Arjun Srinivasan
gtID: 903027003

Machine Learning For Trading
ML Project 4

Method used for learning, indicators created:

I used a linear regression model for the prediction in this case. This was based on evaluation against the results of KNN and the Random Forests for multiple feature sets. The evaluation was similar to what it was in the previous assignments, comparing the RMS errors and ensuring avoiding overfitting, comparing coefficient of correlations etc. The final set of features used for training the linear model were (based on the given information of the data being in the form of a sine wave):

-Peak-to-peak Amplitude

The amplitude of the previous stock prices gives an indication of the magnitude of the price of the stock.

-Frequency
This feature gives an indication of deviation in the price of the stock, giving a measure of volatility.
-Phase

The phase, gives an indication of the price of a stock back at a particular point in time.

- Price of the symbol lookback period days ago (for the purpose of this section, called oldPrice)

This helped get the historical value of the symbol

- First derivative of oldPrice

This helps understand the change in price of the symbol based on its previous day's value

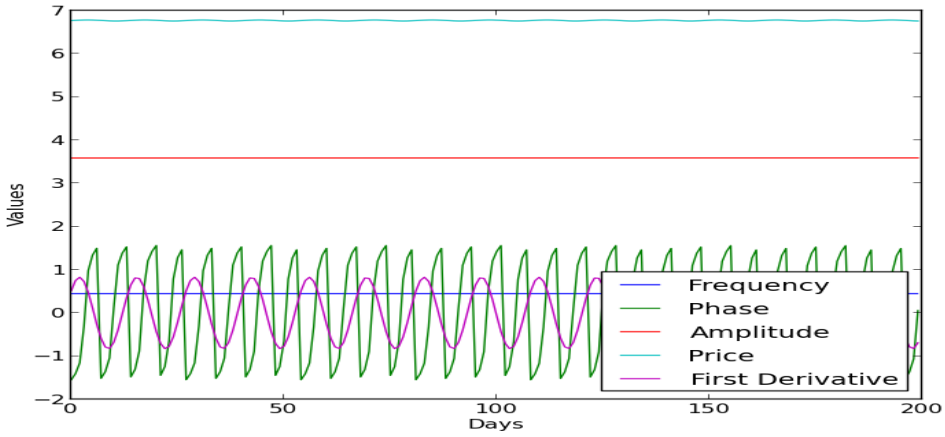
- oldPrice^2 and oldPrice^3 (Polynomial features used for regression)

These are used because the dataset at hand is not a direct case of linear regression and takes the shape of a sine wave (known beforehand)

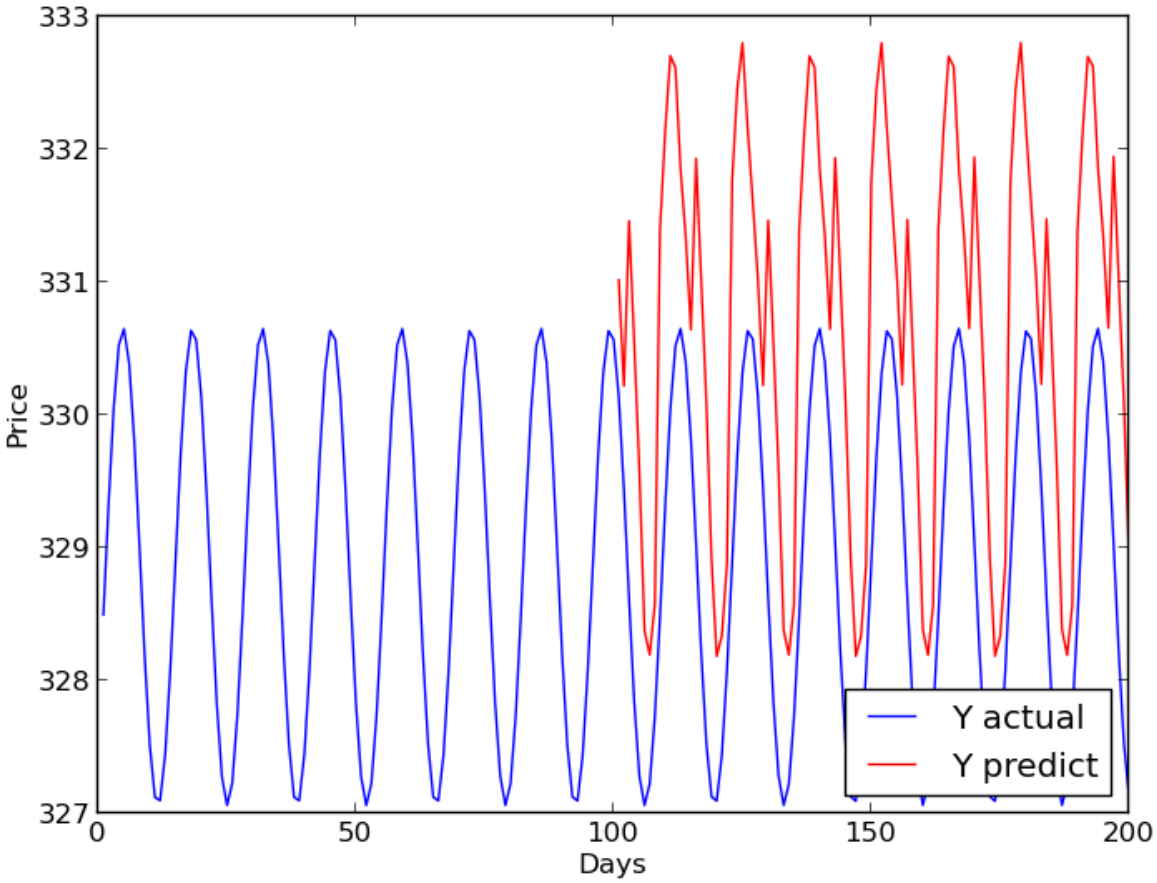
Results:

for the test file **ML4T-292**.csv:

first 5 Features
for first 200 days
(in the



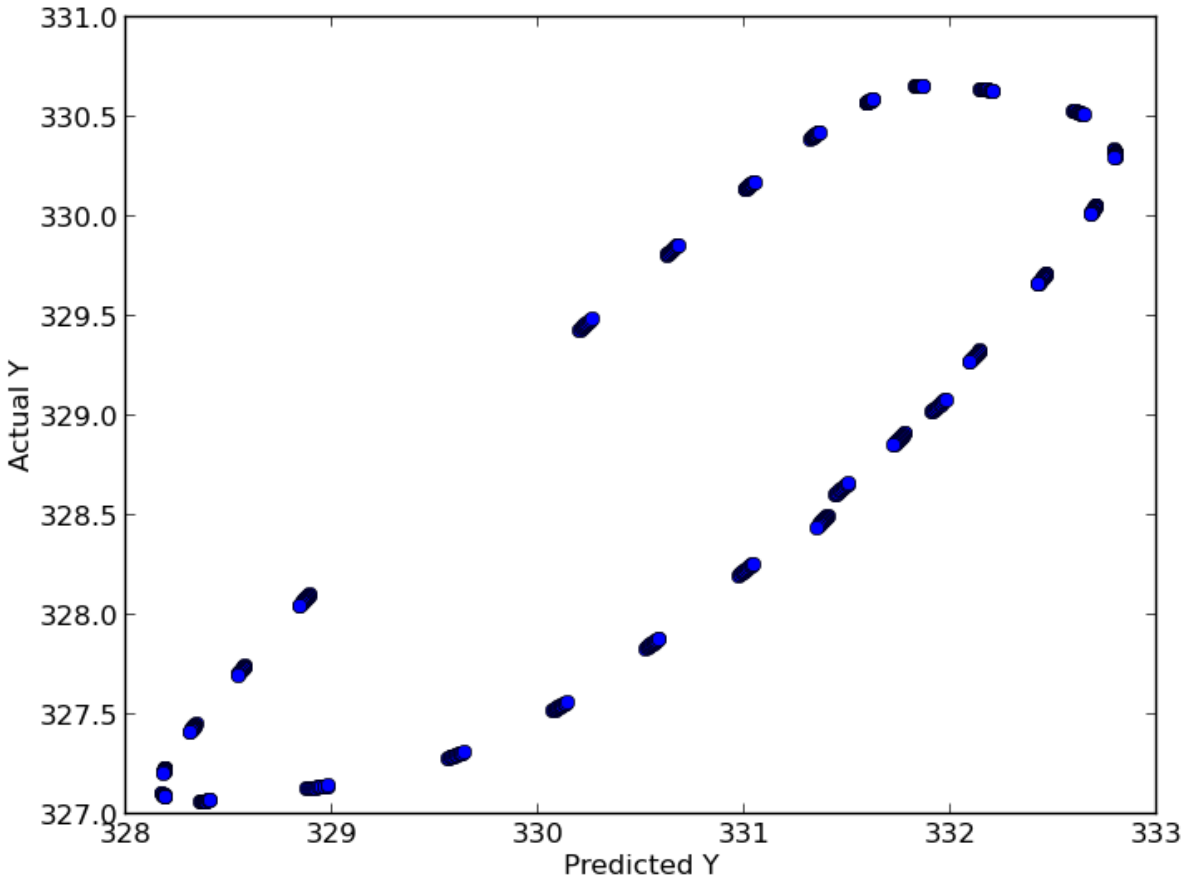
screenshot, price has been depicted as the cube root just to scale the image down to show other values too)



Y actual vs Y predict for first 200 days

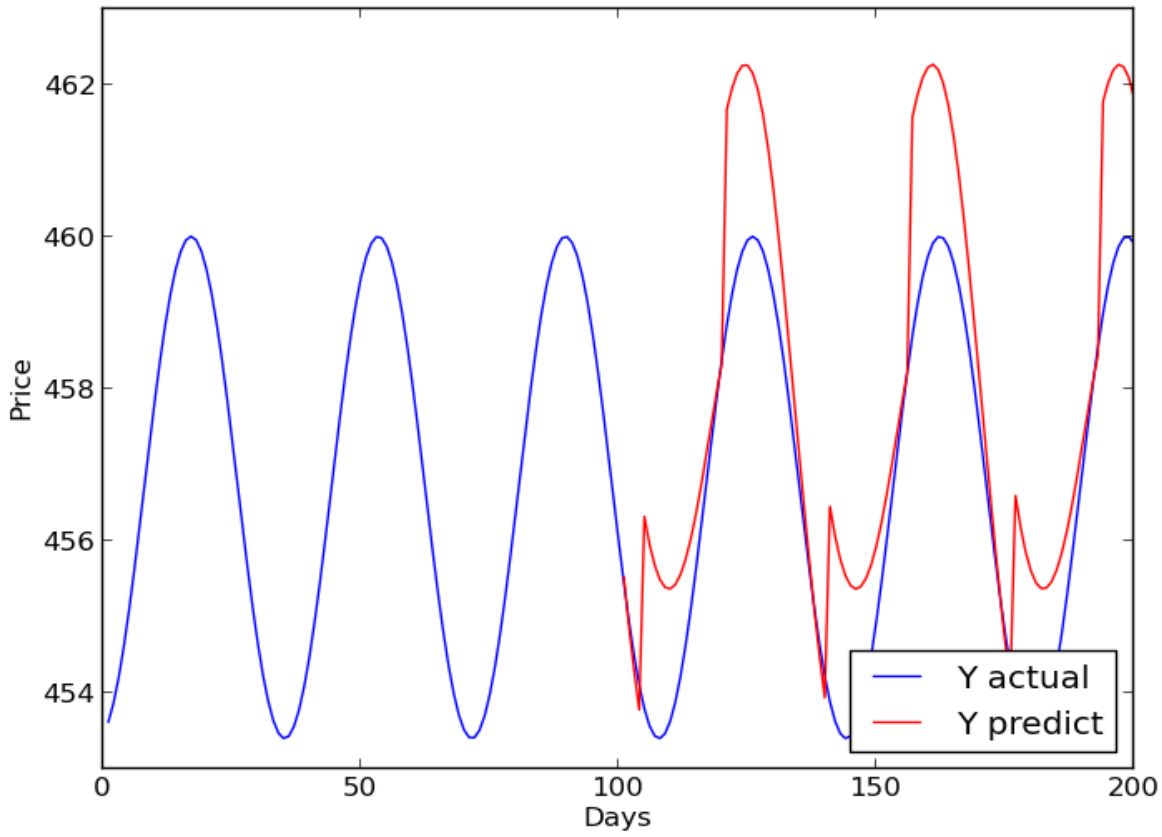
Y

actual vs Y predict for last 200 days

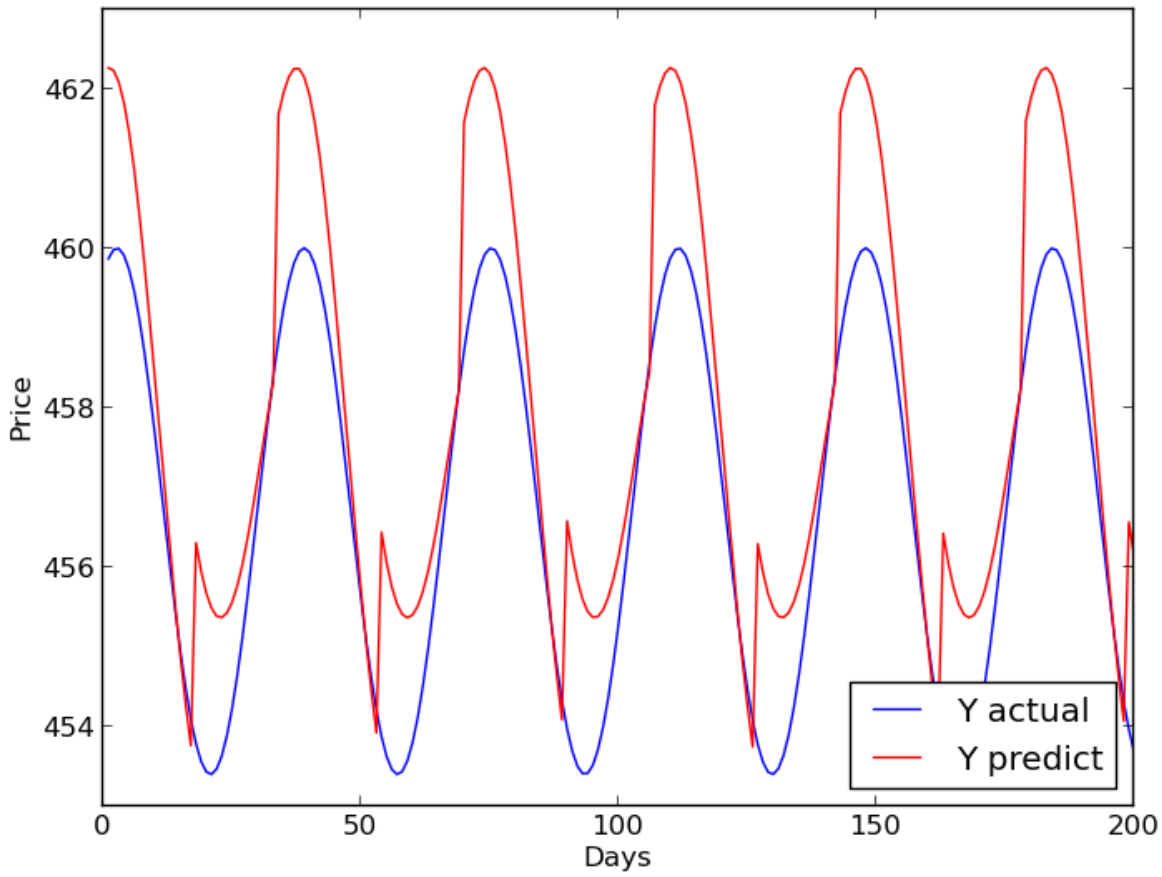


actual Y vs predicted Y

For **ML4T-320**.csv (300 + 1(A) + 19(S))



Y actual vs Y predict for first 200 days



Y actual vs Y predict last 200 days

RMS and correlation coefficients:

for 292.csv
RMS = 2.03474805609
correlation coef. = ***0.830934440069***
for 320.csv
RMS = 1.57865464314
correlation coef. = **0*.937476753415***

Why you think your method worked well (or did not)?

In my opinion, the model performed average. The features selected allowed fitting values relatively well for the sine wave (frequency , amplitude and phase). The correlation coefficients results for both the files are relatively high but in the first dataset, the RMS is a bit on the higher side which is why there isn't a perfect overlapping of the waves. But, the major reason for the existing performance of the model, I believe are the polynomial featured selected in this case – x2, x3). In my opinion, features including further derivatives and time based filtering of data to learn can improve the performance of the model – but we have to be careful to not use too many exponential features because this can lead to the risk of over fitting.

References:

http://stackoverflow.com/questions/16716302/how-do-i-fit-a-sine-curve-to-my-data-with-pylab-and-numpy

https://piazza.com/class/hx3t2m1k1pv8i?cid=1146

https://piazza.com/class/hx3t2m1k1pv8i?cid=1144

https://piazza.com/class/hx3t2m1k1pv8i?cid=1147