Evaluation of Machine Learning Models for Pre-market NIFTY % Gain Prediction — Shivansh Bansal

Data:
Pre-market Data for 31st May, 1st June, 2nd June
SGX NIFTY snapshot at our market opening (9:08 AM)

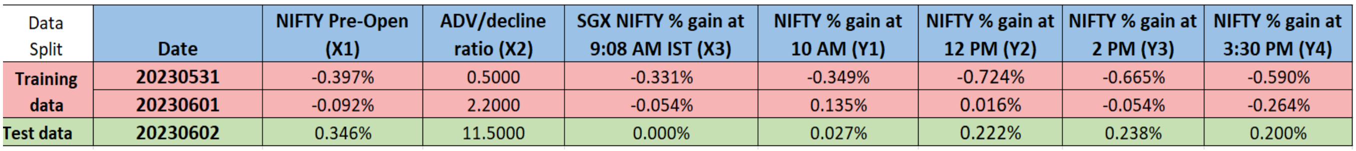
Variables I have:
X1: NIFTY Pre-market snapshot %
X2: ADV/Decline Ratio
X3: SGX NIFTY value snapshot at NSE's premarket time

Values to forecast:
Y1 : NIFTY % at 10:00 AM
Y2: NIFTY % at 12:00 PM
Y3: NIFTY % at 2:00 PM
Y4: NIFTY % at 3:30 PM

Aim:
To understand how well the NIFTY pre-market snapshot sustains as the market progresses, and build a model upon independently predicting the four values from the 3 variables.

$$Y4 = f(X1, X2, X3)$$
$$Y3 = f(X1, X2, X3)$$
$$Y2 = f(X1, X2, X3)$$
$$Y1 = f(X1, X2, X3)$$

Expectation:
To tune models to predict the Y1, Y2, Y3, and Y4. The predictive power of the three variables should be strong for Y1 but should decrease as I move from Y1 to Y4 because as the market progresses, other factors might start affecting the market movement.

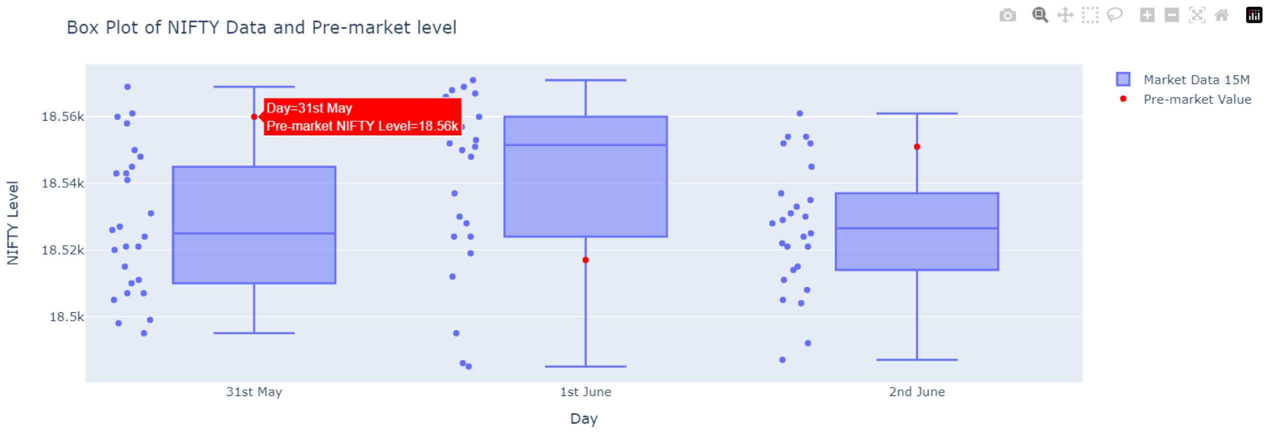


Data Split	Date	NIFTY Pre-Open (X1)	ADV/decline ratio (X2)	SGX NIFTY % gain at 9:08 AM IST (X3)	NIFTY % gain at 10 AM (Y1)	NIFTY % gain at 12 PM (Y2)	NIFTY % gain at 2 PM (Y3)	NIFTY % gain at 3:30 PM (Y4)
Training data	20230531	-0.397%	0.5000	-0.331%	-0.349%	-0.724%	-0.665%	-0.590%
	20230601	-0.092%	2.2000	-0.054%	0.135%	0.016%	-0.054%	-0.264%
Test data	20230602	0.346%	11.5000	0.000%	0.027%	0.222%	0.238%	0.200%

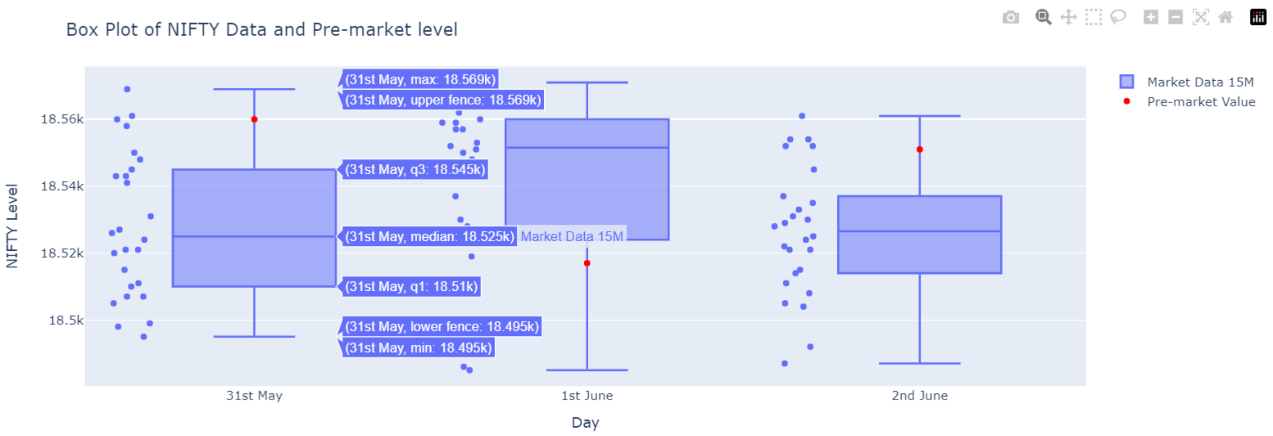
Exploratory Data Analysis:

Feature selection: It was performed by me to identify the most relevant features for predicting the values of Yi. The goal of feature selection is to improve model performance by eliminating irrelevant or redundant features.Over here, X1 and X3 had better correlation with the output variables. But since we already had less data to train upon, i didn't drop X2 and have kept the number of feature-to-select to three only, but this can be changed based our requirement.

Interactive Box Plot to understand in what quartile does pre-market Nifty Value look as compared to the whole day's 15-minute data points.
Red one - Pre-market NIFTY value; Purple one - NIFTY 15-minute data points for the three days
Snapshot 1:



Snapshot 2:



Methodology:

Data Split: The dataset consisted of three rows, which I divided into two parts: a training set and a test set. I used two rows for training and one row for testing, representing a 2:1 split (two red rows in the above table are training data and the green row is test data). This method allows us to do cross-validation of our trained model.
Over here, the dataset was pretty small, otherwise, the stationarity of the data might also have to be checked.

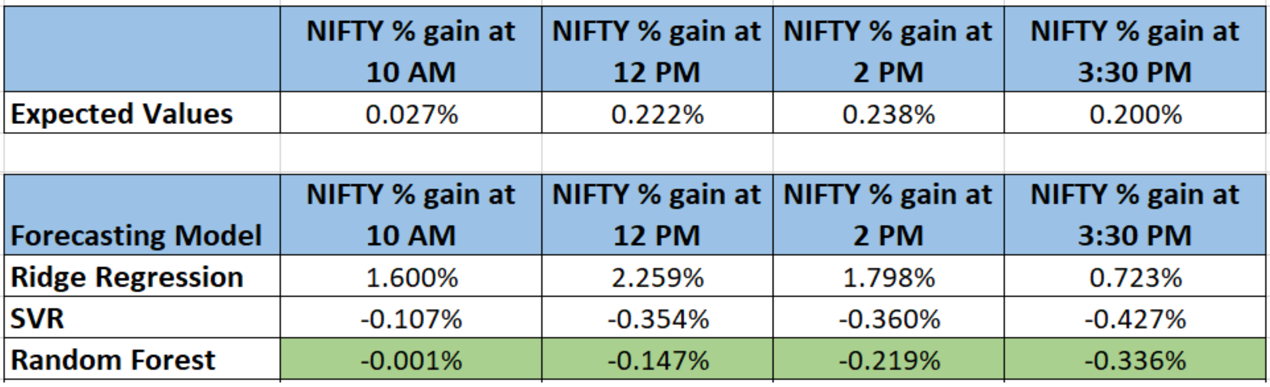
I implemented and evaluated the following machine learning models:

1. Ridge Regression: Ridge regression is a linear regression model with a regularization term. I chose this model as it can handle multicollinearity among features and prevent overfitting. It uses L2 regularization to control the magnitude of the coefficients.

2. Support Vector Regression (SVR): SVR is a regression variant of Support Vector Machines. I selected this model because it can handle non-linear relationships by mapping the input features to a higher-dimensional space.

3. Random Forest Regression: Random Forest is an ensemble learning method that combines multiple decision trees. I utilized this model as it can capture non-linear relationships and handle feature interactions quite effectively.

Results and Insights:



	NIFTY % gain at 10 AM	NIFTY % gain at 12 PM	NIFTY % gain at 2 PM	NIFTY % gain at 3:30 PM
Expected Values	0.027%	0.222%	0.238%	0.200%

Forecasting Model	NIFTY % gain at 10 AM	NIFTY % gain at 12 PM	NIFTY % gain at 2 PM	NIFTY % gain at 3:30 PM
Ridge Regression	1.600%	2.259%	1.798%	0.723%
SVR	-0.107%	-0.354%	-0.360%	-0.427%
Random Forest	-0.001%	-0.147%	-0.219%	-0.336%

Upon training and evaluating the models, I obtained the following insights:

1. Ridge Regression: The Ridge Regression model demonstrated a very high error in predicting the NIFTY % gain at 10 AM. However, due to the limited training data, the model's performance is not trustable. Further evaluation with more extensive datasets could have resulted in better results

2. SVR: This model certainly portrayed better results as compared to ridge regression probably because of its ability to capture non-linear relationships as well.

3. Random Forest Regression: The Random Forest model exhibited the **lowest error** among all the evaluated models. This suggests that the ensemble approach and the ability to capture non-linear relationships were beneficial, even with a limited dataset. Random Forest is also expected to perform better at time when it comes to having a smaller training set. Moreover, as expected, the error increased as we moved from 10:00 AM to the end of the day which might be a result of weakness of predictive power as we move towards the end of the day.

Further Scope:

1. Increasing the training dataset to achieve good tuning of hyperparameters of our model
2. Building more features in addition to X1, X2, X3; for example % contribution from top gainer and if the contribution is very high then we can expect nifty's movement had very high contribution from that stock & the probability of NIFTY sustaining the move might be less
3. Doing classification instead of regression based on the opening numbers on where we expect the market to end, say bullish or bearish or flat as compared to opening.
4. We can also take say top-5 movers of NIFTY and apply the same model upon them to understand if those heavily traded shares had good predictive power of what is going to happen in them along the day. Taking top-5 will help us neglect the ones which didn't have much activity in pre-market session and hence their signal might not be reliable to withstand throughout the day.

Please note that this assignment is more on the methodology side and the results might be misleading since training on just 2 days doesn’t tune the hyperparameters very well and hence we might be getting misdirected results. In our training, we seem to underfitting our data with very less data to train and hence having high bias. We need to have an optimal level of complexity & dataset to ensure that we don’t underfit or overfit our model on the data.