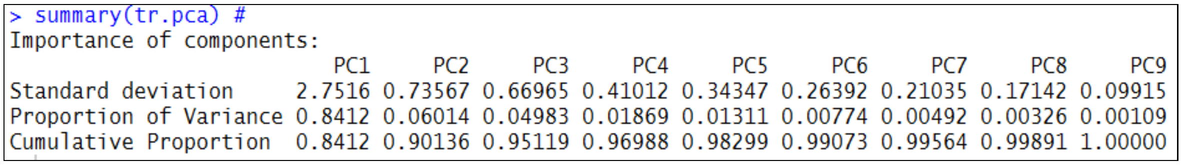
Correlation One: Machine Learning Challenge

Q.1 Which variables matter for predicting S1?

Ans. In this machine learning challenge the data given is stock returns over the period of time. That means it is a time series data which always has a problem of multicollinearity. If we try to predict the stock returns using all the other variables, then we will end up in overfitting which cause poor prediction results. To avoid multicollinearity and overfitting issues we need to shrink the data by removing some of the columns which may not be important in prediction. I have used two different methods to deal with this issues namely Principal Component Analysis and Feature Selection Methods.

Principal Component Analysis

Principal Component Analysis is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. It reduces the dimensionality of the data set by removing the principal components that explains negligible amount of variance. For that remove date and S1 columns and run PCA on S2 through S10 on first 50 rows of training data set.



```
> summary(tr.pca) #
Importance of components:
                          PC1     PC2     PC3     PC4     PC5     PC6     PC7     PC8     PC9
Standard deviation     2.7516 0.73567 0.66965 0.41012 0.34347 0.26392 0.21035 0.17142 0.09915
Proportion of Variance 0.8412 0.06014 0.04983 0.01869 0.01311 0.00774 0.00492 0.00326 0.00109
Cumulative Proportion  0.8412 0.90136 0.95119 0.96988 0.98299 0.99073 0.99564 0.99891 1.00000
```

Observe the values of proportion of variance and select the principal components having threshold value of 1%. Therefore, only first five components meet the threshold keep them and remove S7 to S10.

<u>Variables matter for predicting S1(PCA) = S2, S3, S4, S5, S6</u>

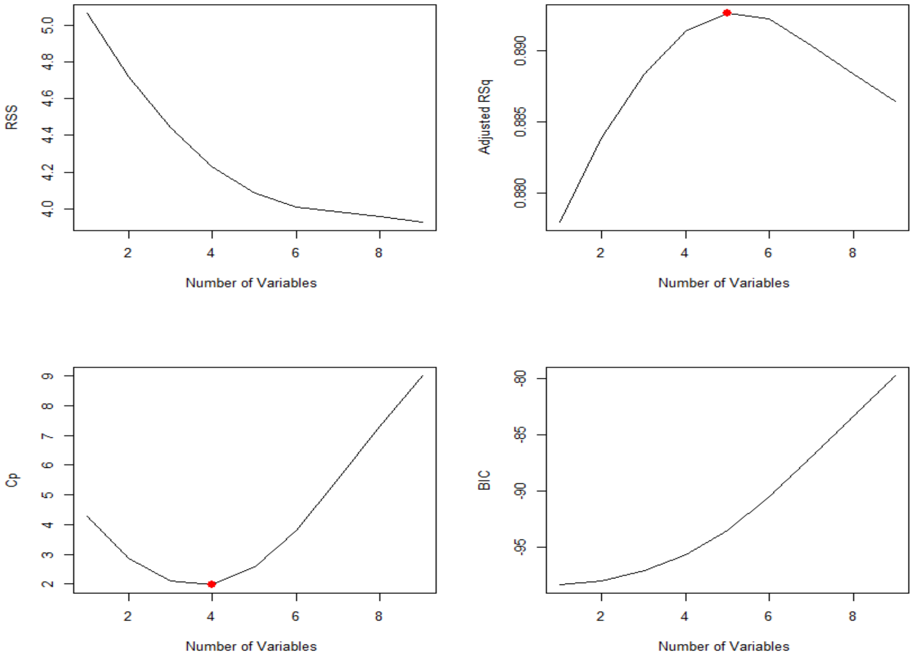
Feature Selection Methods

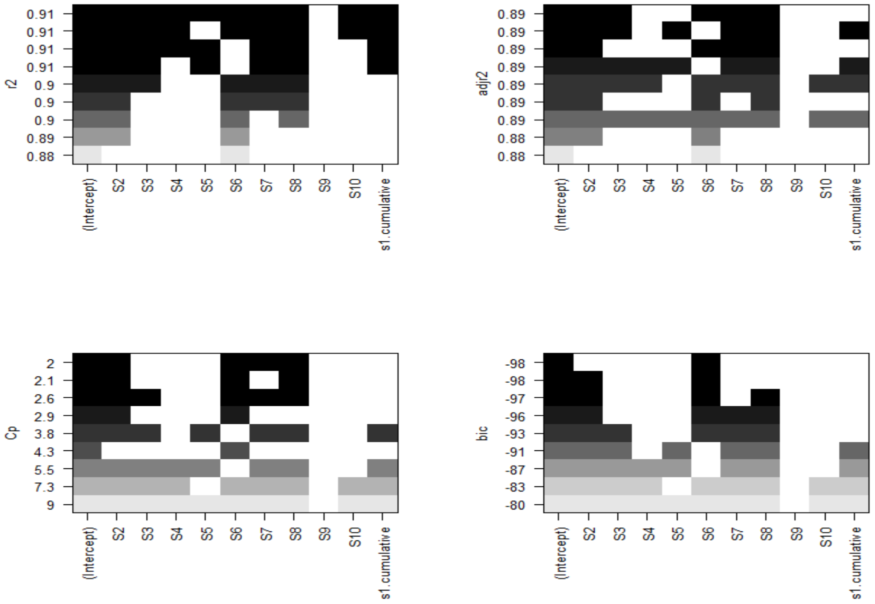
Feature selection is different from dimensionality reduction. Both methods seek to reduce the number of attributes in the dataset, but a dimensionality reduction method do so by creating new combinations of attributes, whereas feature selection methods include and exclude attributes present in the data without changing them. In this case I have used Forward and Backward stepwise collection method. In order for these approaches to yield accurate estimates of the test error, we must use only the training observations to perform all aspects of model-fitting—including variable selection. Therefore, the determination of which model of a given size is best must be made using only the training observations.

I choose the features based on R-square, Cp, AIC and BIC criteria.

<u>Variables matter for predicting S1(Feature Selection Method) = S2, S3, S6, S7, S8</u>

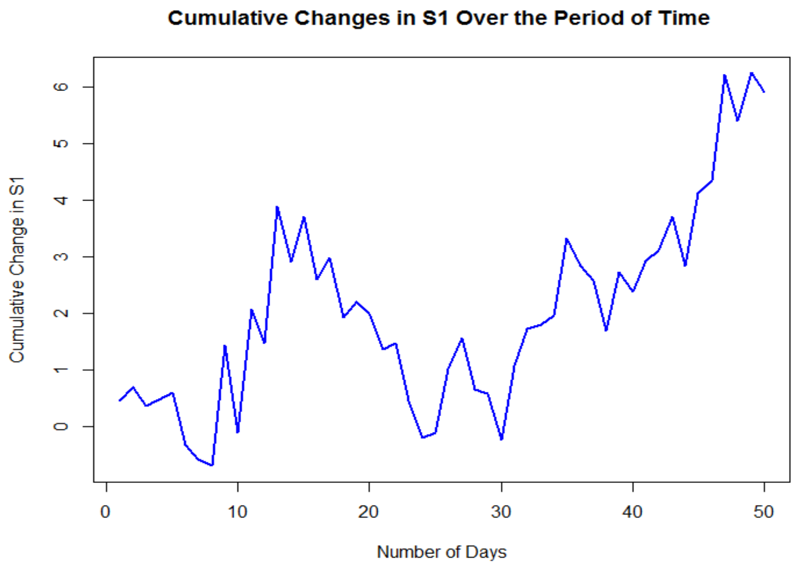
I have used mostly penalized regression methods for prediction by using only S2, S3, S6, S7, S8 these variables and removing the others in this entire project





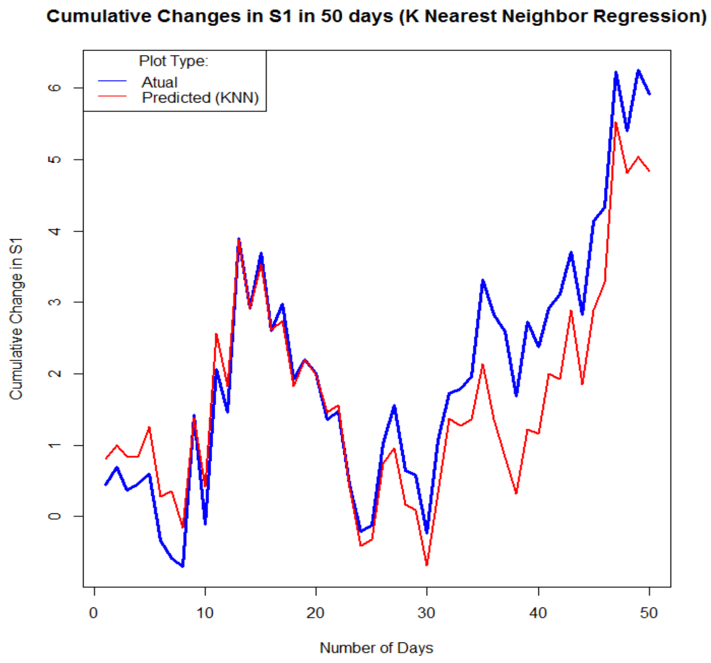
Q.2 Does S1 go up or down cumulatively (on an open-to-close basis) over this period?

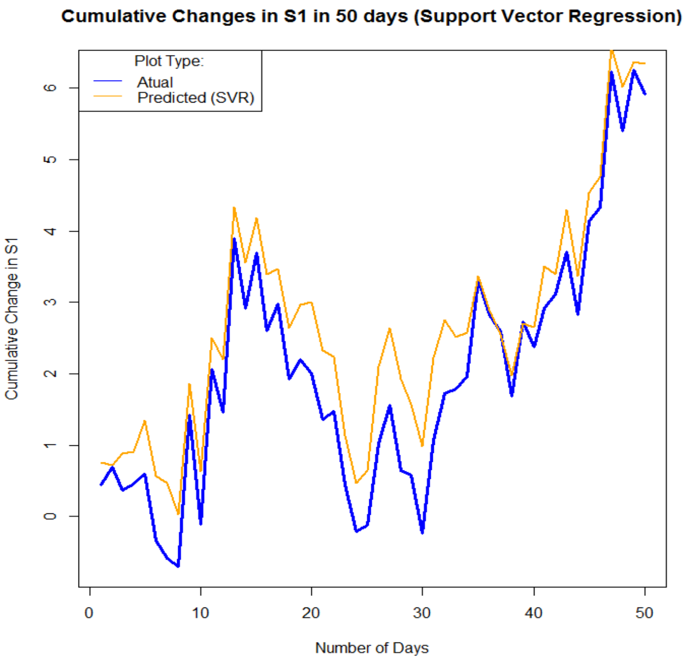
Ans. In the stock returns dataset, S1 represents daily open to close changes of a stock. We find that s1 increases cumulatively over this 50 days' period by 5.92 points. The following graph represents cumulative changes in stocks over the first 50 days:



Q.3 How much confidence do you have in your model? Why and when would it fail?

Ans. As this is a time series data that to stocks data it came with lot of issues like multicollinearity, overfitting and many more. After doing volatility analysis on S1 I got some results. Friday to Monday are 20 days period and Tuesday to Thursday are next 30 days period. We can plot actual cumulative value and predicted cumulative value of S1 and observe the results.


Cumulative Changes in S1 in 50 days (K Nearest Neighbor Regression)


Cumulative Changes in S1 in 50 days (Support Vector Regression)

Here, I showed the prediction on training data using K-Nearest Neighbor Regression (first plot) and Support Vector Regression (second plot). The algorithm does perform well with respect to its predictive abilities, however there are still shortcomings to this technique. In the first graph there are variations in the 30 -50 days i.e Tuesday – Thursday i.e. model is slightly underestimated from the second half of the data. On the other hand, in second graph the model is slightly overestimated throughout the 50 days' period of training dataset. Though these two models have their own shortcomings they have the lowest root mean square error (RMSE = 0.2557318) compared to all the other models I have used in this project.

However, I need to tune the support vector regression model for better performance. This second plot shows the prediction after tuning the model. The given forecast period is only 50 days and with these methods of forecasted values, the final two models is expected to work fine. It may have some variations in long run over the large forecasting periods.

Q.4 What techniques did you use? Why?

Ans. Firstly, I have used Principal Component Analysis to eliminate unnecessary explanatory variables in the training dataset. After running this and considering 1% threshold I have removed S7 – S10 variables. As stated prior, the benefit of this technique is that we preserve the variance of the data set, but are also able to transform it in a manner that allows us to understand the contribution of each principal component to the total variance within the data.

Then, I found that I am using most of the penalized regression methods for prediction with no overfitting and multicollinearity issues, I decided to run feature section method of Forward and Backward Stepwise Section method to choose appropriate explanatory variables. After running this on training data I have got 5 variables namely S2, S3, S6, S7 & S8 for prediction and I eliminated the rest and used this training set for model building.

Models Used to Predict S1

1. Ridge Regression

Ridge Regression is a remedial measure taken to alleviate multicollinearity amongst regression predictor variables in a model. Often predictor variables used in a regression are highly correlated. When they are, the regression coefficient of any one variable depend on which other predictor variables are included in the model, and which ones are left out. So the predictor variable does not reflect any inherent effect of that particular predictor on the response variable, but only a marginal or partial effect, given whatever other correlated predictor variables are included in the model.

2. Lasso Regression

In statistics and machine learning, lasso (least absolute shrinkage and selection operator) is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces.

3. Elastic Net

In statistics and, in particular, in the fitting of linear or logistic regression models, the elastic net is a regularized regression method that linearly combines the L1 and L2 penalties of the Lasso and Ridge regression methods.

4. Support Vector Regression

Support Vector Machine can also be used as a regression method, maintaining all the main features that characterize the algorithm (maximal margin). The Support Vector Regression (SVR) uses the same principles as the SVM for classification, with only a few minor differences. First of all, because output is a real number it becomes very difficult to predict the information at hand, which has infinite possibilities. In the case of regression, a margin of tolerance (epsilon) is set in approximation to the SVM which would have already requested from the problem. But besides this fact, there is also a more complicated reason, the algorithm is more complicated therefore to be taken in consideration. However, the main idea is always the same: to minimize error, individualizing the hyperplane which maximizes the margin, keeping in mind that part of the error is tolerated.

5. K-Nearest Neighbor Regression

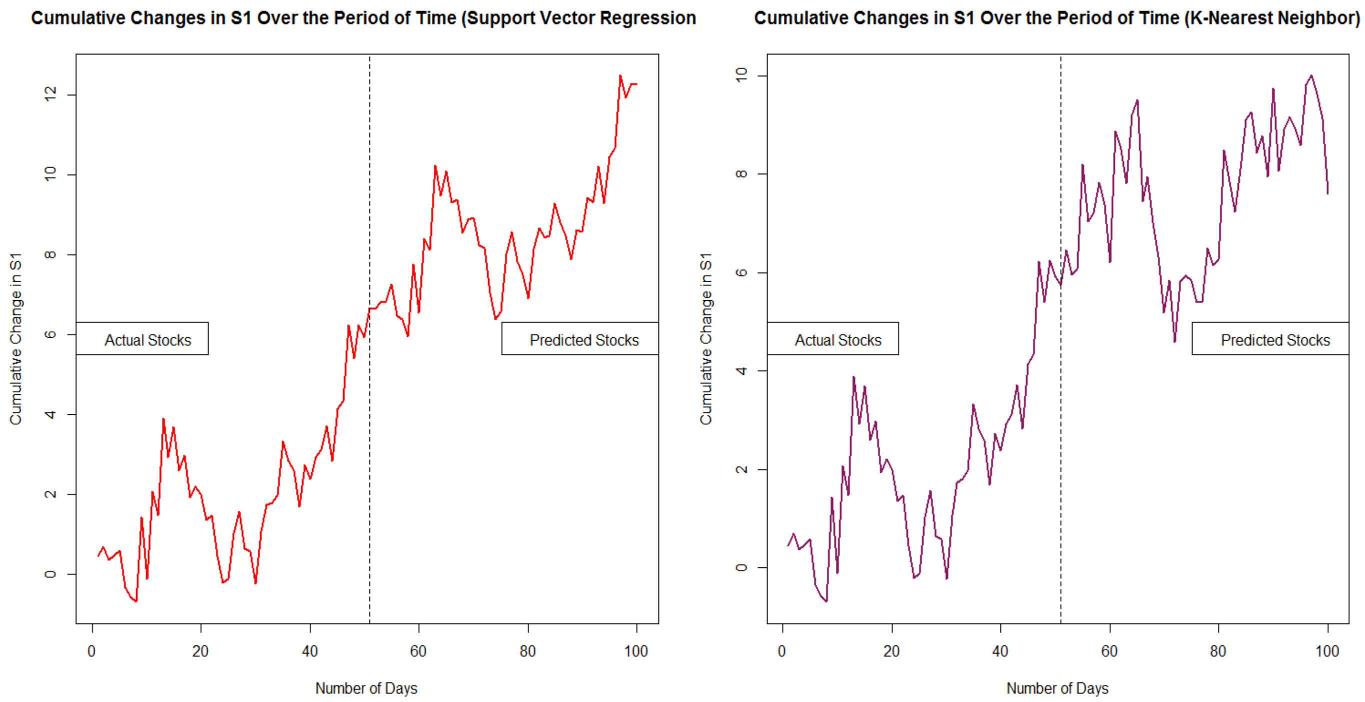
K nearest neighbors is a simple algorithm that stores all available cases and predict the numerical target based on a similarity measure (e.g., distance functions). A simple implementation of KNN regression is to calculate the average of the numerical target of the K nearest neighbors.

I ran all these models on training data set and predicted the stocks of test data set. The root mean square error of all these models are listed in the below table. The models are arranged from highest to lowest RMSE values. Last two models with RMSE = 0.2557318 are the best models with lowest error.

No.	**Model Used**	**RMSE**
1	Ridge Regression	0.2926605
2	Lasso Regression	0.2858645
3	Elastic Net	0.2858711
4	Support Vector Regression	0.2557318
5	K-Nearest Neighbor Regression	0.2557318

Determining The Model to Choose and Why

The K-Nearest Neighbor Regression and Support Vector Regression are performing best in this case. Based on RMSE value these two models are the best models with equal RMSE but different predictions. Let’s see overall cumulative change in S1 over 100 days of period.



As per the prediction of Support Vector Regression cumulative change in S1 rises near 100 days period and it falls in case of K-Nearest Neighbor Regression.

On comparison I found that the model K – Nearest Neighbor Regression model is the best model with the R – Squared value of 91%.

```
> reg.knn$R2Pred
[1] 0.913536
```