2020

Project Report on Finance Risk Analysis

Company Default and Stock Market Analysis

This report covers the default model of company dataset using logistic Regression. It covers EDA, Model building and and model validation.In secod set market risk analysis is done.

Project Report By

Mr.Jaydeep Bhaskar Ashtekar

04/09/2020

Part 1:Credit Risk

Part 2 -Market Risk

Part 1: Credit Risk

1. Exploratory Data Analysis

1.1Basic information of data:

1. Importing libraries:

Imported important basic libraries like pandas, numpy, matplotlib and seaborn

2. Read data:

CSV file data is imported using pandas

3. Since data has many columns so **display option** is set to see 100 columns and 999 rows.

4. Data Size:

Data set has **3586** observations and **24** variables.

5. Deleted company code by dropping it from dataset.

6. All columns names are **cleaned** by replacing space by underscore and removed dots and other special character using str.replace command.

7. **No duplicate** rows are present.

8. Datatype of all variables is **float64.**

9. Description of data:

	Networth_Next_Year	Capital_Employed	Net_Working_Capital	Current_Assets	Total_Sales	Retained_Earnings	PBDT	PBIT	PBT	PBT_as_perc_of_total_income	PAT
count	3586	3586	3586	3576	3586	3586	3586	3574	3586	3586	3586
mean	725	2800	411	1965	1080	49	116	218	86	6	61
std	4770	26975	6301	22609	9997	426	956	1854	800	465	620
min	-8022	-1825	-13162	-1	-63	-449	-5875	-4813	-6032	-13522	-6032
25%	4	8	1	4	1	0	0	0	0	0	0
50%	19	39	10	25	30	0	1	1	0	0	0
75%	124	227	61	135	234	4	13	17	7	4	6
max	111729	714001	223258	721166	443775	14143	23215	41403	16798	15370	13383

	CP	Book_Value_Adj._Unit_Curr	EPS_Annualised_Adjusted_Unit_Curr	APATM_percLatest	Creditors_Velocity_Days	Total_Liabilities	Total_Equity	Market_Value_of_Equity	total_assets	Cost_of_Production	Current_Ratio	Gross_Block
count	3586	3569	3582	3585	3578	3527	3586	3586	3586	3586	3585	3586
mean	92	2251	313	-365	2062	2674	63	1664	2058	799	12	594
std	781	128517	18069	12500	54230	27361	779	12805	23726	9077	108	4872
min	-5875	-33716	0	-688600	0	-1	0	0	0	-23	0	-41
25%	0	7	0	0	8	4	4	0	7	1	1	1
50%	1	19	0	2	39	17	8	8	21	26	1	16
75%	11	60	4	7	89	122	20	111	101	190	3	132
max	20760	7677600	1081400	15267	2034145	652824	42263	260865	653267	419914	4813	128478

Above description gives idea about how is the spread of data.

It can be concluded that **data is spread with outliers** because there is large difference in the mean, maximum and minimum values.

1.2 Outlier and missing value Treatment:

1. Following are imputation for NA values

	Total	Percent
APATM_percLatest	1	0.000279
Book_Value_Adj._Unit_Curr	17	0.004741
Creditors_Velocity_Days	8	0.002231
Current_Assets	10	0.002789
Current_Ratio	1	0.000279
EPS_Annualised_Adjusted_Unit_Curr	4	0.001115
PBIT	12	0.003346
Total_Liabilities	59	0.016453

Strategy to impute the NA/Missing values values is by imputing NA by median. As th e percentage of the NA values is very less so all variables are significant. We can't re move the any variable.

2. Percentage of outliers:

	Number of outliers	perc of outliers
Networth_Next_Year	676	18.85109
Capital_Employed	596	16.62019
Net_Working_Capital	625	17.42889
Current_Assets	576	16.06247
Total_Sales	556	15.50474
Retained_Earnings	603	16.81539
PBDT	815	22.72727
PBIT	722	20.13385
PBT	941	26.24094
PBT_as_perc_of_total_income	874	24.37256
PAT	959	26.74289
CP	816	22.75516
Book_Value_Adj._Unit_Curr	484	13.49693
EPS_Annualised_Adjusted_Unit_Curr	510	14.22197
APATM_percLatest	934	26.04573
Creditors_Velocity_Days	389	10.84774
Total_Liabilities	602	16.78751
Total_Equity	449	12.52092
Market_Value_of_Equity	639	17.8193
total_assets	572	15.95092
Cost_of_Production	560	15.61629
Current_Ratio	565	15.75572
Gross_Block	540	15.05856

Conclusion:

-So it can be said that percentage of outliers is very high so we need to treat these outliers by some imputation.

-All these data points can't be deleted.

-Imputation used here is **capping.** So values above Q3+1.5*IQR will be capped by upper whisker value and values lower than Q1-

1.3. Creating New variables:

Return_on_Total_Asse, Profit_Margin, Debt_to_Equity_Ratio were created and made part of dataset itself.

df['Return_on_Total_Asset']=df['PBT']/df['total_assets']

df['Profit_Margin']=df['PBT']/df['Total_Sales']

df['Debt_to_Equity_Ratio']=df['Total_Liabilities']/df['Total_Equity']

1.4. Treating infinite values:

While generating new variable by ratio, it's necessary to check whether denominator is non-zero or not?

If denominator is zero then we need to treat these zeros.

Total sales have 306 values as zero.

So profit margin is getting –infinity and +infinity values.

The values which gives –infinity is replaced by -0.01 and +infinity by 0.01.

Why 0.01 only? Because nearest value to zero is 0.01.

So all infinite values are removed. Infinite values are checked by command **np.isfinite(df).sum()**

1.5 Transform target variable into 0 and 1:

To identify default we used **Networth_Next_Year.**

If Networth_Next_Year > 0 (Positive) then company will not default and if Networth_Next_Year < 0 (Negative) company will default.

So additional default column is created.

df['Default']=np.where((df['Networth_Next_Year'] > 0), 0, 1)

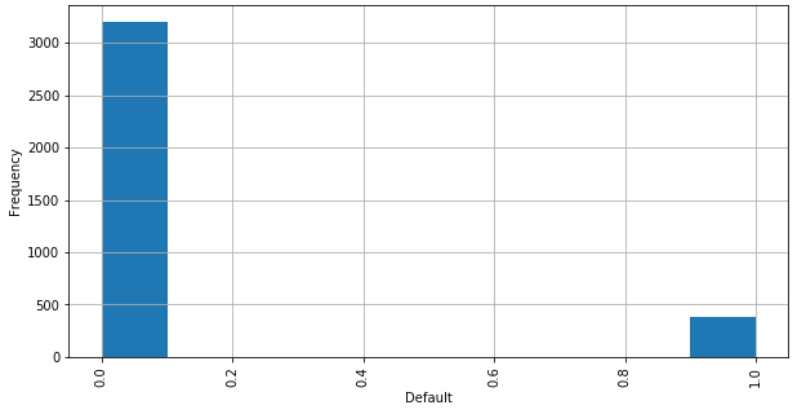
df['Default'].value_counts()

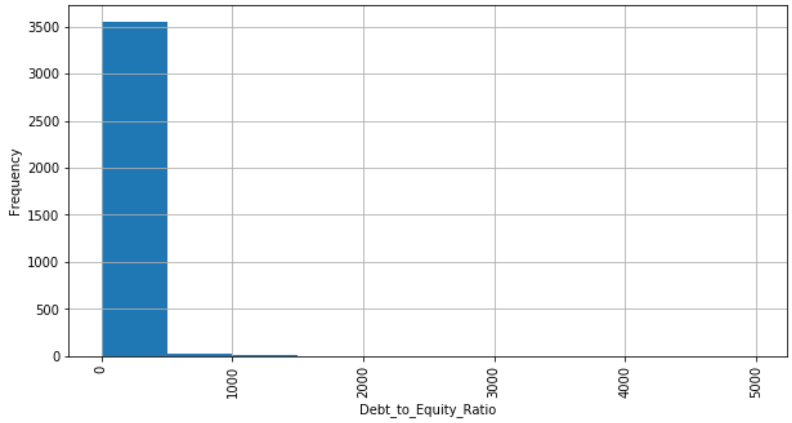
Non defaulters are : 3198(89.18%)

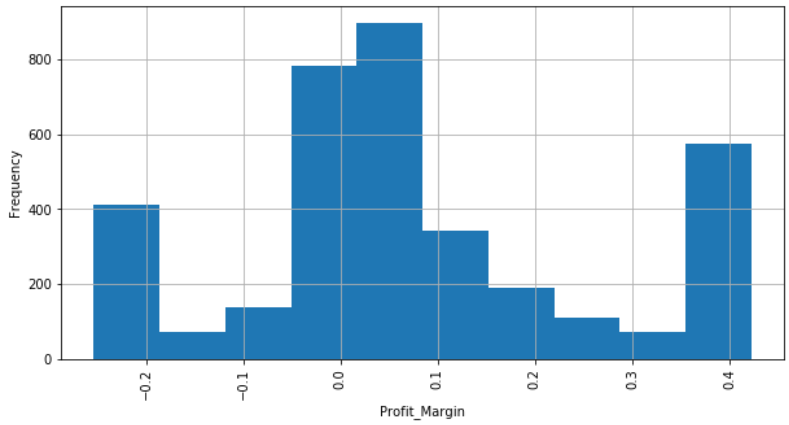
Defaulters are : 388(10.81%)

1.6 Univariate and bivariate analysis:

A.Bar plot(Univariate analysis)







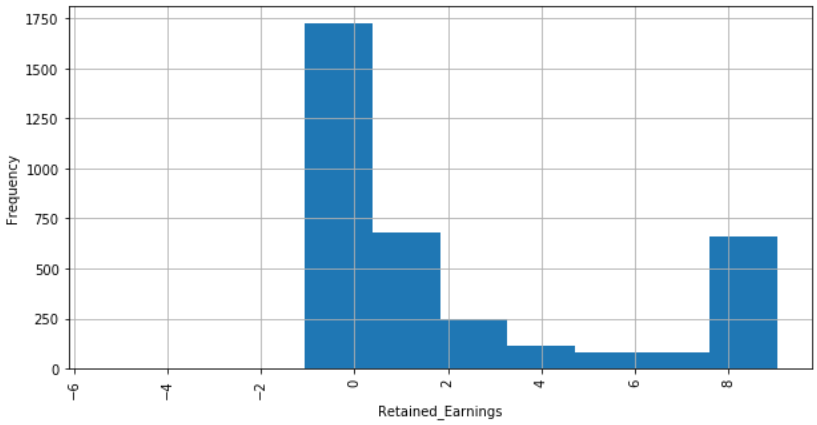
Frequency bar chart:

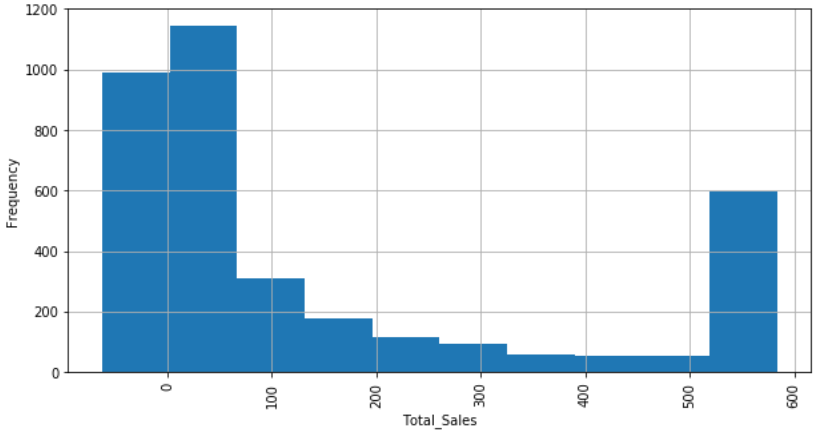
Non Defaulters are more than 3000 and defaulters are less than 500.

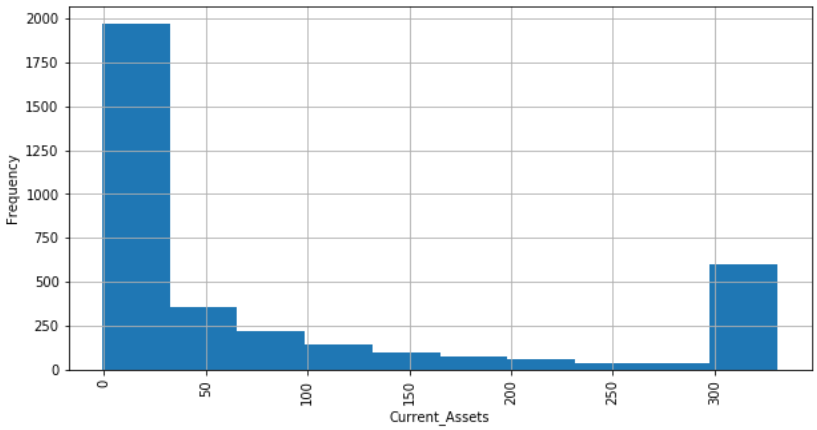
Debt_to_equity_ratio :Almost all values lie in the range of 0-500

Profit margin is partially uniformly distributed from -0.18 to 0.35.

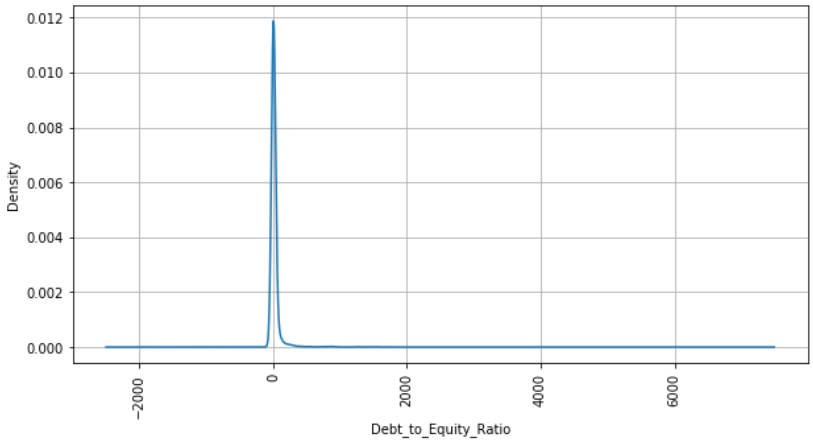
Same can be concluded from following distributions


Frequency
Retained_Earnings

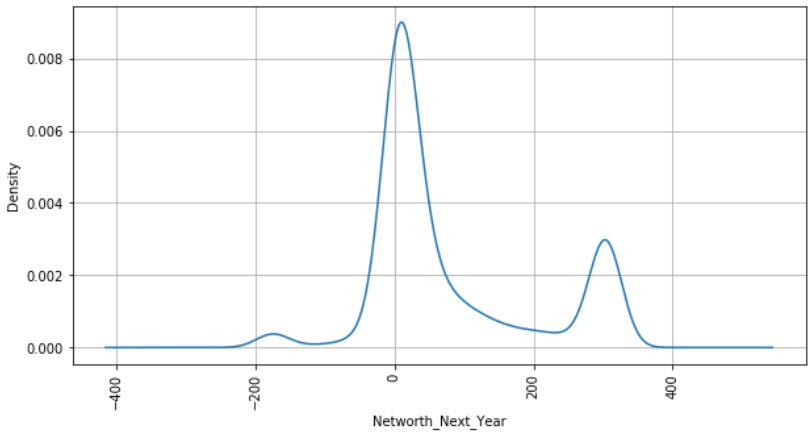

Frequency
Total_Sales


Frequency
Current_Assets

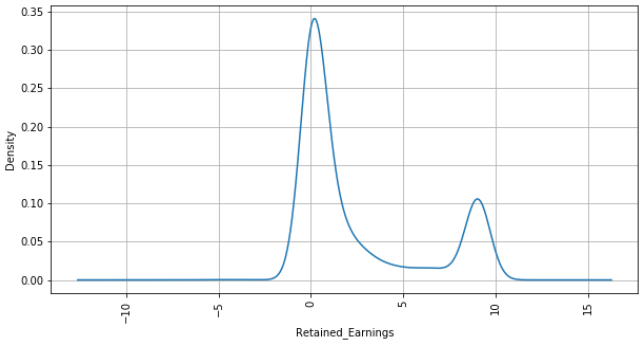
B.KDE plot:(Univariate)

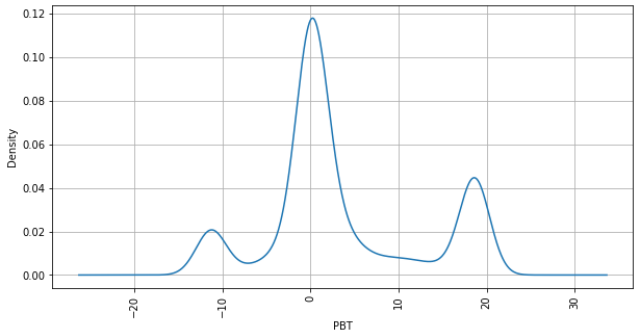


Debt to equity ratio has single mode near to zero and right skewed.

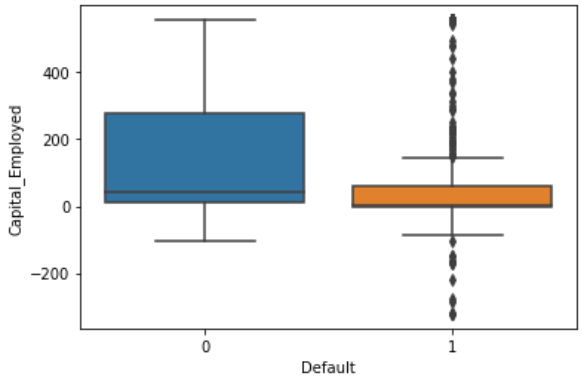


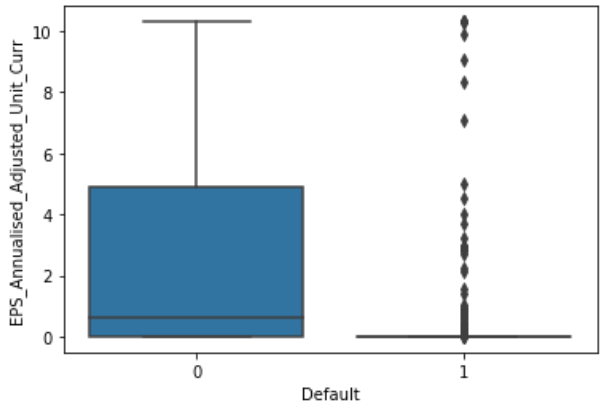
Networth next year is normally distributed with multimode.(3 modes)

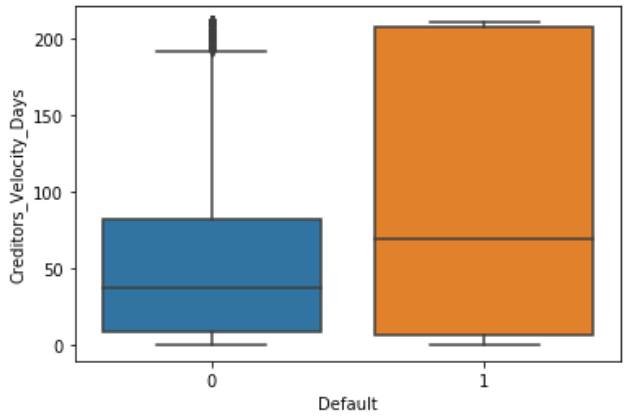




C.Boxplot(Univariate)



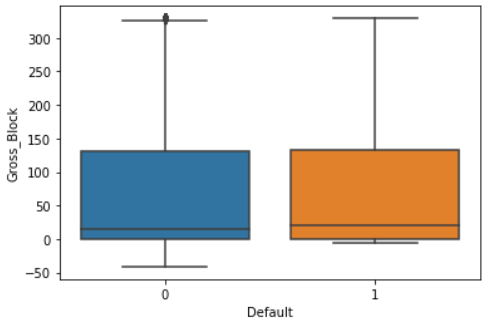




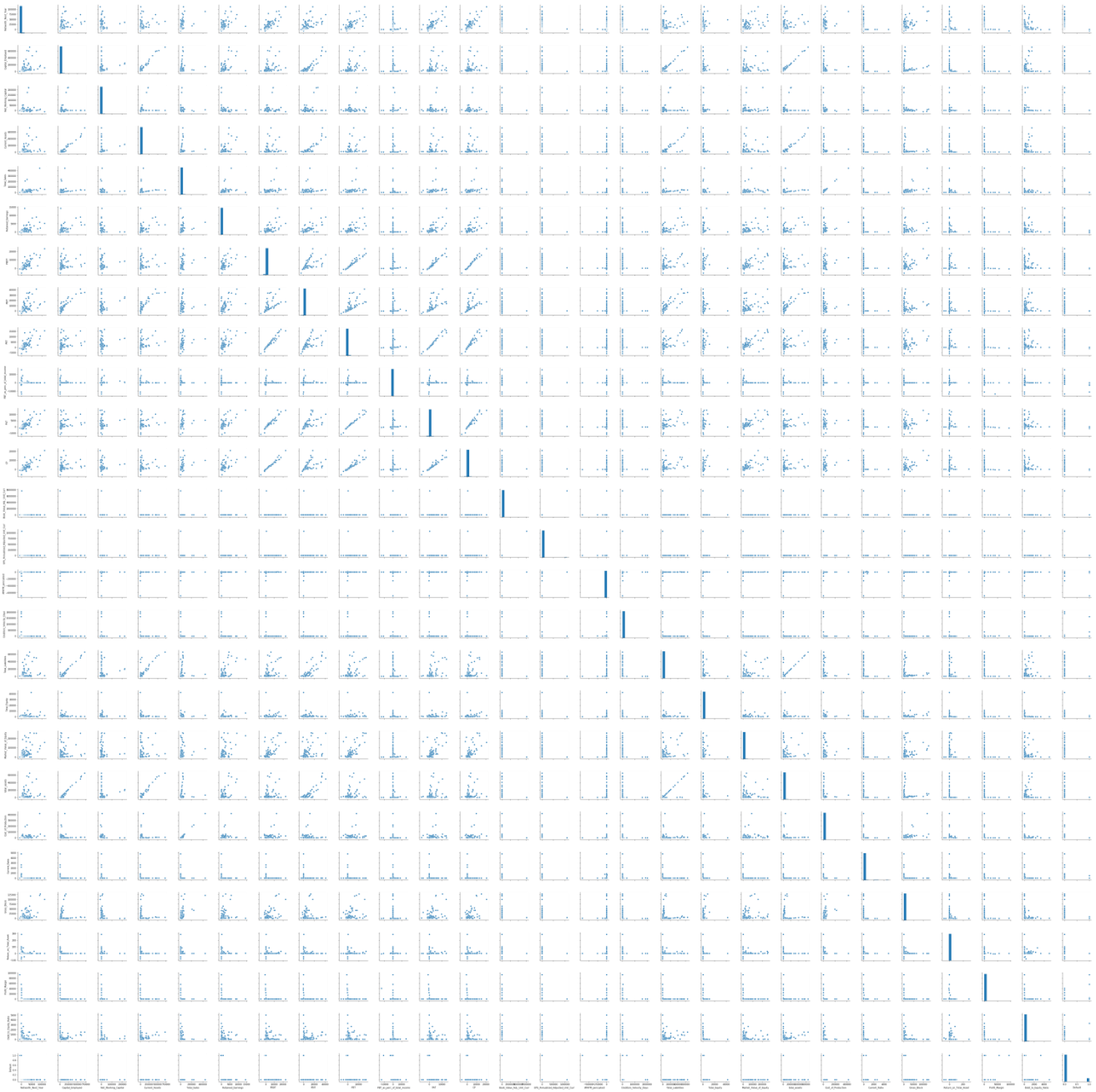
Conclusion:

Almost all variables have outliers for default as shown in figure.

But some variables like creditors velocity days don't have outliers.



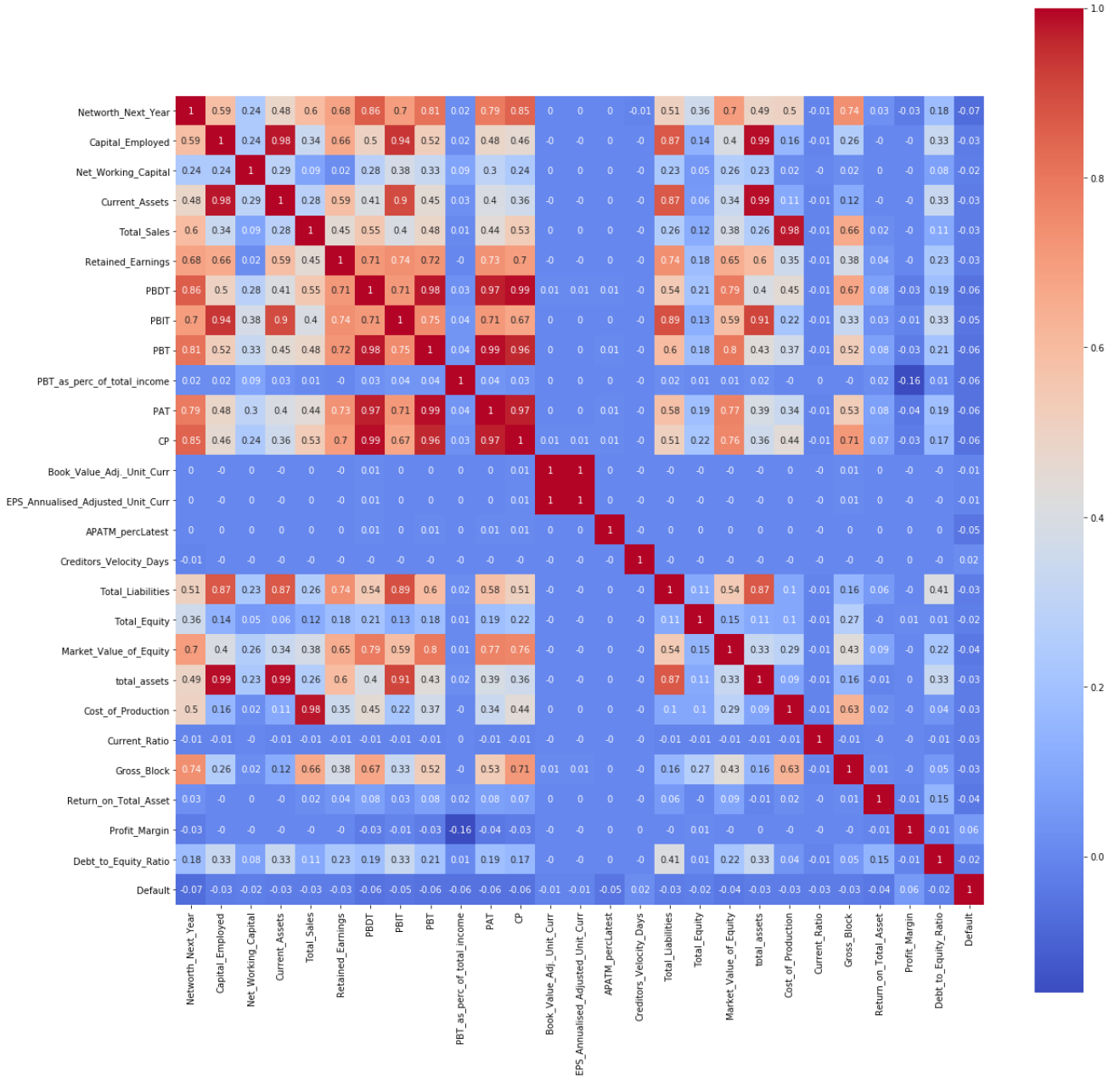
E.Pairplot(Bivariate analysis)



Conclusion:

Here diagonal shows the bar chart of each variable. So before removing outliers distribution was just a bar. After removing outliers distribution is different. It also shows the relation amongst each other.Some variables are related so shows straight relation and some are not related so shows scatters only.

F.Correlation Heatmap:



Following correlation can be concluded:

1.Networth_Next_Year correlated with:	
PBDT	0.86
CP	0.85
PBT	0.81
PAT	0.79

2.Capital_Employed correlated with	
total_assets	0.99
Current_Assets	0.98
PBIT	0.94
Total_Liabilities	0.87

3.Current_Assets is correlated with	
total_assets	0.99
Capital_Employed	0.98
PBIT	0.90
Total_Liabilities	0.87

4.Total_Sales is correlated with	
Cost_of_Production	0.98

5.Retained_Earnings is correlated with	
Total_Liabilities	0.74
PBIT	0.74

6.PBT is positively correlated with	
PAT	0.99
PBDT	0.98
CP	0.96
Networth_Next_Year	0.81
Market_Value_of_Equity	0.80
PBIT	0.75

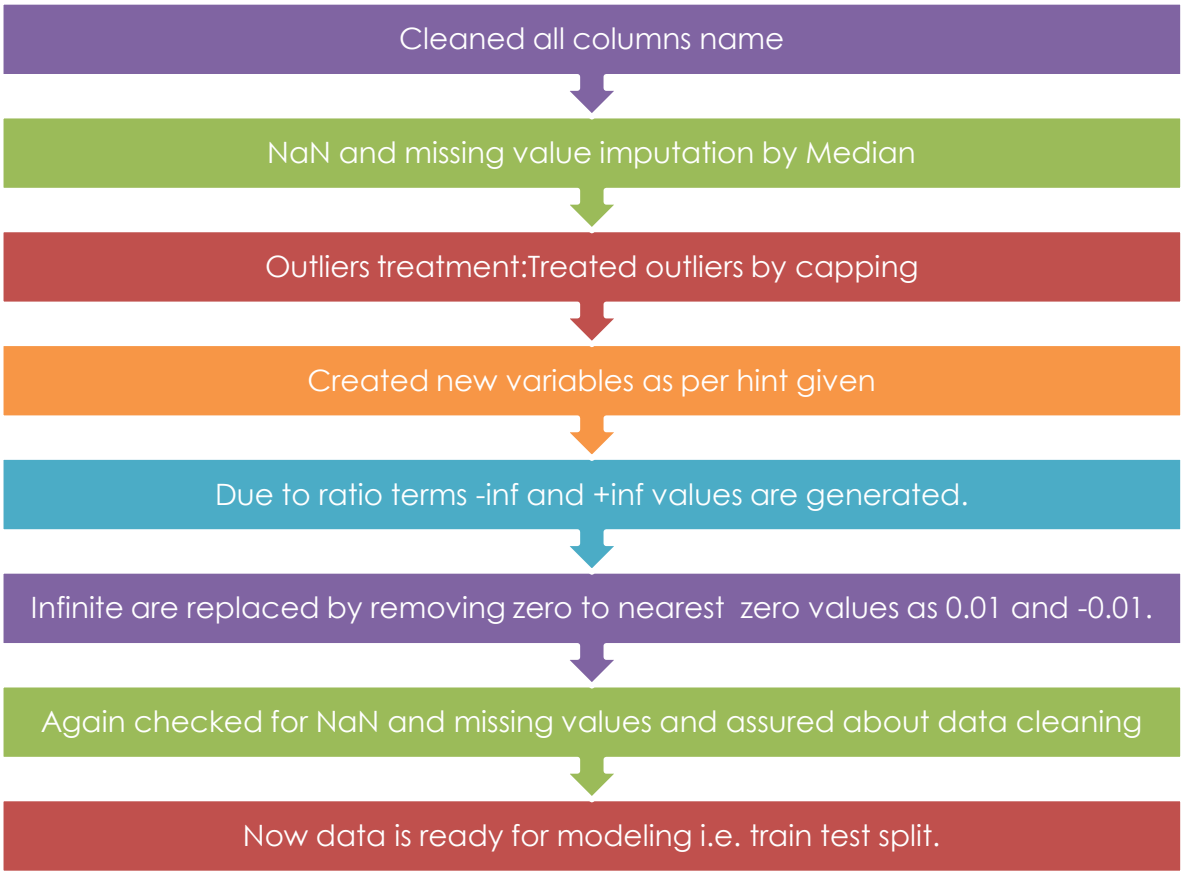
7.EPS_Annualised_Adjusted_Unit_Curr is completely correlated wth

Book_Value_Adj._Unit_Curr	1.00

8.Market_Value_of_Equity is correlated with	
PBT	0.80
PBDT	0.79
PAT	0.77
CP	0.76

1.7 Perform Train Test split:

Before doing train test split we did:



Data is split in X and y data:

X = df.drop('Default', axis = 1)

```
y= df[ 'Default']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.33, random_state =
42)
```

Ration of train test is 67:33.

2. Modeling

2.1. Build a Logistic Regression Model on most important variables on Train Dataset:

Selection of Important Variable:

As we saw there are many variables which are correlated amongst each other. In model if we consider all variables which may lead to multicolinearity.

To avoid this we are removing variables with respect to VIF score.

```
X = df.drop('Default', axis = 1)

calc_vif(X).sort_values(by = 'VIF', ascending = False)
```

Using Above code we got VIF Score of each variable.PBT has highest score of 107.89 so removed it.Again same procedure is followed and finally we got following important set of variables

	variables	VIF
2	Book_Value_Adj__Unit_Curr	1.966202
0	Net_Working_Capital	1.639187
5	Current_Ratio	1.622968
3	APATM_percLatest	1.571040
1	PBT_as_perc_of_total_income	1.534274
6	Profit_Margin	1.533158
4	Creditors_Velocity_Days	1.335486
7	Debt_to_Equity_Ratio	1.063417

Using above **8** variables model is made as:

formula = 'Default ~ Net_Working_Capital+PBT_as_perc_of_total_income+Book_Value_Adj__Unit_Curr+APATM_percLatest+Creditors_Velocity_Days+Current_Ratio+Profit_Margin+Debt_to_Equity_Ratio'

Reults of above logit model is as follows:

Logit Regression Results

Dep. Variable:	Default	**No. Observations:**	2402
Model:	Logit	**Df Residuals:**	2393
Method:	MLE	**Df Model:**	8
Date:	Sun, 06 Sep 2020	**Pseudo R-squ.:**	0.6042
Time:	14:02:31	**Log-Likelihood:**	-313.24
converged:	True	**LL-Null:**	-791.34
Covariance Type:	nonrobust	**LLR p-value:**	4.215e-201

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
Intercept	-0.7060	0.198	-3.575	0.000	-1.093	-0.319
Net_Working_Capital	0.0034	0.003	1.341	**0.180**	-0.002	0.008
PBT_as_perc_of_total_income	-0.1790	0.036	-5.035	0.000	-0.249	-0.109
Book_Value_Adj__Unit_Curr	-0.1159	0.011	-10.889	0.000	-0.137	-0.095
APATM_percLatest	-0.0610	0.015	-3.959	0.000	-0.091	-0.031
Creditors_Velocity_Days	0.0007	0.001	0.539	**0.590**	-0.002	0.003
Current_Ratio	-0.5800	0.089	-6.514	0.000	-0.755	-0.406
Profit_Margin	1.2201	0.477	2.560	0.010	0.286	2.154
Debt_to_Equity_Ratio	0.0003	0.003	0.122	**0.903**	-0.005	0.006

Again it is found that P>|z| is greater than 0.05(significance value) for Net_Working_Capital,Creditors_Velocity_Days and Debt_to_Equity_Ratio.So these variables are not significant as far as default prediction is concerned.

So finally important variables are:

PBT_as_perc_of_total_income,

Book_Value_Adj__Unit_Curr

APATM_percLatest,

Current_Ratio

Profit_Margin.

2.2 State the accuracy, specificity, and sensitivity of the model based upon the optimized cutoff value.

Three logit Models are considered for the dataset:

1. Model with Threshold 0.5

2. Model with optimum threshold

3. Model with SMOTE and optimum threshold

Sr. No.	Model	Accuracy	Specificity	Sensitivity
1.	Model with Threshold 0.5(Train Data Set)	0.959	0.661	0.993
2.	Model with Threshold 0.1512(Optimum)(Train Data Set)	0.917	0.894	0.92
3.	SMOTE Model with Thres hold 0.6196 (Optimum) (Train Data Set)	0.925	0.914	0.936

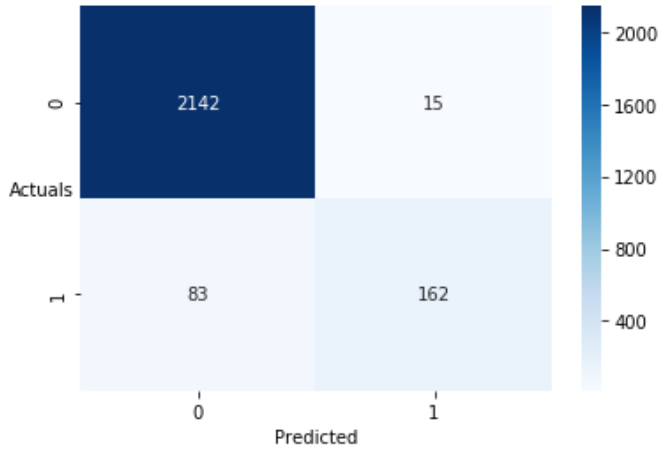
3. Model validation

3.1Validate the Model on Test Dataset and state the performance matrices:

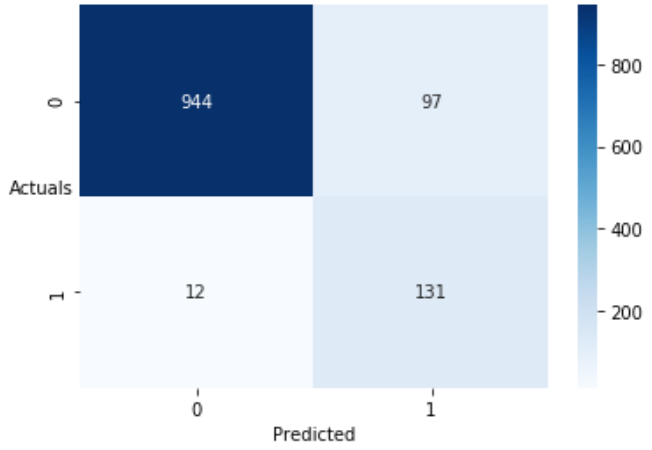
Above model is used on test data and results are predicted as below:

1. Performance Matrices:

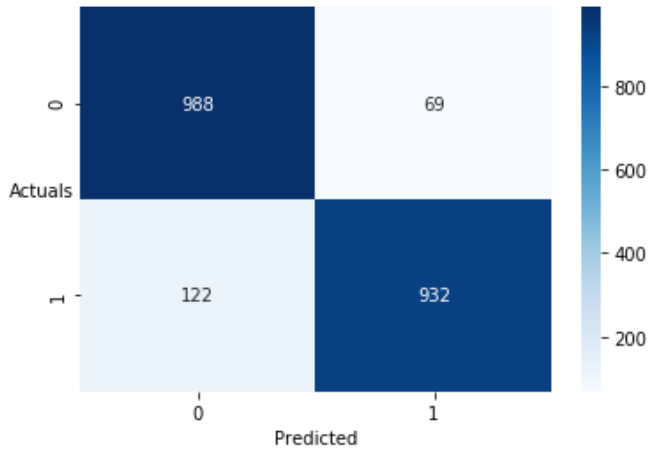
With 0.5 as threshold:



With 0.1527 as threshold on test data without SMOTE:



With 0.6196 as threshold on test data with SMOTE:



Comparison of Each Model:

Sr. No.	Model	Accuracy	Specificity	Sensitivity
1.	Model with Threshold 0.5(Train Data Set)	0.959	0.661	0.993
2.	Model with Threshold 0.1512(Optimum)(Train Data Set)	0.917	0.894	0.92
	Model with Threshold 0.1512(Optimum)(Test Data Set)	0.908	0.916	0.907
3.	SMOTE Model with Thres hold 0.6196 (Optimum) (Train Data Set)	0.925	0.914	0.936
	SMOTE Model with Thres hold 0.6196 (Optimum)(Test Data Set)	0.910	0.884	0.935

Conclusion:

Model with SMOTE gives good results with improved accuracy, specificity and Sensitivity.

Sensitivity and Specificity:

As the value of threshold was changed and model used is with SMOTE sensitivity and specificity will be changed.

Ideally we need to think both specificity and sensitivity.But not always possible.There exists some trade off and we need to select optimum value.Here our model really worked well.

When Sensitivity is a High Priority

When we have to predict bad customers for non approving loan at that time sensitivity will be a high priority.

Sensitivity=TP/(TP+FN)

TP means who will default/bad customer. Means we have to identify the bad customer so sensitivity will be an important facort here.

Our model identifies bad customer as bad 94% accurate and good customer as good 88% accurate.

3.2 Find out the Altzman's Score on test and train dataset:

For whole dataset Altzman's score is calculated:

df['A']=df['Net_Working_Capital']/df['total_assets']

df['B']=df['Retained_Earnings']/df['total_assets']

df['C']=df['PBIT']/df['total_assets']

df['D']=df['Market_Value_of_Equity']/df['Total_Liabilities']

df['E']=df['Total_Sales']/df['total_assets']

and

df['z_Altzman_score']=1.2*df['A'] + 1.4*df['B'] + 3.3*df['C'] + 0.6*df['D'] + 1.0*df['E']

and depending upon above score following is obtained:

Healthy	57.7803
Unhealthy	30.8701
Intermediate	11.3497

So Healthy+Intermediate are Non Default Companies=57.78+11.35=69.13% of total companies will default as per Altzman Score.

and 30.87% companies will default because their status is unhealthy.

3.3 Compare the Altzman's Score with the Logistic Regression Model:

Sr.No.	Alzman's Score	LR model
1.	Altzman Score uses the A,B,C,D,E as values for the prediction state of the company as Healthy Unhealthy and intermediate.	It uses the most significant variables to predict default or not
2.	This model consider only values which are related to find A,B,C,D,E etc.	It may happen that some of the parameters of the Altman's Model may be insignificant while building model.
3.	It just gives idea about the health of company only.	It gives good accuracy upto 90%.

3.4 Scope:

Many trial and error methods can be used to improve model. Some of them are described as below:

The further extension to the project can be:

A. Cluster Analysis: Clustering companies and can be segmented into certain categories.

B. Improving Model:

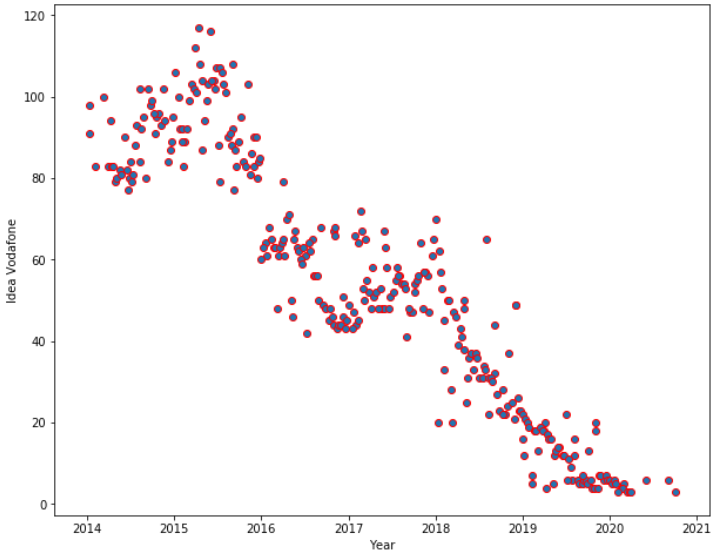
1. Model can be further improved by various classification techniques.

2. PCA technique can be used to reduce columns

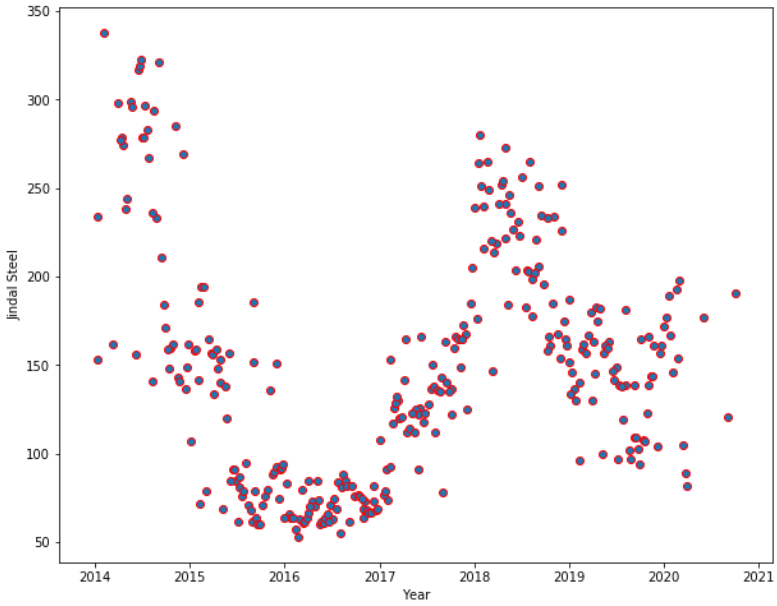
3. Combining Altzman's score and then analyzing data

Part 2 -Market Risk

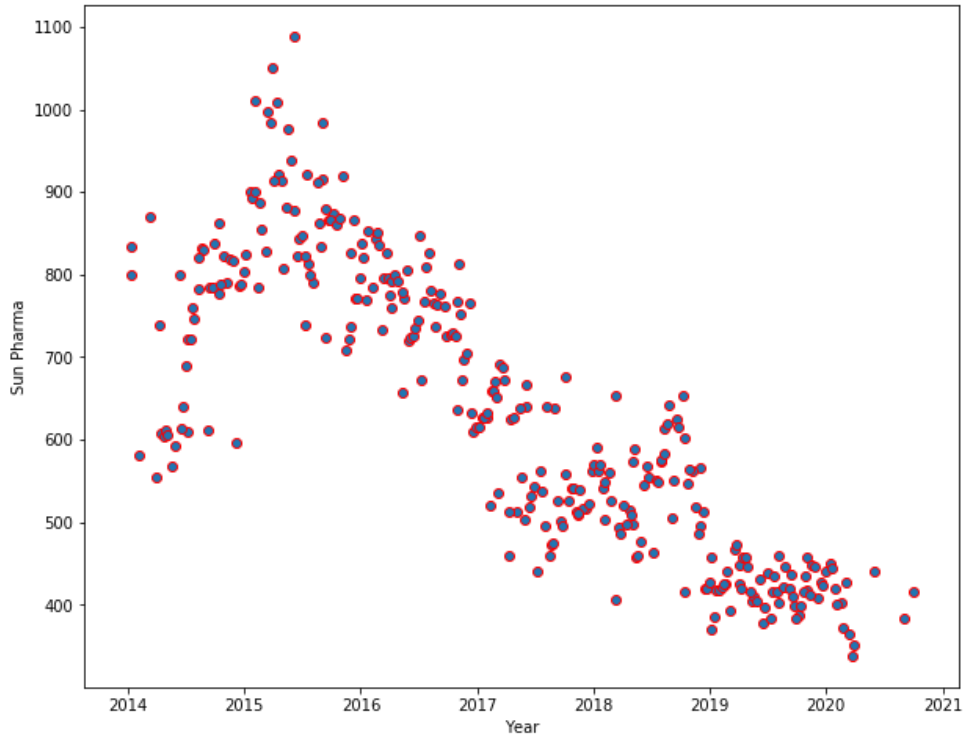
1. Draw Stock Price Chart for any 2 variables

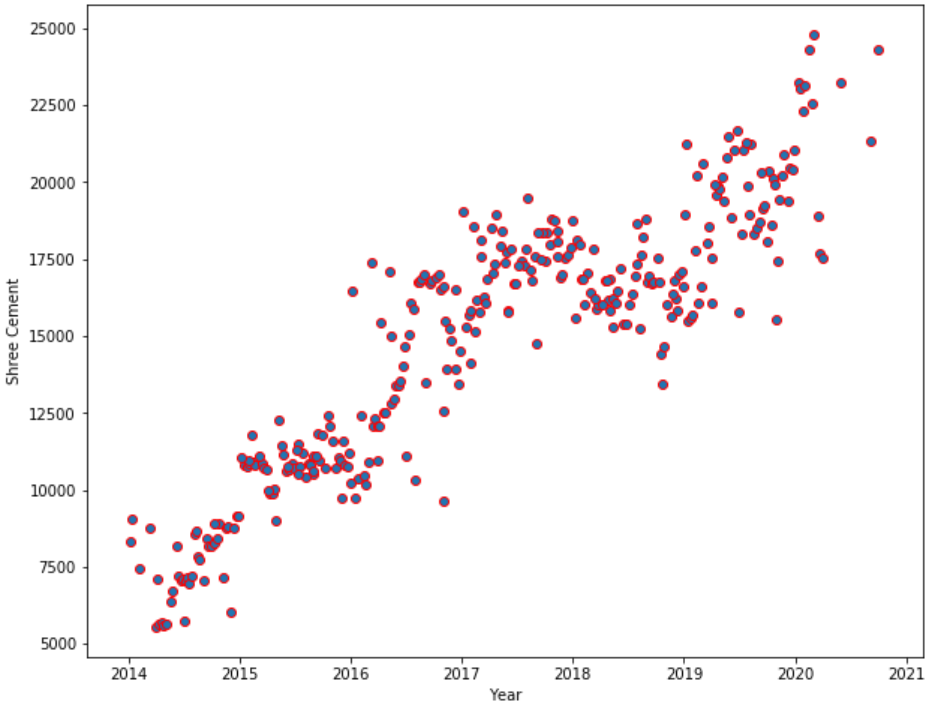


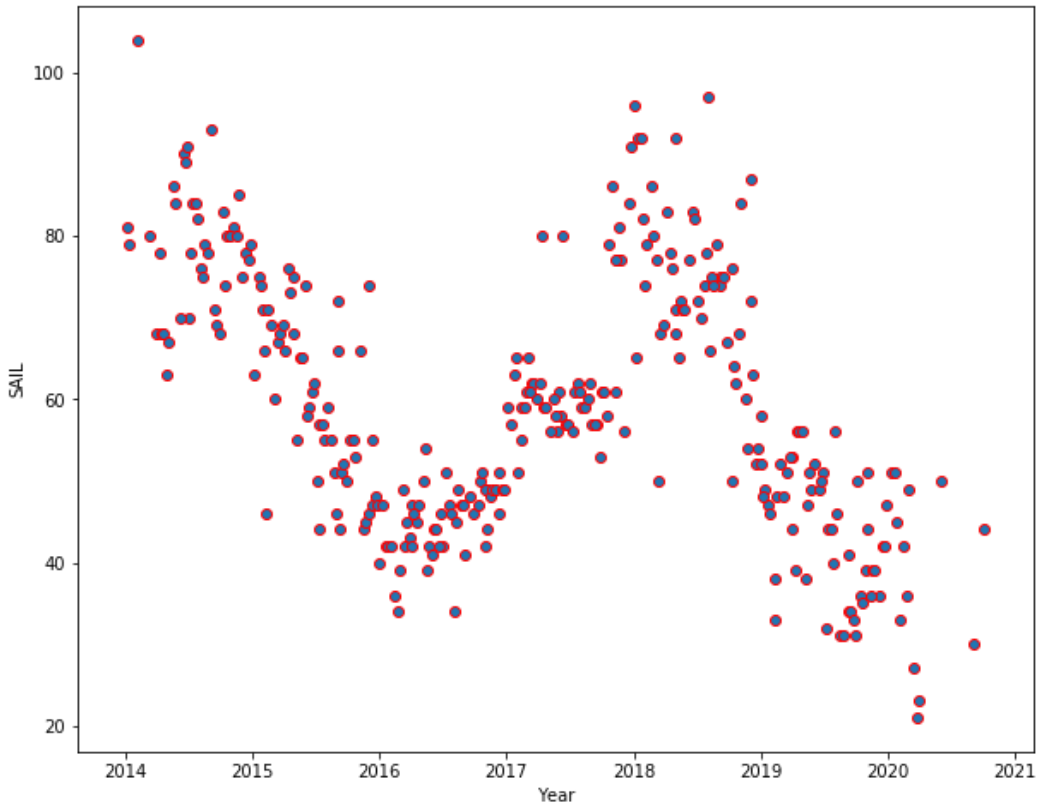
Idea Vodafone Stock value is reducing continuously

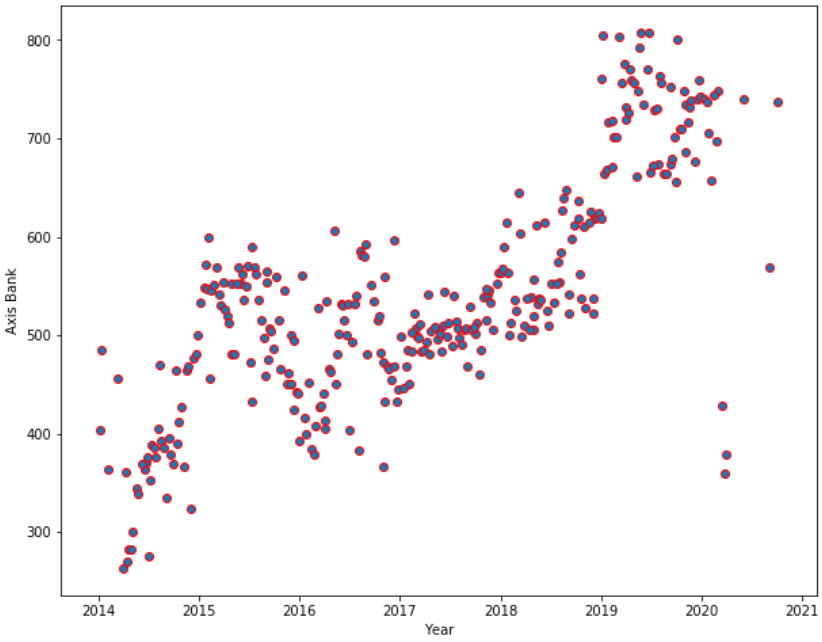


Jindal steel got fall in 2016 and raised in 2018 mid.then Again start fall and now raising.

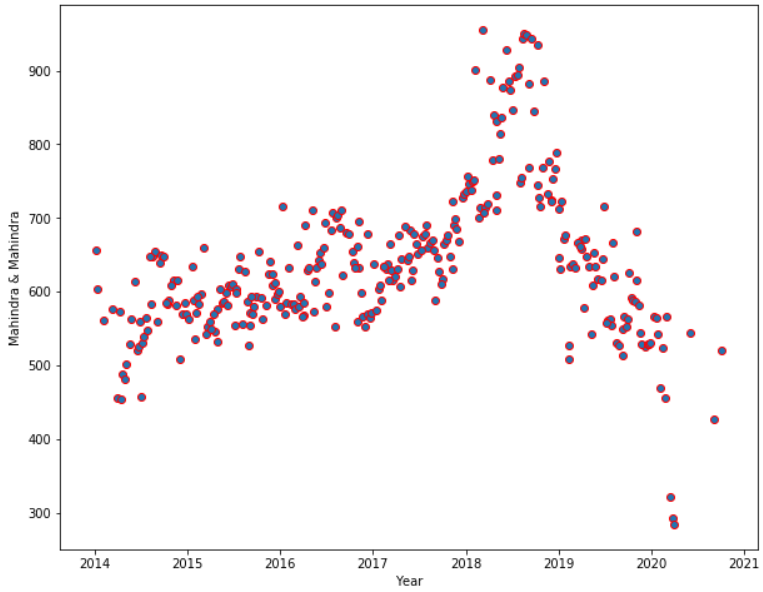

Sun Pharma
Year
1100
1000
900
800
700
600
500
400
2014
2015
2016
2017
2018
2019
2020
2021

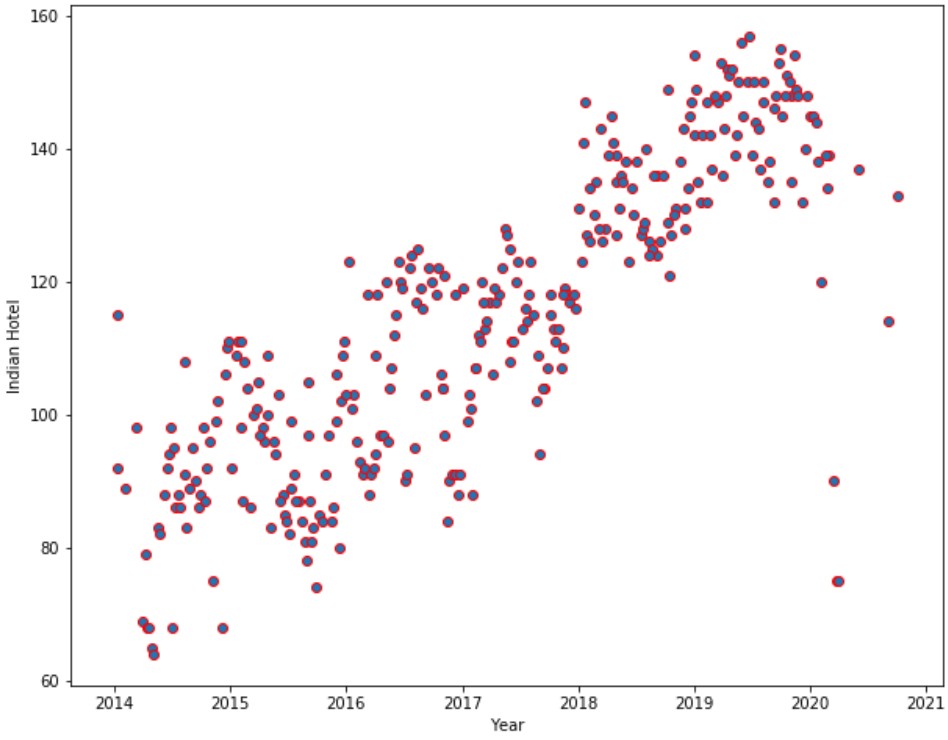

Shree Cement
Year
25000
22500
20000
17500
15000
12500
10000
7500
5000
2014
2015
2016
2017
2018
2019
2020
2021

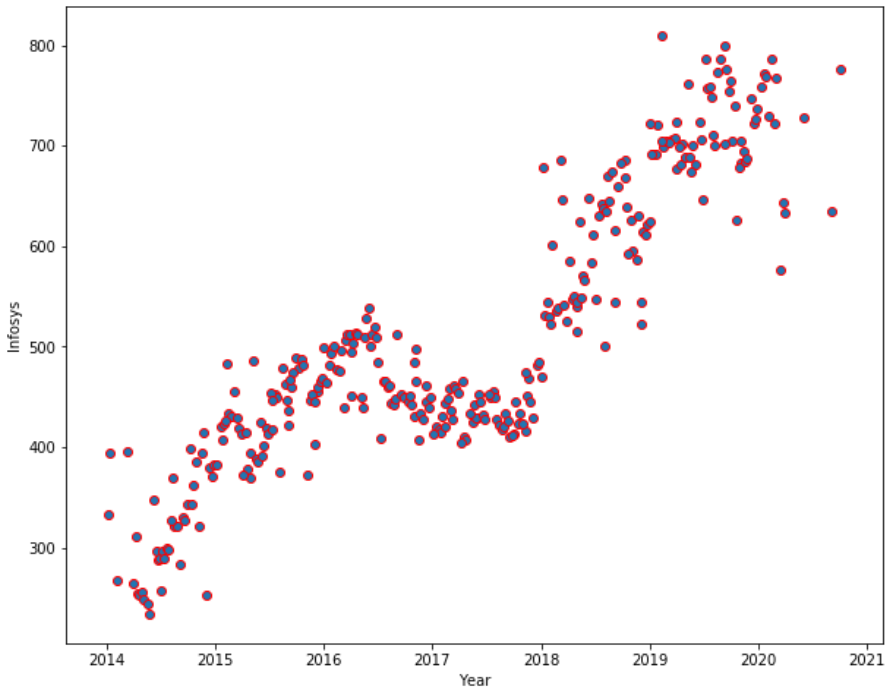


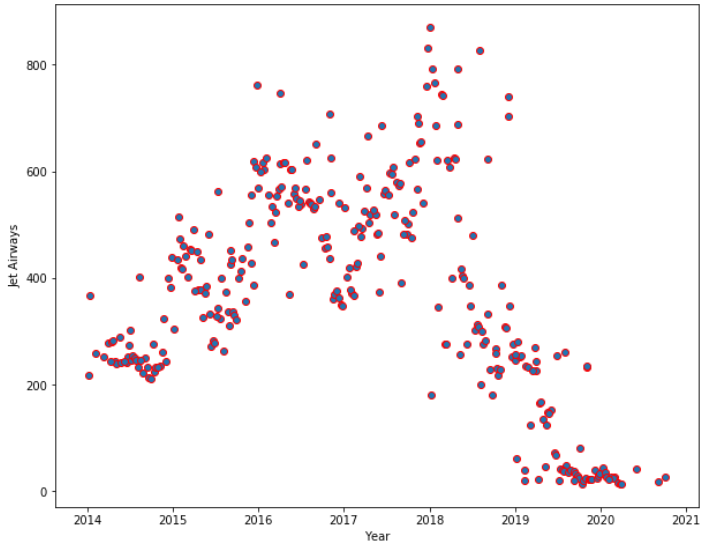


For Axis bank stock price is increasing and fail down for some weeks in 2020


Mahindra & Mahindra
900
800
700
600
500
400
300
2014
2015
2016
2017
2018
2019
2020
2021
Year


Indian Hotel
160
140
120
100
80
60
2014
2015
2016
2017
2018
2019
2020
2021
Year





2.Calculate Returns

Returns are calculated using log difference as follows:

stock_returns = np.log(stock_prices.drop(['Date','dates'],axis=1)).diff(axis = 0, periods = 1)

Positive returns means we get returns from the investments and negative value means we will be in loss.

3.Calculate Stock Means and Standard Deviation.

Stock Means:

Infosys	0.002794
Indian Hotel	0.000266
Mahindra & Mahindra	-0.001506
Axis Bank	0.001167
SAIL	-0.003463
Shree Cement	0.003681
Sun Pharma	-0.001455
Jindal Steel	-0.004123
Idea Vodafone	-0.010608
Jet Airways	-0.009548

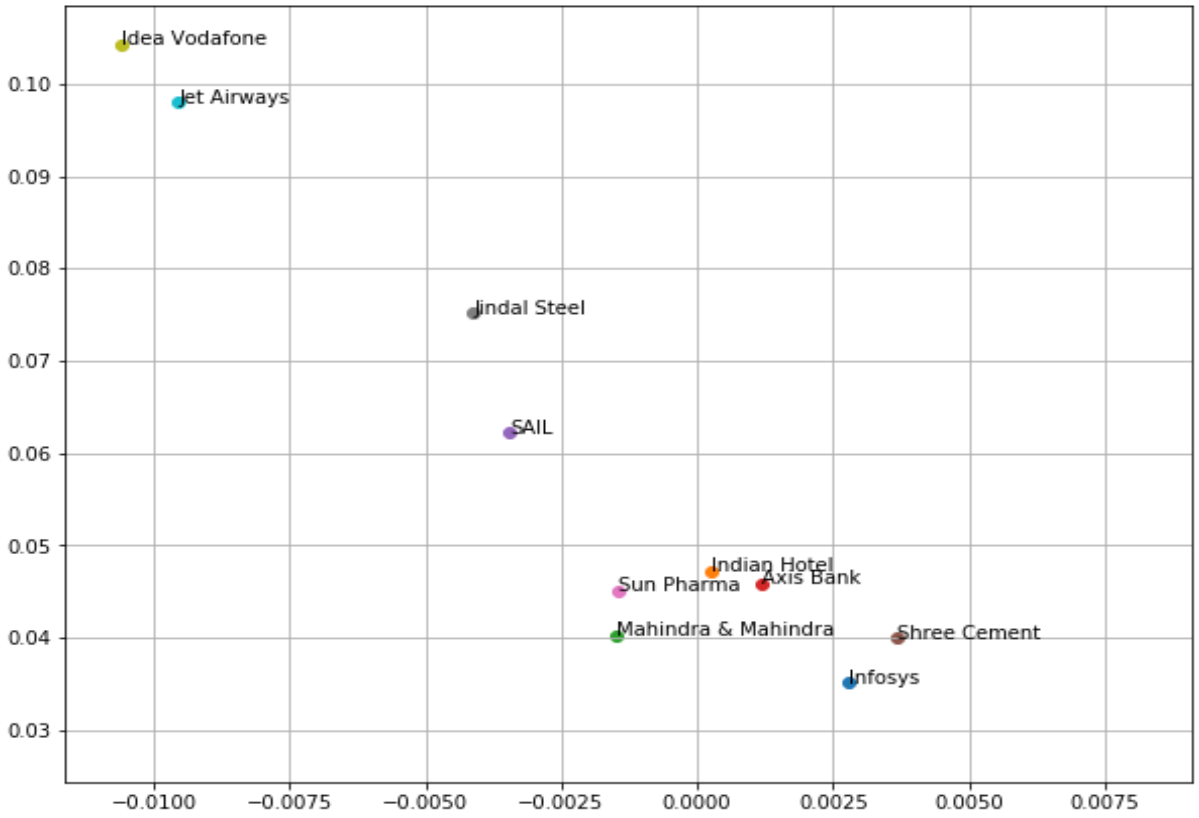
Stock Deviation:

Infosys	0.035070
Indian Hotel	0.047131
Mahindra & Mahindra	0.040169
Axis Bank	0.045828
SAIL	0.062188
Shree Cement	0.039917
Sun Pharma	0.045033
Jindal Steel	0.075108
Idea Vodafone	0.104315
Jet Airways	0.097972

4. Draw a plot of Stock Means vs Standard Deviation and share insights.

```
col=Index(['Infosys', 'Indian Hotel', 'Mahindra & Mahindra', 'Axis Bank', 'SAIL',
       'Shree Cement', 'Sun Pharma', 'Jindal Steel', 'Idea Vodafone',
       'Jet Airways'],
      dtype='object')

plt.figure(figsize = (10, 8))
plt.grid()
for i in range(len(col)):
    plt.scatter(stock_means[col[i]], stock_sd[col[i]])
    plt.annotate(col[i], (stock_means[col[i]], stock_sd[col[i]]))
```



So from above graph it can be concluded that investment is beneficiary in following order: