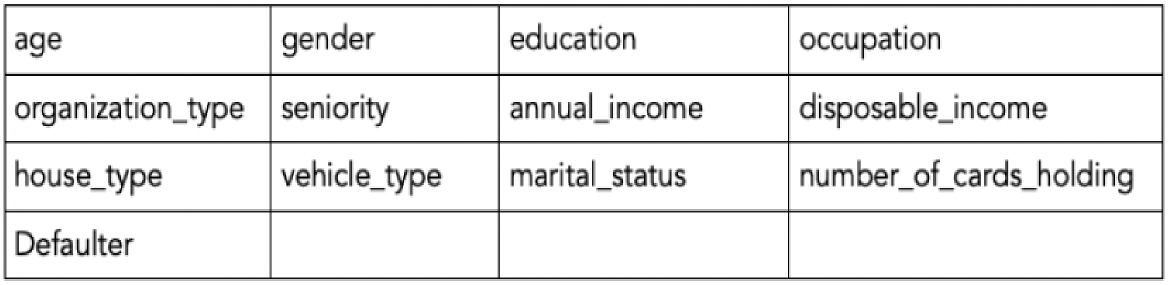
Banks need to protect their interest before it can take risk on you and issue credit card to you. Banks use their previous credit card holders' records for understanding the patterns of the card holders. It is a lot more complex process to predict whether a person who they do not know at personal level, will be a defaulter or not. Banks, along with the data from their own records, also use CIBIL data. CIBIL plays an important role here as it collects data from all your loan types, across financial institutes, over a period of time. As it is defined, "CIBIL Score is a three-digit numeric summary of your credit history. The score is derived using the credit history found in the CIBIL Report (also known as CIR i.e. Credit Information Report). A CIR is an individual's credit payment history across loan types and credit institutions over a period of time. A CIR does not contain details of your savings, investments or fixed deposits." Higher the score better it is. The banks also consider market conditions, their current risk appetite etc. before they issue a credit card to you. Based on all this data, banks want to develop a pattern that will tell them who are likely to be a defaulter and who are not. All data are fed into the analyzer and based on the rules set in the analyzer an applicant's application is accepted or rejected

OBJECTIVE:

Now that we know what credit risk is, and the usefulness of a credit risk analyzer to help banks and financial institutes minimize their credit risk, let us first talk about the methods and techniques we used in developing our credit risk analyzer. First, we build a decision tree credit risk analyzer, based on a portion of the data. Second, we test the credit risk analyzer based on the remaining data. If we have found an analyzer that is tested ok, we will use the analyzer to classify new applicants as per their likelihood of being a defaulter.

DATA:

In this section, we will see the dataset we used in this project. The dataset has 13 features with 50636 observations. The features are:



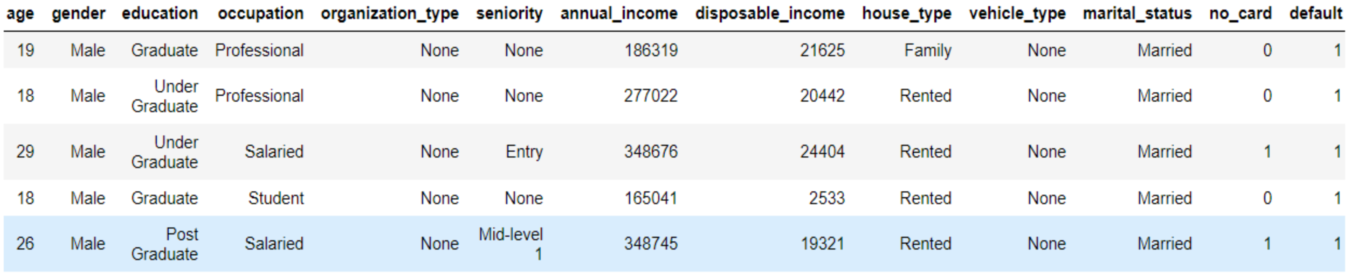
age	gender	education	occupation
organization_type	seniority	annual_income	disposable_income
house_type	vehicle_type	marital_status	number_of_cards_holding
Defaulter			

Here **age, gender** are the age and gender of the card holder.
education is the last acquired educational qualification of the cardholder. The person can be Graduate, Under-Graduate, Post Graduate or Other (10th,12th or something else).
occupation can be salaried, or self-employed or business etc.
organization_type can be Tier 1, Tier 2, Tier 3 or None etc.
seniority denotes at which career level the card holder is in. It can be Entry level, Mid-level 1, Junior, Mid-level 2, Senior etc.

annual_income is the gross annual income of the card holder.
disposable_income is annual income - recurring expenses.
house_type is owned or rented or company provided etc.
vehicle_type is 4-wheeler or two-wheeler or none.
marital_status is marital status of the card holder.
no_card has the information of the number of other credit cards that the card holder already holds.
And the last column is **defaulter** indicating whether the card holder was a defaulter or not. It is 1 if the card holder is a defaulter, 0 otherwise. This is our target variable.

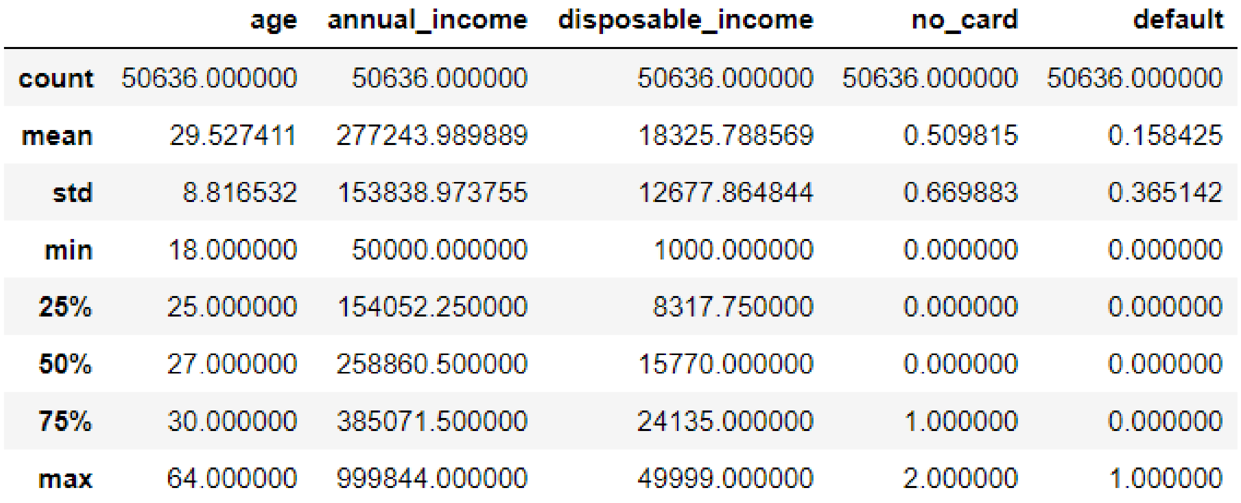
DATA EXPLORATORY ANALYSIS:

First, we will take a look at how the data looks.



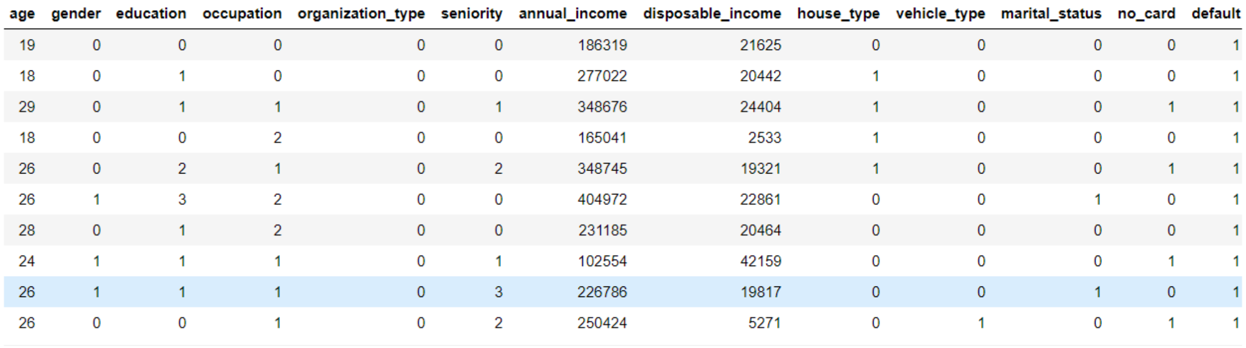
age	gender	education	occupation	organization_type	seniority	annual_income	disposable_income	house_type	vehicle_type	marital_status	no_card	default
19	Male	Graduate	Professional	None	None	186319	21625	Family	None	Married	0	1
18	Male	Under Graduate	Professional	None	None	277022	20442	Rented	None	Married	0	1
29	Male	Under Graduate	Salaried	None	Entry	348676	24404	Rented	None	Married	1	1
18	Male	Graduate	Student	None	None	165041	2533	Rented	None	Married	0	1
26	Male	Post Graduate	Salaried	None	Mid-level 1	348745	19321	Rented	None	Married	1	1

Some standard descriptive analysis of numerical features:



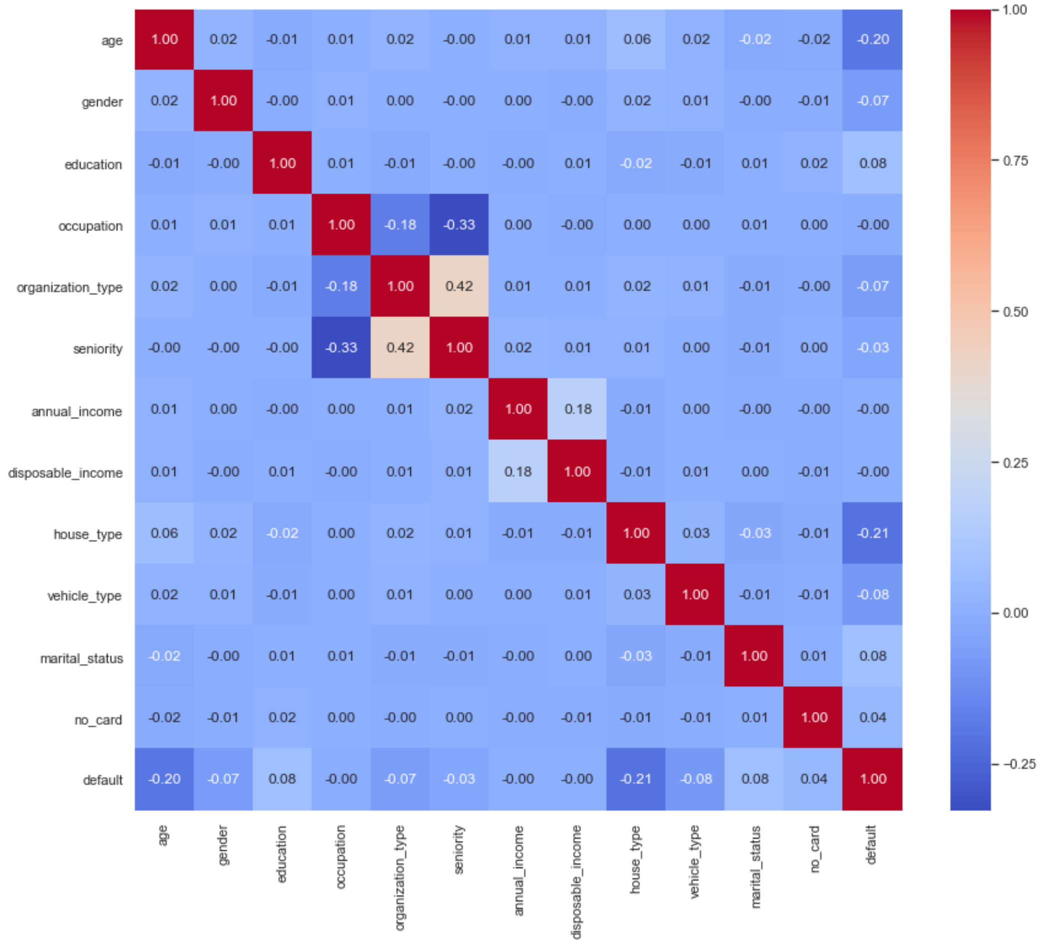
	age	annual_income	disposable_income	no_card	default
count	50636.000000	50636.000000	50636.000000	50636.000000	50636.000000
mean	29.527411	277243.989889	18325.788569	0.509815	0.158425
std	8.816532	153838.973755	12677.864844	0.669883	0.365142
min	18.000000	50000.000000	1000.000000	0.000000	0.000000
25%	25.000000	154052.250000	8317.750000	0.000000	0.000000
50%	27.000000	258860.500000	15770.000000	0.000000	0.000000
75%	30.000000	385071.500000	24135.000000	1.000000	0.000000
max	64.000000	999844.000000	49999.000000	2.000000	1.000000

First, we will preprocess the data for further diagnostics. As we saw earlier, this dataset contains some categorical data and we need to convert them to numerical data. We do this using pandas’ factorize() method.



age	gender	education	occupation	organization_type	seniority	annual_income	disposable_income	house_type	vehicle_type	marital_status	no_card	default
19	0	0	0	0	0	186319	21625	0	0	0	0	1
18	0	1	0	0	0	277022	20442	1	0	0	0	1
29	0	1	1	0	1	348676	24404	1	0	0	1	1
18	0	0	2	0	0	165041	2533	1	0	0	0	1
26	0	2	1	0	2	348745	19321	1	0	0	1	1
26	1	3	2	0	0	404972	22861	0	0	1	0	1
28	0	1	2	0	0	231185	20464	0	0	0	0	1
24	1	1	1	0	1	102554	42159	0	0	0	1	1
26	1	1	1	0	3	226786	19817	0	0	1	0	1
26	0	0	1	0	2	250424	5271	0	1	0	1	1

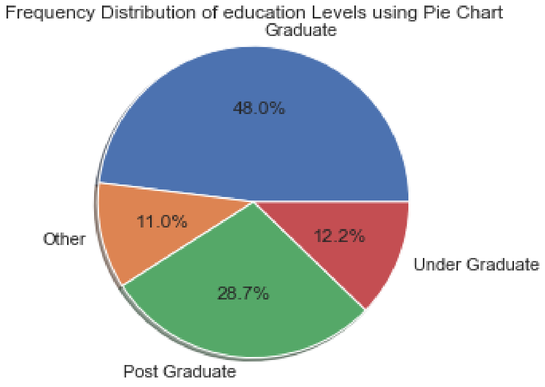
Checking for Correlation after converting categorical data to numerical data:

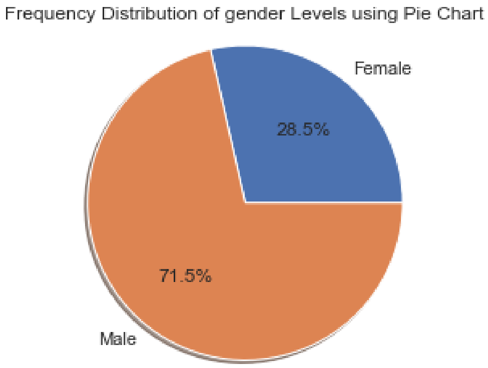


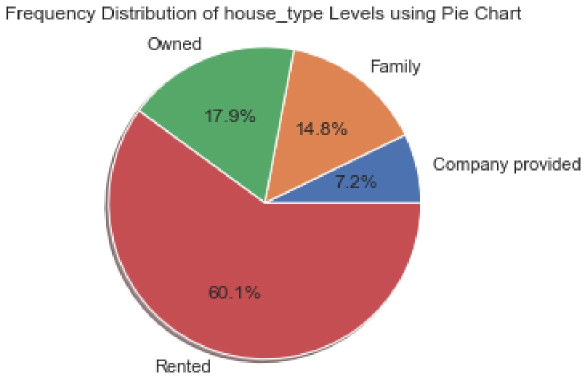
	age	gender	education	occupation	organization_type	seniority	annual_income	disposable_income	house_type	vehicle_type	marital_status	no_card	default
age	1.00	0.02	-0.01	0.01	0.02	-0.00	0.01	0.01	0.06	0.02	-0.02	-0.02	-0.20
gender	0.02	1.00	-0.00	0.01	0.00	-0.00	0.00	-0.00	0.02	0.01	-0.00	-0.01	-0.07
education	-0.01	-0.00	1.00	0.01	-0.01	-0.00	-0.00	0.01	-0.02	-0.01	0.01	0.02	0.08
occupation	0.01	0.01	0.01	1.00	-0.18	-0.33	0.00	-0.00	0.00	0.00	0.01	0.00	-0.00
organization_type	0.02	0.00	-0.01	-0.18	1.00	0.42	0.01	0.01	0.02	0.01	-0.01	-0.00	-0.07
seniority	-0.00	-0.00	-0.00	-0.33	0.42	1.00	0.02	0.01	0.01	0.00	-0.01	0.00	-0.03
annual_income	0.01	0.00	-0.00	0.00	0.01	0.02	1.00	0.18	-0.01	0.00	-0.00	-0.00	-0.00
disposable_income	0.01	-0.00	0.01	-0.00	0.01	0.01	0.18	1.00	-0.01	0.01	0.00	-0.01	-0.00
house_type	0.06	0.02	-0.02	0.00	0.02	0.01	-0.01	-0.01	1.00	0.03	-0.03	-0.01	-0.21
vehicle_type	0.02	0.01	-0.01	0.00	0.01	0.00	0.00	0.01	0.03	1.00	-0.01	-0.01	-0.08
marital_status	-0.02	-0.00	0.01	0.01	-0.01	-0.01	-0.00	0.00	-0.03	-0.01	1.00	0.01	0.08
no_card	-0.02	-0.01	0.02	0.00	-0.00	0.00	-0.00	-0.01	-0.01	-0.01	0.01	1.00	0.04
default	-0.20	-0.07	0.08	-0.00	-0.07	-0.03	-0.00	-0.00	-0.21	-0.08	0.08	0.04	1.00

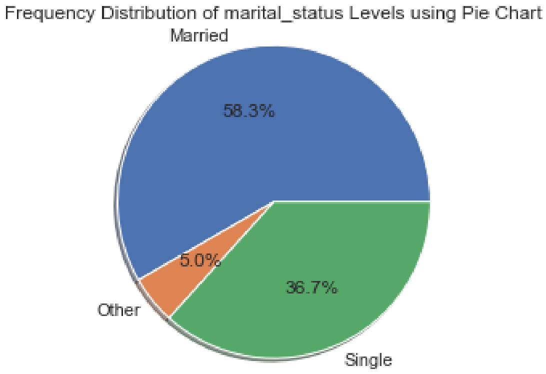
We see a mild correlation of occupation and organization type with seniority. However, they differ in their correlation. While occupation shows a negative correlation, organization type is positively correlated with seniority.

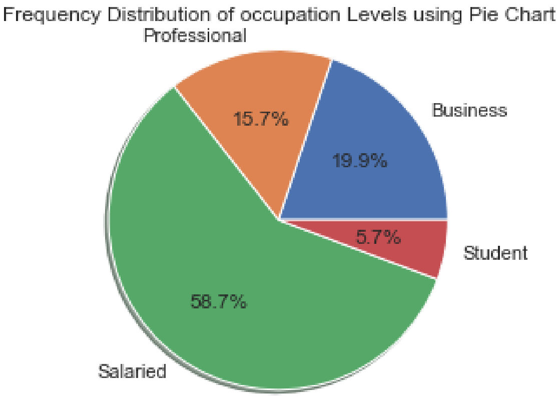
DATA VISUALIZATION:

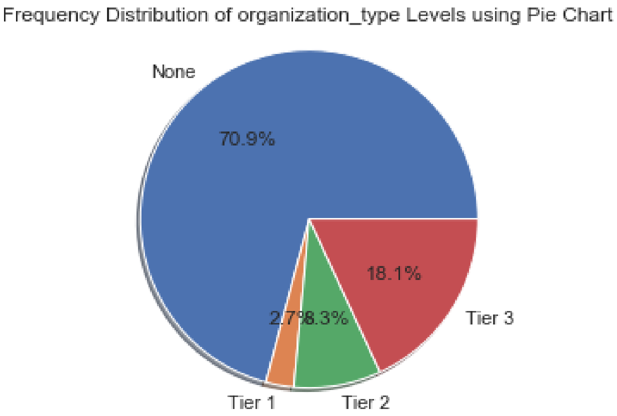


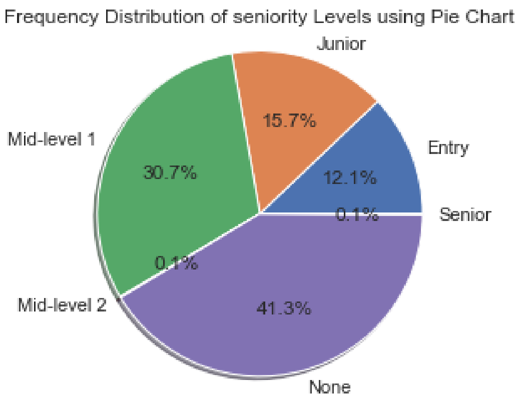


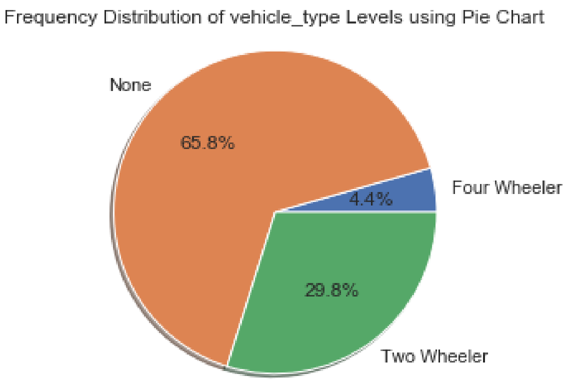












We have used a pie chart to show the frequencies of different categories. The summary is as follows:

Gender: 'Male' – 71.5%, 'Female' – 28.5%

Education: 'Graduate' – 48%, 'Under Graduate' – 12.2%, 'Post Graduate' – 28.7%, 'Other' – 11%

Occupation: 'Professional' – 15.7%, 'Salaried' – 58.7%, 'Student' – 5.7%, 'Business' – 19.9%

Organization Type: 'None' – 70.9%, 'Tier 3' – 18.1%, 'Tier 2' – 8.3%, 'Tier 1' – 2.7%

Seniority: 'None' – 41.3%, 'Entry' – 12.1%, 'Mid-level 1' – 30.7%, 'Junior' – 15.7%, 'Mid-level 2' – 0.1%, 'Senior' – 0.1%

House Type: 'Family' – 14.8%, 'Rented' – 60.1%, 'Company provided' – 7.2%, 'Owned' – 17.9%

Vehicle Type: 'None' – 65.8%, 'Two-Wheeler' – 29.8%, 'Four-Wheeler' – 4.4%

Marital Status: 'Married' – 58.3%, 'Single' – 36.7%, 'Other' – 5%

METHODS AND TECHNIQUES:

We have already converted the categorical data to numerical data.

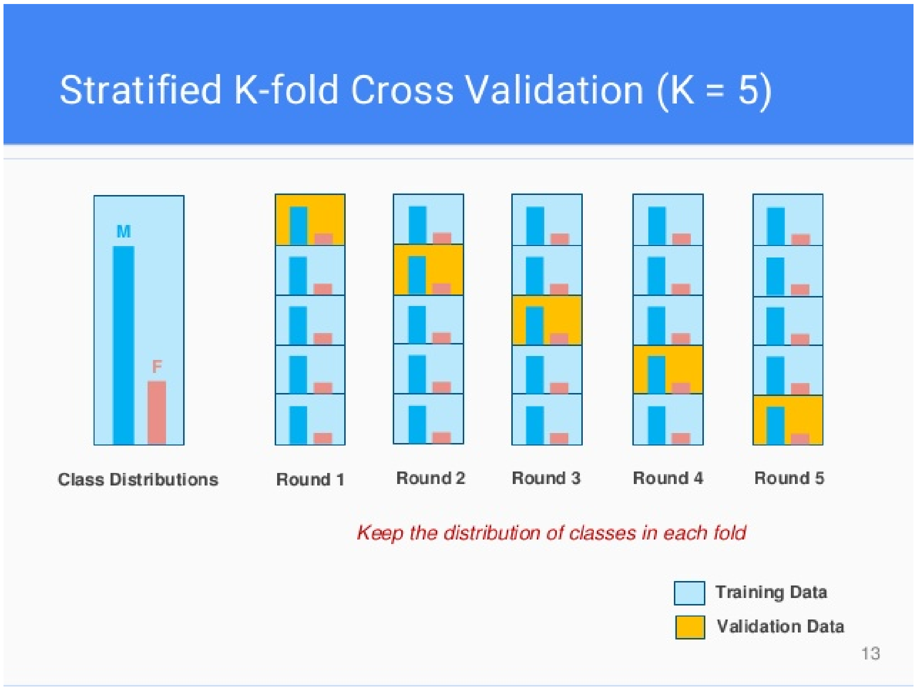
Now we split the data using train_test_split () from sklearn.model_selection in 80:20 ratio.

We are going to use Decision Tree Classification with:

a) Entropy Criterion
b) Gini Criterion

This is a pretty straight-forward way of doing it. You reserve a part of your original dataset for testing (or validation), and the other half for training. Once you get an estimate of the model's error, you may also use the portion previously used for training for testing now, and vice versa. Effectively, this gives you two estimates of how well your model works. However, there are a couple of problems with this approach. First off, you get only one (or two) estimates of the model's accuracy. Though this is better than having no test dataset, it still leaves a lot of room for improvement. Secondly, since you reserved half of the original dataset for testing, you induced a significant amount of *bias* into the whole process- in plain English, do you think the framework trained with half of the dataset will be comparable to the one trained with the complete dataset?

To overcome this, we use Decision Tree Classifier with Stratified K-Fold Cross Validation for both Entropy criterion as well as Gini criterion. Now what exactly is Stratified K-Fold. In stratified k-fold cross-validation, the folds are selected so that the mean response value is approximately equal in all the folds. In the case of a dichotomous classification, this means that each fold contains roughly the same proportions of the two types of class labels. This is generally a better scheme, both in terms of bias and variance, when compared to regular cross-validation.



Stratified K-fold Cross Validation (K = 5)

Keep the distribution of classes in each fold

RESULTS:

Accuracy of Decision Tree with Entropy Criterion with TTS: 84.37
Accuracy of Decision Tree with Gini Criterion with TTS: 84.24

Mean Accuracy of Decision Tree with Entropy Criterion with Stratified K-Fold: 84.19

Max Accuracy of Decision Tree with Entropy Criterion with Stratified K-Fold: 86.10

Min Accuracy of Decision Tree with Entropy Criterion with Stratified K-Fold: 80.22

Mean Accuracy of Decision Tree with Gini Criterion with Stratified K-Fold: 84.20

Max Accuracy of Decision Tree with Gini Criterion with Stratified K-Fold: 86.22

Min Accuracy of Decision Tree with Gini Criterion with Stratified K-Fold: 80.24

As is evident from the results, with train_test_split, decision tree with entropy criterion gave better results but with stratified k-fold cross validation, gini criterion outperformed entropy criterion. The difference in mean and min accuracies is mild but max accuracies reached show a significant difference.

CONCLUSION:

We built a custom Credit Risk Analyzer and reached highest accuracy of 86.22%. This was achieved with a decision tree classifier with gini criterion and stratified K-Fold cross validation.