Risk, Return and Prediction of Stock Prices using Regression Analysis

A Project Report on topic: "*Data Science with Python*"

By Sidharth Rai

Student, (UID: 16BCS7045), Bachelor of Engineering in Computer Science and Engineering with specialization in Cloud Computing in association with IBM, Chandigarh University, Mohali, Punjab (140-413). E-mail: raisidharth97@gmail.com

Abstract

Being the topic of interest for those who are interested in investing in share market and share exchanges because benefits here are not easy to calculate but easy to earn. Technical Analysis of jumping prices, sometimes high sometimes low, which not only depends on company businesses but also depends on company related news, political, social, economic conditions and natural disasters. Many researchers and financers have been acknowledged for the daily behavior and prediction of future prices, but some or the other way their models have not been totally accurate. This project is based on a complete mathematical analysis using Ordinary Least Squares model [4] which is used to calculate the unknown parameters in a linear regression model because of its simplicity and wide acceptability. This model is not highly accurate but this model gives the idea itself how much accurate it will be.

1. Introduction

Prediction of Stock Prices is already a complex issue in financial institutions. These prices depend on various factors like history of the Company, its present news, countries policies, social and economic conditions, natural disasters and many other factors. So, the prediction tasks require not only mathematical analysis but a neural network to make it more accurate. Here in this project I had compared top 5 companies of top 10 sectors of Bombay Stock Exchange(BSE) in India from 2017 on the basis of their Market Capitalization. [1] The comparison is done on historical closed prices from 1st Jan 2018 to 13th June 2018, this prediction can also be done in real time also. This project mainly uses the *pandas* [5] library for analysis and matplotlib [6] library for plotting in Python. With the use of Ordinary Least Squares Model the prediction can be analyzed easily just observing some values.

This project can completely be downloaded from the provided link:

https://github.com/SidharthRai/Regression_Analysis_Project

1.1 List of Sectors and Companies analyzed [2]

Rank	Sectors	Companies	Symbol BSE	Market Cap (Rs. cr)
1	Bank – Private	HDFC Bank	*500180*	530331.19
		Kotak Mahindra	*500247*	253504
		ICICI Bank	*532174*	183043.34
		Axis Bank	*532215*	136496.98
		IndusInd Bank	*532187*	118075.7
2	Engineering - Construction	DLF	*532868*	36698.63
		Godrej Prop	*533150*	17595.8
		Oberoi Realty	*533273*	17272.17
		Prestige Estate	*533274*	10318.13
		ISGEC Heavy Eng	*533033*	4053.28
3	Pharmaceuticals & Drugs	Sun Pharma	*524715*	137013.43
		Cipla	*500087*	47338.22
		Piramal Enter	*500302*	43802.71
		Cadila Health	*532321*	40867.8
		Lupin	*500257*	40630.5
4	Cement & Construction Materials	UltraTechCement	*532538*	103598.13
		Shree Cements	*500387*	56958.86
		Ambuja Cements	*500425*	40894.36
		ACC	*500410*	24545.67
		Dalmia Bharat	*533309*	22496.05
5	Finance – NBFC	Bajaj Finserv	*532978*	95920.27
		ICICI Prudentia	*540133*	57501.16
		L&T Finance	*533519*	33468.81
		Bajaj Holdings	*500490*	32690.24
		Indiabulls Vent	*532960*	26161.9
6	Automobiles	Maruti Suzuki	*532500*	271091.18
		M&M	*500520*	114516.68
		Hind Motors	*500500*	150.23
		Bajaj Auto	*532977*	83585.11
		Hero Motocorp	*500182*	73575.7
7	Chemicals	Pidilite Ind	*500331*	55026.33

		UPL	512070	34839.04
		Tata Chemicals	500770	18863.43
		Solar Ind	532725	10602.27
		Aarti Ind	524208	10174.7
8	Metal - Non Ferrous	Hind Zinc	500188	127710.27
		Hind Copper	513599	6744.84
		Tinplate	504966	1827.5
		Gravita India	533282	1075.57
		Precision Wires	523539	649.99
9	Telecommunication Services	Bharti Airtel	532454	150362.21
		Idea Cellular	532822	27336.05
		Tata Comm	500483	17392.13
		Reliance Comm	532712	4397.2
		MTNL	500108	1121.4
10	Tyres & Allied	MRF	500290	31919.01
		Balkrishna Ind	502355	21637.03
		Apollo Tyres	500877	15768.56
		Ceat	500878	5439.12
		JK Tyre & Ind	530007	2922.49

1.2 Risk and Return Calculation

Plotting a graph of several companies in the same sector between their mean and standard deviation gives the middle term behavior of the stock prices out there. The mean value gives the central term tendency which helps to analyze the return of investment in the company and the standard deviation gives the deviating prices deviation from there mean value which results in observing the risk of investment in the particular company.

An example from the project explains the above context.

- The Risk and Return analysis graph with Risk values on Y-axis and Return Values on X-axis.

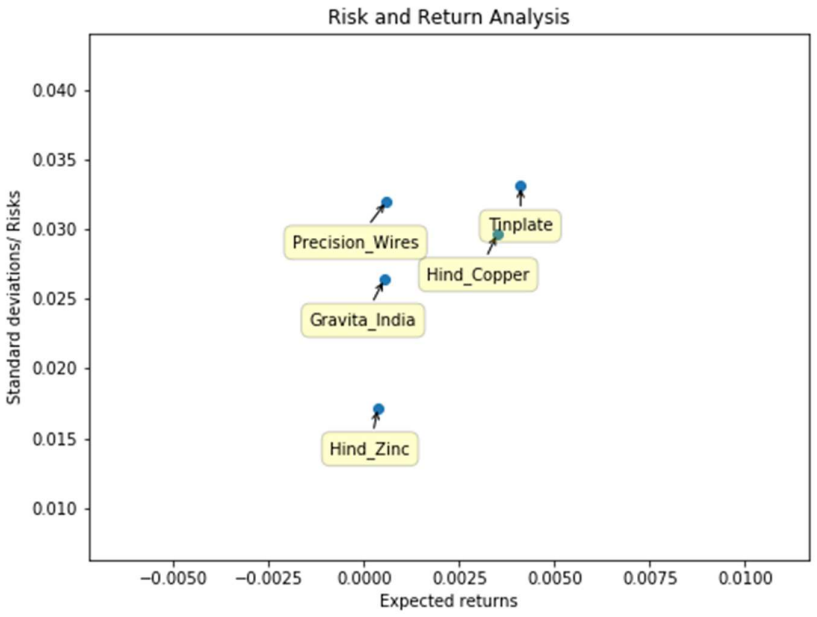

Figure 1 Risk and Return Analysis

- These labeled points clearly explain that Hind Zinc has low Return values as well as shows low risk in investment.
- The Tinplate company gives high return values and also has high risks to invest.
- Similar results on Risk and Return can be analyzed from the graph.
- This Results can help an investor to see if the particular company is worth to invest or not.

1.3 Ordinary Least Square Model (OLS) [4]

It is also known as linear least squares model. This model is a module of Statsmodel [7] library of Python statistically-oriented approaches to data analysis, with an emphasis on econometric analyses. It integrates well with the pandas [5] and numpy libraries. It is used for estimation of parameters which are unknown in any linear regression model. OLS chooses the parameters of a linear function from the dataset of variables by reducing the sum of the squares of the differences between the observed dependent variable to those predicted by the chosen linear function. Geometrically, it is observed as the smaller the differences between the sum of squared values, the better the model fits the data. The estimation is then plotted on the graph with the estimated values and original values. In this project the prediction and calculations are done on the basis of

An example from the project explains the above context:

- The Y-axis in the graph below shows the closed prices and the X-axis below shows the monthly dates.
- The Blue line shows the plotted curve according to the closed prices of Company.
- The Yellow line shows the plotted curve according to the fitted values from the model according to the linear regression formula.

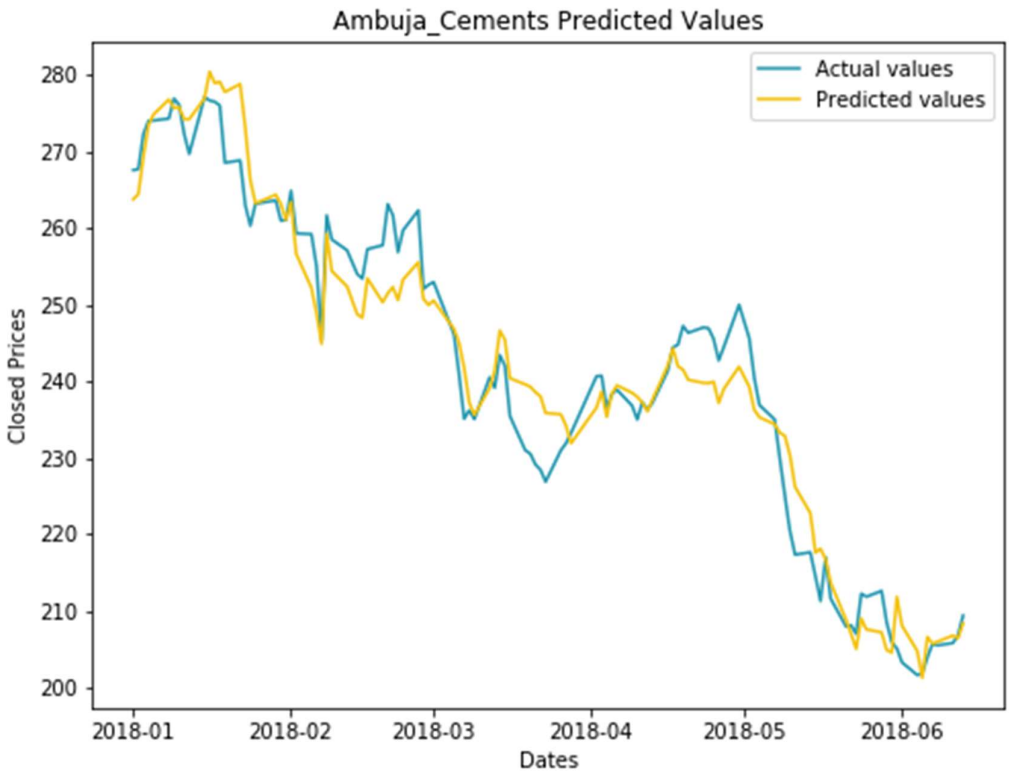

Figure 2 Predictive Values Example

The Accuracy of the model depends on difference between R-squared value and adjusted R-squared value.

1.3.1 Explanation of the result of the OLS model which is of the following way [4]:

An example from the project explains the model and its output:

OLS Regression Results

Dep. Variable:	Ambuja_Cements	**R-squared:**	0.952
Model:	OLS	**Adj. R-squared:**	0.950
Method:	Least Squares	**F-statistic:**	526.3
Date:	Sat, 16 Jun 2018	**Prob (F-statistic):**	2.18e-69
Time:	15:25:19	**Log-Likelihood:**	-333.64
No. Observations:	112	**AIC:**	677.3
Df Residuals:	107	**BIC:**	690.9
Df Model:	4		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	32.0070	14.953	2.141	0.035	2.365	61.649
UltraTechCement	-0.0056	0.006	-0.910	0.365	-0.018	0.007
Shree_Cements	0.0008	0.001	0.807	0.421	-0.001	0.003
ACC	0.1486	0.009	15.776	0.000	0.130	0.167
Dalmia_Bharat	-0.0051	0.005	-0.975	0.332	-0.015	0.005

Omnibus:	1.700	**Durbin-Watson:**	0.302
Prob(Omnibus):	0.427	**Jarque-Bera (JB):**	1.741
Skew:	-0.250	**Prob(JB):**	0.419
Kurtosis:	2.648	**Cond. No.**	5.79e+05

Figure 3 Example OLS Result

The left part of the first table provides basic information about the model fit:

- **Dep. Variables**

It is abbreviated as Depth Variable, here it is column name of the dataframe made using the Pandas[5] Library.

- **Model**

This gives the model name used which is Oridinary Least Square Model.

- **Method**

The method used for calculating the regression values here is Least Squares Method.

- **Date**

Gives the date when the model is used.

- **Time**

Gives the time of implementation of the model.

- **No. of Observations**

Give the number of rows evaluated from dataframe.

- **Df Residuals**

Degrees of freedom of the residuals = (Number of Observations - Number of Parameters)

- **Df Model**

Number of parameters in the model (not including the constant term if present)

- **Covariance Type**

Type of Covariance Estimation by the model.

The right part of the first table shows the goodness of fit:

- **R-Squared**

  The coefficient of determination. A statistical measure of how well the regression line approximates the real data points.

- **Adj. R-squared**

  The above value adjusted based on the number of observations and the degrees-of-freedom of the residuals

- **F-statistic**

  A measure how significant the fit is. The mean squared error of the model divided by the mean squared error of the residuals

- **Prob (F-statistic)**

  The probability that you would get the above statistic, given the null hypothesis that they are unrelated

- **Log-Likelihood**

  The log of the likelihood function.

- **AIC**

  The Akaike Information Criterion. Adjusts the log-likelihood based on the number of observations and the complexity of the model.

- **BIC**

  The Bayesian Information Criterion. Similar to the AIC but has a higher penalty for models with more parameters.

The second table reports for each of the coefficients

- **Name and Const.**

  The Names of the Constant term and the column which will be operated upon

- **Coef**

  The estimated value of the coefficient

- **Std err**

  The basic standard error of the estimate of the coefficient. More sophisticated errors are also available.

- **t**

The t-statistic value. This is a measure of how statistically significant the coefficient is.

- **P > |t|**

P-value that the null-hypothesis that the coefficient = 0 is true. If it is less than the confidence level, often 0.05, it indicates that there is a statistically significant relationship between the term and the response.

- **[95.0% Conf. Interval]**

The lower and upper values of the 95% confidence interval

Finally, there are several statistical tests to assess the distribution of the residuals

- **Skewness**

A measure of the symmetry of the data about the mean. Normally-distributed errors should be symmetrically distributed about the mean (equal amounts above and below the line).

- **Kurtosis**

A measure of the shape of the distribution. Compares the amount of data close to the mean with those far away from the mean (in the tails).

- **Omnibus**

D'Angostino's test. It provides a combined statistical test for the presence of skewness and kurtosis.

- **Prob(Omnibus)**

The above statistic turned into a probability

- **Jarque-Bera**

A different test of the skewness and kurtosis

- **Prob(JB)**

The above statistic turned into a probability

- **Durbin – Watson**

A test for the presence of autocorrelation (that the errors are not independent.) Often important in time-series analysis

- **Cond. No**

A test for multicollinearity (if in a fit with multiple parameters, the parameters are related with each other).

1.4 Correlation

The correlation between two companies on a scattered plotted graph is done in this project using the daily positive or negative percent change. The plots shows how much one company is affected by another company. The relationship between two companies is depicted then.

An example from the project explains the above context:

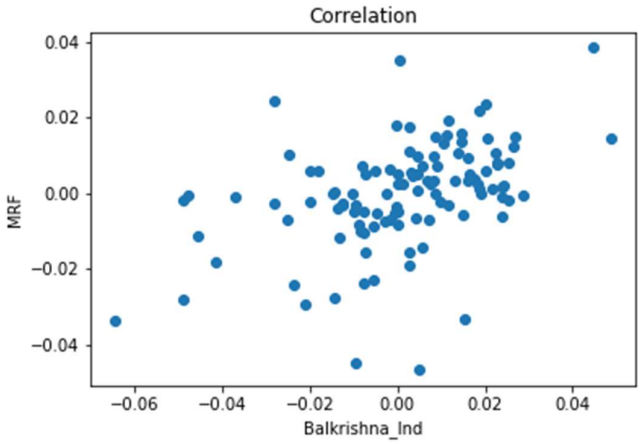

Figure 4 Example Correlation Values

- We can observe that less scattering means high correlation
- A correlation here means similar deviation in prices which helps in analyzation and prediction of the stock prices.
- Similar results are also obtained from a correlation matrix from the result.

	HDFC_Bank	Kotak_Mahindra	ICICI_Bank	AXIS_Bank	Indusind_Bank
HDFC_Bank	1.000000	0.624962	0.142868	0.186834	0.430485
Kotak_Mahindra	0.624962	1.000000	0.015529	0.212057	0.474334
ICICI_Bank	0.142868	0.015529	1.000000	0.607516	0.091036
AXIS_Bank	0.186834	0.212057	0.607516	1.000000	0.220274
Indusind_Bank	0.430485	0.474334	0.091036	0.220274	1.000000

Figure 5 Example Correlation Matrix

- These values show how each company is correlated to each other.
- Higher the value near to 1.0 more similar is the behavior of the companies.

1.5 Top 10 Sectors of India on the basis of Market Capitalization in 2017[3]

1.5.1 Bank-Private

1.5.2 Engineering – Construction

1.5.3 Pharmaceuticals & Drugs

1.5.4 Cement & Construction Materials

1.5.5 Finance – NBFC

1.5.6 Automobiles

1.5.7 Chemicals

1.5.8 Metal - Non Ferrous

1.5.9 Telecommunication Services

1.5.10 Tyres & Allied

1.6 Conclusion

We have shown the comparison between the predicted price and actual price. As it clearly visible from the graph that, our prediction price is near to with the actual stock price. This method of predicting the return on investment will help in a great way to financial institutions and stock brokers to predict the future price in such uncertain conditions. This method is mathematically reliable but can't be fully accepted as many factors on which the model depends have been untouched in this project. Further developments may see the neural networks of how the prices are really affected. Economic Times have recently launched this analysis on their websites.

1.7 Acknowledgements

I would like to acknowledge my trainer Er. Jitendra Yadav, TCIL-IT, for his valuable support in Data Science. Also I would like to thank my family and friends who guided me with this project.

1.8 REFERENCES

1. Stock Price Prediction Using Regression Analysis by Dr. P. K. Sahoo, Mr. Krishna Charlapally; https://www.ijser.org/researchpaper/Stock-Price-Prediction-Using-Regression-Analysis.pdf

2. Money Control companies with their Market Cap; https://www.moneycontrol.com/stocks/marketinfo/marketcap/bse/abrasives.html

3 List of top 10 Sectors with top 5 compaines of each (as on 13 JUN, 2018); https://www.indiainfoline.com/article/news-top-story/top-10-sectors-and-stocks-bought-and-sold-by-fund-managers-in-dec-2017-118011200377_1.html

4. OLS Model; https://blog.datarobot.com/ordinary-least-squares-in-python

5. Pandas; https://pandas.pydata.org/

6. Matplotlib; https://matplotlib.org/