IQP Report

How to Predict Price of Stocks

2013 A Term ~ 2014 C Term

Worcester Polytechnic Institution

Zhouxiao Wu

Jiali Gao

Advisor: Mayer Humi

Content

1. Abstract

In this project, we developed two models by using technics like ARIMA and Fourier Expansion to predict stock price in Matlab, Tisean Arima and excel. In this way, we can help investors who are new to the market or have little information and sources, giving them suggestions and indications. We use historical stock data from yahoo.com to build, test and refine our models for prediction. The outcomes are pleasing and we believe our models can help some investors out.

2. Executive summary

Risks are ubiquitous in stock market especially for new investors. Rookie investors or people with little information would be very thankful if anyone or anything can provide them with suggestions and indications. Hence, developing a tool that can predict the price of stocks will make a far-reaching impact on investing. However, because our goal of this project is to help particularly people with no special source of information, we need to come up with something that only takes easily accessible data and make relatively accurate prediction to help them to make better decisions.

There are other tools or websites, which do predictions or investment suggestions as well. But many of them are expensive to purchase and some give very bad predictions. Our project is to build a model that can be run by Matlab, which is a widely used tool, to help investor. Anyone has Matlab in his or her computer can use this tool for free and gets relatively accurate prediction at the same time. More importantly, to make prediction, no secret information is ever needed. One can find everything necessary online and then make money out of it.

So, what we did in the project was to make a model, which can separate a historical data of stock price into several parts, each part was represented by a function and if we extend those functions and put each new part back together, we can then obtain the future stock price for a few days. Therefore, the real difficulty was to find which functions could work better and what composition would give us solid outcomes.

After three terms' working, we have found two models that give overall satisficing results. We chose twenty stocks to test models, half was from energy field and half was from technology field because we had to take account of field factors. The results were very pleasing since more than half of the chosen stocks were predicted at a certain level of accuracy.

Having achieved those mentioned above, we strongly believe that although our models can be further developed by adding more factors, they can help people predict stock price and make money.

3. Introduction

Being able to predict the price of stocks can make anyone a big fortune. But only a few people of the smartest can have this ability. Thanks to this IQP and Professor Humi, we have the chance explore this amazing ability in this project and even develop it by ourselves.

"A stock in essence is a share of ownership in the company."(Assets Primer) says the background that our advisor gave us. For someone that doesn't know economy well, stock is just something one can invest easily. Once the price goes up, investors earn money, and vise versa. But for those who want to fully understand stocks, there are just too much information and too many concepts to understand. For example, there are different kinds of stocks as preferred stock and common stock as well as bonds that are similar to stocks. Also bonds, Currency Pairs and Commodities could have been options to be studied and predicted, but we passed them for reasons.

The reason why we passed bonds was that bonds could be easily customized with different rates and times. In other wards, bonds have incredible variety. What's more, bonds are not traded frequently. Thus, the value of market tends to change slowly. Due to these reasons, it was ruled out.

Also, we didn't consider Currency Pairs to be the best choice. Currency Pairs are relationships between the values of different currencies. "The basic idea is that the quoted value for a currency pair is how many units of the quote currency it would cost to purchase one unit of the base currency."(Assets Primer) Although, speculating currency pairs and stocks are almost the same, currency pairs are more concerned with macroeconomics, which makes them less variety.

As for Commodities, it is a type of asset that exists for probably the longest time. Professor told us in the background, "There are a variety of commodities traded in the market with various price behaviors, not so different from what is seen in the stock market."(Assets Primer) But we still ruled it out because its lack of diversity. There are many Commodities are strongly related to weather and some of them are correlated to each other.

According to all the reasons mentioned above, our goal was set to predict the price of stock. First we decided to approach our goal by studying the figures, which were historic price data. At the beginning of the project, our advisor introduced the first model and then we developed it with his help. Then, we were introduced with ARIMA and we built a new model with it.

4. Why this is an IQP

"WPI believes that in order to become the best engineers and scientists they can be, students should have a broad understanding of the cultural and social contexts of those fields, and thus be more effective and socially responsible practitioners and citizens," says in WPI webpage. As we can see through the titles themselves, MQP enhance our depth of our major field, while IQP concerns more on developing teamwork and getting to know the relationship between science and society. Our project aims to predict the price of stocks. The most important and obvious reason why this is an Interactive qualifying project is that predicting stock value can reduce the risk of buying stock, which can powerfully help shareholders have a better understanding of stock price's direction, upward or downward, and make a safer choice on selling or buying stocks to maintain their original fortune and get a positive return at a time. One of the way shareholders can earn money is that company pay a dividend or a portion of earnings to its shareholders on a regular basis. Shareholders also can reinvest the dividends to build their portfolio or use it as income. In addition, shareholders also can sell the stocks at anytime they want. If the selling price is higher than they bought, they earn the extra money. The ownership of the stock gives shareowners the right and flexibility to sell or hold on stocks. The only disadvantage may have a negative return in some years rather than a positive one. That could reduce your income and the value of your portfolio.

As we all known, saving money in the bank is a common way to accumulate fortune. However, saving money in the bank is no longer keeping the true value of the money by time. Because the bank has little annual interest rate, which approximately to 0.02% per year, but the country's economy's inflation is getting higher year by year, the true value of the money exceedingly shrinks and the very little saving interest rate has no help for keeping money's original value. So the true value of the money saving in the bank keeps depreciating, the inflation goes up consistently. Thus, the traditional way of accumulating fortune, saving large amount of money in bank, is not an ideal way to maintain its original value and shrink its value upon the level of the inflation.

Since the little annual saving rate has no effect on keeping value of money in the big environment of high inflation, the consumption is usually the alternative option for people to accumulate fortune on getting more goods, including capital assets, like house. According to the macroeconomics' models, they show that money demand decrease, which people will hold less money and spend more on goods, resulting LM curve shifts backwards with higher interest rate and less production. However, irrational consumption unquestionably does not make consumers wealthier. In addition, even if consumers buy goods with carefully thoughts and second thinking, less money saving is still putting them in a difficult situation, because less saving may weaken its ability to deal with emergencies of large pavements. Goods or capital assets can be exchanged to cash, but not liquid enough for a quick and large amount of cash immediately.

As mentioned above, either traditional saving money or exchanging money for goods is not a perfect plan for protecting fortune from shrinking in the economy. Although buying stock is not one hundred percentage guarantees for getting more money back, using "Stock price predicting" can largely reduce the risk on making wrong decision on stock's purchase and give users more confident on its stock. Furthermore, the high interest rate is not only maintaining stockholders' original value of money, but also increasing stockholders' money value at the time.

5. Research

5.1 First model -- Trend- Fourier expansion model

a. The mathematics of the model

Mathematical theorem is needed to develop our model. We would like to introduce the mathematics that we will use in our model.

1) Least Square(Least Squares, 2013)

The best fitting line, according to the Sum of squared errors (SSE) measure, will be the line whose intercept and slope, b_0 and b_1, respectively, minimize SSE (b_0, b_1). These values are called the **least squares estimators** of intercept and slope.

Formulas for least square estimators:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(X_i - \hat{X})(Y_i - \hat{Y})}{\sum_{i=1}^{n}(X_i - X)^2}$$

$$\hat{\beta}_0 = \hat{Y} - \hat{\beta}_1 \hat{X}$$

2) Fourier analysis(ForierSeries, 2013)

"A Fourier series is an expansion of a periodic function $f(x)$ in terms of an infinite sum of sines and cosines. Fourier series make use of the orthogonality relationships of the sine and cosine functions."

This example fits the ENSO data using several custom nonlinear equations. The ENSO data consists of monthly averaged atmospheric pressure differences between Easter Island and Darwin, Australia. This difference drives the trade winds in the southern hemisphere.

The ENSO data is clearly periodic, which suggests it can be described by a Fourier series:

$$y(x) = a_0 + \sum_{i=1}^{\infty} a_j \cos\left(2\pi \frac{x}{c_i}\right) + b_i \sin\,(2\pi \frac{x}{c_i})$$

where a_i and b_i are the amplitudes, and c_i are the periods (cycles) of the data. The question to answer here is how many cycles exist?

As a first attempt, assume a single cycle and fit the data using one cosine term and one sine term.

$$y_1(x) = a_0 + a_1 \cos\left(2\pi \frac{x}{c_1}\right) + b_1 \sin\,(2\pi \frac{x}{c_1})$$

3) Confidence interval (Confidence interval, 2013)

If the sample distribution of an estimator is known, this knowledge can be used to compute a likely range of values for the parameter being estimated. Such a range of values is called an interval estimate or a confidence interval.

4) Autocorrelation(Autocorrelation and Partial Autocorrelation, 2013)

Autocorrelation is the linear dependence of a variable with itself at two points in time. For stationary processes, autocorrelation between any two observations only depends on the time lag *h* between them. Define $Cov(y_t, y_{t-h}) = \gamma_h$. Lag-*h* autocorrelation is given by

$$\rho_h = Corr(y_t, y_{t-h}) = \frac{Y_h}{Y_0}$$

5) Correlation(Corrcoef, 2013)

For data that exhibit linear association, it may be of interest to ask how strong that association is. The Pearson correlation coefficient is a measure of the strength of linear association between two quantitative variables.

R = corrcoef(X) returns a matrix R of correlation coefficients calculated from an input matrix X whose rows are observations and whose columns are variables. The matrix R = corrcoef(X) is related to the covariance matrix C = cov(X) by

$$R(i,j) = \frac{C(i,j)}{\sqrt{C(i,i)C(j,j)}}$$

corrcoef(X) is the zeroth lag of the normalized covariance function, that is, the zeroth lag of xcov(x,'coeff') packed into a square array.

b. Data and Stocks

In order to predict the price of stock, we need to find the data for the stocks in the past years. Since the data for stocks in more than past 2 years may not be that correlated to the current stock prices, we want to focus on only two years and change the time period later on.

There are several paths for us to collect the historical data for our stocks. We could have gone to Library to get the Data, but unfortunately the data there has a few months delay. Meanwhile the historical data from Yahoo, in particular, is the most up-to-date one and together with all open, closed, highest, lowest and adjusted closed prices for stocks. Thus, we get the historical prices for the stocks we need.

We cannot just arbitrarily choose 10 stocks for each of us, since each stocks has their own field and impact factors, which may ends up demanding different types of model. Therefore we decide to find 10 stocks from just one field. Then we take Energy and Technology as our options. The reason why we need to have 10 stocks is that the stocks we choose have to contain both large and small companies. By doing so, we could find out how different factors would influence the stocks with high price like Apple Inc. or with price lower than 10 dollars.

Here are the brief description of the 10 stocks we choose from each field.

From energy field:

Alon USA Energy, Inc. (2005) (ALJ)

Alon is an independent refiner and marketer of petroleum products focused on growth and innovation to meet both the energy and environmental needs of today. With refining, asphalt and retail/branded marketing operations across the western and south-central regions of the United States.

Duke Energy Corporation (1983) (DUK)

Duke Energy is a leading energy company focused on electric power and gas distribution operations, and other energy services in the Americas – including a growing portfolio of renewable energy assets.

Enterprise Products Partners L.P. (1998) (EPD

Enterprise Products Partners L.P. is one of the largest publicly-traded energy partnerships and a leading North American provider of midstream energy services. (Main: natural gas, NGL crude oil, refined products)

Halliburton Company (1981) (HAL)

Founded in 1919, Halliburton is one of the world's largest providers of products and services to the energy industry. The company serves the upstream oil and gas industry throughout the lifecycle of the reservoir – from locating hydrocarbons and managing geological data, to drilling and formation evaluation, well construction and completion, and optimizing production through the life of the field.

Kinder Morgan Management LLC (2001) (KMR)

Kinder Morgan is the largest midstream and the third largest energy company in North America. Their pipelines transport natural gas, refined petroleum products, crude oil, carbon dioxide (CO_2) and more. They also store or handle a variety of products and materials at their terminals such as gasoline, jet fuel, ethanol, coal, petroleum coke and steel.

Northwest Natural Gas Company (1990) (NWN)

NW Natural buys natural gas from suppliers in the Western U.S. and Canada and distributes it to residential, commercial, and industrial customers throughout our service territory.

Otter Tail Corporation (1990) (OTTR)

Otter Tail Corporation is a growing company. Our diversified operations include an electric utility and infrastructure businesses which include manufacturing, construction and plastics.

Total SA (1991) (TOT)

Total is a major energy operator, active in every segment of the oil and gas industry. They also produce chemicals and develop and market solutions involving new energies.

WGL Holdings Inc. (1987) (WGL)

WGL Holdings is a holding company that was established on November 1, 2000 as a Virginia corporation to own subsidiaries that sell and deliver natural gas and provide a variety of energy-related products and services to customers primarily in the District of Columbia and the surrounding metropolitan areas in Maryland and Virginia.

Exxon Mobil Corporation (1970) (XOM)

We are the world's largest publicly traded international oil and gas company, providing energy that helps underpin growing economies and improve living standards around the world.

From technology field:

Apple Inc. (1984)(AAPL)
Apple Inc. and its wholly-owned subsidiaries design, manufacture, and market mobile communication and media devices, personal computers, and portable digital music players worldwide.

Bridgeline Digital, Inc. (2007)(BLIN)
Bridgeline Digital, Inc. develops iAPPS Web engagement management product platform and related digital solutions in the United States. Its iAPPS platform enables companies and developers to create Websites, Web applications, and online stores.

Google Inc. (2004)(GOOG)
Google Inc., a technology company, builds products and provides services to organize the information. The company offers Google Search, which provides information online.

Mellanox Technologies, Ltd. (2007)(MLNX)
Mellanox Technologies, Ltd., a fabless semiconductor company, produces and supplies semiconductor interconnect products for computing, storage, and communications applications in the high-performance computing, Web 2.0, storage, financial services.

Microsoft Corporation (1986)(MSFT)
Microsoft Corporation develops, licenses, and supports software, services, and hardware devices. Its Windows division offers Windows operating system; Windows Services suite of applications and Web services, including Outlook.com and SkyDrive.

RCM Technologies Inc. (1995)(RCMT)
RCM Technologies, Inc. engages in the design, development, and delivery of business and technology solutions to commercial and government sectors in the United States, Canada, and Puerto Rico.

VMware, Inc. (2007)(VMW)

VMware, Inc. provides virtualization infrastructure solutions in the United States and internationally. The company's virtualization infrastructure solutions include a suite of products designed to deliver a software-defined data center.

Western Digital Corporation (1987)(WDC)

Western Digital Corporation, through its subsidiaries, develops, manufactures, and sells storage products and solutions that enable people to create, manage, experience, and preserve digital content.

Wave Systems Corp. (1999)(WAVX)

Wave Systems Corp. develops, produces, and markets products for hardware-based digital security. Its products are based on the Trusted Platform Module (TPM), a hardware security chip that enables secure protection of files and other digital secret.

c. Development of model and the performance of stocks

1) First version of our model

In the first version of our model, we suppose to find out the trend for our data and the trend for the difference, which is the difference between original price and the trend. Then by using these two functions, we can have an initial prediction by simply extending the date.

We have the data for the last 2 years (from to Sep. 2nd) and then we could use the matlab to plot the data and in a least square sense, we use polyfit in matlab to find the coefficients of a polynomial of degree n that fits the data.

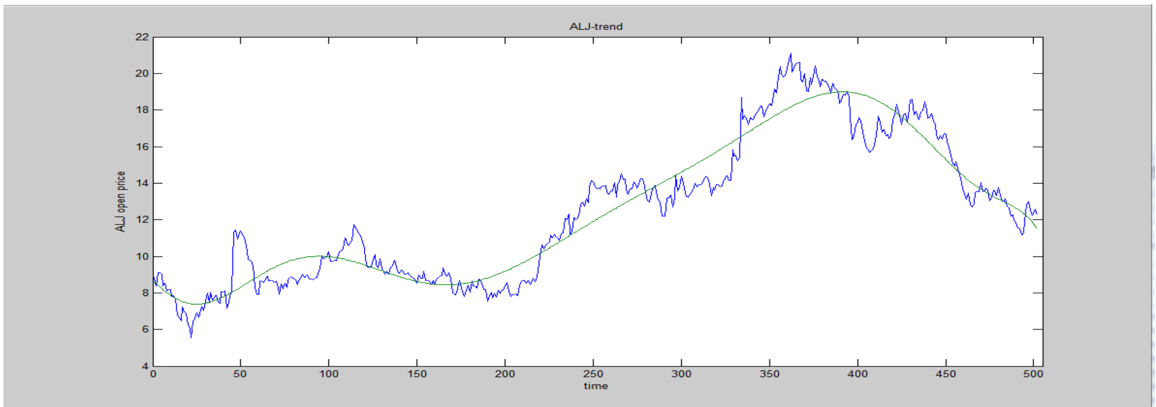


Figure1

We tried the degree n= 10 for the first time, and it come out with the trend showed in figure 1. When we tried to look at the coefficient for this polynomial, we found out that most of the coefficients were less than 0.00001, which were too small. Therefore, knowing that we can't let those coefficients be too small, we tried to make the degree n less than 4. So when we make n < 4, we can get the following figure (figure 2). Even it was not perfectly fit, it made the prediction better.

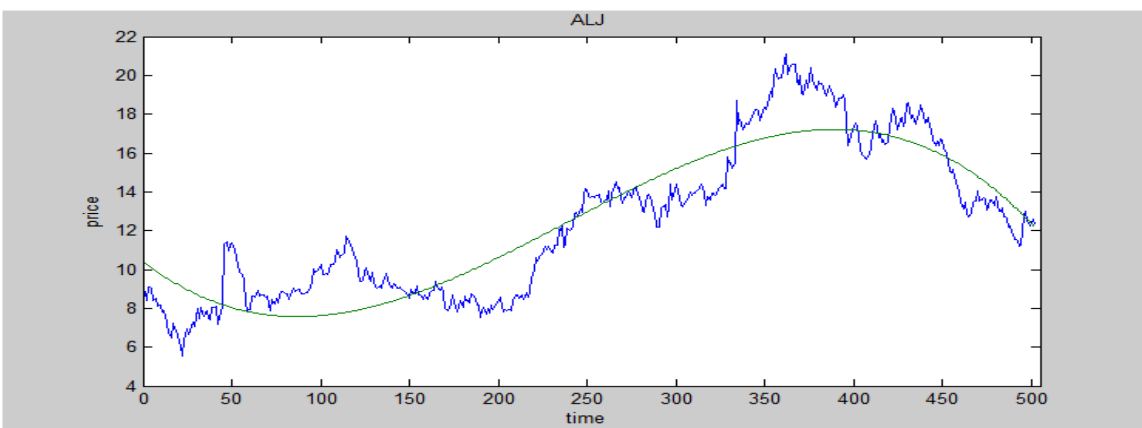
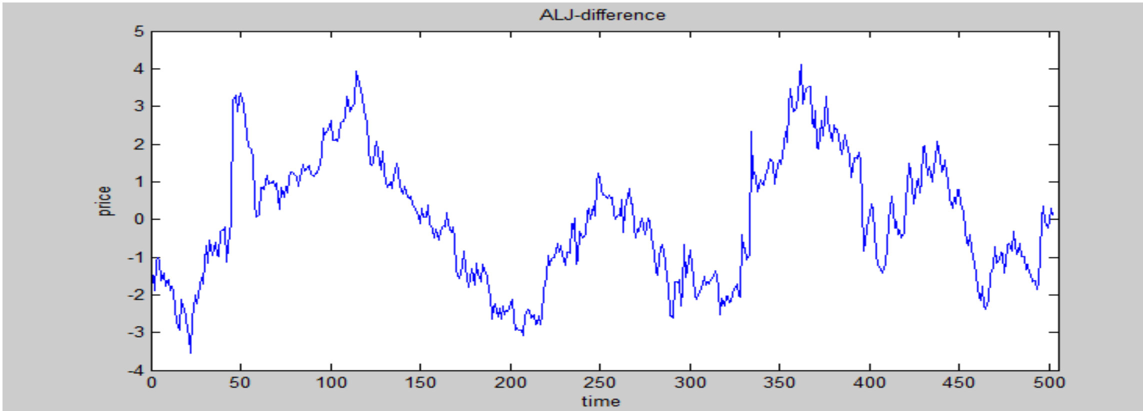
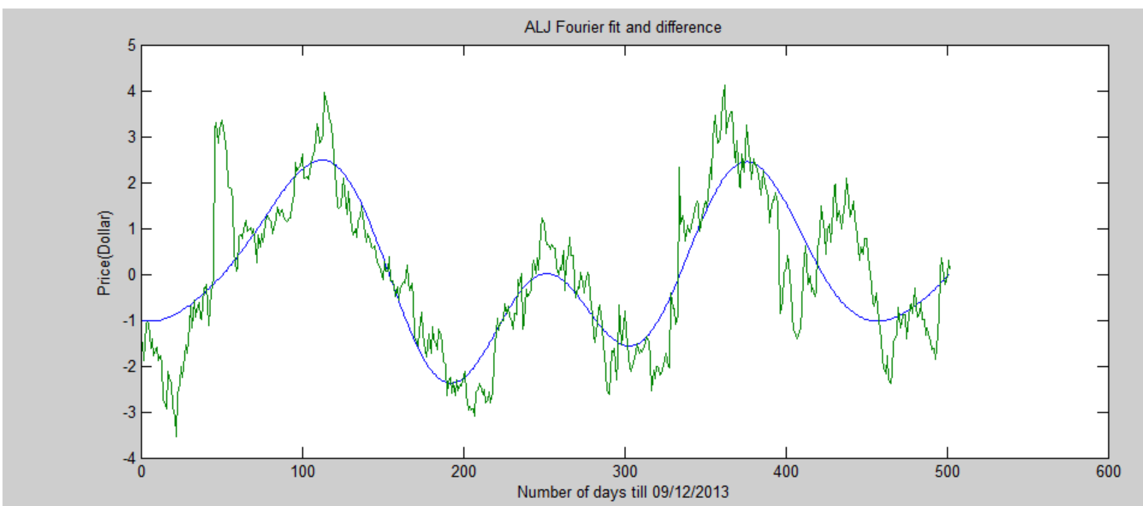


Figure 2

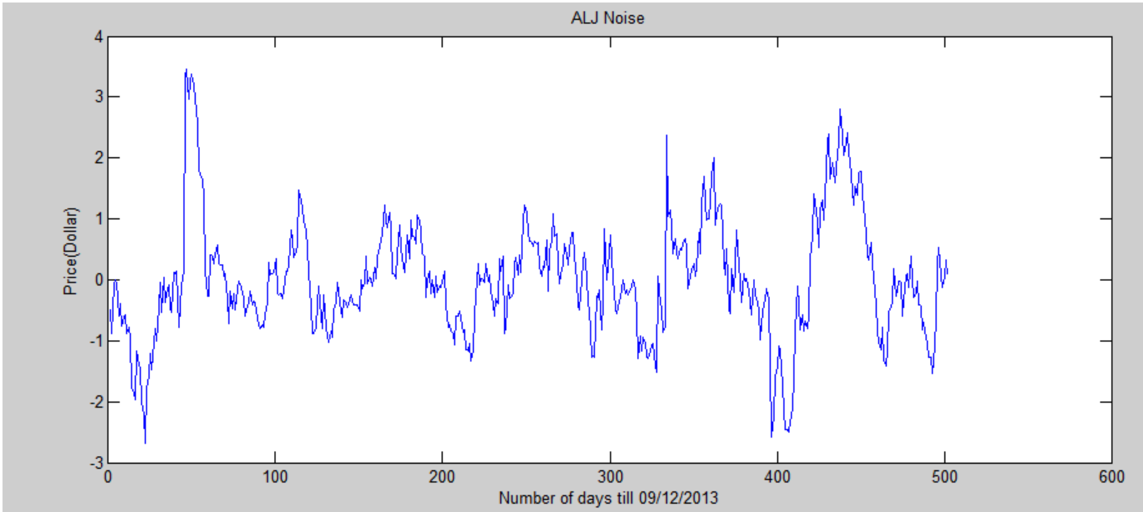
After we calculated the difference for the data and the polynomial we get from polyfit.



The difference we get here was quite similar to sin or cos function. So we decided to fit our difference with the Fourier expansion and it turned out to be well fitted. Here is the example plot for difference and Fourier expansion.



After taking the difference between difference and the Fourier expansion, we can obtain the noise.



Since the highest and the lowest noise was kind of big when we compared it to the original price of the stock, we need to improve our model to get less noise and a better fit. And here comes our second version of our model.

2) Second version of our model

In order to get smaller noise and a better fit, we consider using the autocorrelation for our data. Autocorrelation would give us data from -1 to 1, and we needed to find out the day that the autocorrelation of the stock prices first approached 0. Then these days we get were the most relative ones. Therefore we updated our data to the date which autocorrelation tells us. Figure 3 is one of the autocorrelation we did for a company.

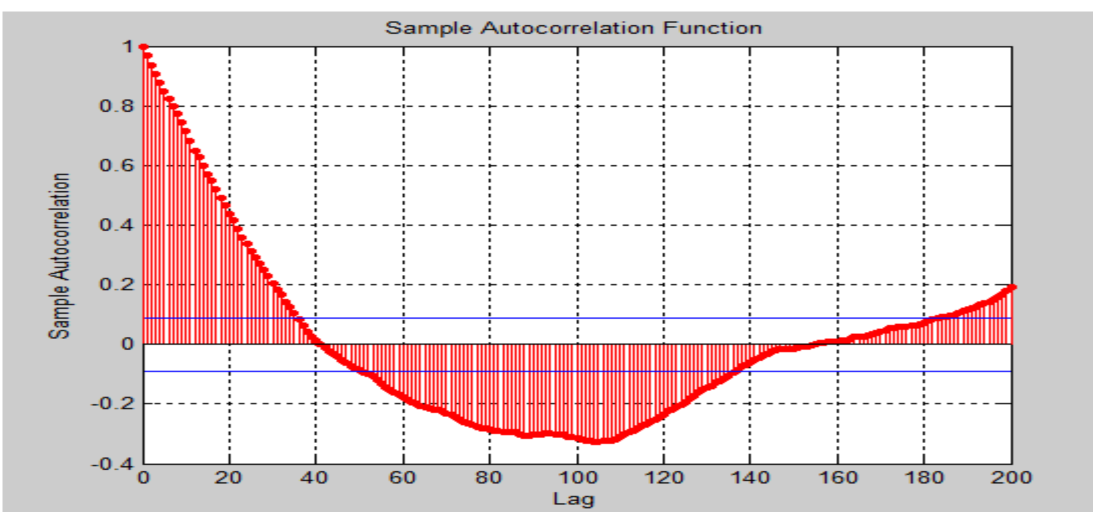


Figure 3

According to the figure 3 above we update the days to 40 instead of 2 years. Figure 4 is a sample with updated data, trend and the difference.

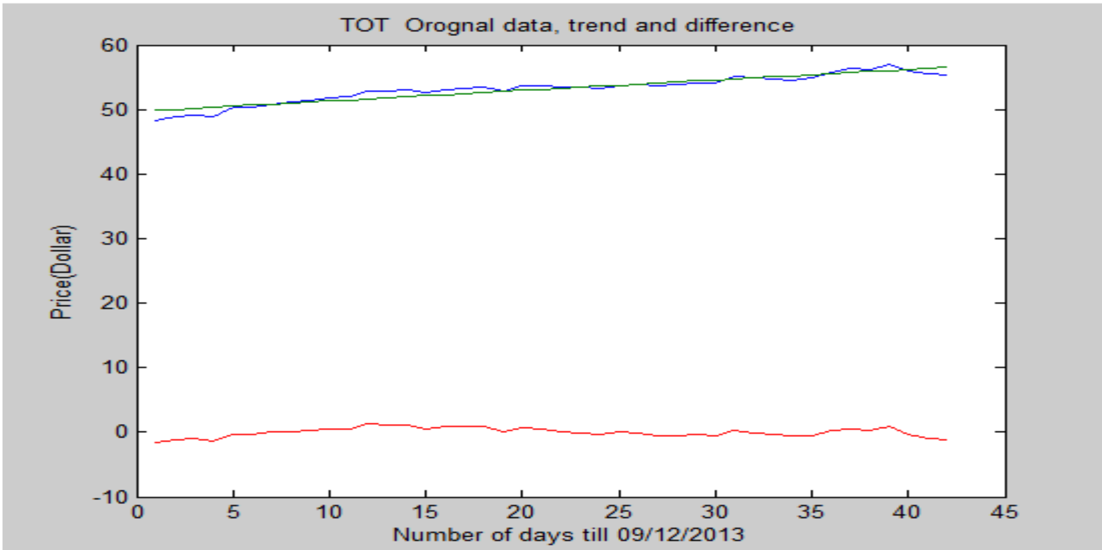


Figure 4 (Blue line: original price; Red: difference; Green: trend)

In the figure 4, we did the polyfit for the updated data and calculated the difference.

Figure 5 is the polynomial fit for the original data. As we can see in the figure 5, the polyfit gives us a better trend comparing to the trend we obtained during the 1st version of our model.

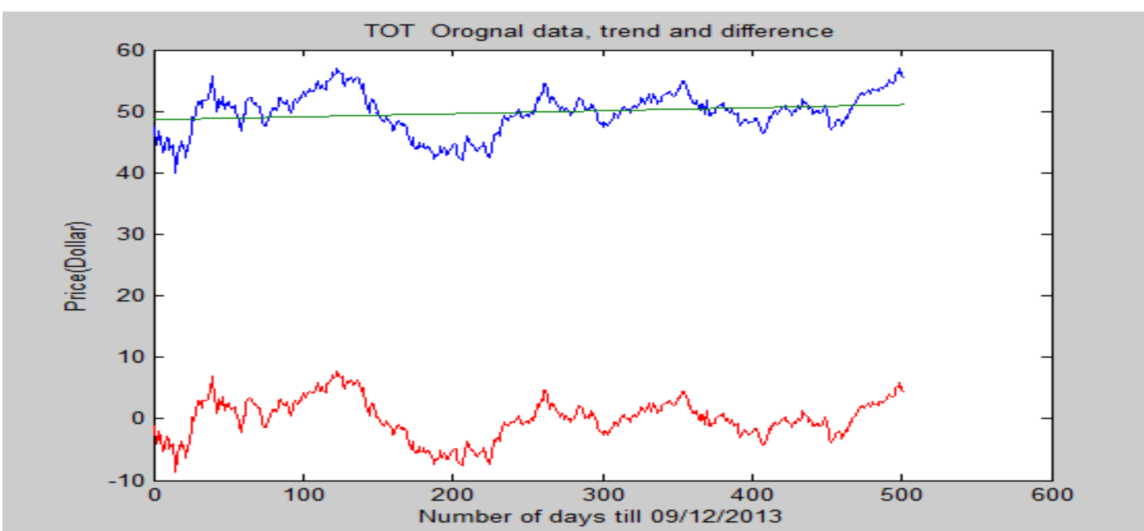


Figure 5 (Blue line: original price; Red: difference; Green: trend)

Additionally, we need to obtain the Fourier expansion for the difference to the updated data. Figure 6 is what we get now.

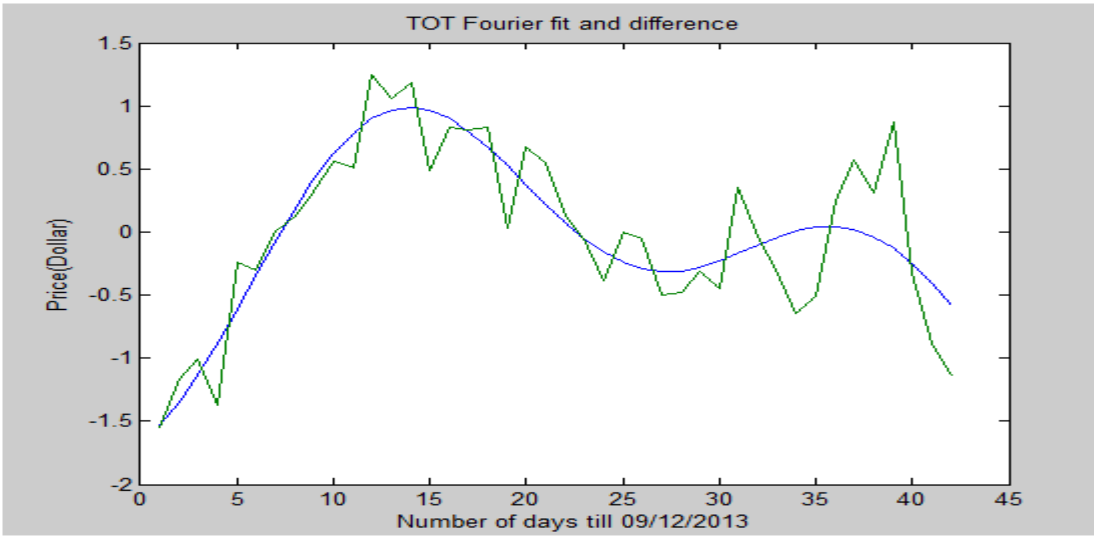


Figure 6(Green: difference; Blue: Fourier fit)

Figure 7 is the noise we get from the given data.

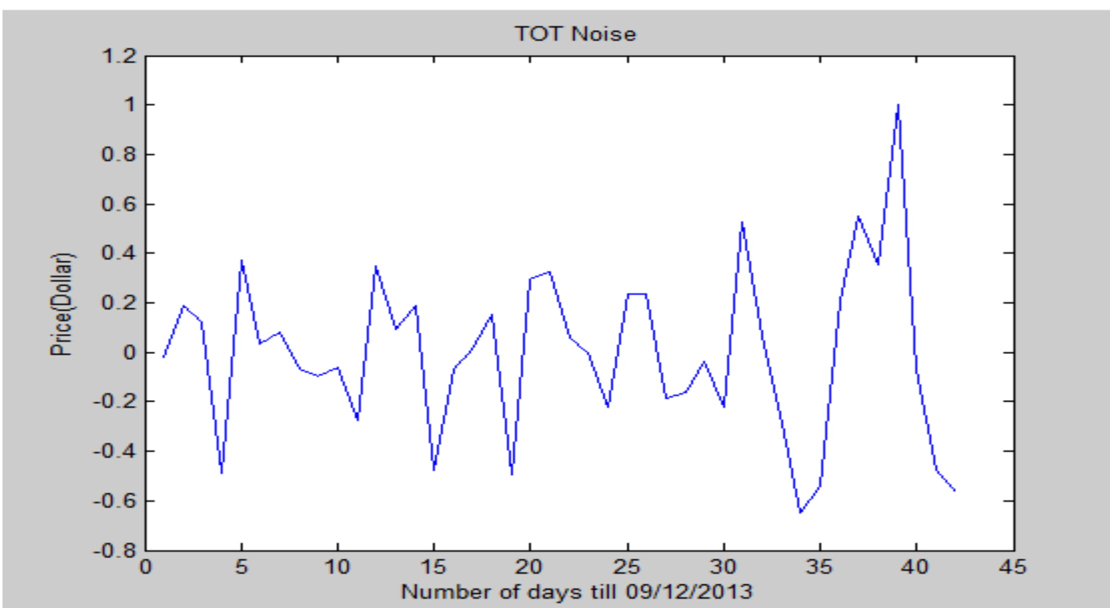


Figure 7

Comparing with the real price of the stock which is 50 dollars, the noise is from -0.8 to 1, which is only 2%. Since figure 7 gives us much smaller noise than the previous one, we may get a closer and better prediction.

After getting all the trends for prices of stocks, Fourier function for differences and the noise, we can get our initial prediction for the stocks. Figure 8 is the prediction of one of our stocks together with the real data we get from Yahoo.

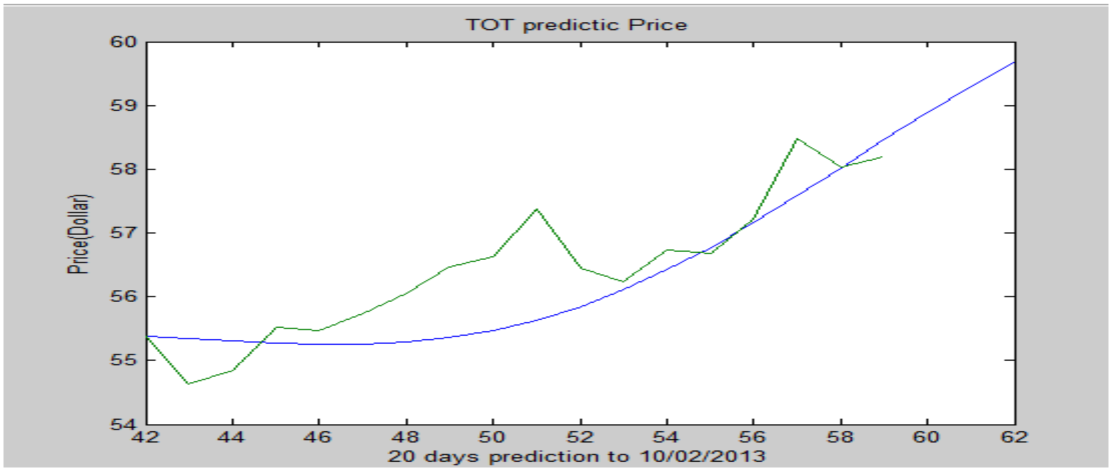


Figure 8(Green: real price; Blue: prediction)

In the figure 8, we could find out the first date for the predict price is the same as the real price of stocks, because we added the noise for the first date to all the predict prices so that the prediction and the real price can start from the same point. As one can see, in the second version, we only considered what the trend for the stock itself would influence the prediction. In the third version, we would like to take the market factors into account.

3) Third version of our model

The market factor we take into account is the Nasdaq index, because all 20 stocks we choose are from Nasdaq. For the Nasdaq index, we did what we have done in the second version: getting the trend, difference and noise. The only thing that was different was the time was based on what autocorrelations tell us from the stocks themselves.

However, the Nasdaq index was from 3000 to 4000, which was too large comparing with our stock prices. Thus, we should normalize the Nasdaq index by using the following formula:

normalized_Nasdaq = (Nasdaq -ave_Nasdaq)/ave_Nasdaq where Nasdaq is the price for Nasdaq on an arbitrary date; ave_Nasdaq is the average of Nasdaq prices over the autocorrelation period that given by each stock.

Furthermore, the normalization of time would also improve our model. The normalization can be done by following equation:

$$T = (t - t\_ave)/ t\_ave$$

in which T is the normalized time for the prices; t is the date we need to normalize; t_ ave =(t_auto+t_predict)/2 in which t_auto is the time from the autocorrelation and t_predict is the prediction period.

Before we normalized the data, if we fit the updated data of stocks with n>=2, the difference from day to day is too large. After we normalized the time, the period of time is from -1 to 1 which makes the difference in polynomial get less impact on our predictions.

How we can combine the impact of market and the price for stock itself is to calculate the correlation between these two factors and with the following formula we could obtain our new prediction:

new_predict_price = old_predict_price *(1+(1-corrcoef)*normalized_Nasdaq)

in which old_predict_price refers to the prediction we get from the second version; corrcoef is the correlation coefficient between prediction for the stock itself and normalized Nasdaq. The reason why we use the formula above is just a hypothesis and if we find the formula which cannot help us improve our prediction, we would to come up with a new one.

Now there is one more question: whether the prediction is good or not. We can not only conclude the model is good or not just seeing these predictions for 20 stocks, we also should get the confidence interval for all the predict price and if the real price is within the confidence interval, we could say that our prediction is good. The confidence interval now for our model is the mean and the mean plus the standard deviation of our predict prices.

Now let us have a look at how the normalization of time and Nasdaq would improve our data and how good our predictions are with the confidence interval.

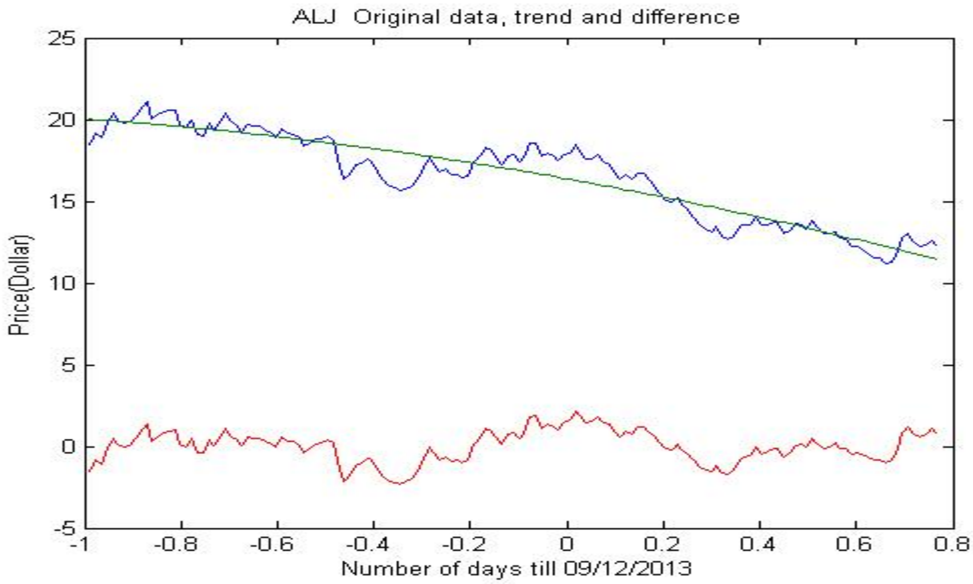


Figure 9 (Blue line: original price; Red: difference; Green: trend)

Clearly in figure 9, we can find the days are normalized from -1 to 1. And the purple line on the bottom stands for the difference between the data and trend.

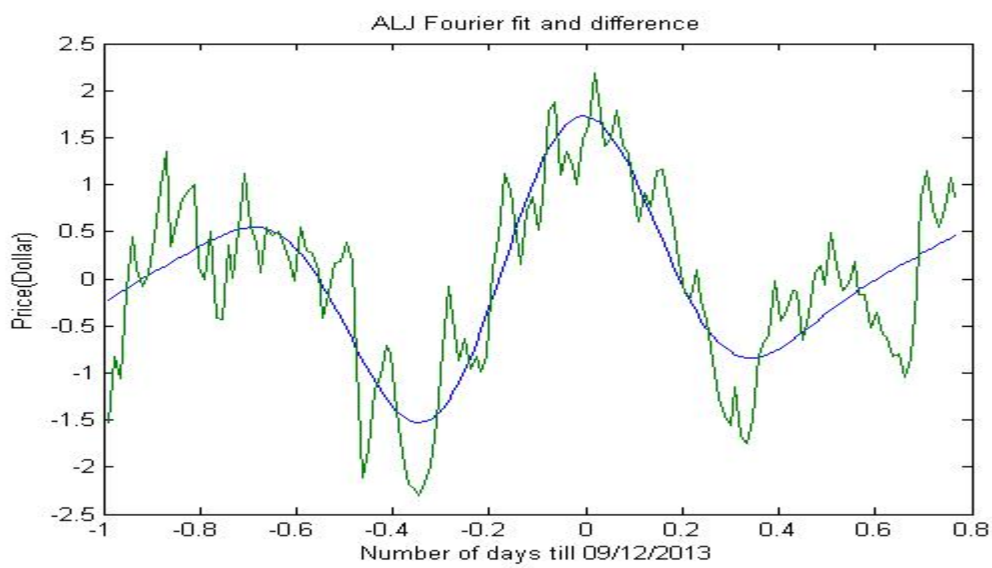


Figure 10(Green: difference; Blue: Fourier Fit)

The Fourier analysis is applied in figure 10 with normalized time.

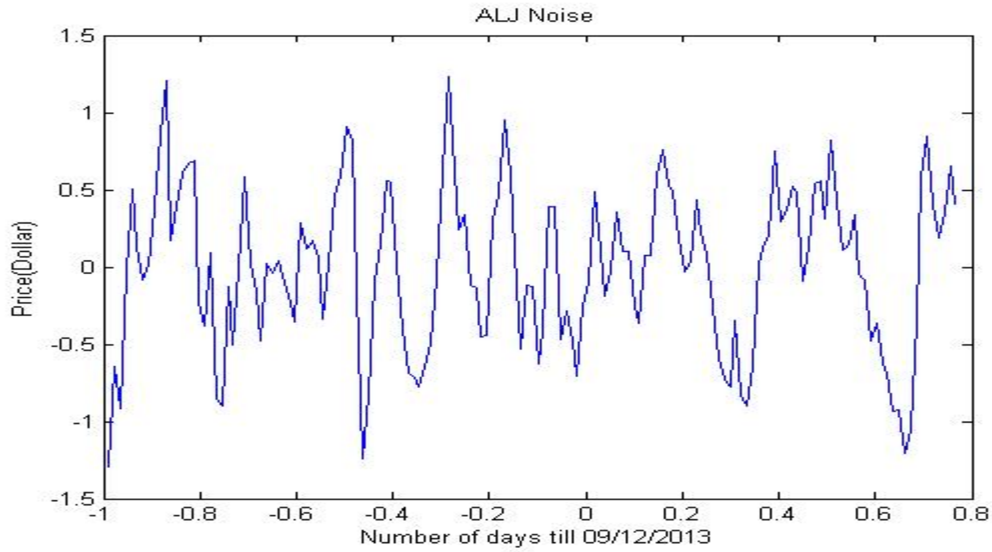


Figure 11

Given in figure 11, noise can obtain from the figure 9.

Then just base on the stock itself we can already obtain a predict price with an confidence interval, which is figure 11. Both purple dots and yellow dots are the interval of the mean of our prediction function. For red and blue dots, they stands for the mean plus the standard deviation of our prediction function.

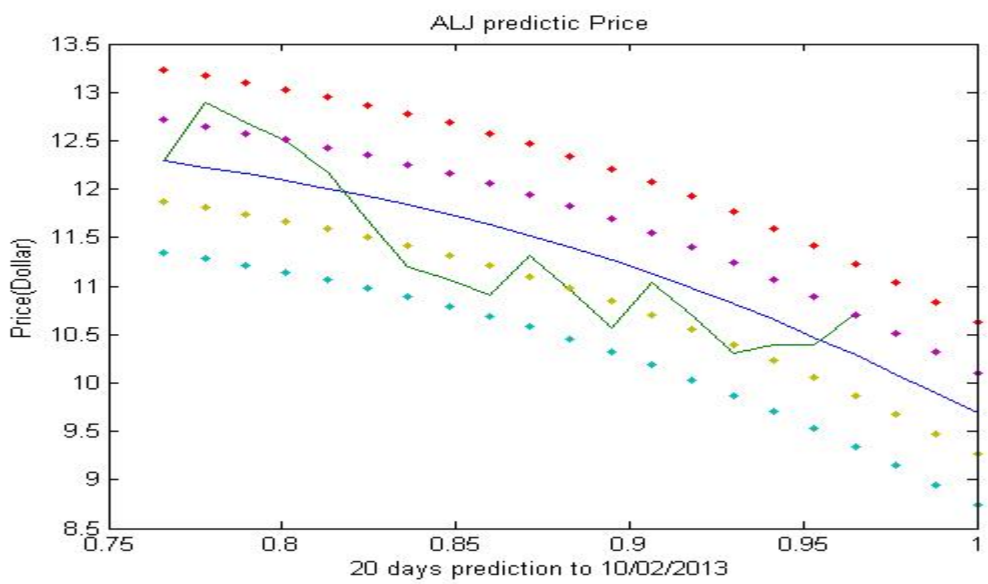


Figure 12(Purple: Prediction; Green: real price)

The prediction which only related to the stock itself is not enough. So we would like to adjust our prediction by calculating the normalized Nasdaq.

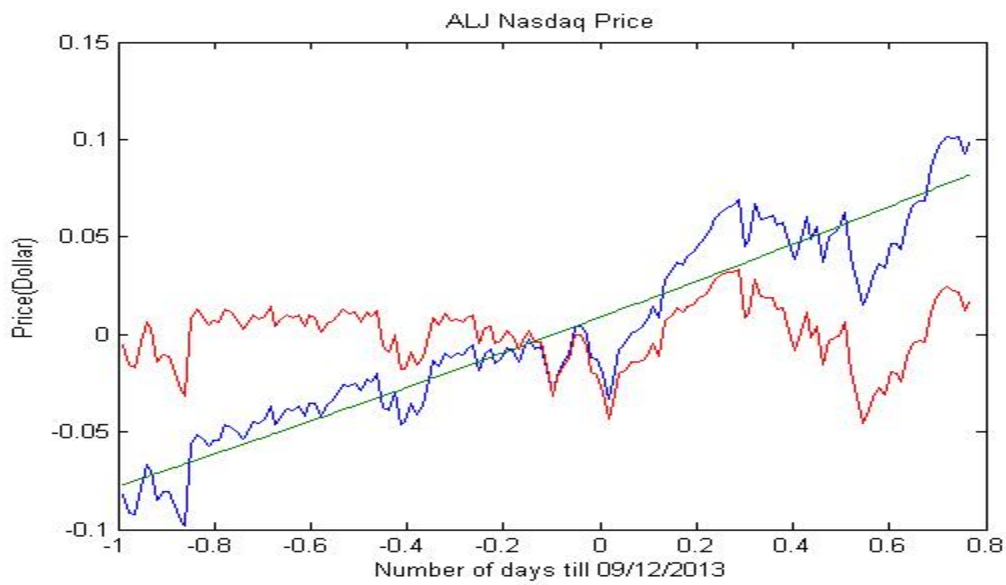


Figure 13 (Blue line: original price; Red: difference; Green: trend)

Then based on the time for stock itself, we find the data for Nasdaq price and normalize it. Figure 13 presents normalized Nasdaq price, trend and differences. What present in blue line is the normalized Nasdaq and the red line stands for the difference of trend and the data.

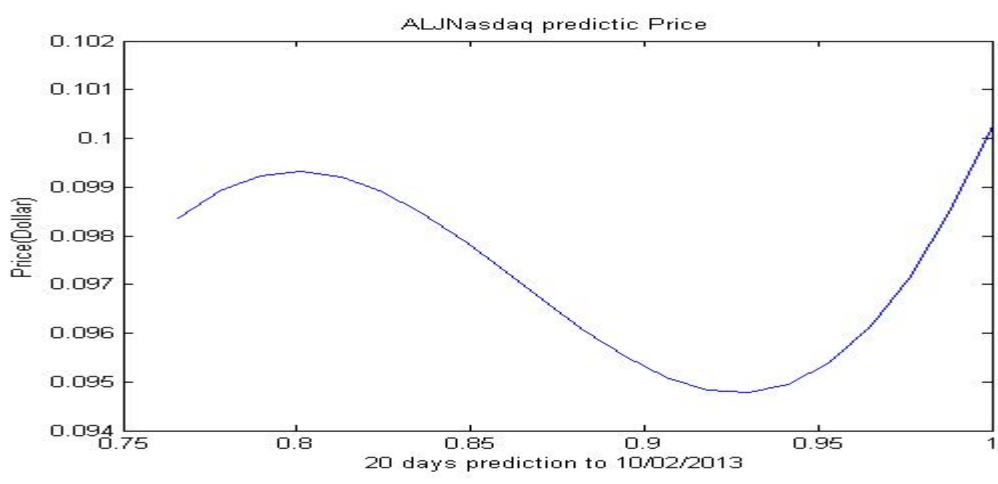


Figure 14

As what we have done for stocks we get the predict price for nasdaq part (figure 14).

Then we could calculate the correlation coefficient with two predictions in matlab. After we get the coefficient we can apply the formula to get our new prediction as what we show in figure 15

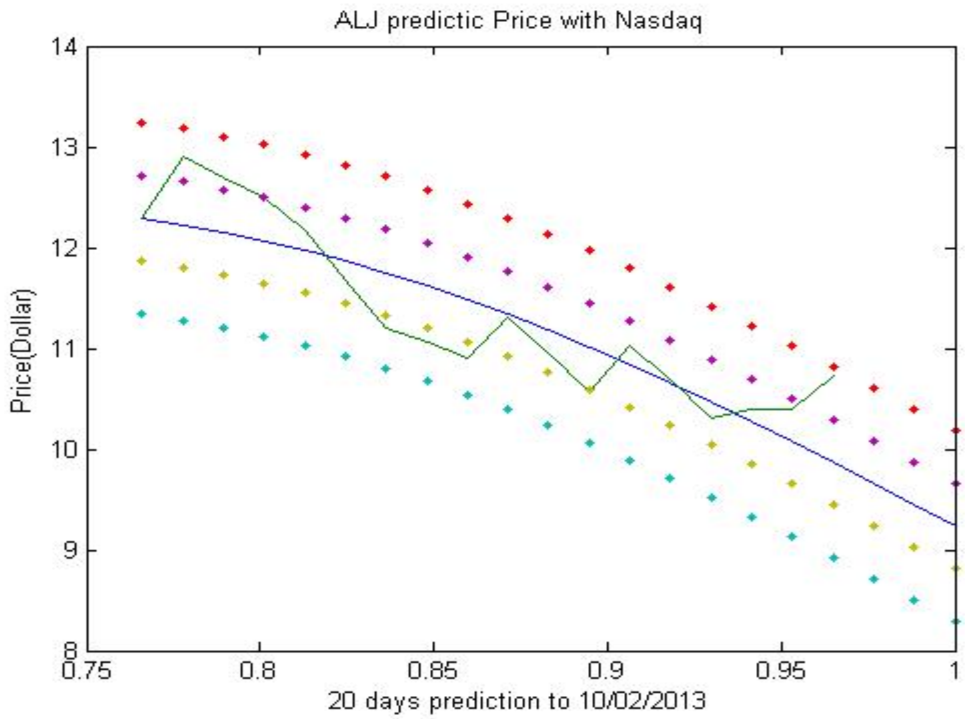


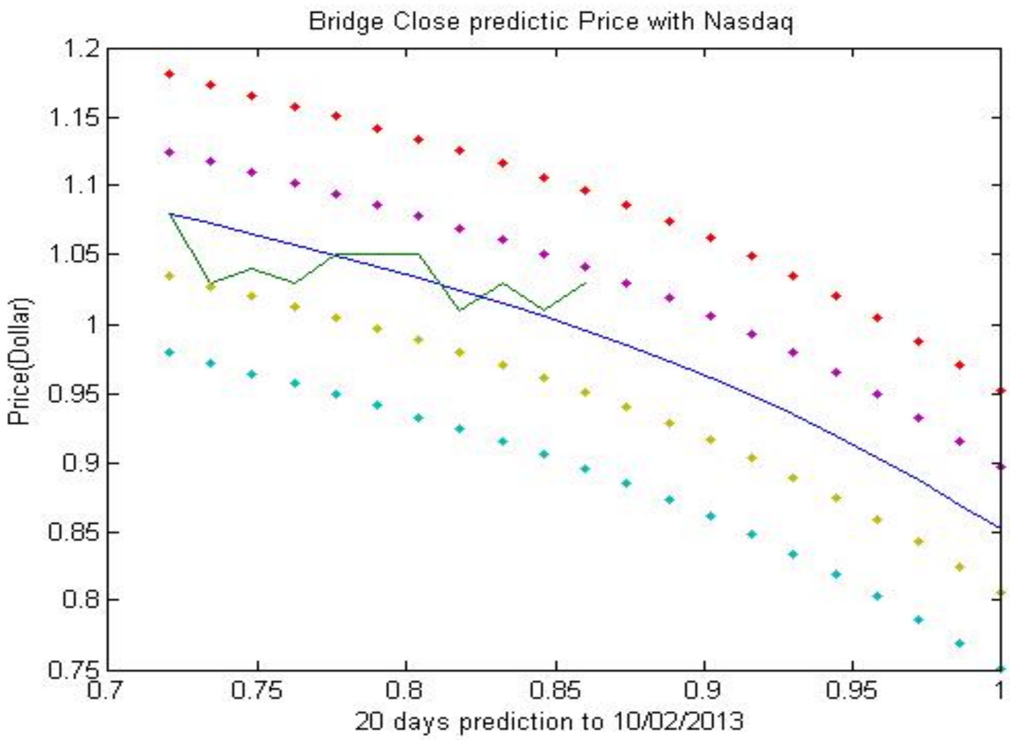
Figure 15

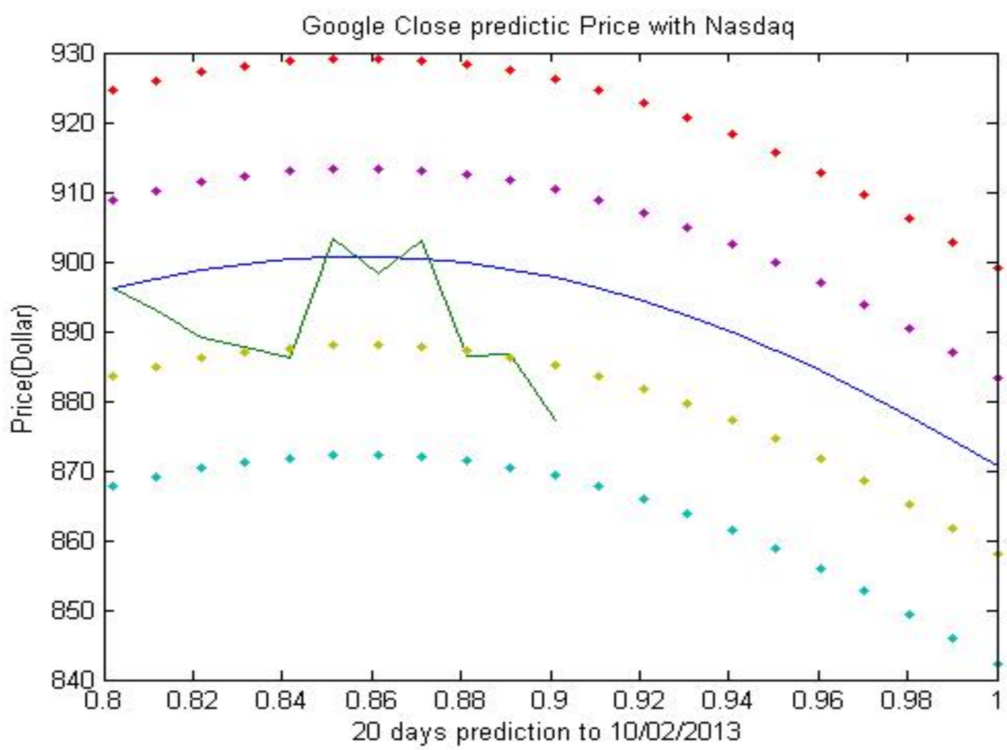
And figure 15 is our prediction for now.

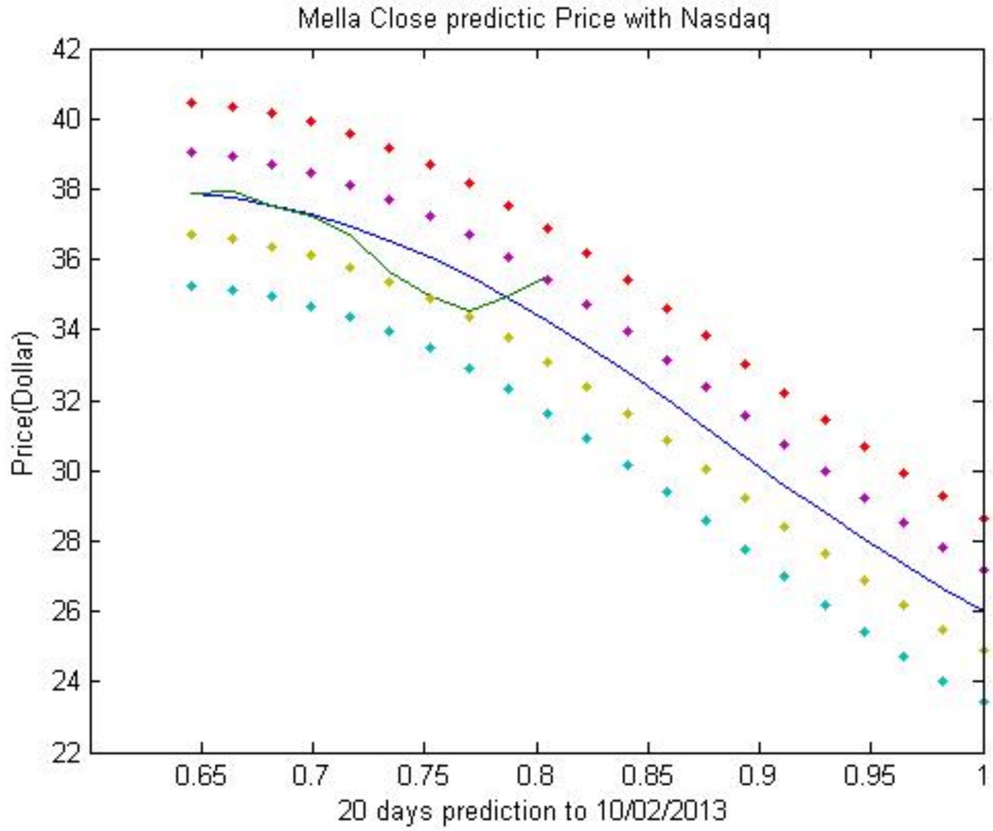
d. Evaluation of models

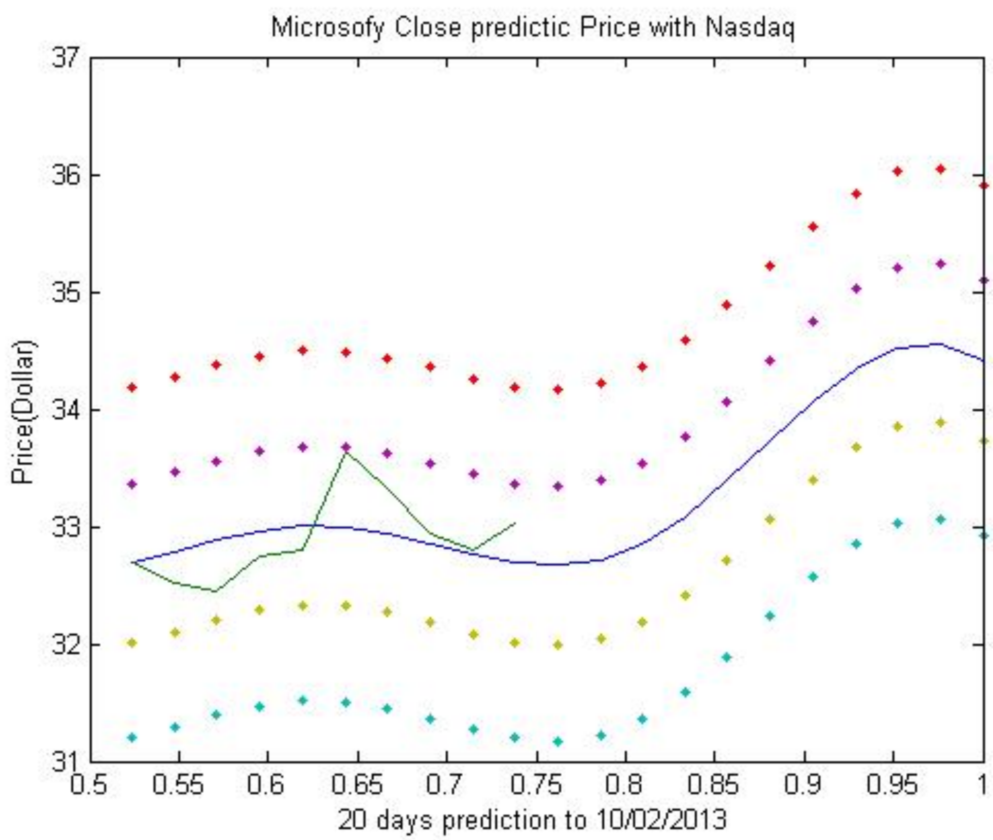
Here are all the final perdition with the real price for all 20 stocks.

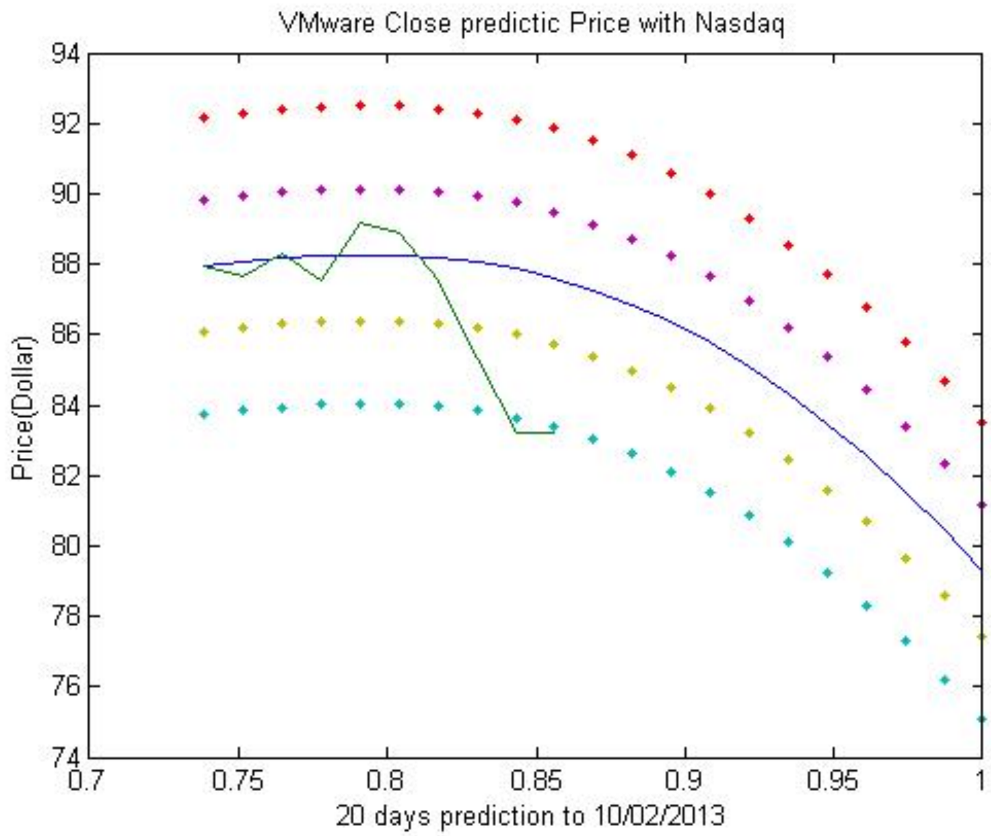
Technology:

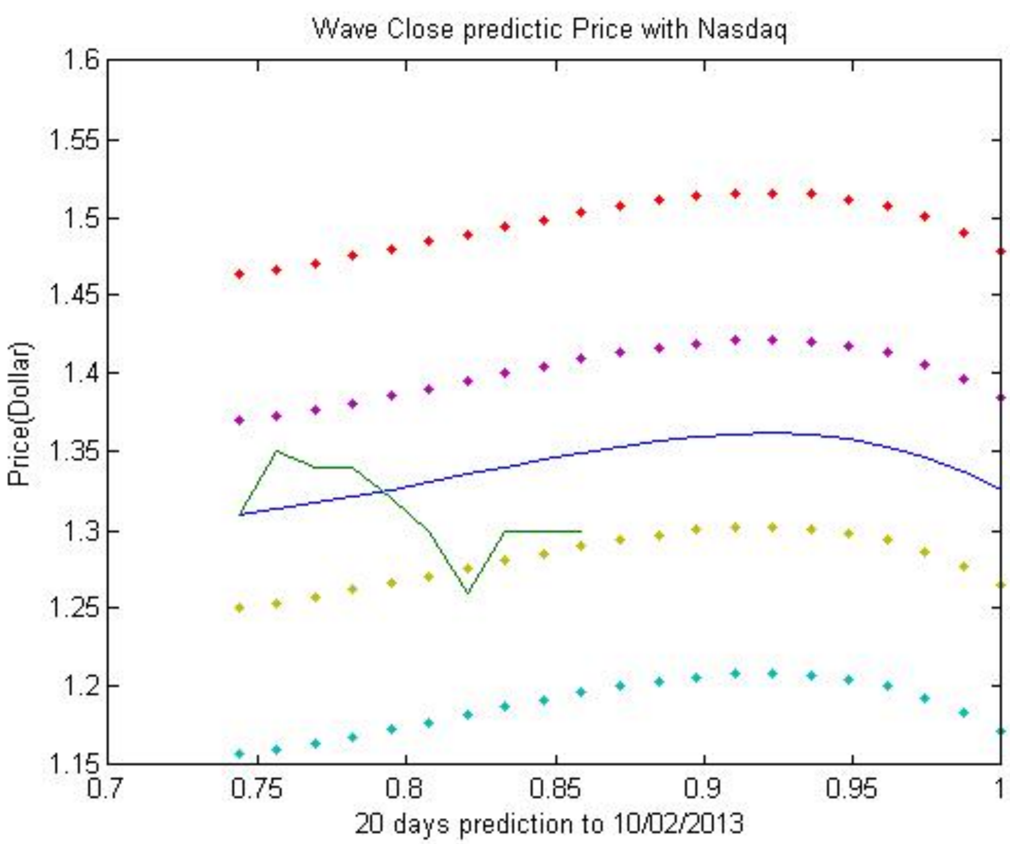

Bridge Close predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
1.2
1.15
1.1
1.05
1
0.95
0.9
0.85
0.8
0.75
0.7
0.75
0.8
0.85
0.9
0.95
1

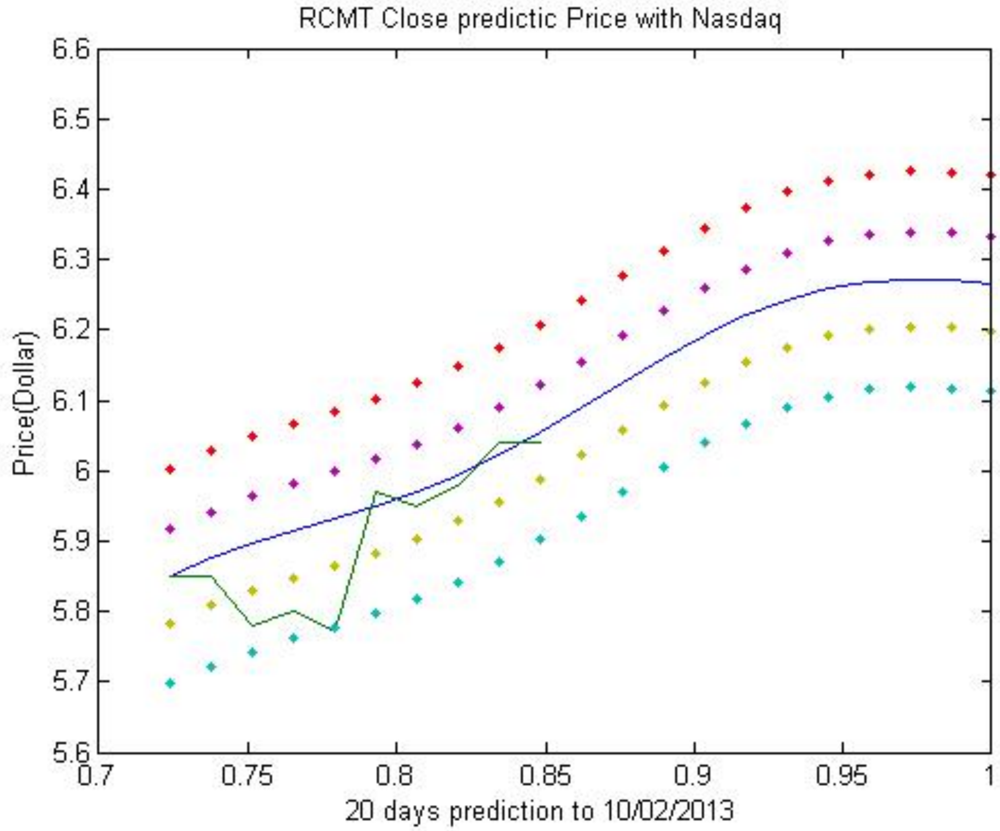

Google Close predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
930
920
910
900
890
880
870
860
850
840
0.8
0.82
0.84
0.86
0.88
0.9
0.92
0.94
0.96
0.98
1

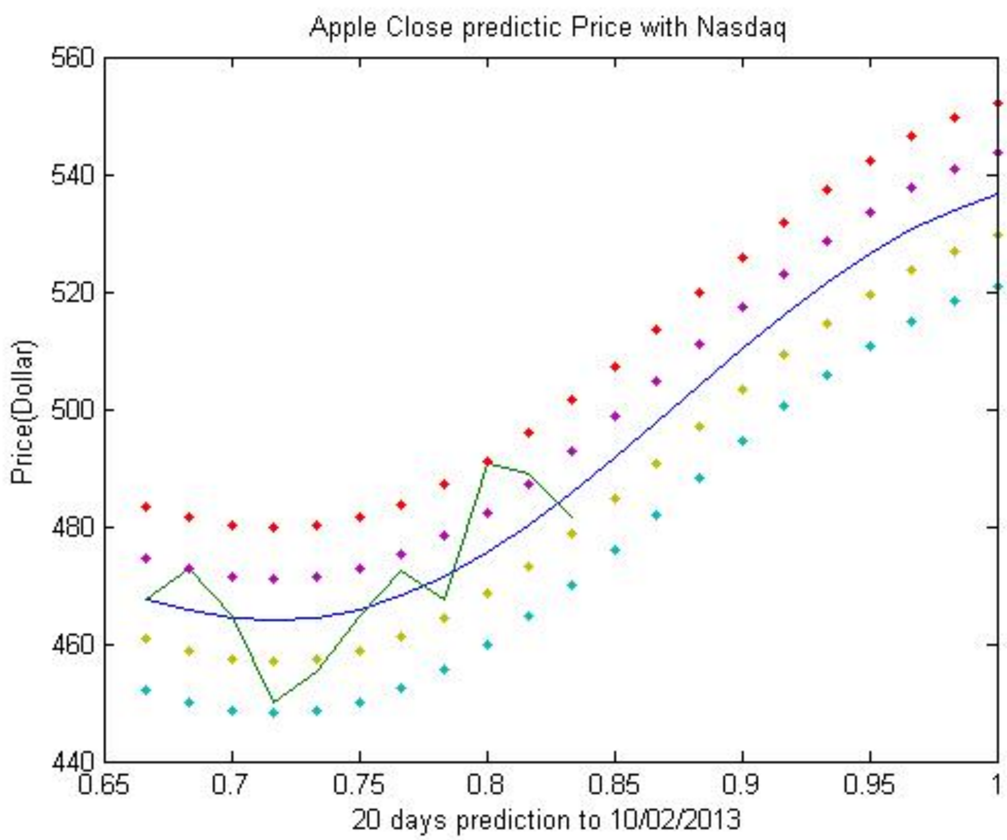

Mella Close predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
42
40
38
36
34
32
30
28
26
24
22
0.65
0.7
0.75
0.8
0.85
0.9
0.95
1


Microsofy Close predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
37
36
35
34
33
32
31
0.5
0.55
0.6
0.65
0.7
0.75
0.8
0.85
0.9
0.95
1

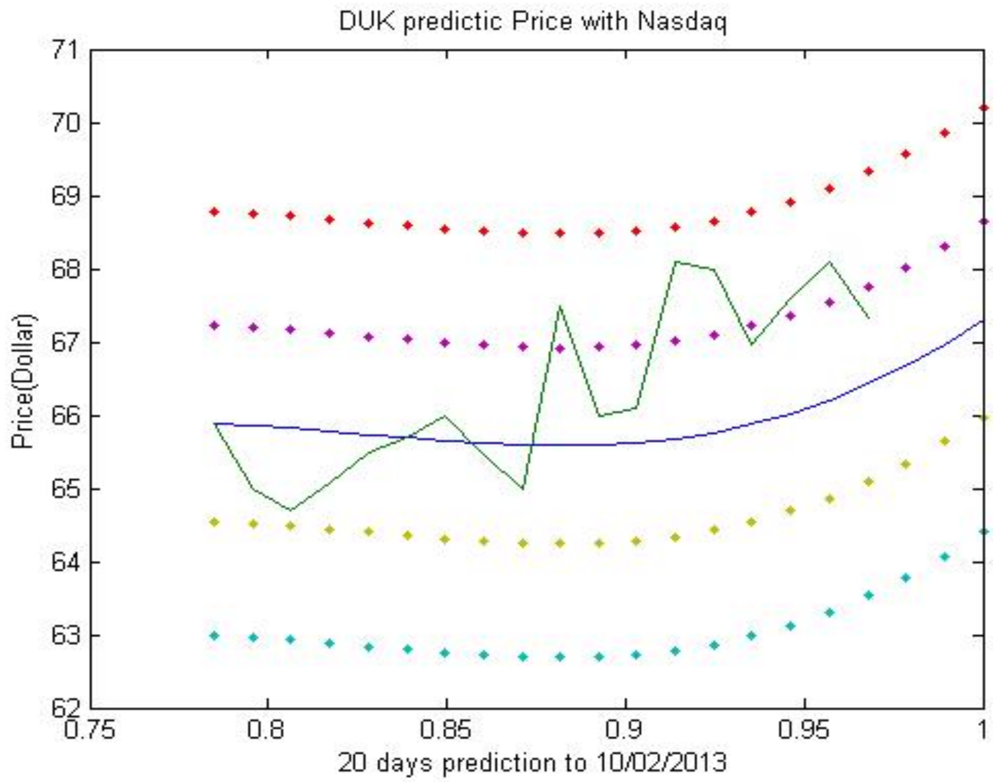

VMware Close predictic Price with Nasdaq
94
92
90
88
86
84
82
80
78
76
74
0.7
0.75
0.8
0.85
0.9
0.95
1
Price(Dollar)
20 days prediction to 10/02/2013

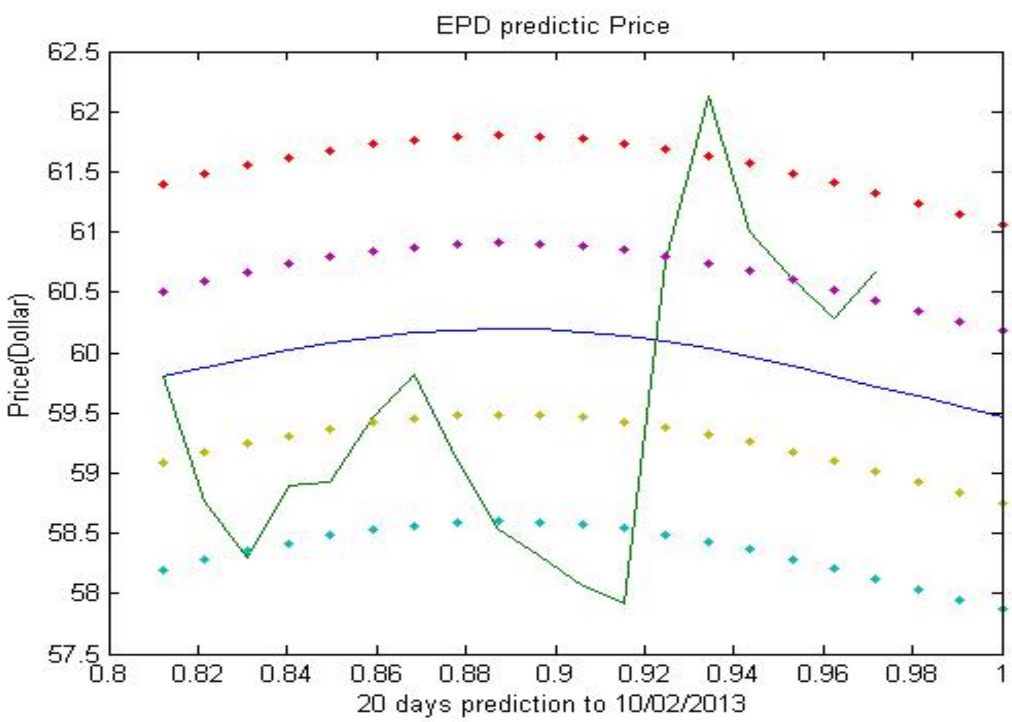

Wave Close predictic Price with Nasdaq
1.6
1.55
1.5
1.45
1.4
1.35
1.3
1.25
1.2
1.15
0.7
0.75
0.8
0.85
0.9
0.95
1
Price(Dollar)
20 days prediction to 10/02/2013

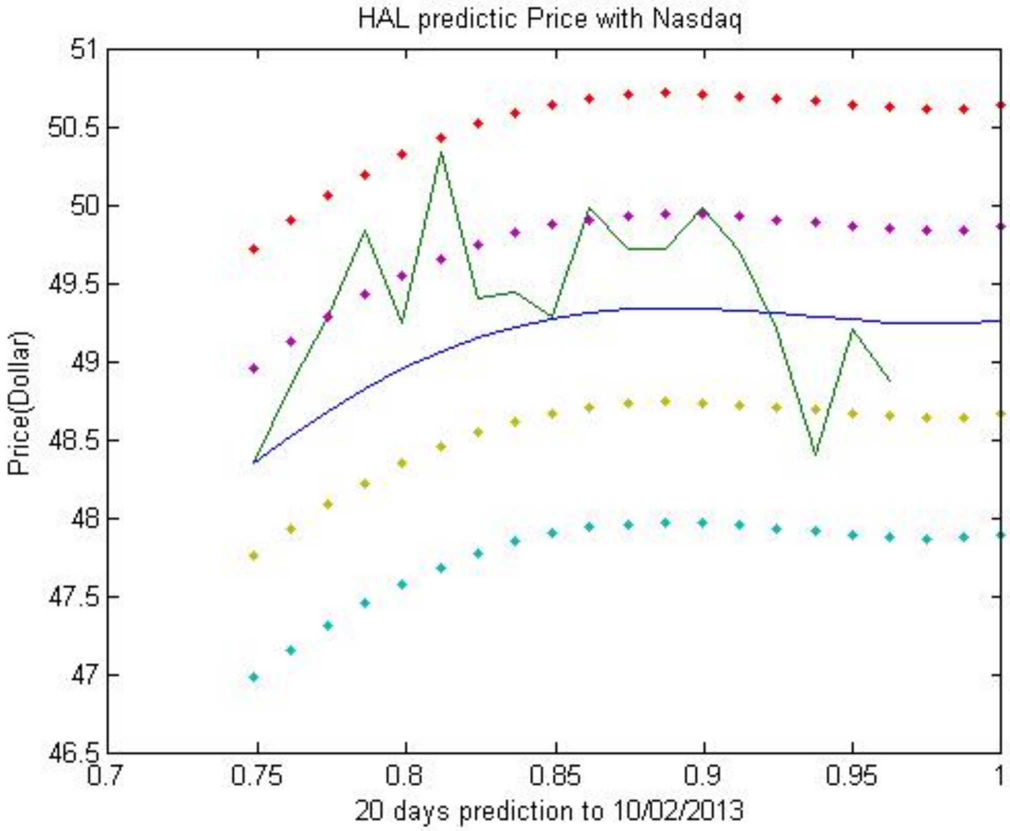

RCMT Close predictic Price with Nasdaq
Price(Dollar)
6.6
6.5
6.4
6.3
6.2
6.1
6
5.9
5.8
5.7
5.6
0.7
0.75
0.8
0.85
0.9
0.95
1
20 days prediction to 10/02/2013

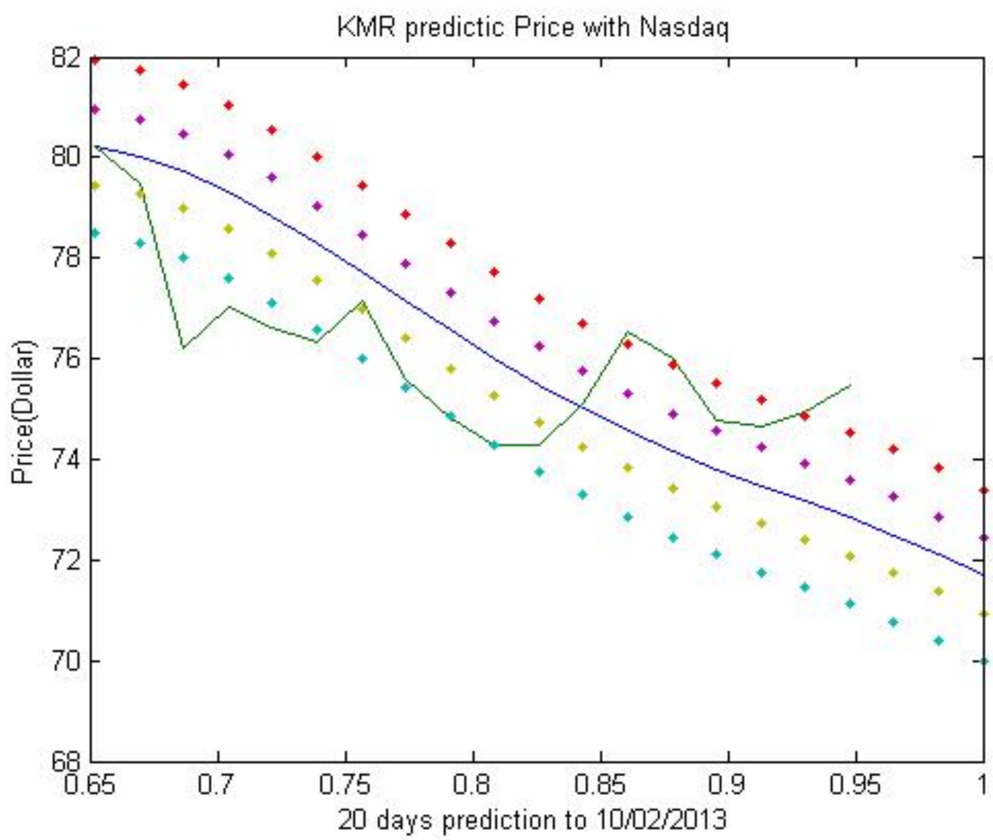

Apple Close predictic Price with Nasdaq
Price(Dollar)
560
540
520
500
480
460
440
0.65
0.7
0.75
0.8
0.85
0.9
0.95
1
20 days prediction to 10/02/2013

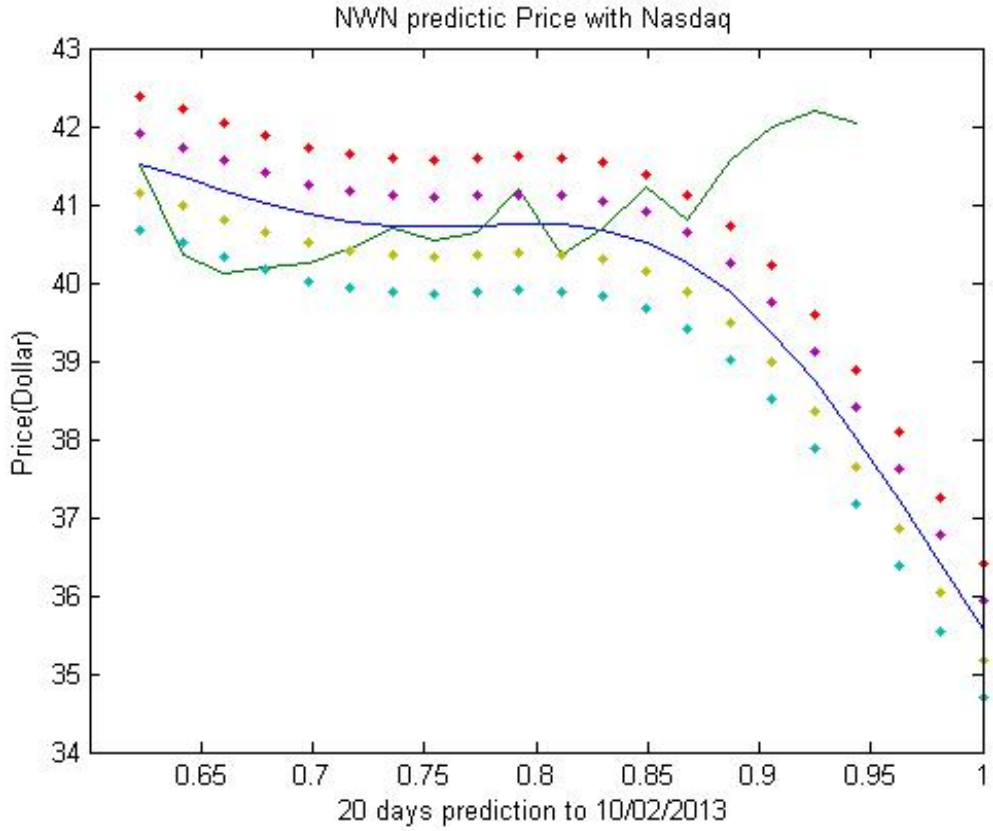
Energy:

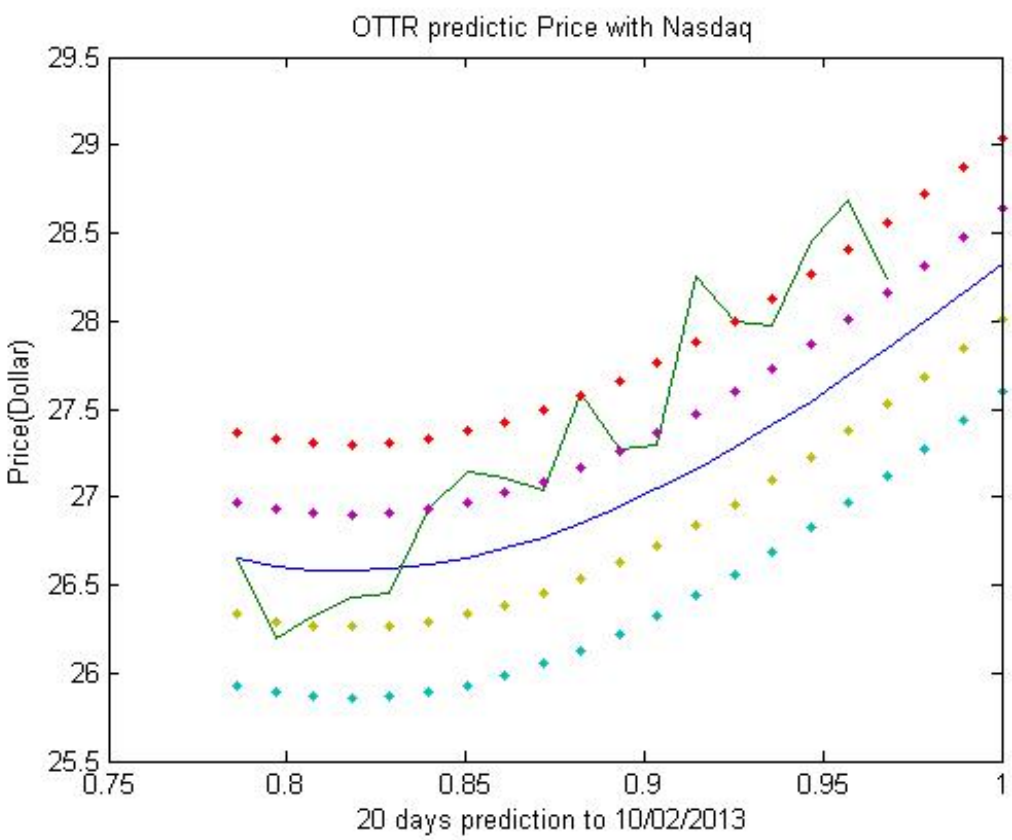

DUK predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
71
70
69
68
67
66
65
64
63
62
0.75
0.8
0.85
0.9
0.95
1

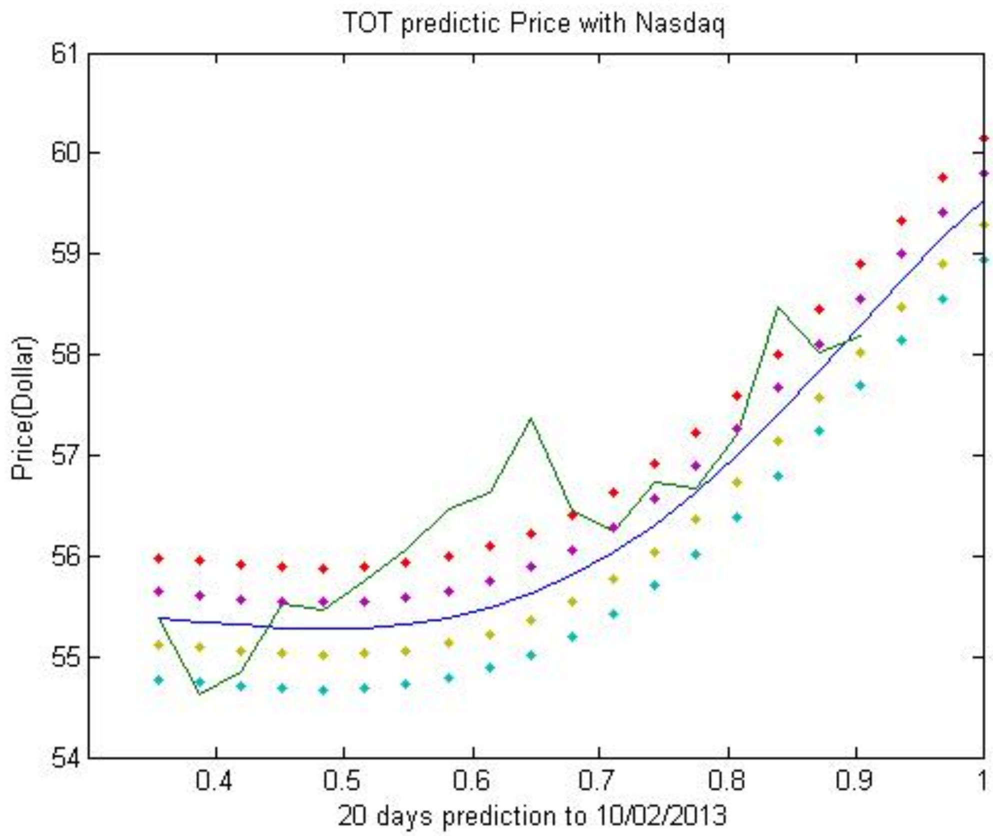

EPD predictic Price
Price(Dollar)
20 days prediction to 10/02/2013
62.5
62
61.5
61
60.5
60
59.5
59
58.5
58
57.5
0.8
0.82
0.84
0.86
0.88
0.9
0.92
0.94
0.96
0.98
1

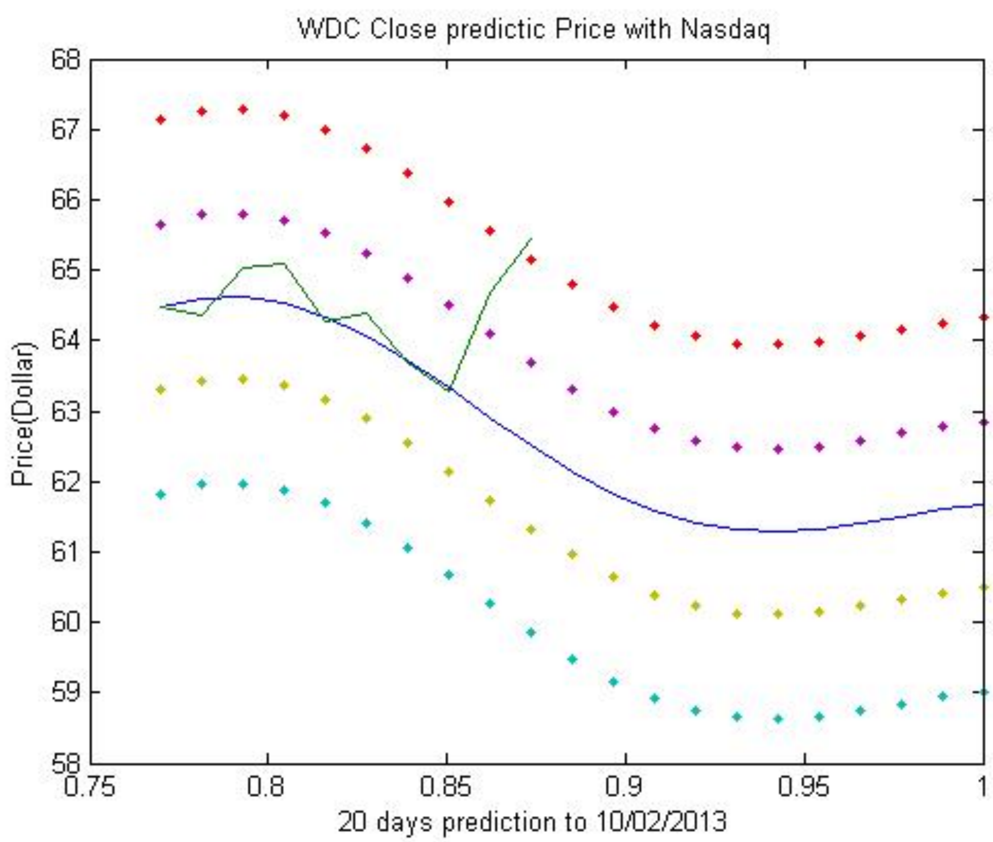

HAL predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
51
50.5
50
49.5
49
48.5
48
47.5
47
46.5
0.7
0.75
0.8
0.85
0.9
0.95
1

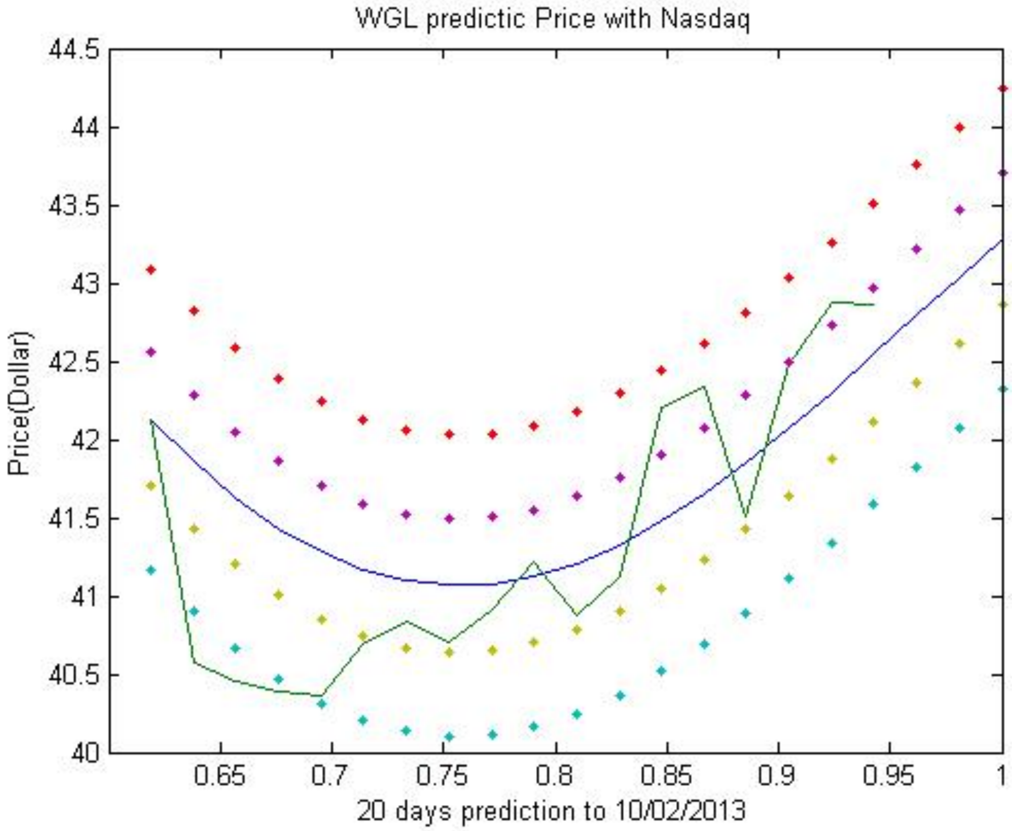

KMR predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
82
80
78
76
74
72
70
68
0.65
0.7
0.75
0.8
0.85
0.9
0.95
1

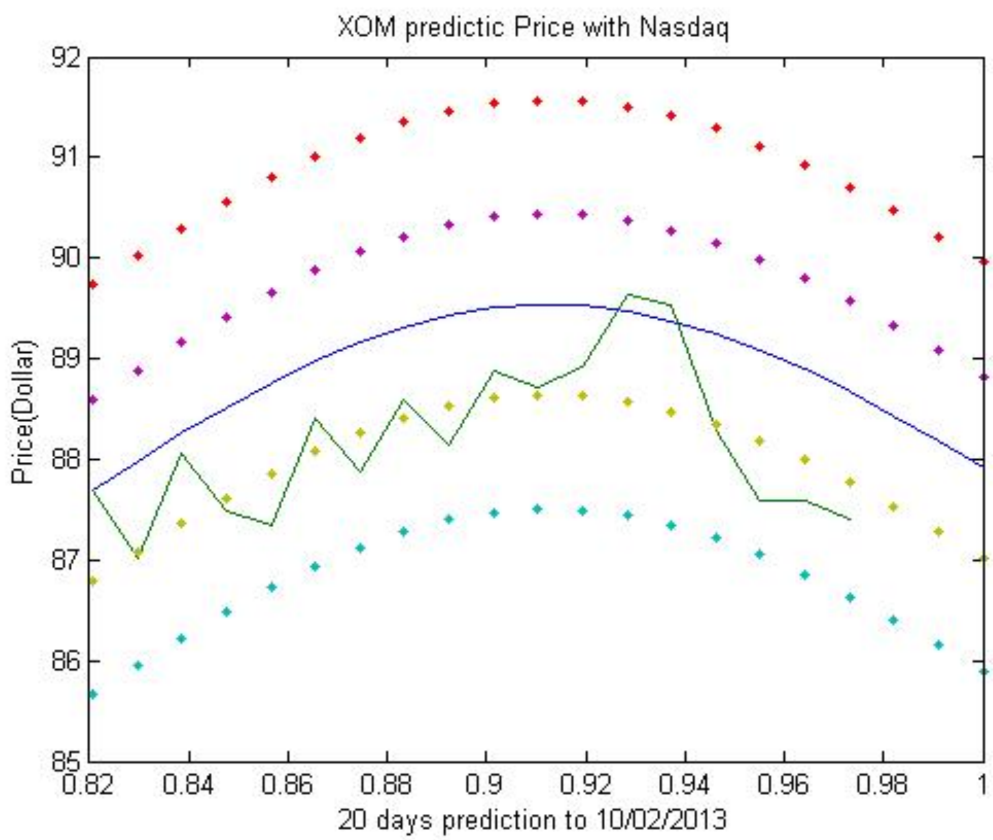

NWN predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
43
42
41
40
39
38
37
36
35
34
0.65
0.7
0.75
0.8
0.85
0.9
0.95
1

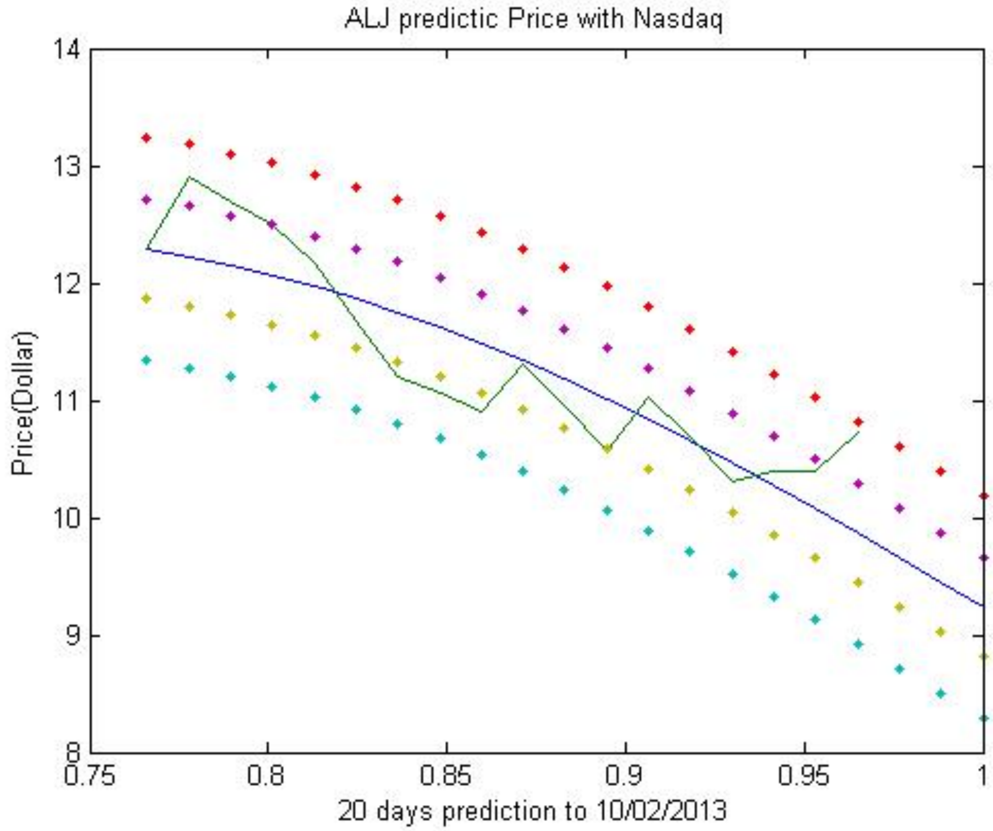

OTTR predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
29.5
29
28.5
28
27.5
27
26.5
26
25.5
0.75
0.8
0.85
0.9
0.95
1


TOT predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
61
60
59
58
57
56
55
54
0.4
0.5
0.6
0.7
0.8
0.9
1


WDC Close predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
68
67
66
65
64
63
62
61
60
59
58
0.75
0.8
0.85
0.9
0.95
1


WGL predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
44.5
44
43.5
43
42.5
42
41.5
41
40.5
40
0.65
0.7
0.75
0.8
0.85
0.9
0.95
1


XOM predictic Price with Nasdaq
Price(Dollar)
20 days prediction to 10/02/2013
92
91
90
89
88
87
86
85
0.82
0.84
0.86
0.88
0.9
0.92
0.94
0.96
0.98
1

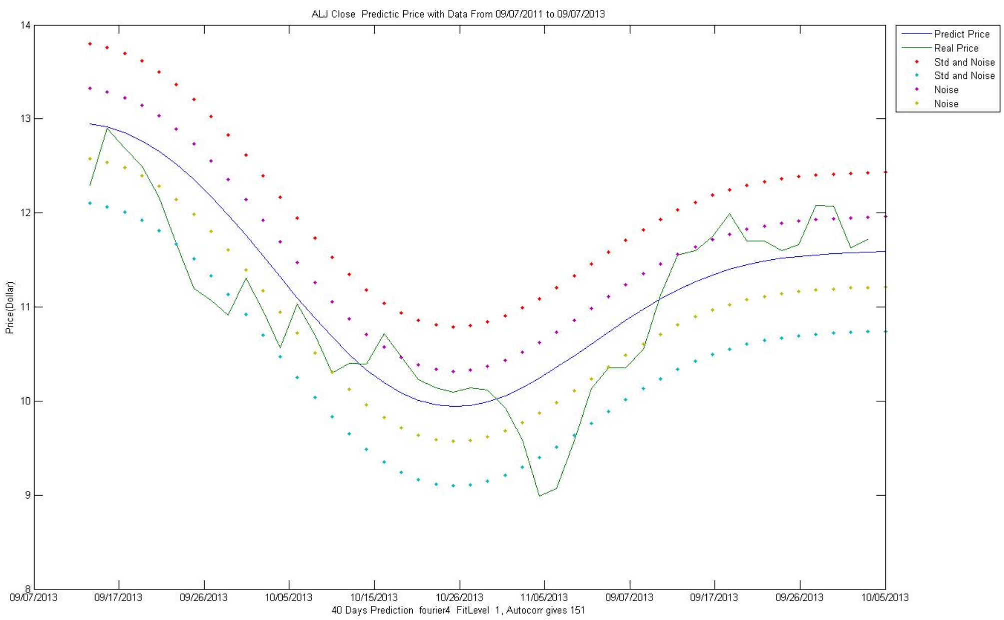


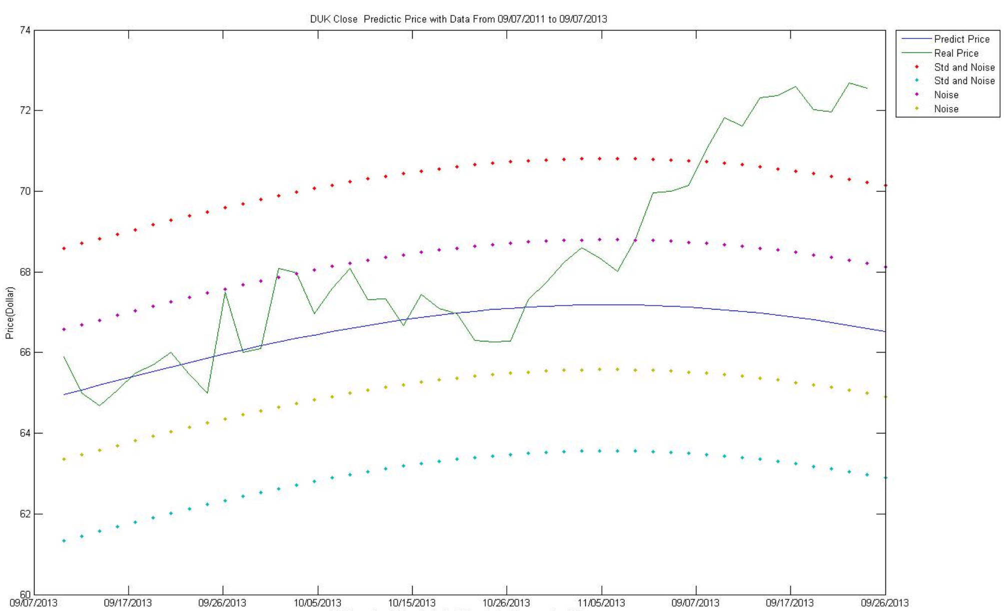
As we can see from all the charts above, some of the stocks that we chose are not showing what we expected.

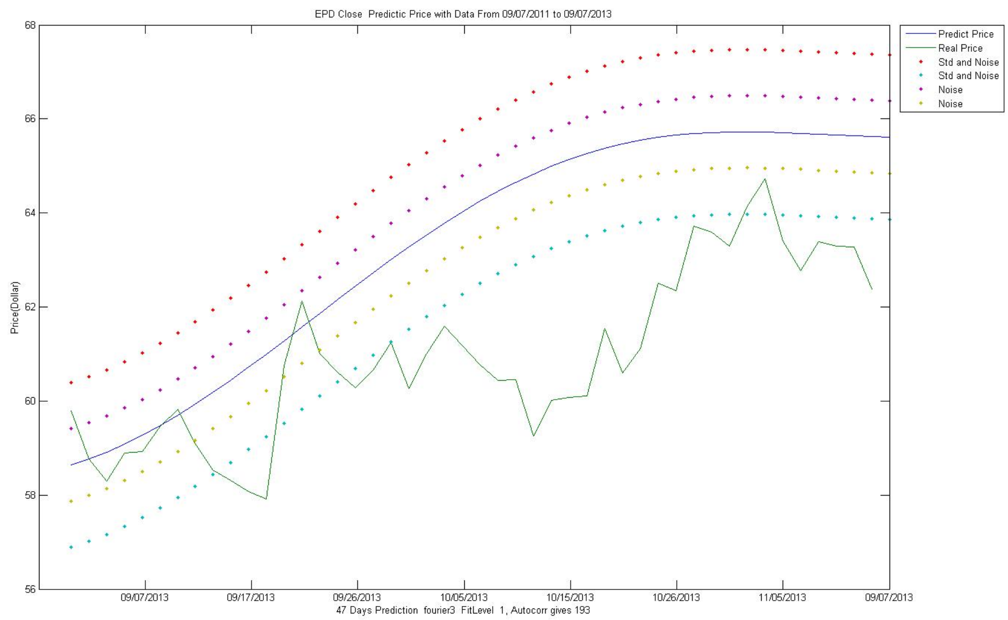
e. 40 business days' and 27 business days' performance of our stocks with modified model

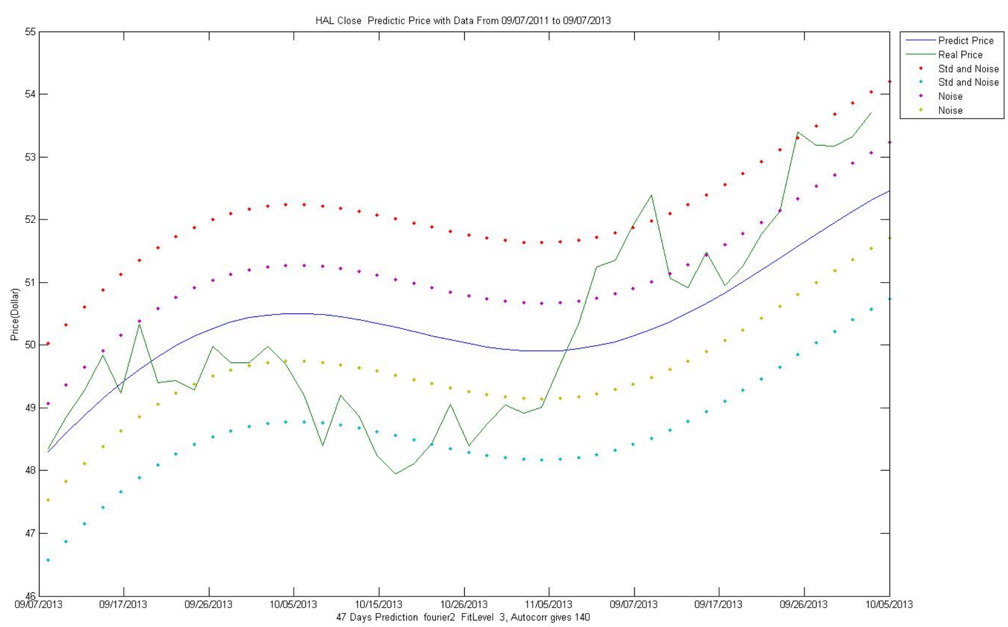
Since the prediction together with the affection of Nasdaq doesn't make so much difference, we would like to show the performance of our stocks with the trend and Fourier expansion just with the stocks themselves. Let us take a look at the predictions.

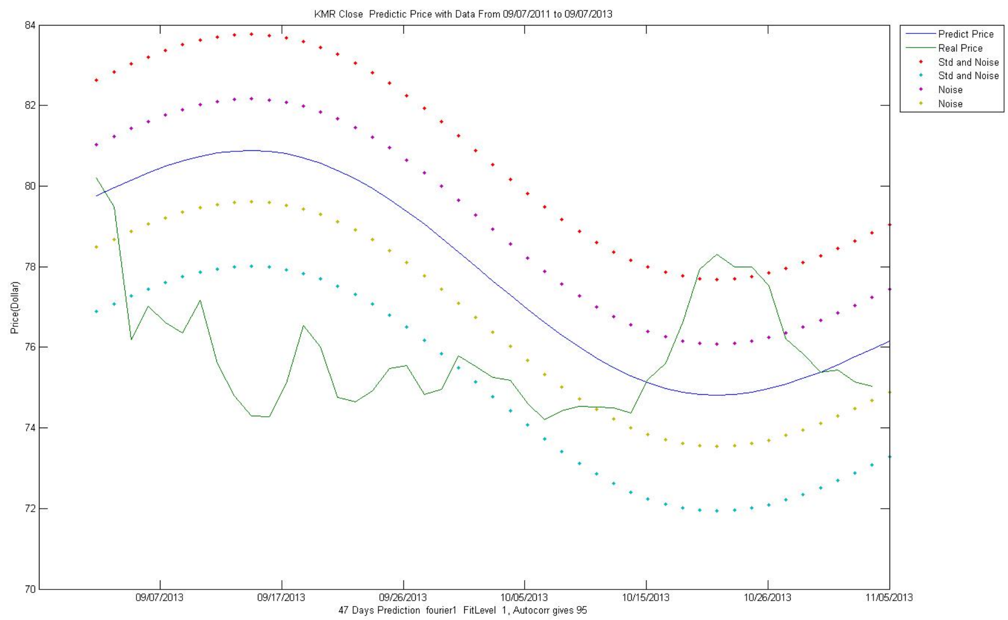
Stocks in energy field


ALJ Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
14
13
12
11
10
9
8
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
09/07/2013
09/17/2013
09/26/2013
10/05/2013
40 Days Prediction fourier4 FitLevel 1, Autocorr gives 151

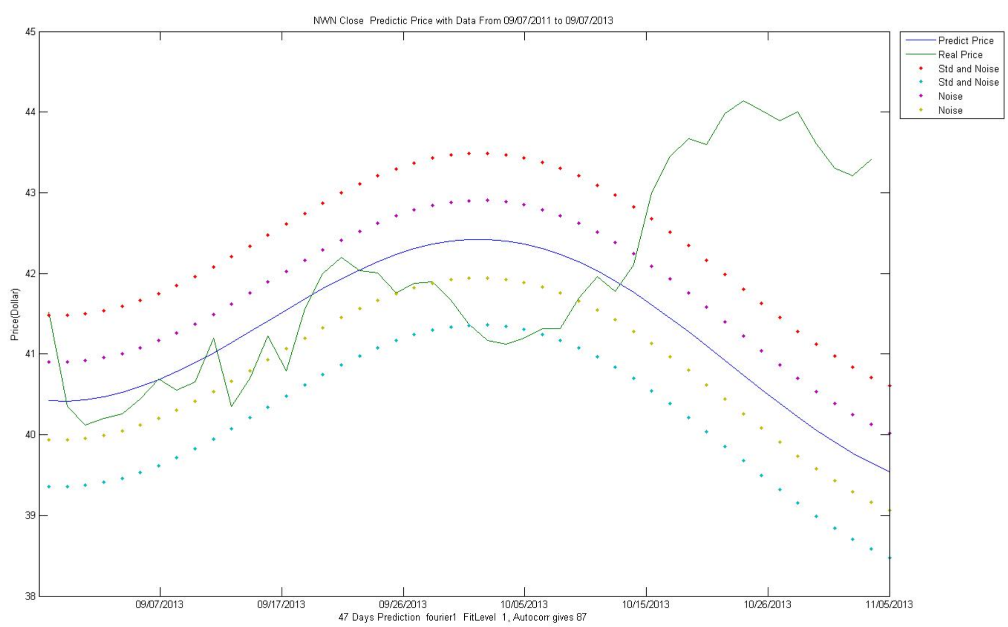

DUK Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
74
72
70
68
66
64
62
60
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
09/07/2013
09/17/2013
09/26/2013
47 Days Prediction fourier1 FitLevel 1, Autocorr gives 166


EPD Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
68
66
64
62
60
58
56
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
09/07/2013
47 Days Prediction fourier3 FitLevel 1, Autocorr gives 193

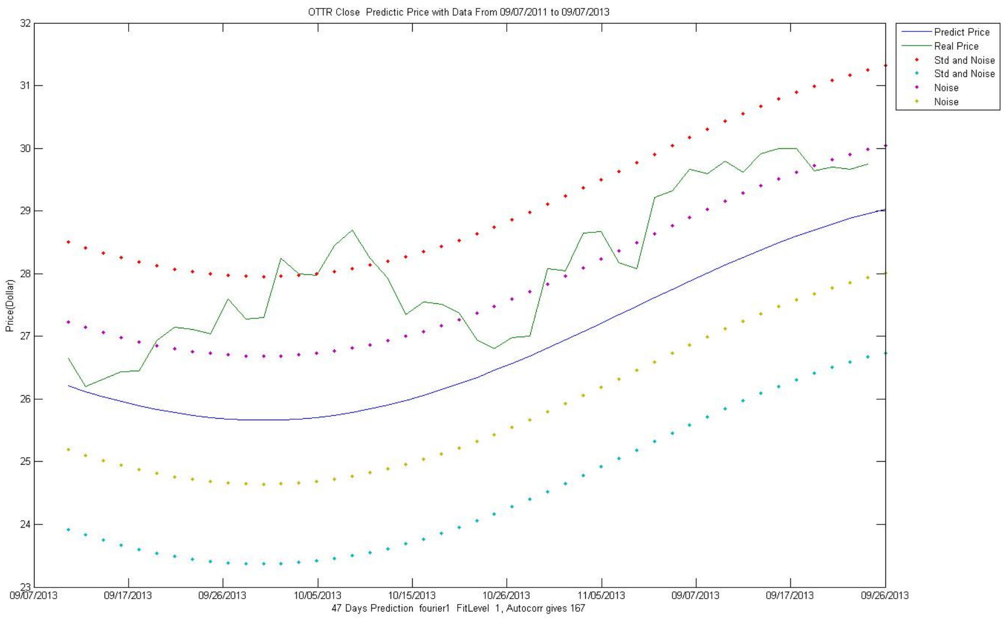

HAL Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
55
54
53
52
51
50
49
48
47
46
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
09/07/2013
09/17/2013
09/26/2013
10/05/2013
47 Days Prediction fourier2 FitLevel 3, Autocorr gives 140



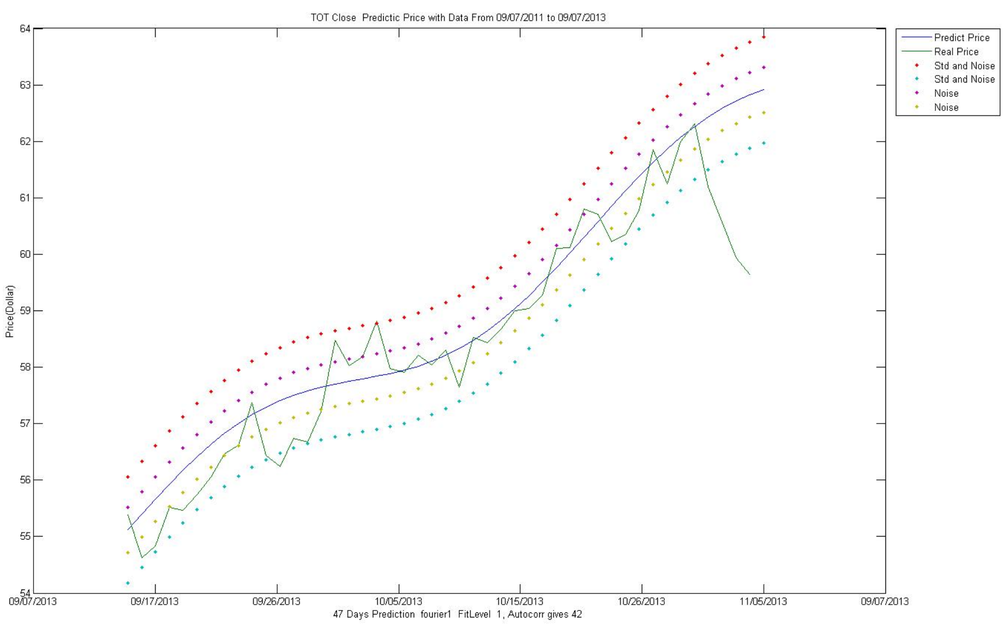
KMR Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
47 Days Prediction fourier1 FitLevel 1, Autocorr gives 95



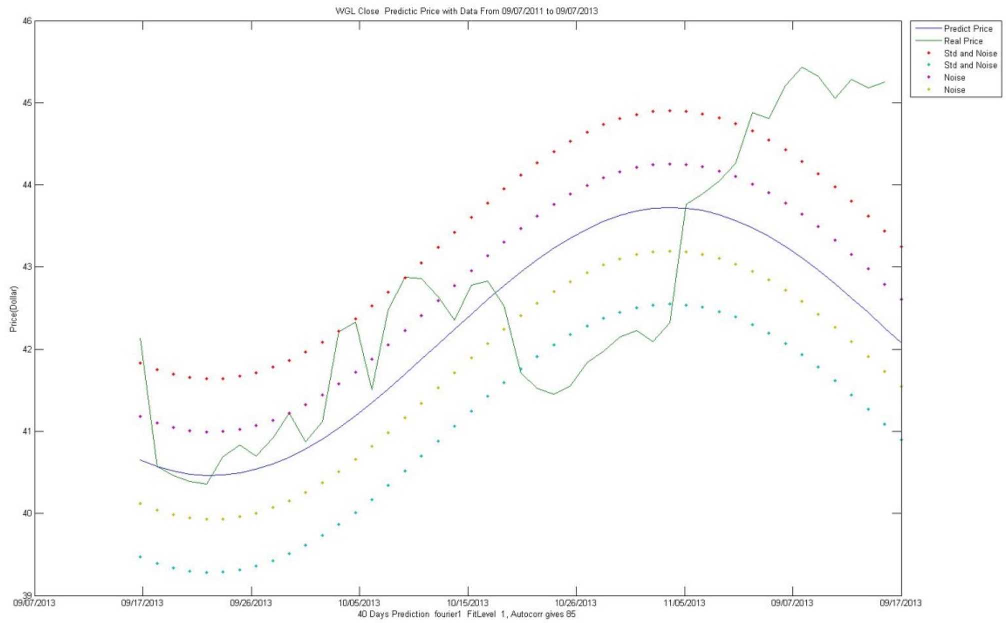
NWN Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
47 Days Prediction fourier1 FitLevel 1, Autocorr gives 87

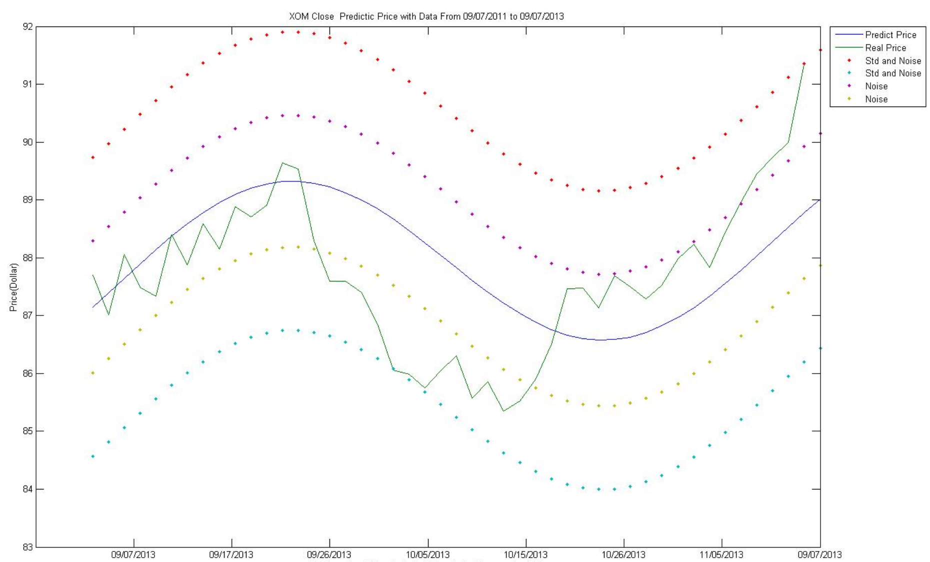


OTTR Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
32
31
30
29
28
27
26
25
24
23
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
09/07/2013
09/17/2013
09/26/2013
47 Days Prediction fourier1 FitLevel 1, Autocorr gives 167



TOT Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
64
63
62
61
60
59
58
57
56
55
54
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
09/07/2013
47 Days Prediction fourier1 FitLevel 1, Autocorr gives 42





For 40 business days Prediction:

5 out of 10 stocks: "ALJ, HAL, KMR, OTTR, XOM" The actual price is within the "Noise+ Std".

2 out of 10 stocks: "DUK, NWN" the actual price is higher than the "Noise+ Std".

2 out of 10 stocks: "TOT, EPD" the actual price is Lower than the "Noise+ Std".

7 out of 10 stocks: “ALJ, DUK, HAL, NWN, OTTR, WGL, XOM” The actual price is higher than the predicted price.

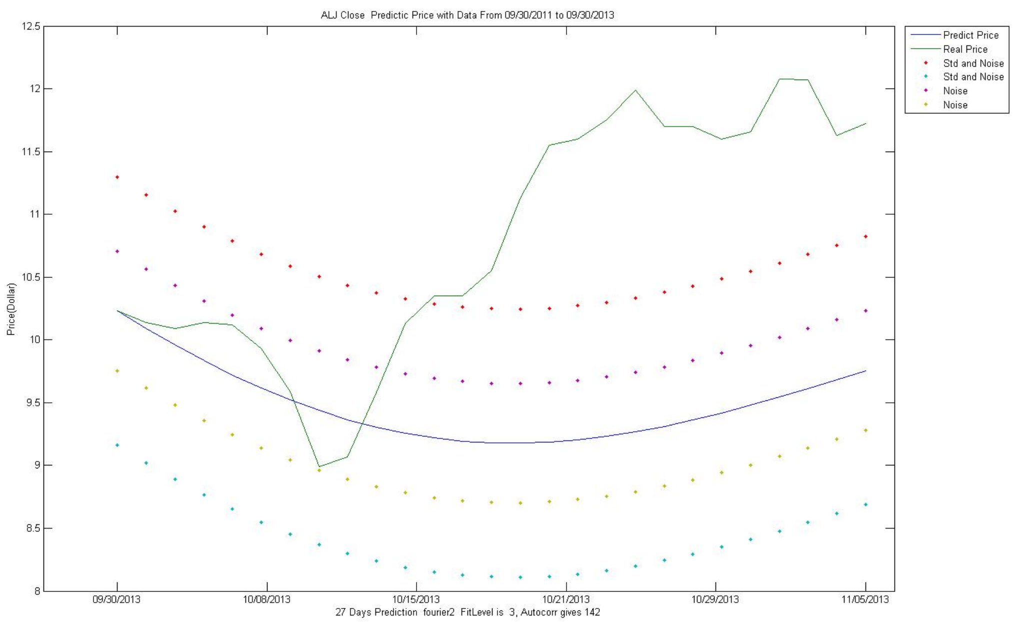
3 out of 10 stocks: “TOT, EPD, and KMR” The actual price is lower than the predicted price.

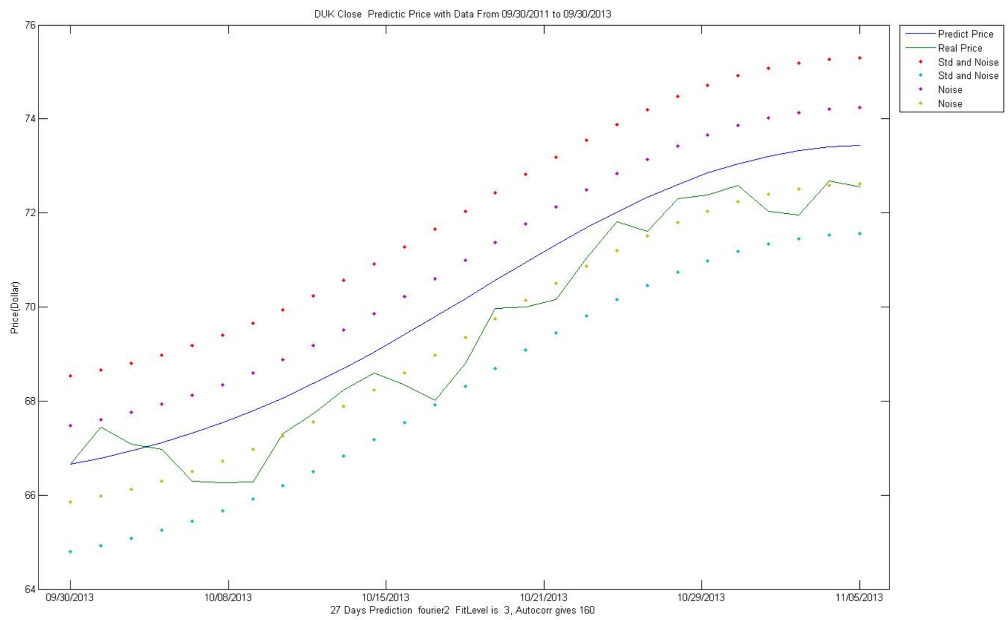
3 stocks “ALJ, OTTR, KMR” The Predicted price is fairly close to the actual price.

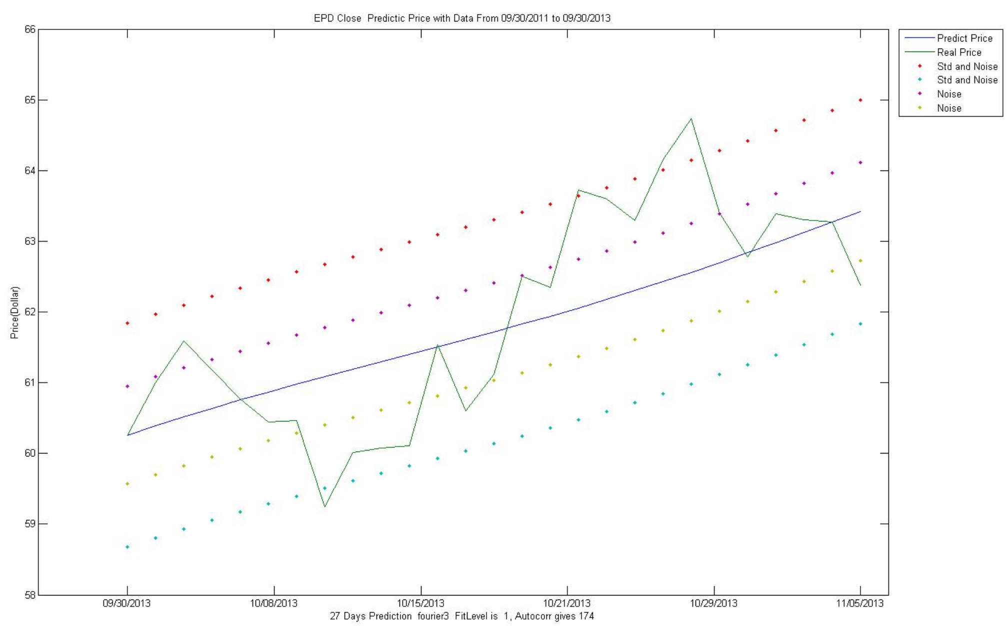
Overall, ALJ, HAL, TOT and XOM fit well.

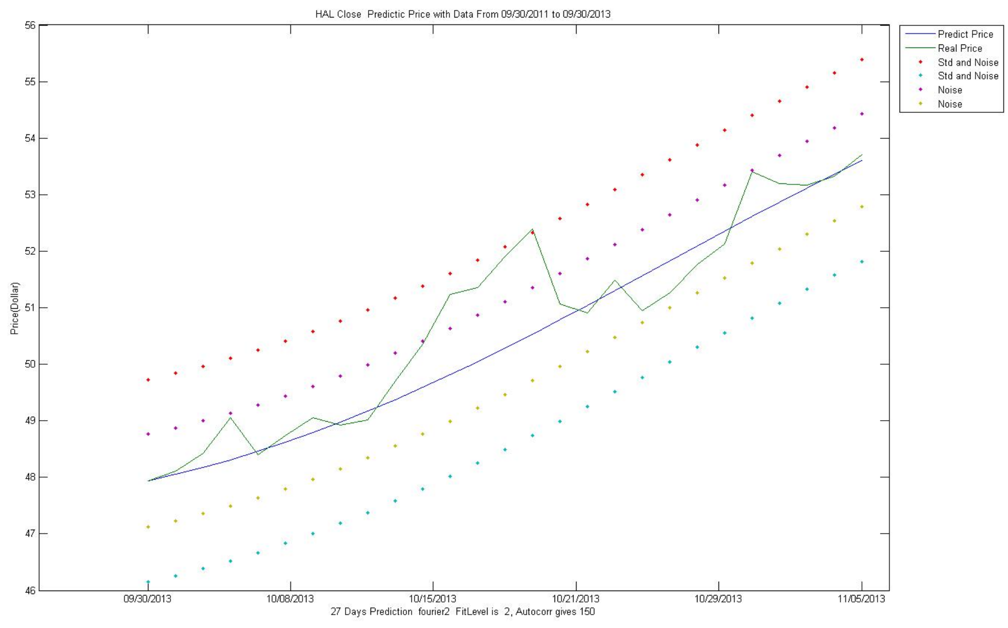
The actual DUK, NWN, WGL OTTR stock price is higher than the prediction

The actual EPD stock price is lower than the prediction

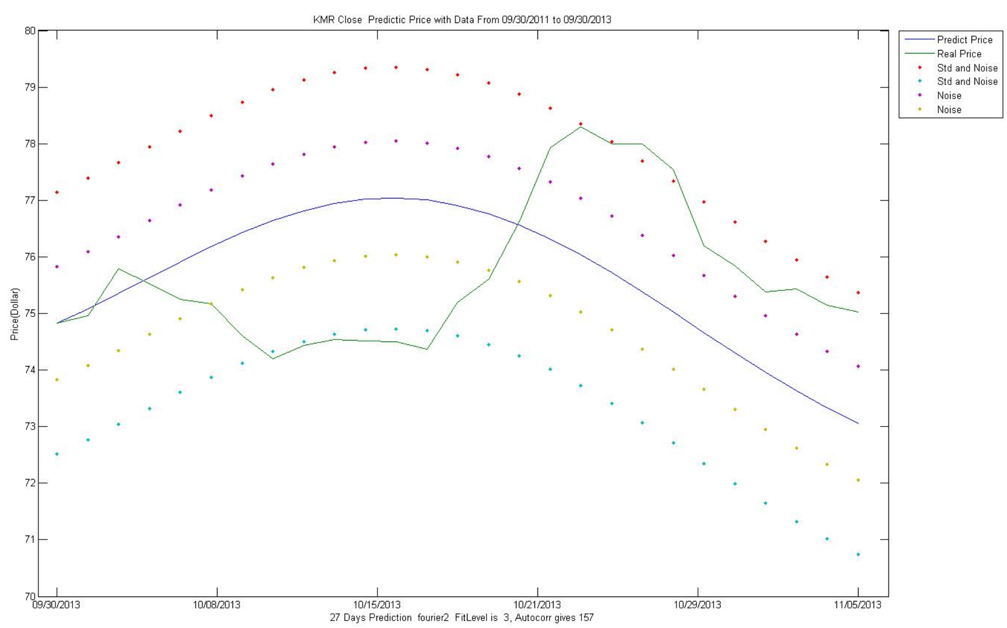



DUK Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
76
74
72
70
68
66
64
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 3, Autocorr gives 160

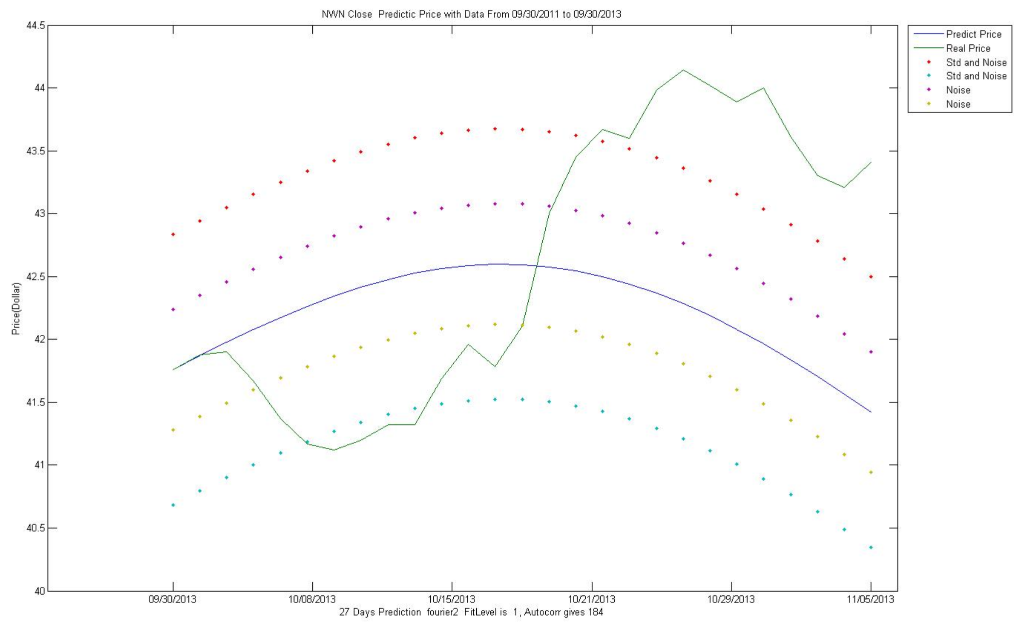

EPD Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
66
65
64
63
62
61
60
59
58
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier3 FitLevel is 1, Autocorr gives 174

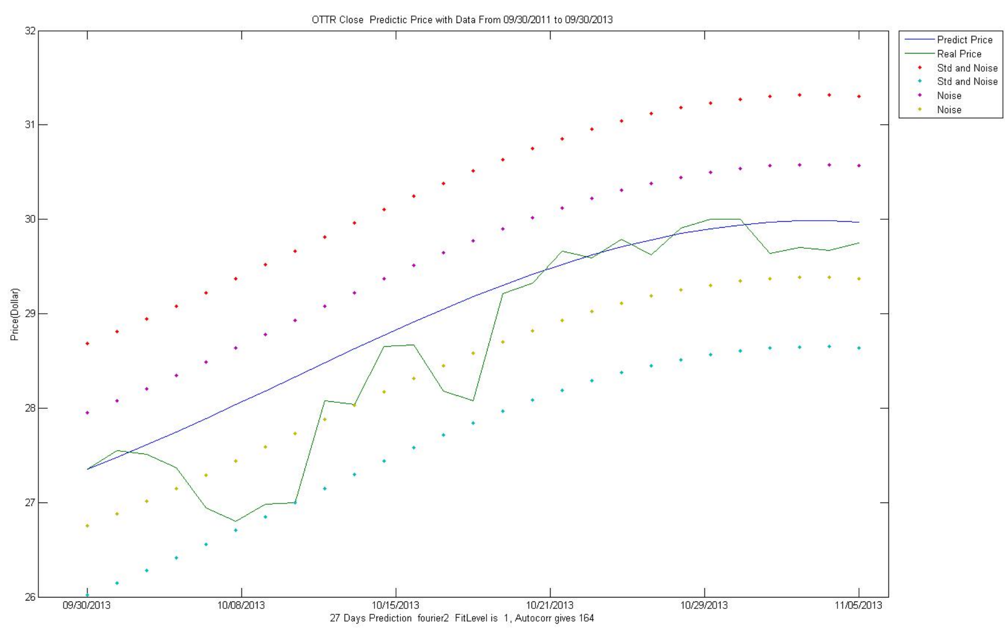


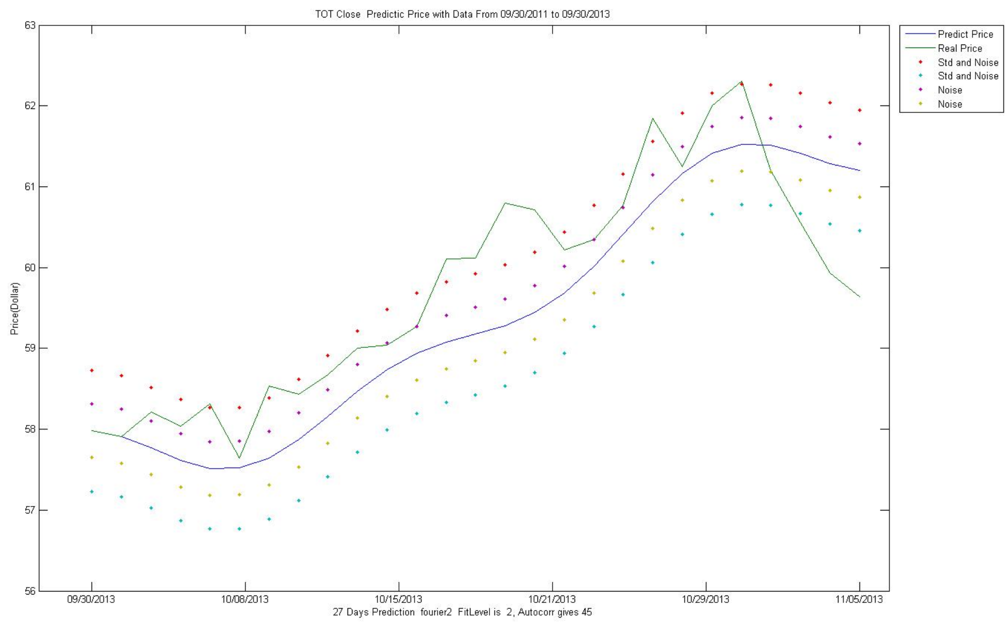
HAL Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
56
55
54
53
52
51
50
49
48
47
46
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 2, Autocorr gives 150

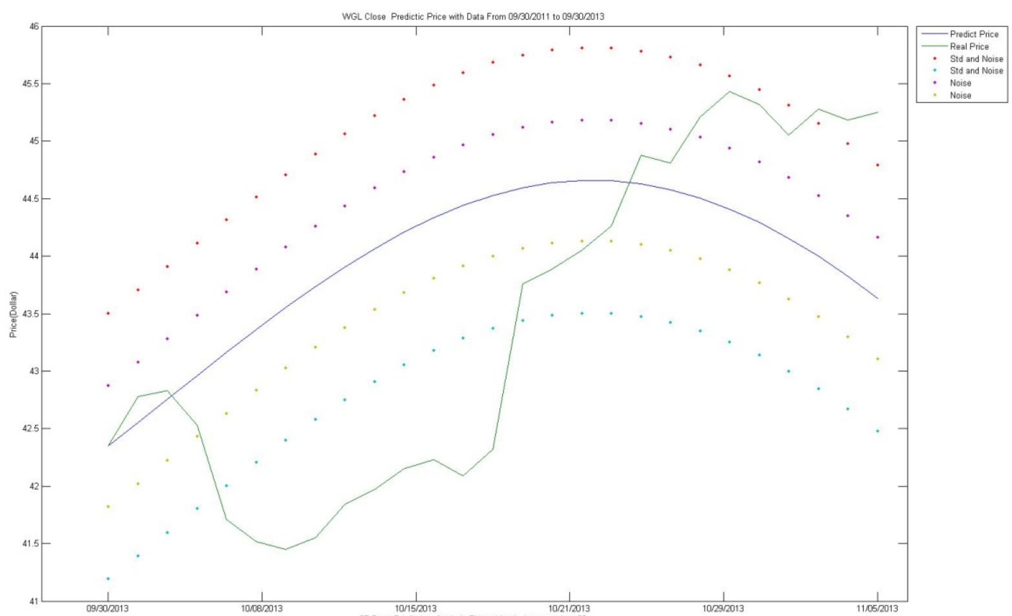


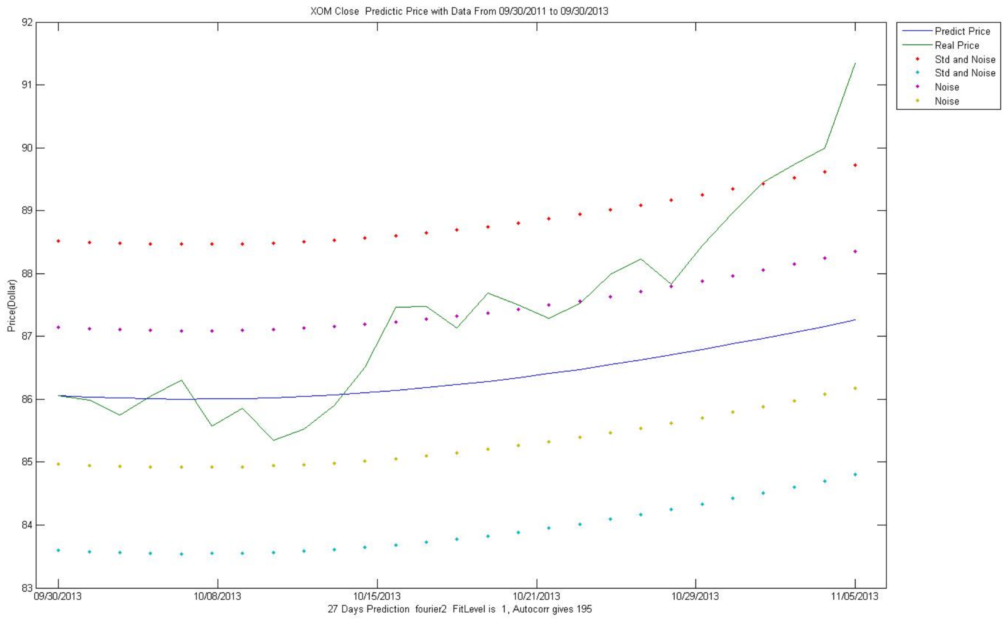
KMR Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
80
79
78
77
76
75
74
73
72
71
70
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 3, Autocorr gives 157


NWN Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
44.5
44
43.5
43
42.5
42
41.5
41
40.5
40
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 1, Autocorr gives 184


OTTR Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
32
31
30
29
28
27
26
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 1, Autocorr gives 164


TOT Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
63
62
61
60
59
58
57
56
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 2, Autocorr gives 45


WGL Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
Std and Noise
Std and Noise
Noise
Noise
Price(Dollar)
46
45.5
45
44.5
44
43.5
43
42.5
42
41.5
41
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier1 FitLevel is 1, Autocorr gives 90



For 27 business days Prediction:

5 out of 10 stocks: “DUK, EPD, HAL, KMR, and OTTR” The actual price is within the “Noise+ Std”.

4 out of 10 stocks: “ALJ, NWN, WGL, and XOM” the final actual price is higher than the “Noise+ Std”.

1 out of 10 stocks: “TOT” the final actual price is lower than the “Noise+ Std”.

5 out of 10 stocks: “ALJ, KMR, NWN, WGL, and XOM” The final actual price is higher than the predicted price.

5 out of 10 stocks: “DUK, EPD, HAL, OTTR, and TOT” The final actual price is lower than the predicted price.

2 stocks “HAL, OTTR” The Predicted price is fairly close to the final actual price.

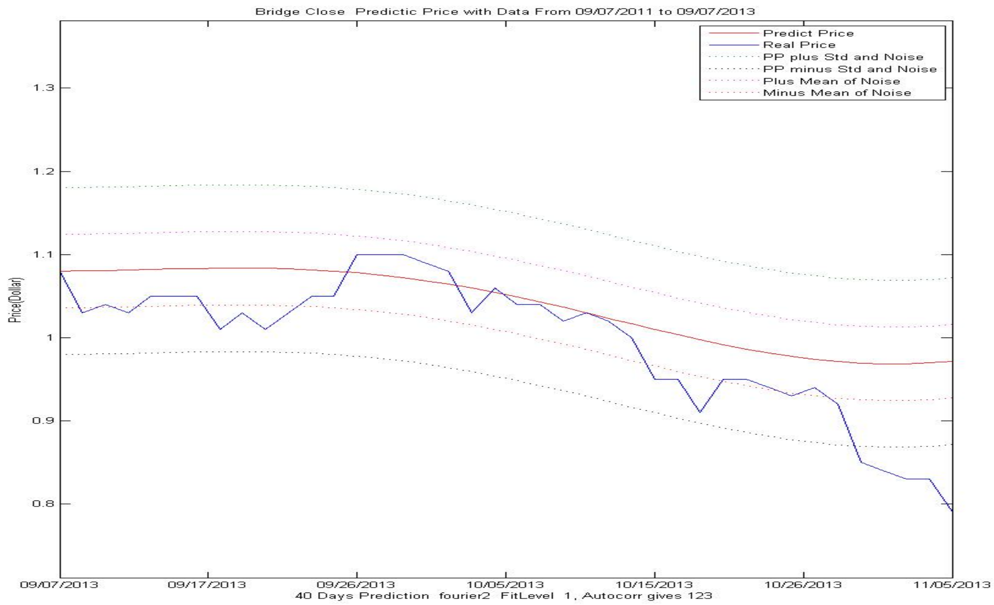
Overall, DUK HAL KMR, OTTR, TOT and XOM fit well.

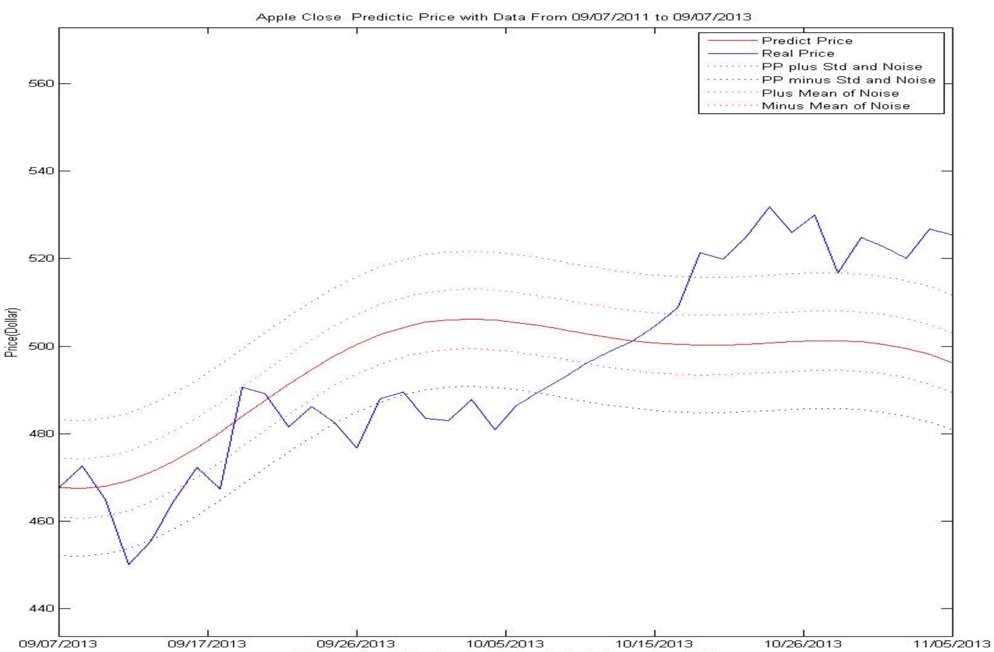
The actual ALJ TOT XOM stock price is higher than the prediction.

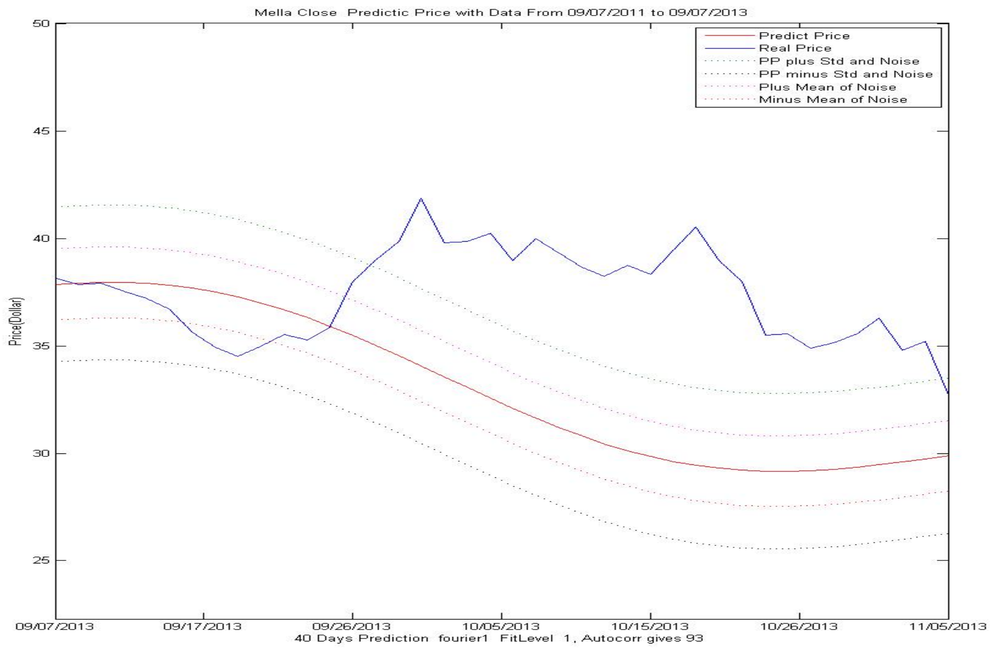
The actual DUK, OTTR stock price is lower than the prediction.

EPD, KMR, NWN and WGL are floating around the prediction.

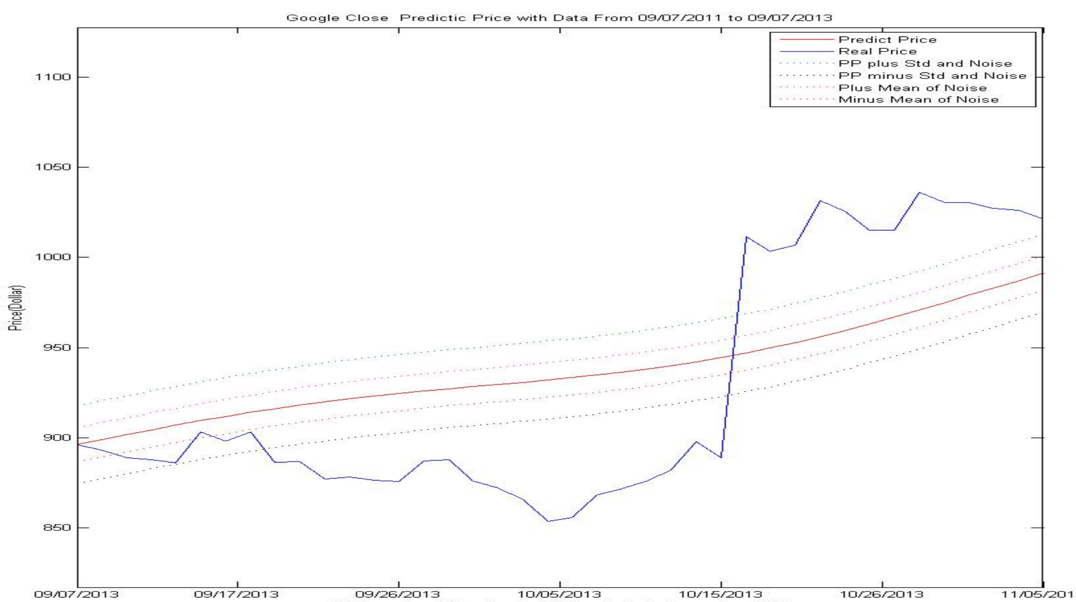
Stocks in technology field



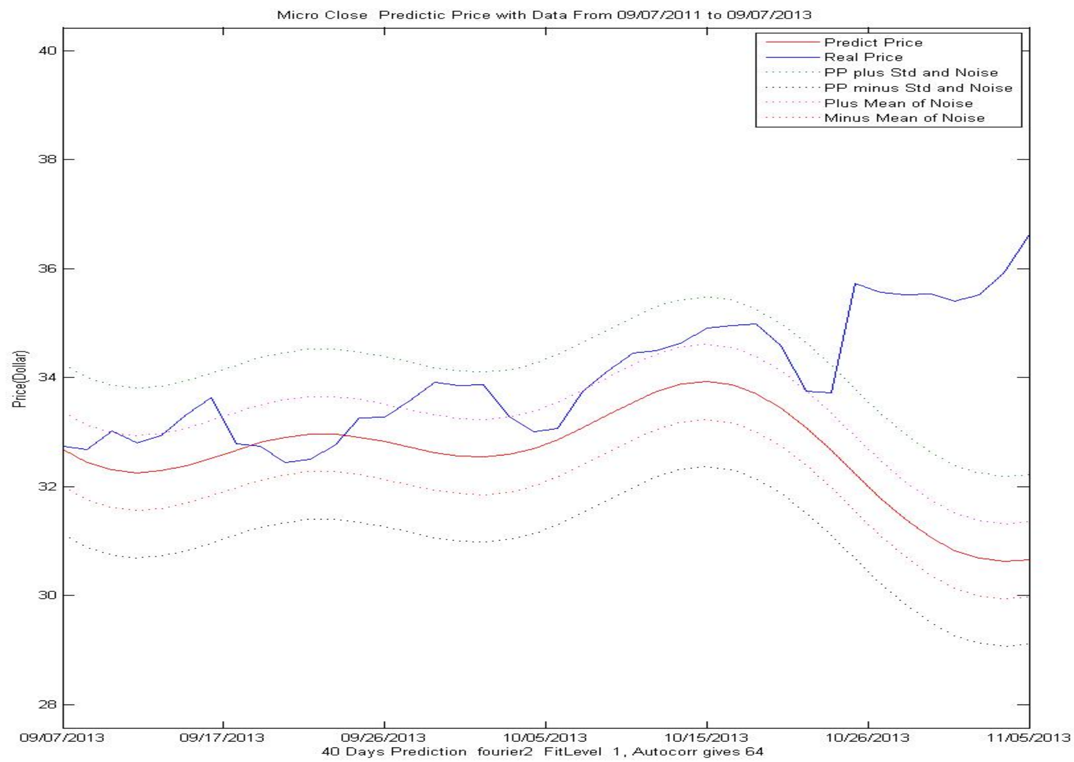


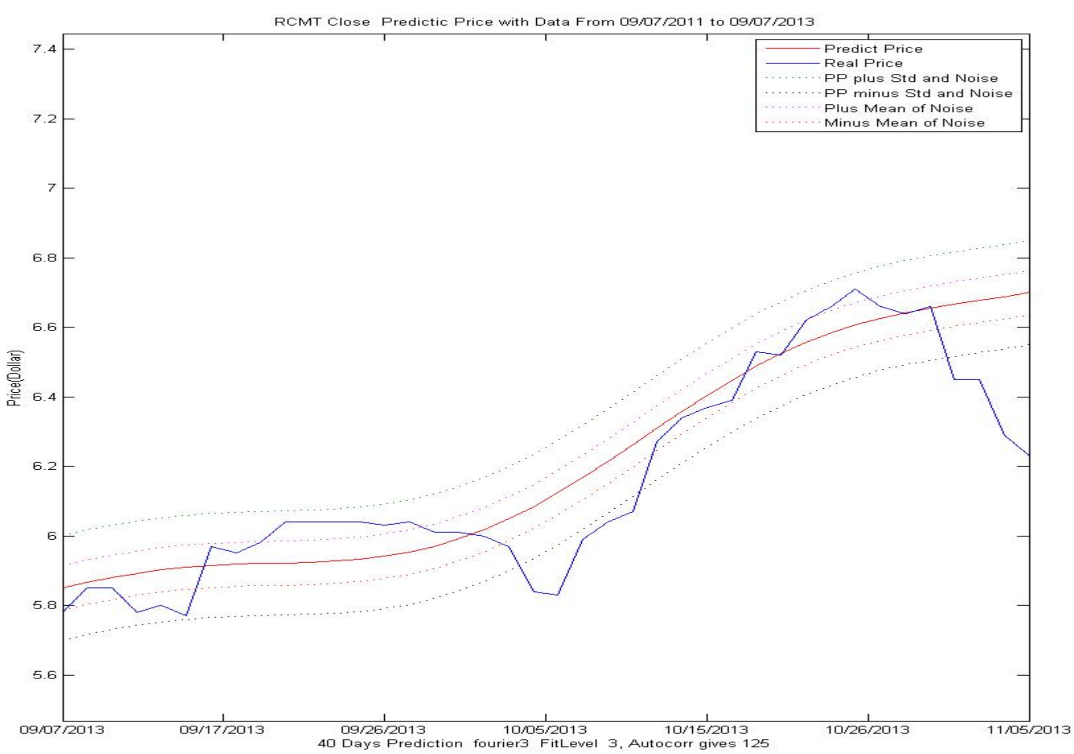


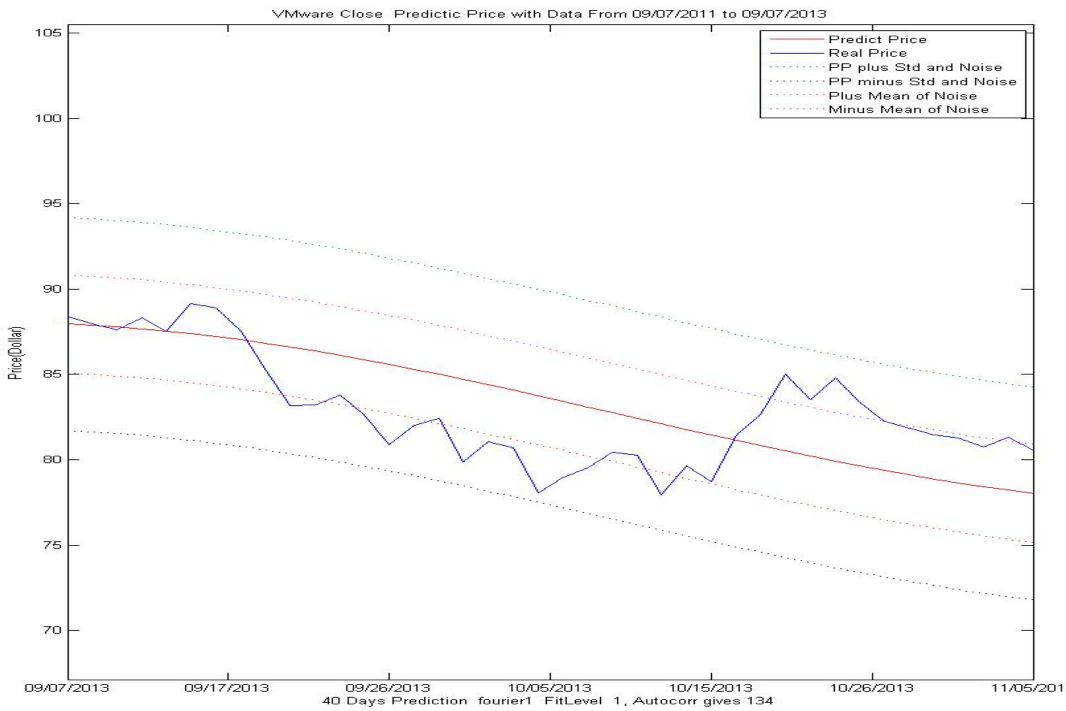
Mella Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
50
45
40
35
30
25
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
40 Days Prediction fourier1 FitLevel 1, Autocorr gives 93

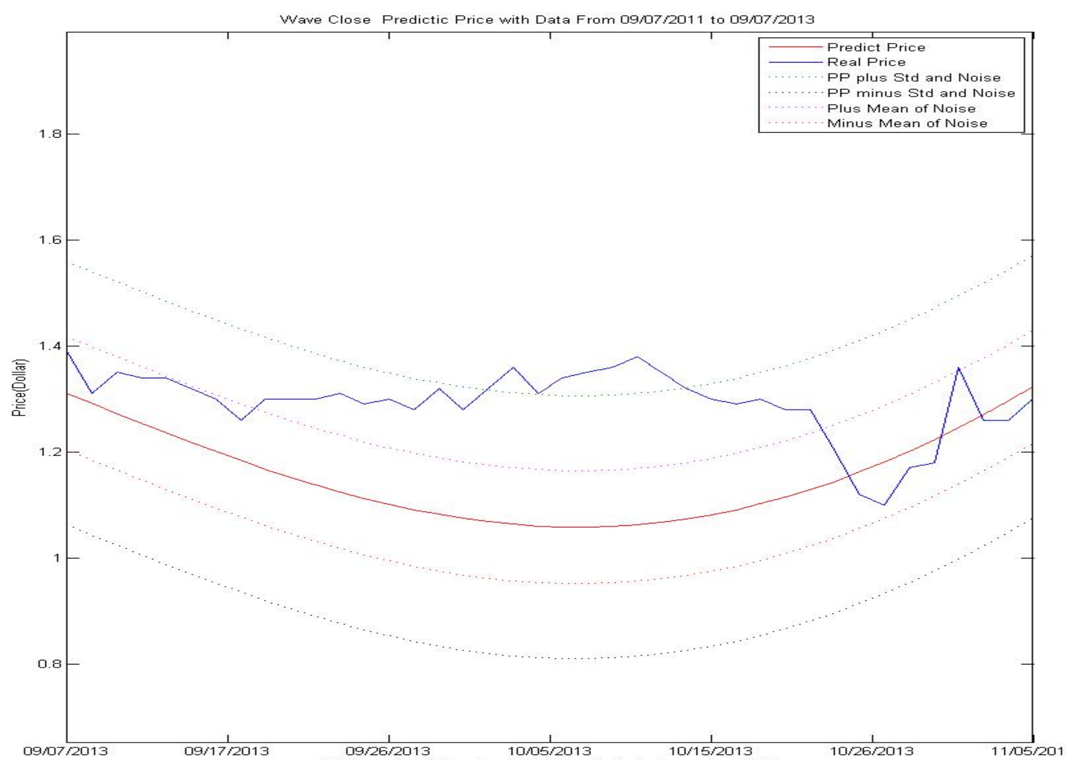


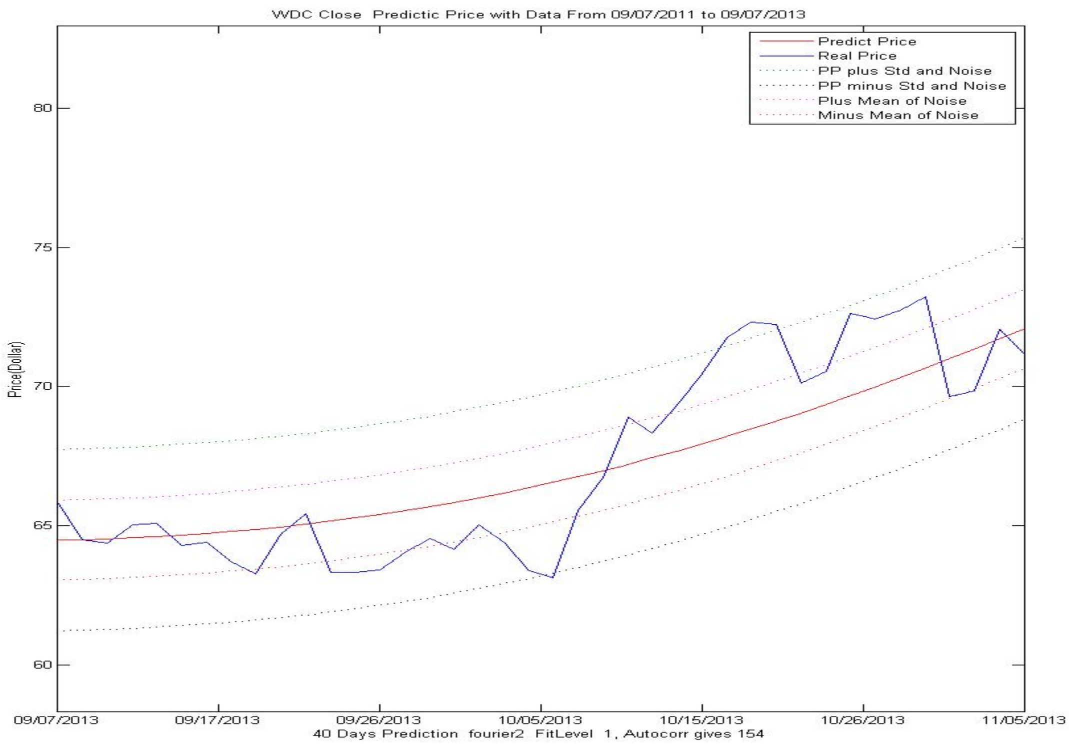
Google Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
1100
1050
1000
950
900
850
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
40 Days Prediction fourier4 FitLevel 1, Autocorr gives 182


Micro Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
40
38
36
34
32
30
28
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
40 Days Prediction fourier2 FitLevel 1, Autocorr gives 64


RCMT Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
7.4
7.2
7
6.8
6.6
6.4
6.2
6
5.8
5.6
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
40 Days Prediction fourier3 FitLevel 3, Autocorr gives 125


VMware Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
105
100
95
90
85
80
75
70
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
40 Days Prediction fourier1 FitLevel 1, Autocorr gives 134


Wave Close Predictic Price with Data From 09/07/2011 to 09/07/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
1.8
1.6
1.4
1.2
1
0.8
09/07/2013
09/17/2013
09/26/2013
10/05/2013
10/15/2013
10/26/2013
11/05/2013
40 Days Prediction fourier1 FitLevel 1, Autocorr gives 136



For 09/07 Prediction:

3 out of 10 stocks: "VMware, Wave, and WDC" The final actual price is within the "Noise+ Std".

4 out of 10 stocks: "Apple, Google, Mella, and Micro" the final actual price is higher than the "Noise+ Std".

2 out of 10 stocks: "Bridge, RCMT" the actual price is lower than the "Noise+ Std".

5 out of 10 stocks: "Apple, Google, Mella, VMware and Micro" The actual price is higher than the predicted price.

3 out of 10 stocks: "Wave, WDC, Bridge and RCMT" The actual price is lower than the predicted price.

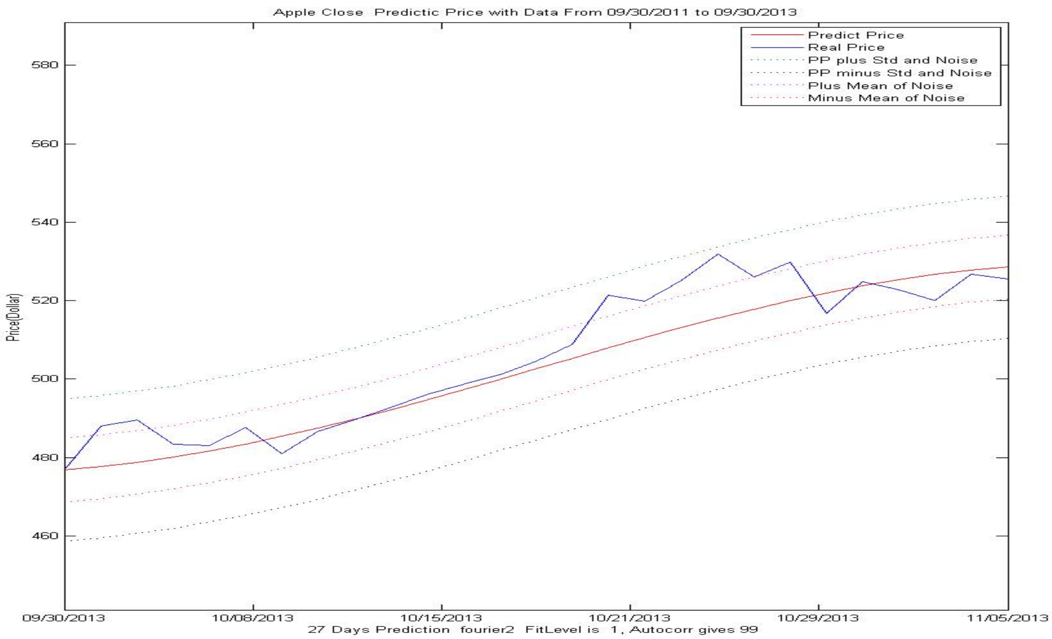
3 stocks "VMware, Wave and WDC" The Predicted price is fairly close to the actual price.

Overall, VMware, Wave and WDC fit well.

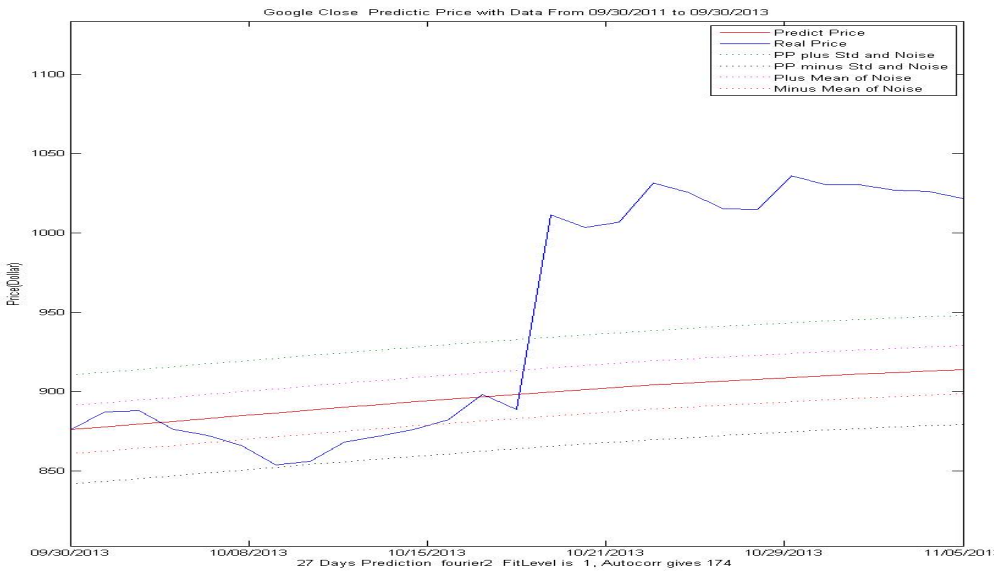
The actual Mella, Micro, Wave stock price is higher than the prediction

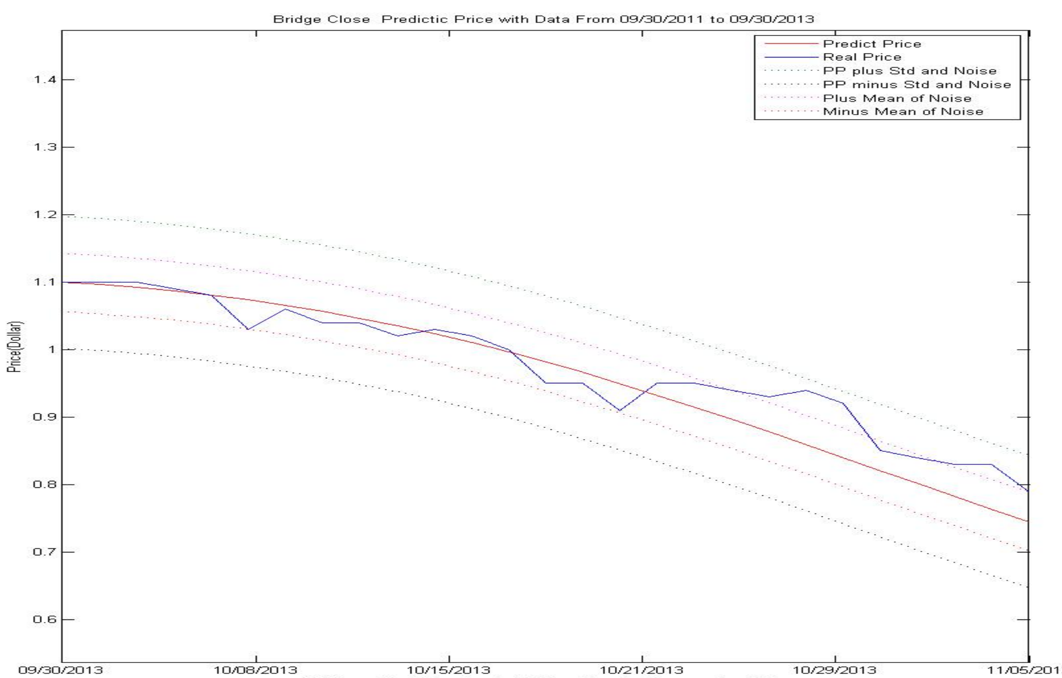
The actual Bridge stock price is lower than the prediction

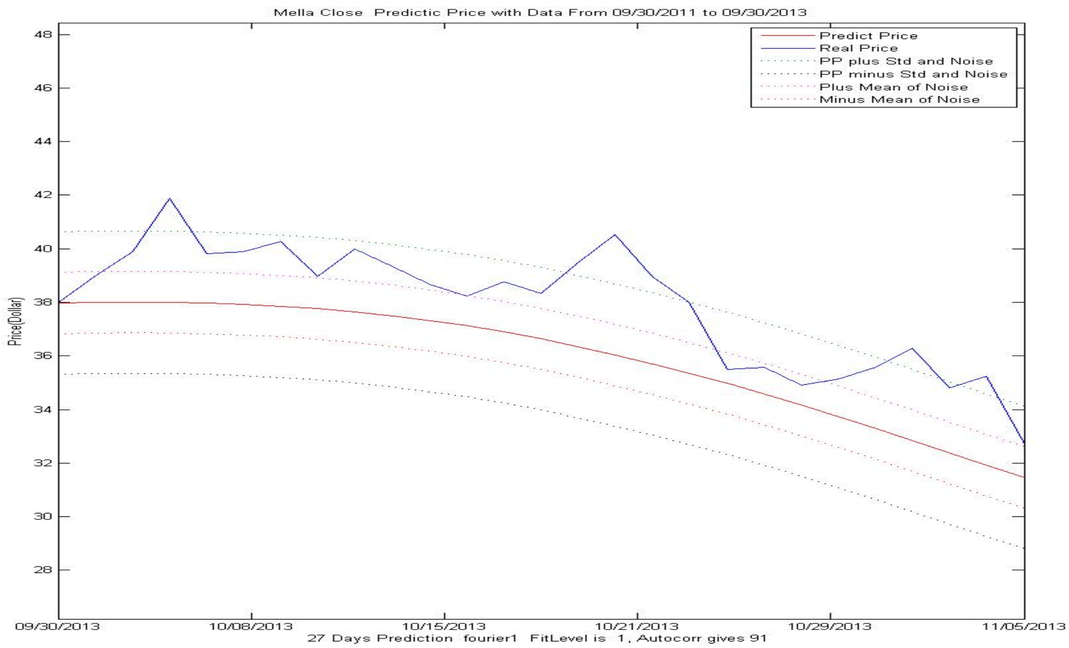
Apple Google RCMT VMware WDC is floating around.



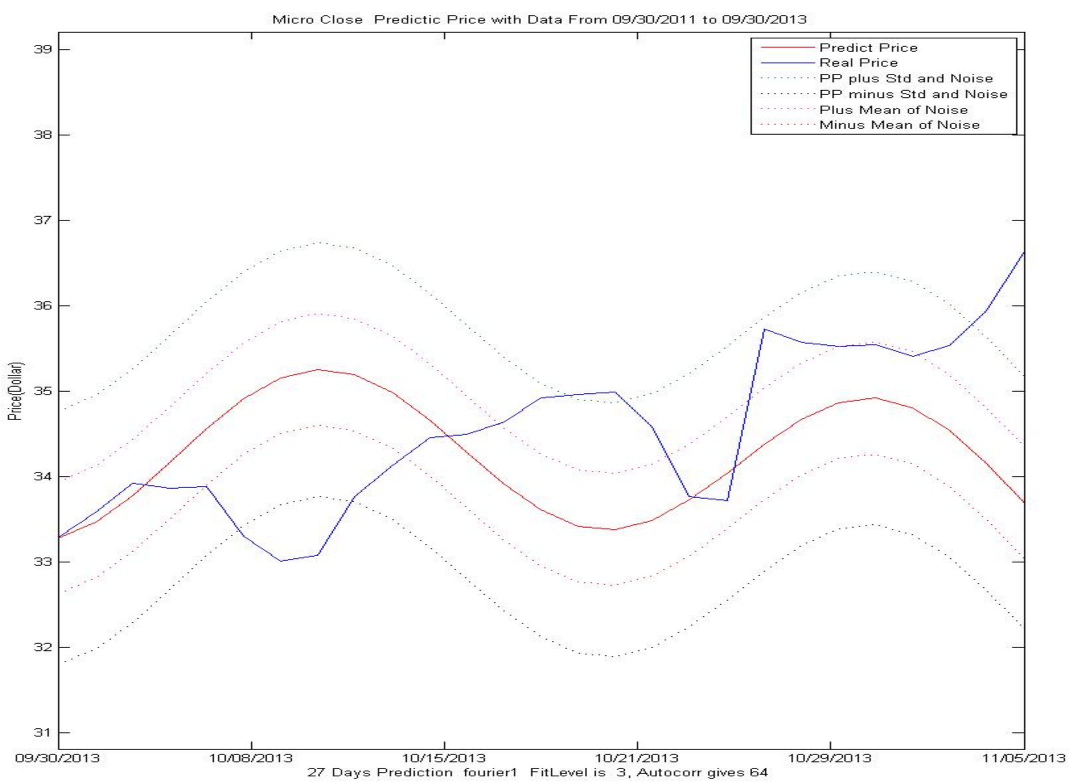
Apple Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
580
560
540
520
500
480
460
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 1, Autocorr gives 99


Google Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
1100
1050
1000
950
900
850
Price(Dollar)
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 1, Autocorr gives 174

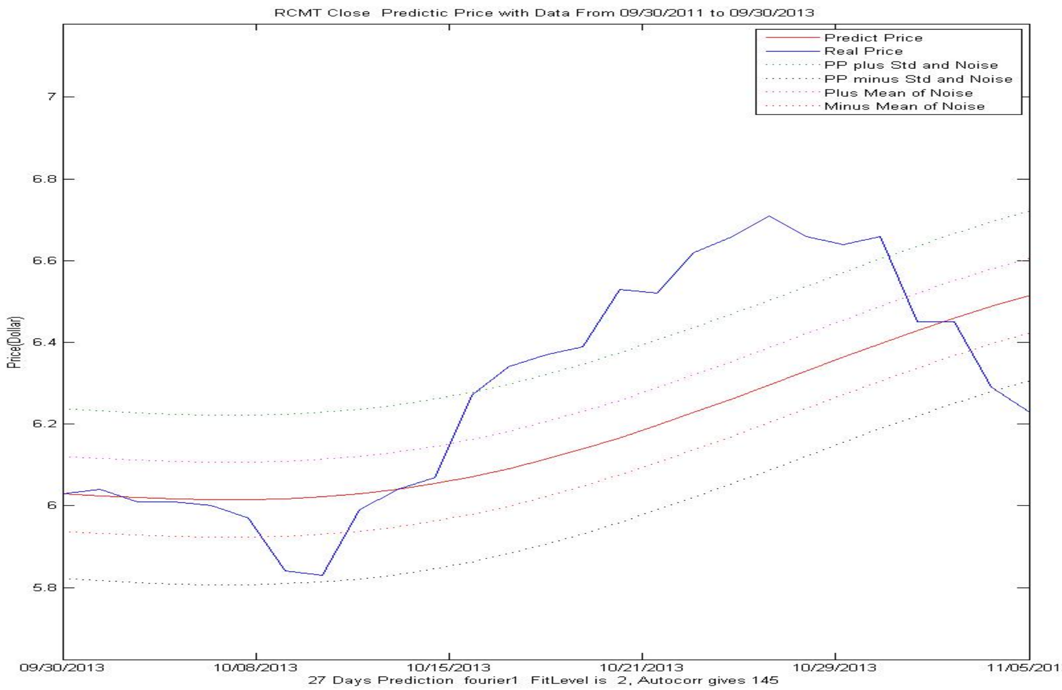

Bridge Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
1.4
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
Price(Dollar)
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier1 FitLevel is 4, Autocorr gives 118

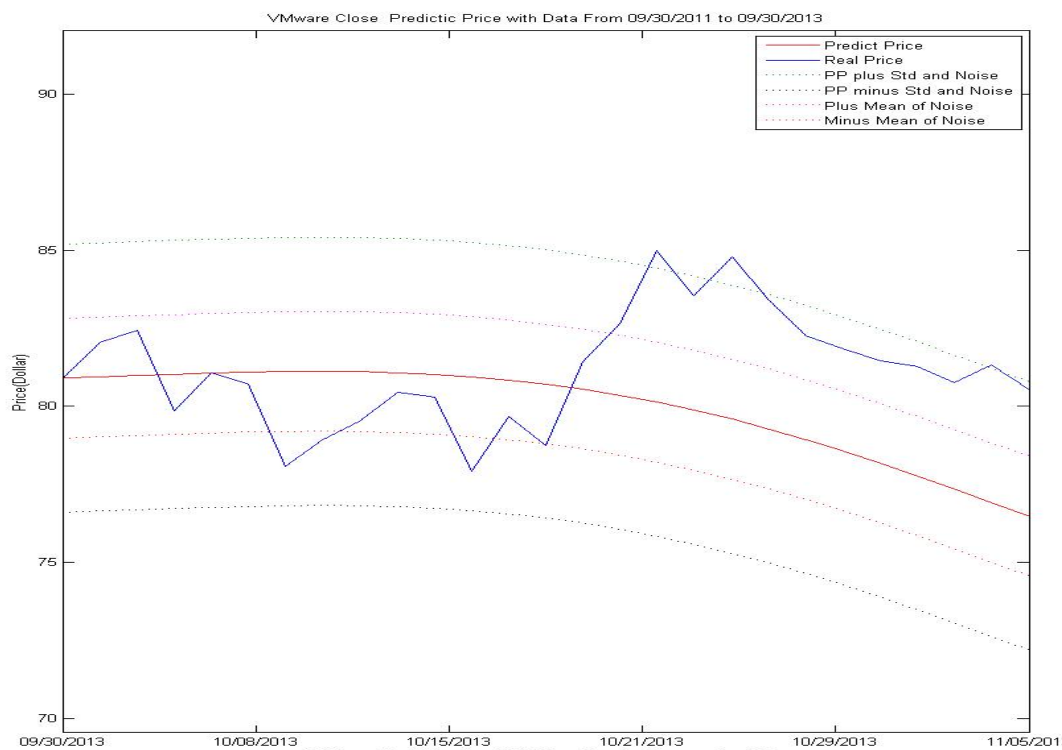


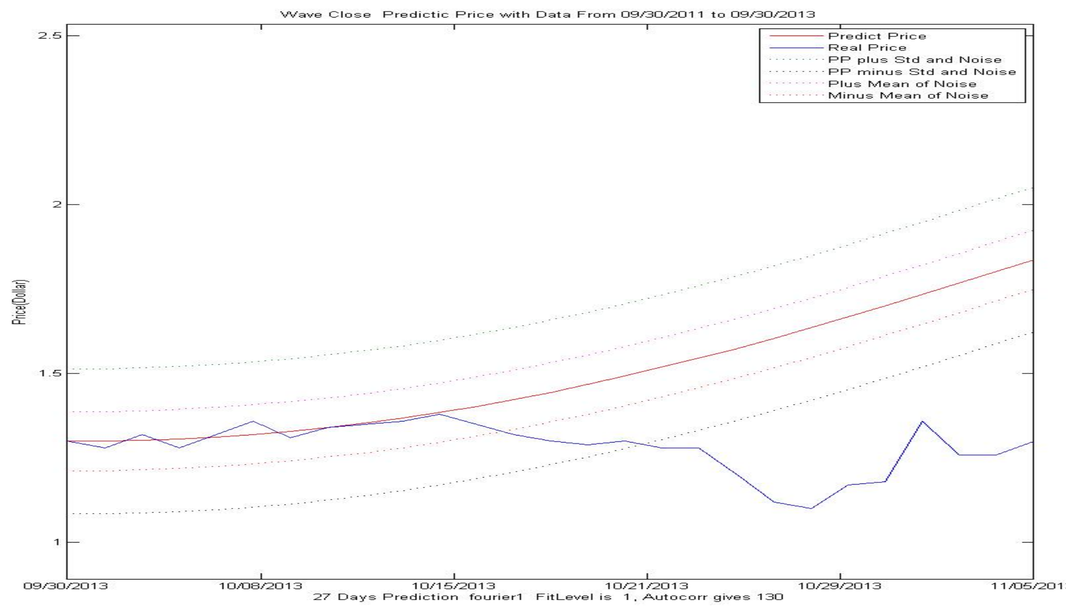
Mella Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
48
46
44
42
40
38
36
34
32
30
28
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier1 FitLevel is 1, Autocorr gives 91

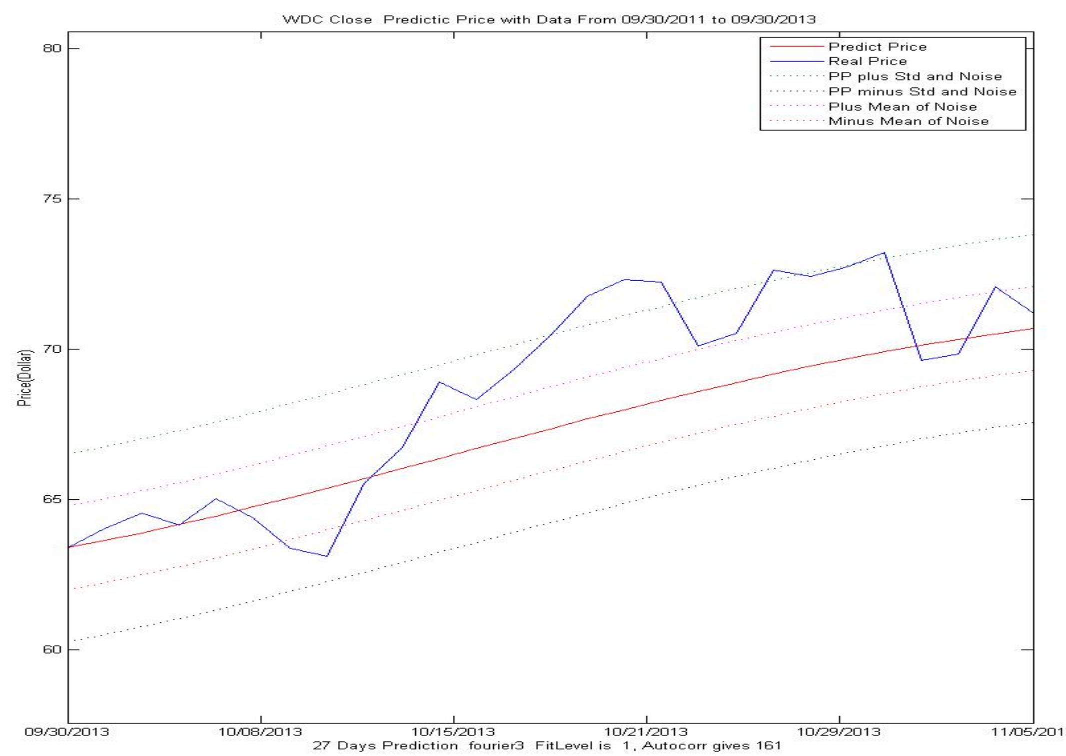


Micro Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
39
38
37
36
35
34
33
32
31
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier1 FitLevel is 3, Autocorr gives 64


RCMT Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
7
6.8
6.6
6.4
6.2
6
5.8
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier1 FitLevel is 2, Autocorr gives 145


VMware Close Predictic Price with Data From 09/30/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
90
85
80
75
70
09/30/2013
10/08/2013
10/15/2013
10/21/2013
10/29/2013
11/05/2013
27 Days Prediction fourier2 FitLevel is 1, Autocorr gives 135





For 09/30 Prediction:

5 out of 10 stocks: “Apple, Bridge, Mella, VMware and WDC” The actual price is within the “Noise+ Std”.

2 out of 10 stocks: “Google, Micro” the final actual price is higher than the “Noise+ Std”.
2 out of 10 stocks: “RCMT, Wave” the final actual price is lower than the “Noise+ Std”.
5 out of 10 stocks: “ALJ, KMR, NWN, WGL, and XOM” The final actual price is higher than the predicted price.
5 out of 10 stocks: “DUK, EPD, HAL, OTTR, and TOT” The final actual price is lower than the predicted price.
2 stocks “HAL, OTTR” The Predicted price is fairly close to the final actual price.
Overall, DUK HAL KMR, OTTR, TOT and XOM fit well.
The actual ALJ TOT XOM stock price is higher than the prediction.
The actual DUK, OTTR stock price is lower than the prediction.
EPD, KMR, NWN, WGL are floating around the prediction.

5.2 Second model -- ARIMA Model

a. Description of ARIMA Model

During B term, we have switched from our model to a more advanced one, which is the arima model. “ARIMA models are, in theory, the most general class of models for forecasting a time series which can be stationarized by transformations such as differencing and logging. In fact, the easiest way to think of ARIMA models is as fine-tuned versions of random-walk and random-trend models: the fine-tuning consists of adding lags of the differenced series and/or lags of the forecast errors to the prediction equation, as needed to remove any last traces of autocorrelation from the forecast errors,”(duck.edu). With professor Humi’s help, we learned and understood what the model is and what it does. The form of the model is Arima(p,d,q), where p is the autoregressive term, d is times of taking the difference and q is the average moving terms. And the outcome of this model is complicated, and also differs from which programs that runs the model.

Because it is a popular and efficient model, there are many applications and programs that can carry this model out. Of course, we could have used this model with

Matlab, but other programs like Tisean and NUMXL that can do a better job and make it easier for us to use the model, thus have better predictions.

1) Tisean

We downloaded the Tisean from http://www.mpipks-dresden.mpg.de/~tisean/. And we used the version 3.0.1. We chose the Unix version and run the program on Mac OS. Here is the command we use.

```
~/bin/arima-model -m2  -P3,1,1  -o AN311m2.outc
```

Here, -m2 means the number of columns of data we want to input. –P3,1,1 means the three inputs that the model requires. The rest is the output name which you can choose whatever you want.

Here is a sample output.

```
#iteration 31 1.659434e-01 5.019685e-01 8.800816e-02
#iteration 32 1.628720e-01 5.031685e-01 8.268400e-02
#iteration 33 2.002682e-01 5.013068e-01 4.342208e-02
#iteration 34 2.429131e-01 5.133799e-01 6.981273e-02
#iteration 35 2.283707e-01 5.047799e-01 1.156633e-01
#iteration 36 1.727743e-01 5.214392e-01 5.104231e-02
#iteration 37 1.779686e-01 4.868546e-01 4.820755e-02
#iteration 38 1.918518e-01 4.211067e-01 2.402596e-02
#iteration 39 1.543657e-01 6.030597e-01 1.190808e-02
#iteration 40 2.023305e-01 6.636383e-01 3.622827e-02
#iteration 41 3.833473e-01 9.051146e-01 6.177648e-02
#iteration 42 2.444026e-01 5.031974e-01 1.798197e-01
#iteration 43 4.489087e-01 7.609636e-01 1.106307e-01
#iteration 44 3.160693e-01 8.049575e-01 2.015966e-01
#iteration 45 2.180041e-01 6.979231e-01 1.830912e-01
#iteration 46 2.589846e-01 1.051439e+00 1.455721e-01
#iteration 47 2.896106e-01 1.113741e+00 2.739458e-01
#iteration 48 3.123517e-01 1.024202e+00 3.602931e-01
#iteration 49 4.010343e-01 1.121178e+00 2.155918e-01
#iteration 50 3.396075e-01 8.680206e-01 1.704056e-01
#average forcast error= 2.237735e+01
#individual forecast errors: 9.854052e+00 3.007307e+01
#Log-Likelihood= -4.273224e+03   AIC= 8.554449e+03
#x_1(n-0) 3.542856e-01 -2.540137e+00
#x_1(n-1) -1.169400e-01 -1.523050e-01
#x_1(n-2) 1.423846e-01 3.970443e-01
#x_2(n-0) 1.938000e-01 1.354296e+00
#x_2(n-1) 3.493044e-02 4.284138e-02
#x_2(n-2) -3.697936e-02 -1.333398e-01
#e_1(n-0) -3.107576e-01 2.502319e+00
#e_2(n-0) -2.110392e-01 -1.382516e+00
```

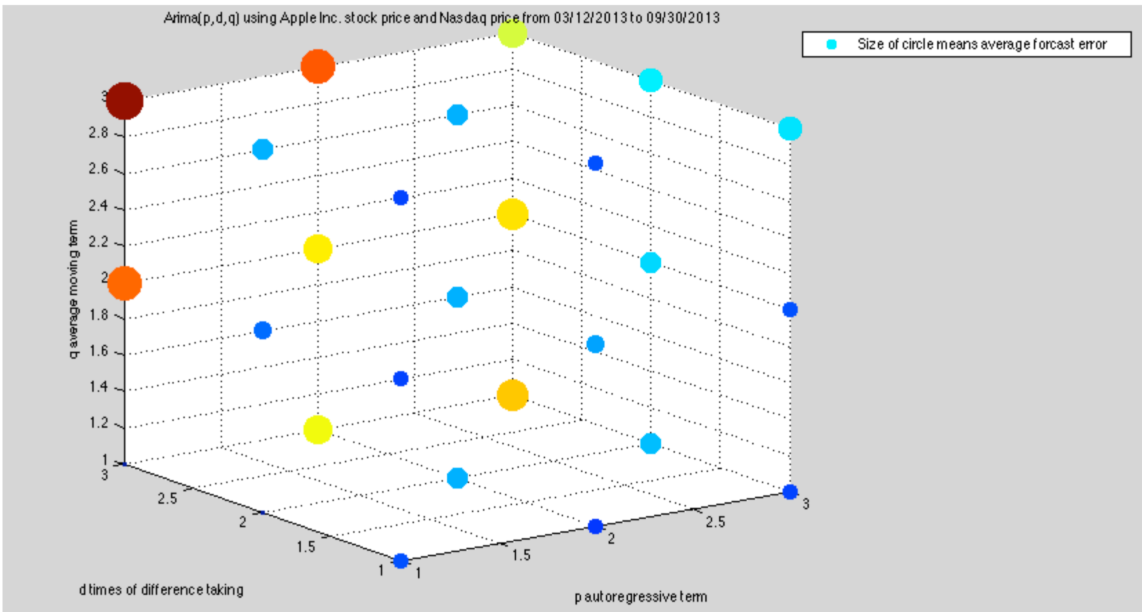
How to find the best prediction

We mainly use the average forecast error to determine how good the prediction is. #x_1(n-0), #x_1(n-1) and #x_1(n-2) are used for prediction using function below.

$$x_n=a_1x_{n-1}+\ldots+a_px_{n-p} + \text{noise}.$$

But the question was that how do we know what kind of input could result in the best prediction. How we defined the best prediction is to get a smallest error in the model. Hence, we came up with a solution to try out all possible inputs. So we tried from arima(1,1,1), arima(1,1,2), arima(1,1,3) to arima(3,3,3). In another word, we tried out q,d and q from 1 to 3 for all the combination. And we collected all the average forecast errors and made a 3D chart using Matlab.

For this chart, we used Apple Inc. stock price and Nasdaq price from 03/12/2013 to 09/30/2013.

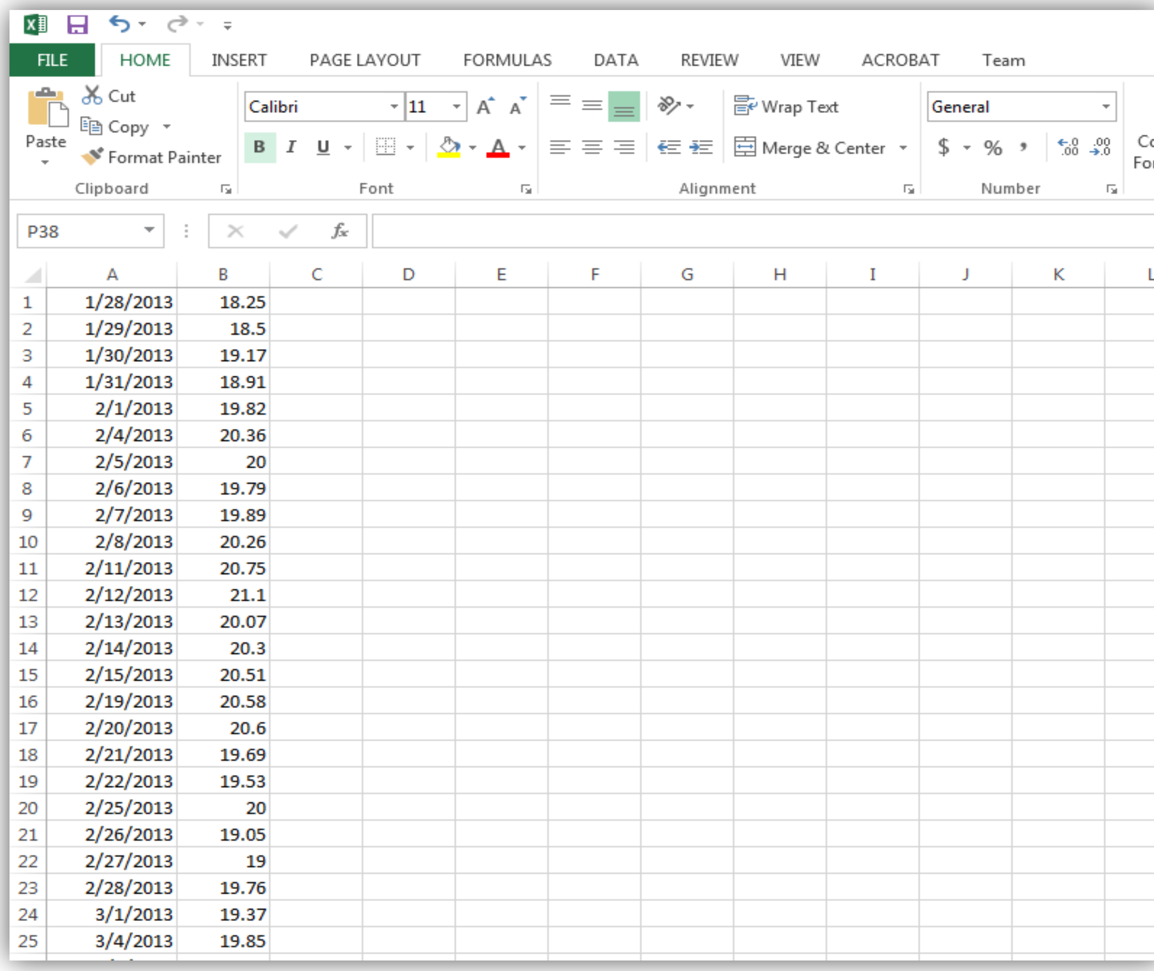


From this chart, we can conclude that arima(1,1,3) had the best result. We can use this technique to make new and better predictions.

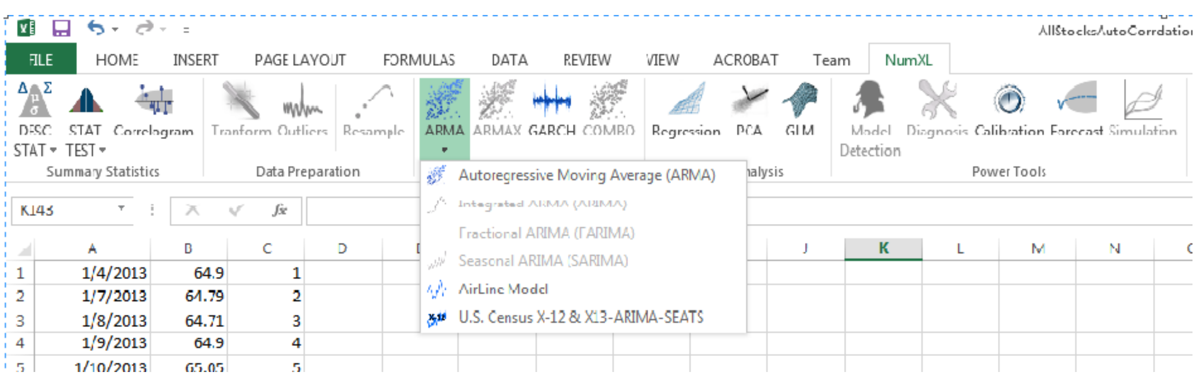
2) NumXL

We have already gotten to know the ARIMA model itself. Tisean model, however, could cause problems to people who are not familiar with the terminal in Linux or Mac operating system. We found a more efficient program, NumXL, to complete our model. NumXL is an add-on to excel. We could download the NumXL from http://www.spiderfinancial.com/products/numxl.

First of all we should input the historical data we got earlier in A term into excel from oldest to newest, as the shown below.



Then, we would like to use the NumXL ARIMA to deal with all historical data.



While dealing with the historical data, we should also find out the best inputs for ARIMA to give us the smallest error. Therefore we tried all points from ARIMA(1,1) to

ARIMA(3,3). In NumXL, sigma (σ) is our error in the model. So after tried out all 9 inputs, we could choose the one with smallest sigma to do our prediction.

ARMA(1,1)

Param	Value
ϕ_0	69.49
ϕ_1	1.00
θ_1	0.92
σ	1.51

Goodness-of-fit

LLF	AIC	CHECK
-263.82	535.65	1.

Residuals (standardized) Analysis

	AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	0.00	0.74	-0.15	-0.08	FALSE	TRUE	TRUE
Target	0.00	1.00	0.00	0.00			
SIG?	FALSE	TRUE	FALSE	FALSE			

ARMA(1,2)

Param	Value
ϕ_0	69.49
ϕ_1	0.92
θ_1	0.70
θ_2	0.39
σ	-1.58

Goodness-of-fit

LLF	AIC	CHECK
#NUM!	#NUM!	1.

Residuals (standardized) Analysis

	AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	#NUM!	#NUM!	#NUM!	#NUM!	#NUM!	#NUM!	#NUM!
Target	0.00	1.00	0.00	0.00			
SIG?	#NUM!	#NUM!	#NUM!	#NUM!			

ARMA(1,3)

Param	Value
ϕ_0	69.49
ϕ_1	0.98
θ_1	0.13
θ_2	0.04
θ_3	0.16
σ	0.56

Goodness-of-fit

LLF	AIC	CHECK
-136.67	285.35	1.

Residuals (standardized) Analysis

	AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	0.00	1.00	-0.63	0.88	TRUE	FALSE	FALSE
Target	0.00	1.00	0.00	0.00			
SIG?	FALSE	FALSE	TRUE	TRUE			

ARMA(2,1)

Param	Value
ϕ_0	69.49
ϕ_1	0.51
ϕ_2	0.49
θ_1	0.48
σ	2.05

Goodness-of-fit

LLF	AIC	CHECK
-275.09	560.18	1.

Residuals (standardized) Analysis

	AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	0.00	0.28	-0.70	1.00	FALSE	FALSE	FALSE
Target	0.00	1.00	0.00	0.00			
SIG?	FALSE	TRUE	TRUE	TRUE			

ARMA(2,2)

Param	Value
ϕ_0	69.49
ϕ_1	0.51
ϕ_2	0.49
θ_1	0.48
θ_2	0.46
σ	1.80

Goodness-of-fit

LLF	AIC	CHECK
-257.06	526.12	1.

Residuals (standardized) Analysis

	AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	0.00	0.36	-0.51	1.01	FALSE	FALSE	FALSE
Target	0.00	1.00	0.00	0.00			
SIG?	FALSE	TRUE	TRUE	TRUE			

ARMA(2,3)

Param	Value
ϕ_0	69.49
ϕ_1	0.51
ϕ_2	0.49
θ_1	0.49
θ_2	0.47
θ_3	0.45
σ	1.57

Goodness-of-fit

LLF	AIC	CHECK
-236.74	487.47	1.

Residuals (standardized) Analysis

	AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	0.00	0.40	-0.24	0.12	FALSE	TRUE	TRUE
Target	0.00	1.00	0.00	0.00			
SIG?	FALSE	TRUE	FALSE	FALSE			

ARMA(3,1)

Param	Value
ϕ_0	69.49
ϕ_1	0.35
ϕ_2	0.33
ϕ_3	0.32
θ_1	0.29
σ	1.80

Goodness-of-fit

LLF	AIC	CHECK
-257.66	527.32	1.

Residuals (standardized) Analysis

	AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	0.00	0.35	-0.61	0.95	FALSE	FALSE	TRUE
Target	0.00	1.00	0.00	0.00			
SIG?	FALSE	TRUE	TRUE	TRUE			

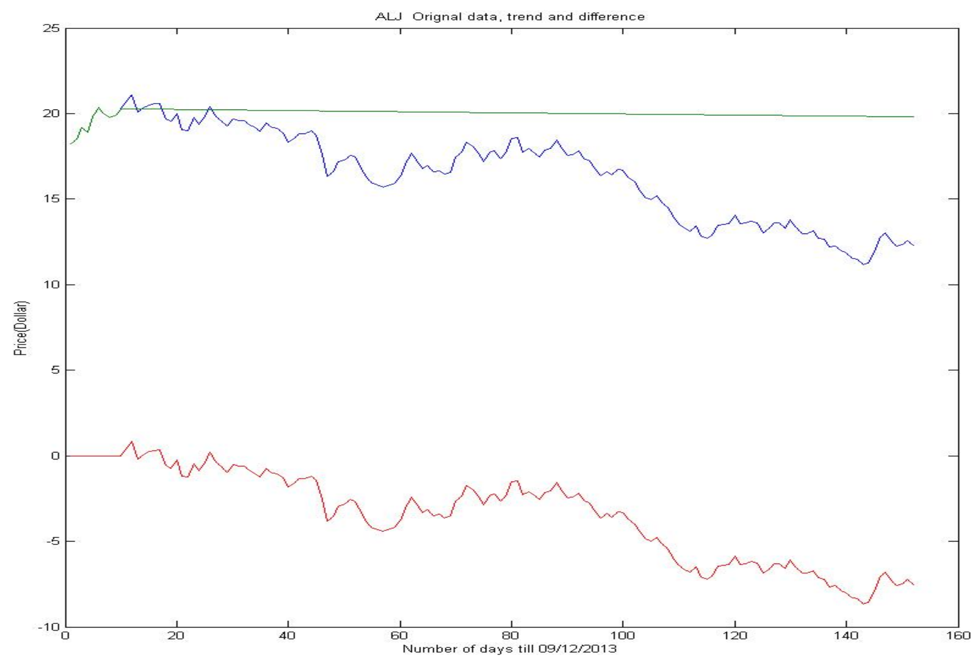
ARMA(3,2)				Goodness-of-fit					Residuals (standardized) Analysis						
	Param	Value		LLF	AIC	CHECK			AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	ϕ_0	69.49		-233.76	481.53	1.			0.00	0.40	-0.53	1.05	FALSE	FALSE	TRUE
	ϕ_1	0.35						Target	0.00	1.00	0.00	0.00			
	ϕ_2	0.33						SIG?	FALSE	TRUE	TRUE	TRUE			
	ϕ_3	0.32													
	θ_1	0.29													
	θ_2	0.28													
	σ	1.54													

ARMA(3,3)				Goodness-of-fit					Residuals (standardized) Analysis						
	Param	Value		LLF	AIC	CHECK			AVG	STDEV	SKEW	KURTOSIS	Noise?	Normal?	ARCH?
	ϕ_0	69.49		-217.03	450.05	1.			0.00	0.46	-0.30	0.75	FALSE	TRUE	TRUE
	ϕ_1	0.35						Target	0.00	1.00	0.00	0.00			
	ϕ_2	0.33						SIG?	FALSE	TRUE	FALSE	FALSE			
	ϕ_3	0.32													
	θ_1	0.30													
	θ_2	0.28													
	θ_3	0.26													
	σ	1.35													

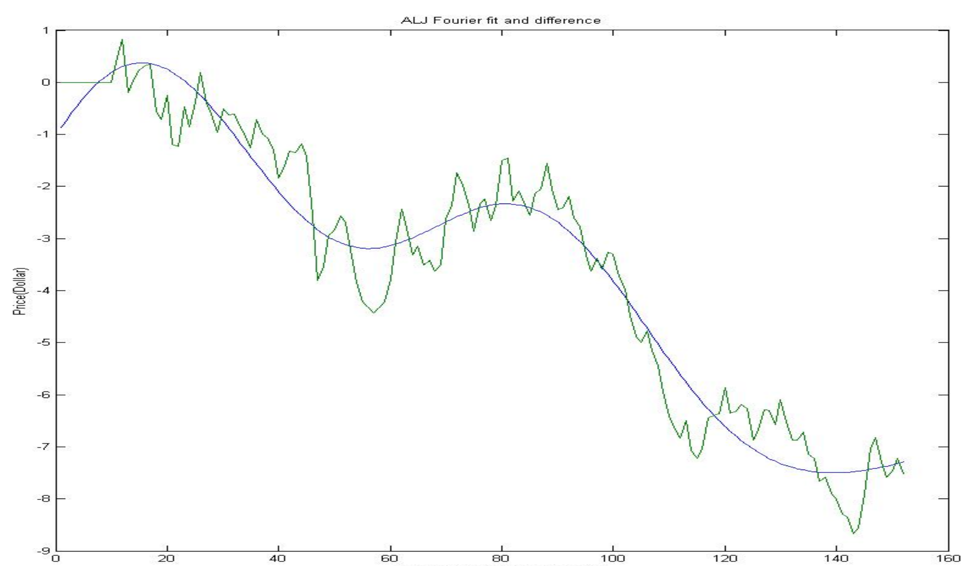
In this example, we can tell that ARMA(1,3) give us smallest sigma. Thus we choose this one to develop our prediction. We could use the Forecast part in NumXL to get the trend. Since 10 initial data would work perfectly to our prediction, we tried to give ARIMA first 10 days' prices of the stock and predict the remaining prices of the dates in the autocorrelation. What we got below is just the prediction, so when we put the our predicted data to Matlab, we should combine the first 10 real prices.

Step	Mean	STD	UL	LL
1	66.00799	0.562954	67.11136	64.90462
2	66.12914	0.843399	67.78217	64.47611
3	66.18556	1.058059	68.25931	64.1118
4	66.2391	1.277837	68.74361	63.73458
5	66.29177	1.45936	69.15207	63.43148
6	66.34359	1.615738	69.51038	63.17681
7	66.39458	1.753858	69.83207	62.95708
8	66.44473	1.877885	70.12532	62.76414
9	66.49407	1.990577	70.39553	62.59261
10	66.54261	2.093876	70.64653	62.43869
11	66.59036	2.189215	70.88114	62.29958
12	66.63734	2.27769	71.10153	62.17315
13	66.68356	2.360162	71.30939	62.05773
14	66.72902	2.437325	71.50609	61.95196
15	66.77375	2.509747	71.69277	61.85474
16	66.81776	2.577903	71.87036	61.76516
17	66.86105	2.642192	72.03965	61.68245
18	66.90364	2.702958	72.20134	61.60594
19	66.94554	2.760495	72.35601	61.53507
20	66.98675	2.815061	72.50417	61.46934
21	67.0273	2.866883	72.64629	61.40832
22	67.0672	2.916161	72.78277	61.35163
23	67.10644	2.963074	72.91396	61.29893
24	67.14505	3.00778	73.04019	61.24991
25	67.18303	3.050424	73.16176	61.20431
26	67.2204	3.091137	73.27892	61.16189
27	67.25716	3.130035	73.39192	61.12241
28	67.29333	3.167227	73.50098	61.08568
29	67.32891	3.202811	73.6063	61.05151
30	67.36391	3.236878	73.70807	61.01974
31	67.39834	3.269511	73.80646	60.99022
32	67.43222	3.300787	73.90164	60.96279
33	67.46554	3.330777	73.99374	60.93734
34	67.49833	3.359547	74.08292	60.91374
35	67.53058	3.387159	74.16929	60.89187
36	67.56231	3.41367	74.25298	60.87164

Furthermore, we could use the Matlab to plot the trend like we did in first model and then apply Fourier expansion to the difference.

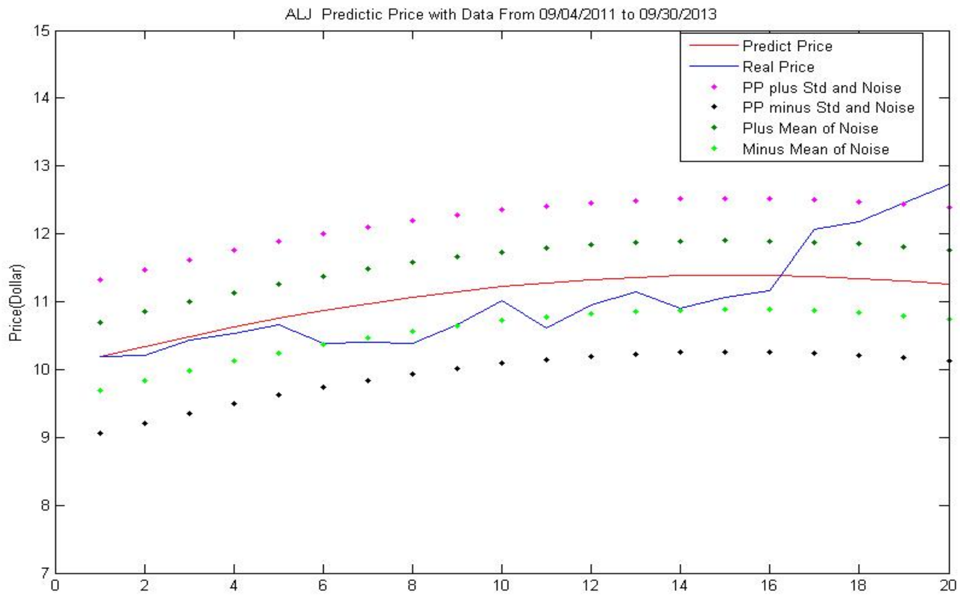


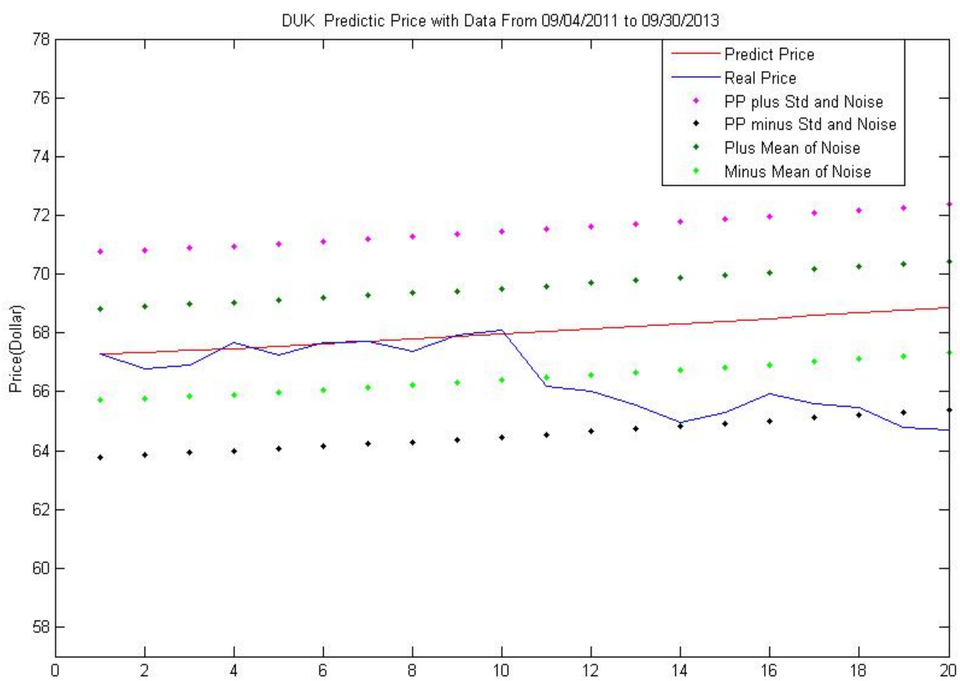
In this graph, the green line stands for our prediction, blue line stands for the real price of the stock and red line is the difference. Then we apply the Fourier expansion to the difference.

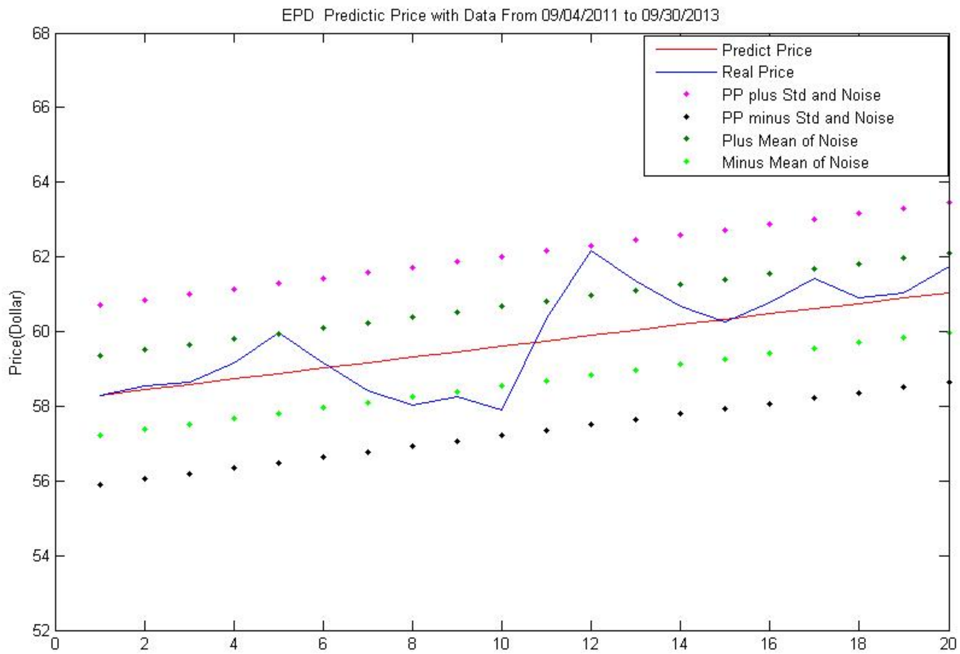


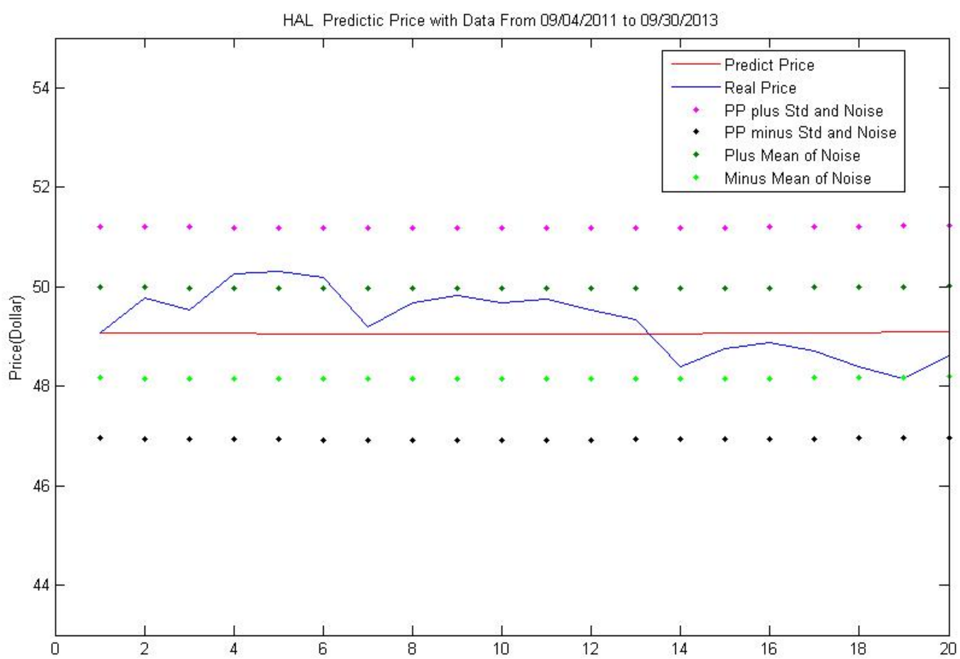
Combining the trend and Fourier expansion, we can get our prediction from date Sep 5th, 2013.

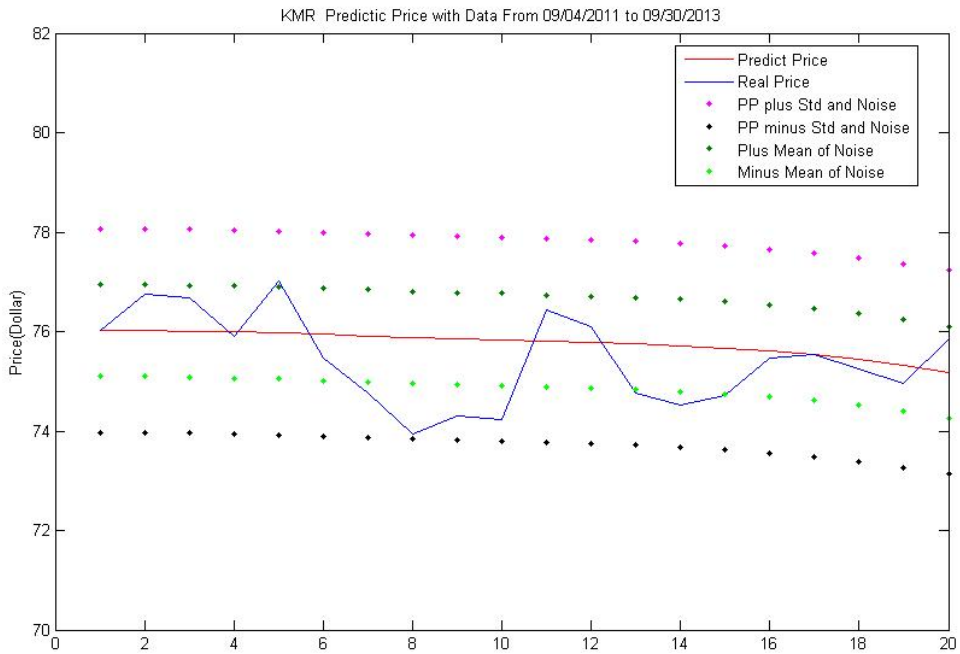
Here are predictions with ARIMA model for10 stocks in the energy field.

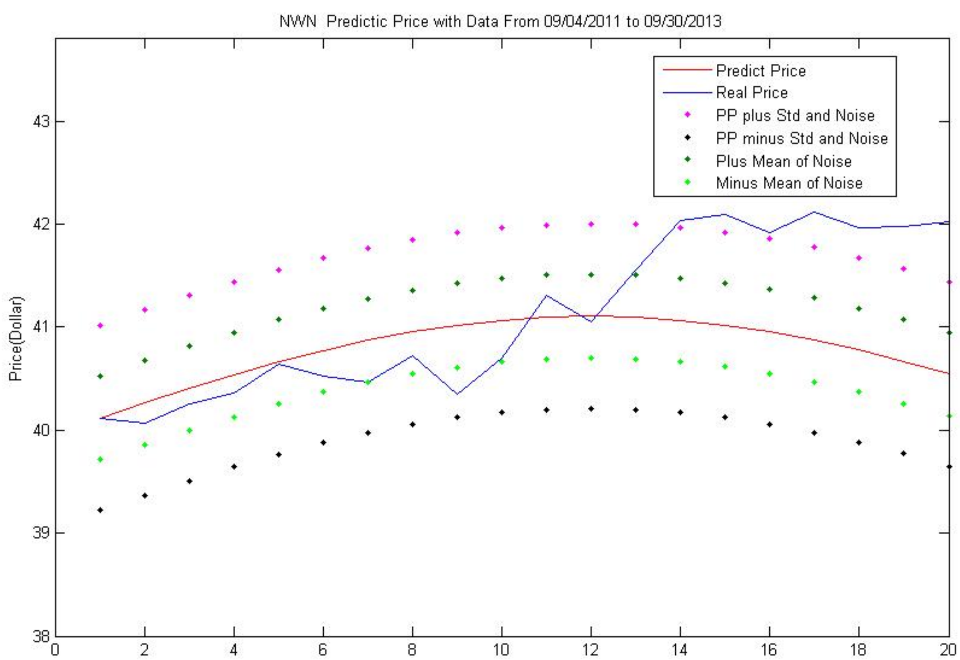


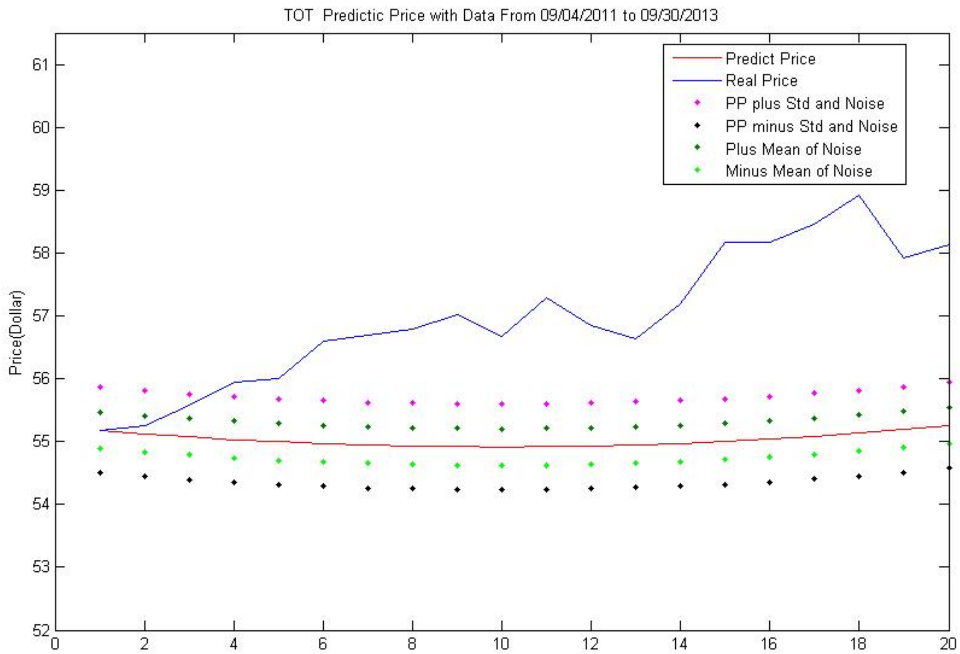


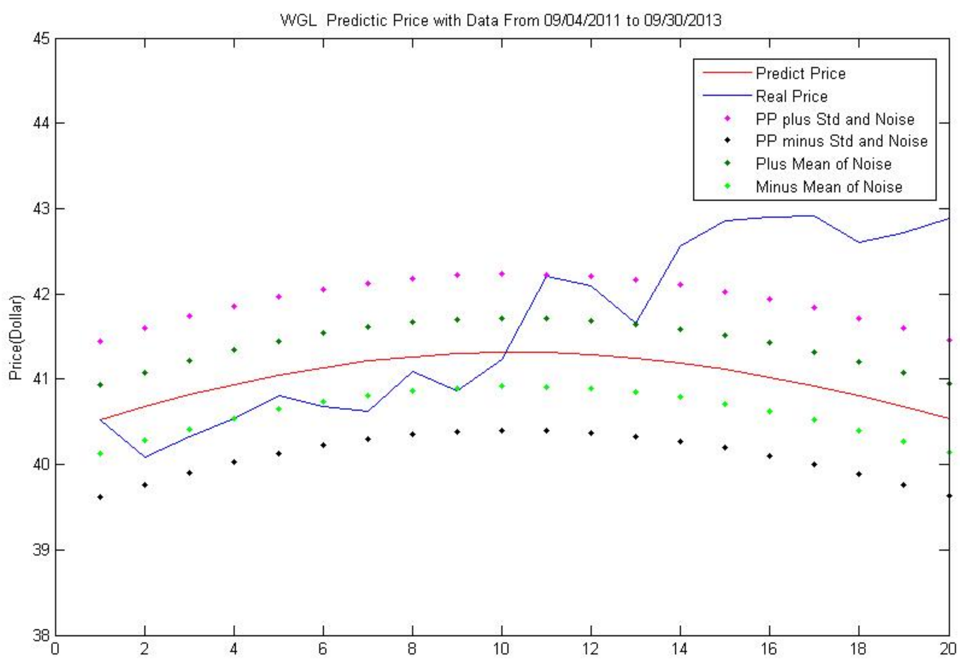

EPD Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)

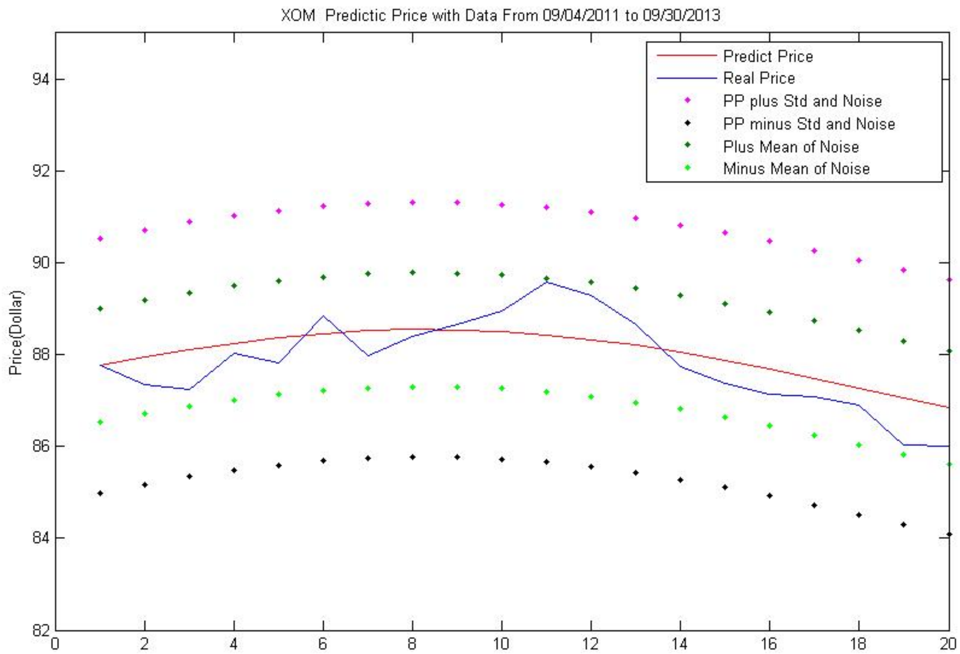

HAL Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)


KMR Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
82
80
78
76
74
72
70
0
2
4
6
8
10
12
14
16
18
20


NWN Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
43
42
41
40
39
38
0
2
4
6
8
10
12
14
16
18
20


TOT Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
61
60
59
58
57
56
55
54
53
52
0
2
4
6
8
10
12
14
16
18
20


WGL Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price(Dollar)
45
44
43
42
41
40
39
38
0
2
4
6
8
10
12
14
16
18
20



As we can see from these 20 days' predictions,

4 out of 10 stocks: " EPD, HAL, KMR, and XOM" The actual price is within the "Noise+ Std".

4 out of 10 stocks: "ALJ, NWN, WGL, and TOT" the final actual price is higher than the "Noise+ Std".

1 out of 10 stocks: "DUK" the final actual price is lower than the "Noise+ Std".

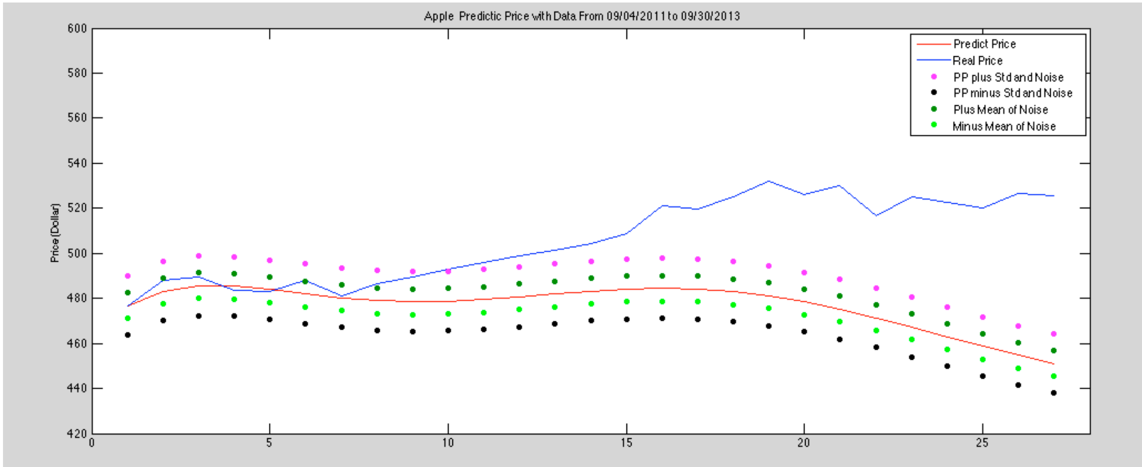
6 out of 10 stocks: "ALJ, EPD, KMR, NWN, TOT and WGL" The final actual price is higher than the predicted price.

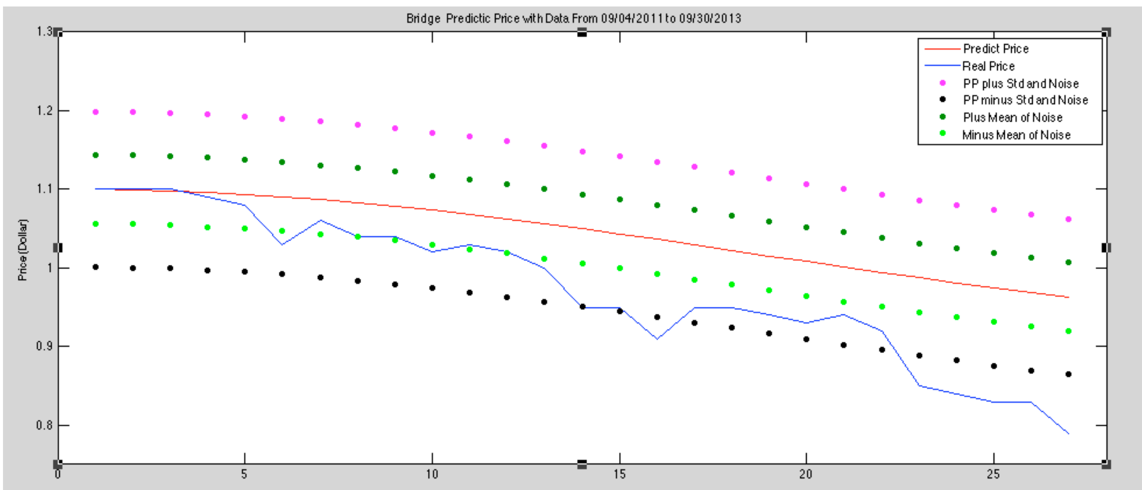
4 out of 10 stocks: "DUK, HAL, OTTR, and XOM" The final actual price is lower than the predicted price.

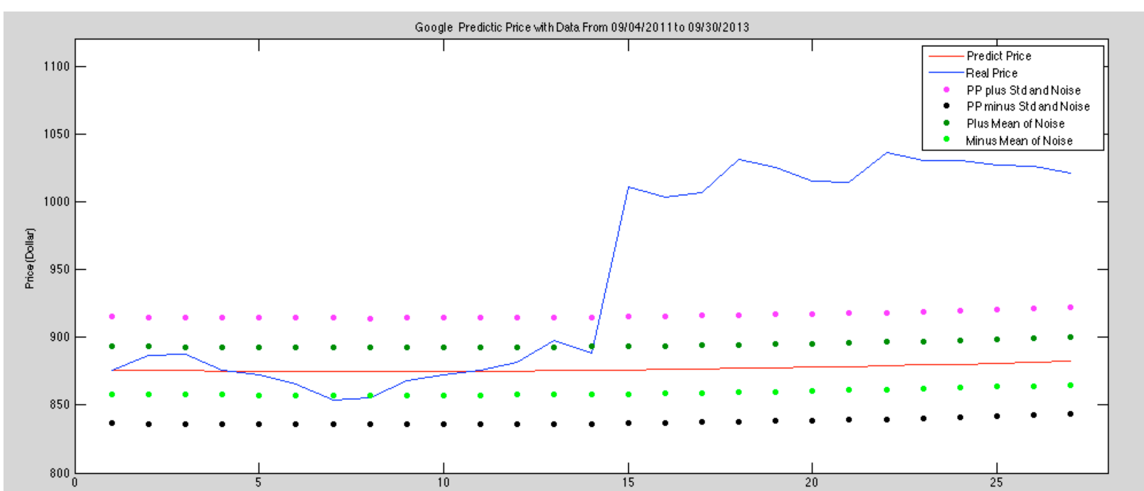
2 stocks "EPD and KMR" The Predicted price is fairly close to the final actual price.

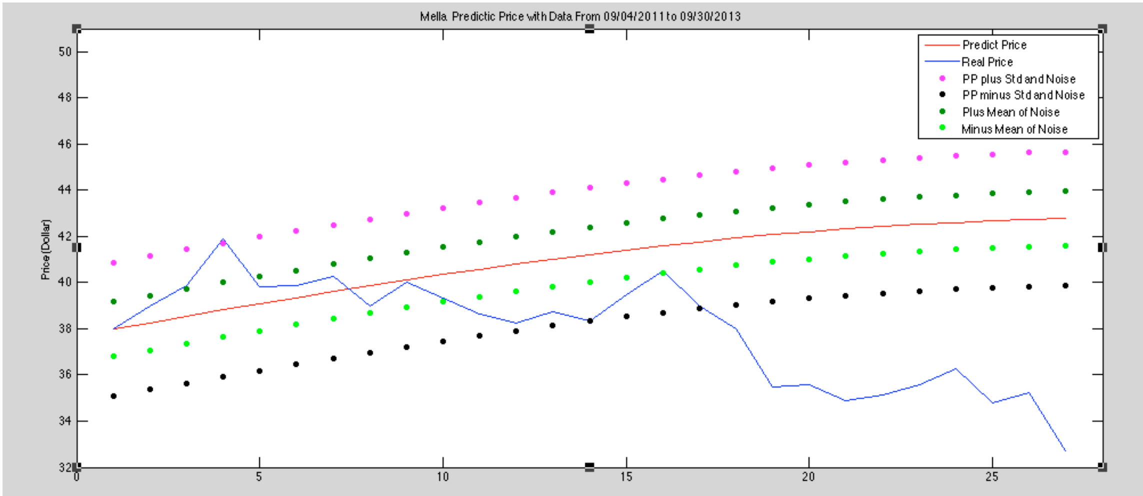
Overall, ALJ, EPD, HAL NWN, WGL and XOM fit well.

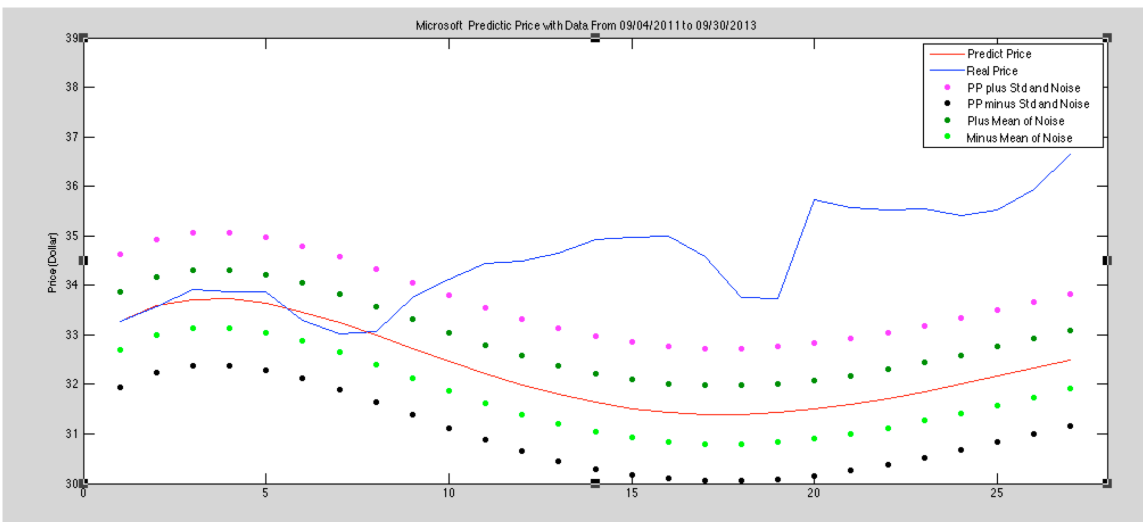
Here are predictions with ARIMA model for10 stocks in the technology field.

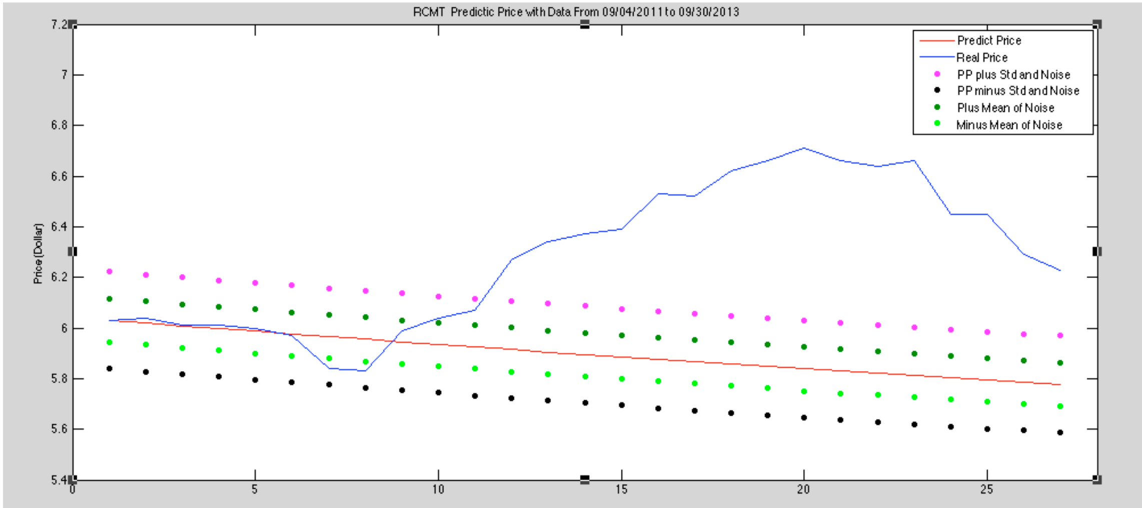

Apple Predictic Price with Data From 09/04/2011 to 09/30/2013
Price (Dollar)
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise

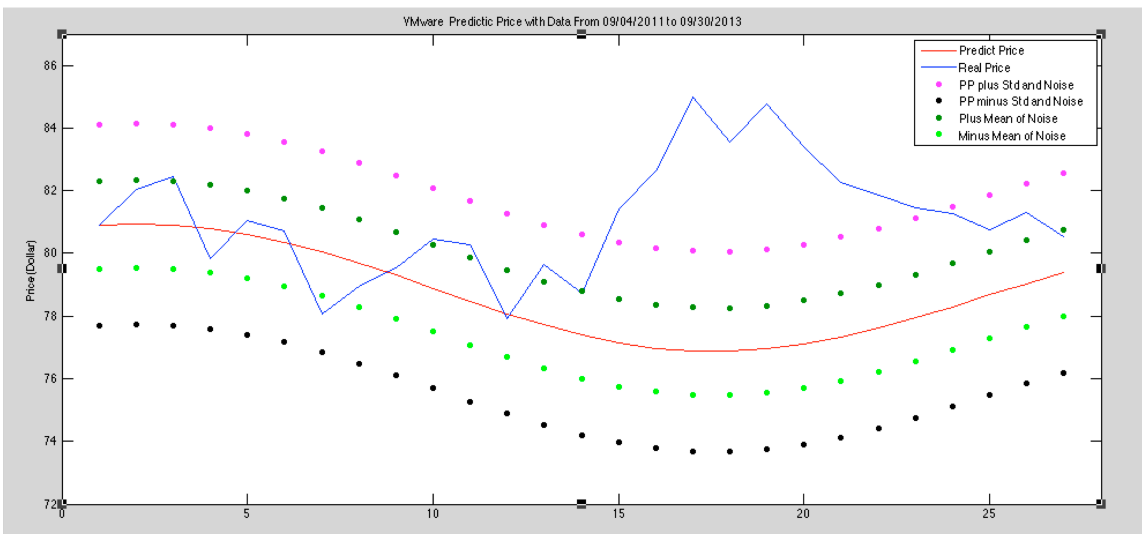

Bridge Predictic Price with Data From 09/04/2011 to 09/30/2013
Price (Dollar)
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise

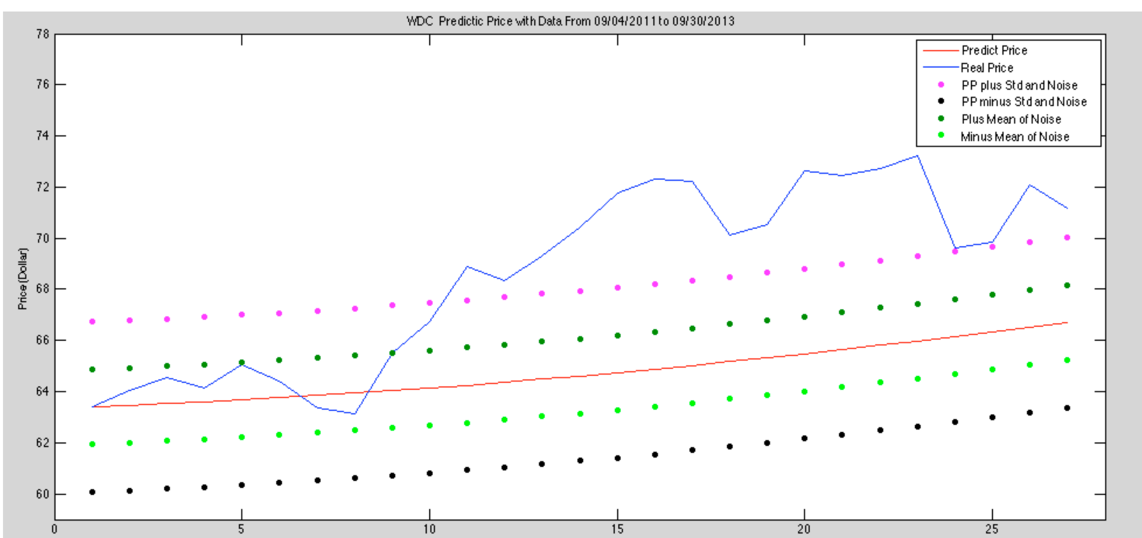

Google Predictic Price with Data From 09/04/2011 to 09/30/2013
Price (Dollar)
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise

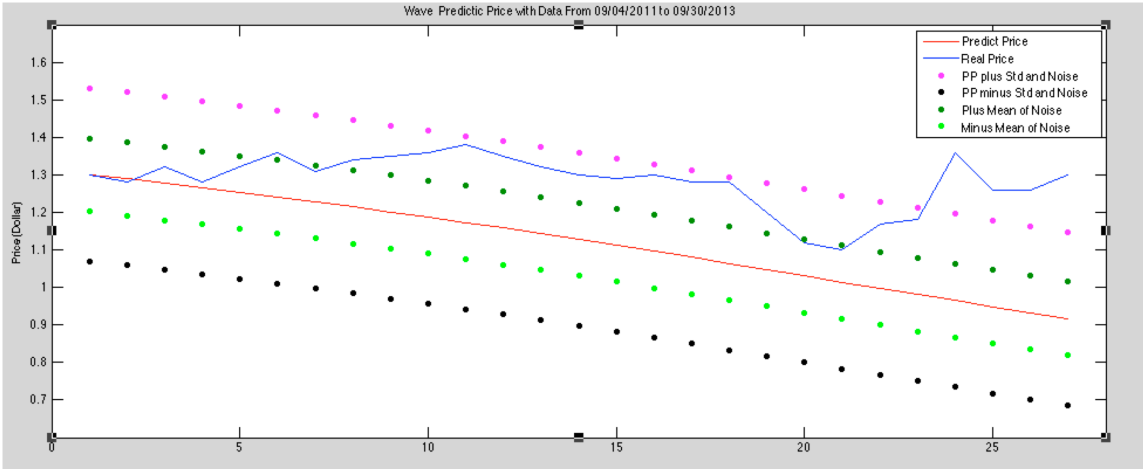

Mella Predictic Price with Data From 09/04/2011 to 09/30/2013
Price (Dollar)
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise


Microsoft Predictic Price with Data From 09/04/2011 to 09/30/2013
Price (Dollar)
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise


RCMT Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price (Dollar)


VMware Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price (Dollar)


WDC Predictic Price with Data From 09/04/2011 to 09/30/2013
Predict Price
Real Price
PP plus Std and Noise
PP minus Std and Noise
Plus Mean of Noise
Minus Mean of Noise
Price (Dollar)



As we can see from these 27 days' predictions,

2 out of 10 stocks: the actual price is within the "Noise+ Std".

6 out of 10 stocks: the final actual price is higher than the "Noise+ Std".

2 out of 10 stocks: the final actual price is lower than the "Noise+ Std".

7 out of 10 stocks: the final actual price is higher than the predicted price.

3 out of 10 stocks: the final actual price is lower than the predicted price.

7 out of 10 stocks: the final actual price is higher than the predicted price.

Overall, only two predictions fit well.

6. Problem and Solution

First model -- Trend- Fourier expansion model

As we can see in the result of the first model, some of the predictions are not good enough, especially for the technology field. Then we should find some other determination which influences the price of the stocks in order to get a better and closer prediction. Now, we find three main considerations which can be taking into account to improve our model: Interest Rates, Unemployment and GDP.

It is said that the increasing of interest rates would hurt the stock market. Since increasing of interest rates would make more investors try a safer means to save their instead of put their income in risk investment. Though after doing the research we cannot find out a doubtless reason how interest rates influence stock market, there must be a relation between them. Thus, we could try to get the data for interest rates daily and also use the correlation between interest rates and stocks to calculate the coefficient.

Unemployment is also a factor could determine the price of stocks. High unemployment rates would make investors try to put their money safely which lead to the less investment in the stock market. Additionally, we can easily find out the predicted sign for the correlation coefficient between unemployment and stock prices is negative.

Although GDP update every three months, it is an essential determination that impact our prediction. Total income which eared domestically is measured by GDP. Therefore, the difference of GDP from one period to the next would make a big difference. A rise of GDP would suggest that the performance of firms as a whole is giving a positive sign to the stock market and a decrease of GDP would suggest that the performance of companies provides a negative sign to the stock market.

Furthermore, after taking the advice from professor Humi, we get to know the Aruoba-Diebold-Scotti Business Conditions Index (ADS) which contains all the determinations we mentioned above. These are the useful indices we should add to our model next term.

Another idea would be the dynamic model.

Try to involve as many factors as possible for the basic model. Then do ten days prediction for each period. Adjust coefficients for each factor or even delete some irrelevant factors, because there are factors that may change the predicted price in a totally opposite direction. Instead of adding factors, we refine the model by reducing factors, since it is hard to add one factor once a time and figuring out whether to delete this factor later.

Second model -- ARIMA Model

In the second model, we can clearly see that the fourier expansion lead the main trend for the prediction. Since the Arima model only fluctuates during the first 10 or less prediction and stays at the same number after that, the fourier expansion at the end of the prediction could make big difference. Thus, if our level of fourier expansion direct a wrong way, our prediction would deviate from our real expectation.

One possible solution would be making the prediction period shorter.

As we can see from the result, the stocks with shorter autocorrelation would give us better prediction than those with longer autocorrelation. So when the period gets shorter, the trend for AMRIMA forecast can be more distinct. In this way, the Arima model can be a solid combination of two function and a better prediction may occur.

7. Conclusions

Our main goal for this Interactive Qualifying Project is to give investors who owned only limited knowledge to the stock market a good guide. Though some of the predictions are not working out that good with our model, for most stocks we can give investors a lead. This would make their investments easier and try to gain money not just by chance. These models are developed by us from very basic ones to complex ones with a high reliability. At the end of this IQP, we got the predictions that seemed to be pretty pleasing, but meanwhile there are a lot to be done to improve the results. We would recommend future IQP students concern more on further developing the first model by adding Aruoba-Diebold-Scotti Business Conditions Index to especially the first model.

Reference

Petruccelli, Joseph D., Balgobin Nandram, and Minghui Chen. Applied Statistics for Engineers and Scientists. Upper Saddle River, NJ: Prentice Hall, 1999. Print.

"Documentation Center." *Polynomial Curve Fitting*. N.p., n.d. Web. 17 Oct. 2013.

"Least Squares." *Wikipedia*. Wikimedia Foundation, 13 Oct. 2013. Web. 17 Oct. 2013.

"Aruoba-Diebold-Scotti Business Conditions Index." - *Weekly Macroeconomic Activity*. N.p., n.d. Web. 17 Oct. 2013.

“Fourier Series” Wolfarm Mathwork, 13 Oct. 2013. Web. 17 Oct. 2013.

“Confidence interval” Wolfarm Mathwork, 13 Oct. 2013. Web. 17 Oct. 2013.

“Autocorrelation and Partial Autocorrelation” Wolfarm Mathwork, 13 Oct. 2013. Web. 17 Oct. 2013.

“Corrcoef” Wolfarm Mathwork, 13 Oct. 2013. Web. 17 Oct. 2013.

“All Historical Data of Stock Price” Yahoo Finance, 13 Oct. 2013. Web. 17 Oct. 2013

“NumXL” Spider Financial, 04 Feb. 2014. Web. 15 Feb. 2014

"TISEANNonlinear Time Series Analysis." *TISEAN: Nonlinear Time Series Analysis*. N.p., n.d. Web. 12 Mar. 2014.

Appendix

Matlab Codes (first model)

```
% This is for IQP ploting for 10 stocks
name = 'Apple Close';
Company = Apple_Close;
truePrice10 = Apple09252013;
Date = Date502;
lag = 200;
fitLevel = 2;
fourierLevel = 'fourier4';

%clear older figures
clf

% Fit without cutting
f = polyfit (Date, Company, fitLevel);
trend = polyval (f, Date);

% differnce after get rid of trend
Diff = Company - trend;

%fourier fit
fourierFunc = fit (Date, Diff, fourierLevel);
fourierFit = fourierFunc (Date);

% noise = difference - fouriserfit
noise = Diff - fourierFit;
```

```
%Plot the original, trend, diffence
plot (Date ,Company, '-', Date, trend ,'-', Date, Diff);
title ([name ' Orognal data, trend and difference']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
%Plot the fourier fit and diff
plot (Date, fourierFit,'-', Date, Diff);
title ([name ' Fourier fit and difference']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
%Plot the noise
plot (Date, noise);
title ([name ' Noise']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause

%corr
autocorr (Company, lag);
%pause
c = autocorr (Company, lag);

% find the index of correlation in the data
corr_index = find (abs(c) < 10^-2.5);

% cut the data and date
cutCompany = Company (end-corr_index + 1 : end);
cpDate = Date502 (1:corr_index+20);
```

```
%resign
Company = cutCompany;
Date = cpDate;

%normalize the date
t_ave = (corr_index(1)+20)/2;
ncpDate = (cpDate - t_ave)/t_ave;
Date = ncpDate (1:corr_index);

% Fit
f = polyfit (Date, Company, fitLevel);
trend = polyval (f, Date);

% differnce after get rid of trend
Diff = Company - trend;

%fourier fit
fourierFunc = fit (Date, Diff, fourierLevel);
fourierFit = fourierFunc (Date);

% noise = difference - fouriserfit
noise = Diff - fourierFit;

%Plot the original, trend, diffence
plot (Date ,Company, '-', Date, trend ,'-', Date, Diff);
title ([name ' Orognal data, trend and difference']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
%Plot the fourier fit and diff
```

```
plot (Date, fourierFit,'-', Date, Diff);
title ([name ' Fourier fit and difference']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
%Plot the noise
plot (Date, noise);
title ([name ' Noise']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause

% Predict price = trend + fourier
newDate = ncpDate(corr_index : corr_index + 20);
newTrueDate = ncpDate(corr_index : corr_index + 10);
predictPrice = polyval(f, newDate) + fourierFunc(newDate) + noise(end);
meanOfNoise = mean(abs(noise));
stdOfNoise = std(noise)+ meanOfNoise;
%Plot the predicted price
plot (newDate, predictPrice,'-',newTrueDate, truePrice10,'-',newDate,
predictPrice+stdOfNoise,'.',newDate, predictPrice-stdOfNoise,'.',newDate,
predictPrice+meanOfNoise,'.',newDate, predictPrice-meanOfNoise,'.');
title ([name ' predictic Price']);
ylabel ('Price(Dollar)');
xlabel ('20 days prediction to 10/02/2013');
pause
```

```
% Nasdaq
Nasdaq = nastiq;
fitLevelNasdaq = 4;
fourierLevelNasdaq = 'fourier2';
cutNasdaq = Nasdaq (end-corr_index + 1 : end);

%resign
aveNas = mean(cutNasdaq);
nNas = (cutNasdaq - aveNas)/aveNas ;
Nasdaq = nNas;

% Fit
fNasdaq = polyfit (Date, Nasdaq, fitLevelNasdaq)
trendNasdaq = polyval (fNasdaq, Date);

% differnce after get rid of trend
DiffNasdaq = Nasdaq - trendNasdaq;

%fourier fit
fourierFuncNasdaq = fit (Date, DiffNasdaq, fourierLevelNasdaq);
fourierFitNasdaq = fourierFuncNasdaq (Date);

% noise = difference - fouriserfit
noiseNasdaq = DiffNasdaq - fourierFitNasdaq;

plot (Date ,nNas, '-', Date, trendNasdaq ,'-', Date, DiffNasdaq);
title ([name ' Nasdaq Price']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
predictNasdaq = polyval(fNasdaq, newDate) + fourierFuncNasdaq(newDate) +
```

```
noiseNasdaq(end);

plot (newDate, predictNasdaq);
title ([name 'Nasdaq predictic Price']);
ylabel ('Price(Dollar)');
xlabel ('20 days prediction to 10/02/2013');

pause

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
alph = corrcoef(cutNasdaq, cutCompany);
alph = alph(1,2);

%
for i = 1:21
croPre(i) = predictPrice(i)* predictNasdaq(i);

end
%Predict
predict2 = predictPrice +(1-alph)*transpose(croPre)-(1-alph)*transpose(croPre(1));
noise2 = noise + (1-alph)*noiseNasdaq;
meanOfNoise2 = mean(abs(noise2))
stdOfNoise2 = std(noise2)+ meanOfNoise2

%Plot the predicted price
plot (newDate, predict2,'-',newTrueDate, truePrice10,'-',newDate,
predict2+stdOfNoise2,'.',newDate, predict2-stdOfNoise2,'.',newDate,
predict2+meanOfNoise2,'.',newDate, predict2-meanOfNoise2,'.');
title ([name ' predictic Price with Nasdaq']);
ylabel ('Price(Dollar)');
```

```
xlabel ('20 days prediction to 10/02/2013');
pause
predict2 = predict2(1:11);
plot (newTrueDate, predict2-truePrice10);
title ([name ' predictic Price difference']);
ylabel ('Price(Dollar)');
xlabel ('10 Days');
pause
```

matlab codes (second version)

```
% This is for IQP ploting

% Here are all the variables that need to be modified to plot
name = 'DUK';
Company = DUK_close;
truePrice10 = DUK11052;

Company2 = DUK0930;
truePrice102 = DUK1105;

Date = Date502;
lag = 200;
fitLevel = 1;
fourierLevel = 'fourier3';
ABSlevel = -2;
fitLevel2 = 1;
fourierLevel2 = 'fourier2';

%clear older figures
```

```
clf

% Fit without cutting
f = polyfit (Date, Company, fitLevel);
trend = polyval (f, Date);

% differnce after get rid of trend
Diff = Company - trend;

%fourier fit
fourierFunc = fit (Date, Diff, fourierLevel);
fourierFit = fourierFunc (Date);

% noise = difference - fouriserfit
noise = Diff - fourierFit;



%corr
autocorr (Company, lag);
%pause
c = autocorr (Company, lag);

% find the index of correlation in the data
corr_index = find (abs(c) < 10^ABSlevel)

% cut the data and date
cutCompany = Company (end-corr_index + 1 : end);
cpDate = Date502 (1:corr_index+46);



%resign
```

```
Company = cutCompany;
Date = cpDate;

%normalize the date
t_ave = (corr_index(1)+ 46)/2;
ncpDate = (cpDate - t_ave)/t_ave;
Date = ncpDate (1:corr_index);

% Fit
f = polyfit (Date, Company, fitLevel);
trend = polyval (f, Date);

% differnce after get rid of trend
Diff = Company - trend;

%fourier fit
fourierFunc = fit (Date, Diff, fourierLevel);
fourierFit = fourierFunc (Date);

% noise = difference - fouriserfit
noise = Diff - fourierFit;

%Plot the original, trend, diffence
plot (Date ,Company, '-', Date, trend ,'-', Date, Diff);
title ([name '  Orignal data, trend and difference']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
%Plot the fourier fit and diff
plot (Date, fourierFit,'-', Date, Diff);
title ([name ' Fourier fit and difference']);
```

```
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
%Plot the noise
plot (Date, noise);
title ([name ' Noise']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause


% Predict price = trend + fourier
newDate = ncpDate(corr_index  : corr_index + 46);
newTrueDate = ncpDate(corr_index  : corr_index + 45);
predictPrice = polyval(f, newDate) + fourierFunc(newDate) + noise(end);


% Mean and std of noise
meanOfNoise = mean(abs(noise));
stdOfNoise = std(noise)+ meanOfNoise;

% period of validation
for i = 1:46
   if truePrice10(i) >= predictPrice(i) +stdOfNoise
      tvald = newTrueDate(i);
      tvaldt = i;
      break
   end
   if truePrice10(i) <= predictPrice(i)-stdOfNoise
      tvald = newTrueDate(i);
      tvaldt = i;
```

```
        break
    end
    tvald = newTrueDate(i);
end



% Axis range
ymin = min(min(predictPrice-stdOfNoise),min(truePrice10));
ymax = max(max(predictPrice+stdOfNoise),max(truePrice10));
yMax = ymax + (ymax - ymin)/1.5;
yMin = ymin - (ymax - ymin)/5;

%Plot the predicted price
plot (newDate, predictPrice,'r-',newTrueDate, truePrice10,'-',newDate,
predictPrice+stdOfNoise,':',newDate, predictPrice-stdOfNoise,'k:',newDate,
predictPrice+meanOfNoise,'m:',newDate, predictPrice-meanOfNoise,':');
line([newTrueDate(tvaldt) newTrueDate(tvaldt)], [yMin
predictPrice(tvaldt)+stdOfNoise],'LineWidth',1.5,'LineStyle',:,'Color',[.5 .5 .5]);
title ([name '  Predictic Price with Data From 09/07/2011 to 09/07/2013' ]);
ylabel (['Price(Dollar)']);
xlabel (['40 Days Prediction  ' fourierLevel '  FitLevel  ' int2str(fitLevel) ', Autocorr gives
' int2str(corr_index(1))]);
axis([newDate(1) newTrueDate(end) yMin yMax])
Day =
{'09/07/2013','09/17/2013','09/26/2013','10/05/2013','10/15/2013','10/26/2013','11/05/201
3'};
x=[newDate(1):(newDate(40)-newDate(1))/6:newDate(40)];
set(gca,'xtick',x);
set(gca,'xticklabel',Day);
legend('Predict Price','Real Price','PP plus Std and Noise','PP minus Std and Noise','Plus
Mean of Noise','Minus Mean of Noise','Validation line','Location','NorthEast');
```

```
pause


% Save the data for later part
compDate = newDate;
compPrice = predictPrice;
aaa = newTrueDate;
bbb = truePrice10;





%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%

Company = Company2;
truePrice10 = truePrice102;
Date = Date502;
lag = 200;
fitLevel = fitLevel2;
fourierLevel = fourierLevel2;

%clear older figures
clf

% Fit without cutting
f = polyfit (Date, Company, fitLevel);
trend = polyval (f, Date);

% differnce after get rid of trend
```

```
Diff = Company - trend;

%fourier fit
fourierFunc = fit (Date, Diff, fourierLevel);
fourierFit = fourierFunc (Date);

% noise = difference - fouriserfit
noise = Diff - fourierFit;

%pause
c = autocorr (Company, lag);

% find the index of correlation in the data
corr_index = find (abs(c) < 10^ABSlevel);

% cut the data and date
cutCompany = Company (end-corr_index + 1 : end);
cpDate = Date502 (1:corr_index+40);



%resign
Company = cutCompany;
Date = cpDate;

%normalize the date
t_ave = (corr_index(1)+ 27)/2;
ncpDate = (cpDate - t_ave)/t_ave;
Date = ncpDate (1:corr_index);

% Fit
f = polyfit (Date, Company, fitLevel);
```

```
trend = polyval (f, Date);

% differnce after get rid of trend
Diff = Company - trend;

%fourier fit
fourierFunc = fit (Date, Diff, fourierLevel);
fourierFit = fourierFunc (Date);

% noise = difference - fouriserfit
noise = Diff - fourierFit;



% Predict price = trend + fourier
newDate = ncpDate(corr_index  : corr_index + 26);
newTrueDate = ncpDate(corr_index  : corr_index + 26);
predictPrice = polyval(f, newDate) + fourierFunc(newDate) + noise(end);


% Mean and std of noise
meanOfNoise = mean(abs(noise));
stdOfNoise = std(noise)+ meanOfNoise;

% period of validation
for i = 1:46
   if truePrice10(i) >= predictPrice(i) +stdOfNoise
      tvald = newTrueDate(i);
      tvaldt = i;
      break
   end
```

```
    if truePrice10(i) <= predictPrice(i)-stdOfNoise
        tvald = newTrueDate(i);
        tvaldt = i;
        break
    end
    tvald = newTrueDate(i);
end



% Axis range
ymin = min(min(predictPrice-stdOfNoise),min(truePrice10));
ymax = max(max(predictPrice+stdOfNoise),max(truePrice10));
yMax = ymax + (ymax - ymin)/1.5;
yMin = ymin - (ymax - ymin)/5;



%Plot the predicted price
plot (newDate, predictPrice,'r-',newTrueDate, truePrice10,'-',newDate,
predictPrice+stdOfNoise,':',newDate, predictPrice-stdOfNoise,'k:',newDate,
predictPrice+meanOfNoise,'m:',newDate, predictPrice-meanOfNoise,':');
line([newTrueDate(tvaldt) newTrueDate(tvaldt)], [yMin
predictPrice(tvaldt)+stdOfNoise],'LineWidth',1.5,'LineStyle',:,'Color',[.5 .5 .5]);
title ([name '  Predictic Price with Data From 09/30/2011 to 09/30/2013' ]);
ylabel ('Price(Dollar)');
xlabel (['27 Business Days Prediction   ' fourierLevel '  FitLevel is  ' int2str(fitLevel) ',
Autocorr gives ' int2str(corr_index(1))]);
axis([newDate(1) newTrueDate(end) yMin yMax])
Day1 = {'09/30/2013','10/08/2013','10/15/2013','10/21/2013','10/29/2013','11/05/2013'};
x=[newDate(1):(newDate(27)-newDate(1))/5:newDate(27)];
set(gca,'xtick',x);
```

```
set(gca,'xticklabel',Day1);
legend('Predict Price','Real Price','PP plus Std and Noise','PP minus Std and Noise','Plus Mean of Noise','Minus Mean of Noise','Location','NorthEast');
pause



% Compare
ymin = min(min(min(predictPrice),min(compPrice)), min(bbb));
ymax = max(max(max(predictPrice),max(compPrice)), max(bbb));
yMax = ymax + (ymax - ymin)/1.5;
yMin = ymin - (ymax - ymin)/5;

plot (newDate, predictPrice,'r--',compDate, compPrice,'r-',aaa, bbb,'b-')
title ([name '  Two Predictions Comparison with Real Price']);
ylabel ('Price(Dollar)');
xlabel ('27 Days Prediction and 40 Days Prediction');
axis([aaa(1) aaa(end) yMin yMax])
Day1 = {'09/06/2013','09/21/2013','10/04/2013','10/12/2013','10/20/2013','10/27/2013','11/05/2013'};
x=[compDate(1):(compDate(40)-compDate(1))/6:compDate(40)];
set(gca,'xtick',x);
set(gca,'xticklabel',Day1);
legend('Predict Price with Newer Data','Predict Price with Older Data','Real Price','Location','NorthEast');
pause
```

Matlab code (ARIMA)

```
name = 'XOM';
truePrice = XOM;
```

```
ARIMApredictori = XOMAtrend
Date = XOMdate;
fourierLevel = 'fourier1';
predictPrice1 = XOMApredict;
truePrice10 = XOM20;

%clear older figures
clf

% our stock and trend
plot(truePrice,'green')

plot(ARIMApredictori,'blue')

% differnce after get rid of trend
Diff = truePrice - ARIMApredictori;
%fourier fit
fourierFunc = fit (Date, Diff, fourierLevel);
fourierFit = fourierFunc (Date);


noise = Diff - fourierFit;

%Plot the original, trend, diffence
plot (Date ,truePrice, '-', Date, ARIMApredictori ,'-', Date, Diff);
title ([name '  Orignal data, trend and difference']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
%Plot the fourier fit and diff
plot (Date, fourierFit,'-', Date, Diff);
```

```
title ([name ' Fourier fit and difference']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause
%Plot the noise
plot (Date, noise);
title ([name ' Noise']);
ylabel ('Price(Dollar)');
xlabel ('Number of days till 09/12/2013');
pause

%standard deviation
meanOfNoise = mean(abs(noise));
stdOfNoise = std(noise)+ meanOfNoise;

predictPrice = predictPrice1 + fourierFunc(newDate)

nnoise = -truePrice10(1) + predictPrice(1);

    predictPrice = predictPrice- nnoise;




%Plot the predicted price
plot (newDate, predictPrice,'r-',newDate, truePrice10,'-',newDate,
predictPrice+stdOfNoise,'m.',newDate, predictPrice-stdOfNoise,'k.',newDate,
predictPrice+meanOfNoise,'.',newDate, predictPrice-meanOfNoise,'gr.');
title ([name '  Predictic Price with Data From 09/04/2011 to 09/30/2013' ]);
ylabel (['Price(Dollar)']);
legend('Predict Price','Real Price','PP plus Std and Noise','PP minus Std and Noise','Plus
Mean of Noise','Minus Mean of Noise','Validation line','Location','NorthEast');
```