**Forcasting Stock Price Using Machine Learning Technique**

|  |  |
| --- | --- |
| Mohammed Nazim Uddin, PhD | Md Tanvir Rahman |
| School of Science, Engineering and Technology | School of Science, Engineering and Technology |
| East Delta University | East Delta University |
| Chattagram, Bangladesh | Chattagram , Bangladesh |
| nazim@eastdelta.edu.bd | Ornobtanvir.git@gmail.com |

**Abstract:**

***the stock market is an emerging sector in any country of the world . Many people directly related to this sector . Along with the development with the stock market ,forecasting become an important topic .Since finance market has become more and more competitive, stock price prediction has been a hot research topic in the past few decades .predicting stock price is regarded a challenging task because stock market is essentially non linear ,non-parametric,noisy,and a chaotic system .Trend of a market depends on many things like liquid money human behavior, news related to stock market etc. All this together controls the behavior of trends in a stock market with the advancement of the computing technology we use machine learning technique,like Support Vector Regression,K-nearest-neighbor,liner Regression Random forest Regressor, for analyzing time series data to predict stock price. In this paper I try to develop a forecasting model with stacking multiple method to find the best forecast of the stock price.***

***‘Keywords— Google stock, , Time Series data, SVR, KNN-Regressor, liner regression,staking regressor,Random forest regression’***

## **1. INTRODUCTION**

The goal is to take time series data, find the equation that best fits the data, and be able forecast out a specific value. Time series data is a continuous data statistical observations recorded

over a specific period of time. This model will try to understand the pattern of the continuous data by combining different method and produce a best fit line that fits the data. The target is to determine the future stock price and improve their strategy for future. regression models are among the most known regression models used in the machine learning community and recently many researchers have examined their sufficiency in ensembles. Although many methods of ensemble design have been proposed, there is as yet no obvious picture of which method is best. One notable successful adoption of ensemble learning is the distributed scenario. In this work, we propose an efficient distributed method that uses the same training set with the parallel usage of an averaging methodology that combines linear regression and KNN regression models,Support Vector Regression,random Forest Regresssion. We performed a comparison of the presented ensemble with other ensembles that use either the linear regression as base learner and the performance of the proposed method was better in most cases. Using averaging methodology, we expect to obtain better results because both theory and experiments show that averaging helps most if the errors in the individual regression models are not positively correlated.

**linear regression** is a linear approach to modelling the relationship between a scalar response dependent variable and one or more explanatory variables (or independent variables). The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called **multiple linear regression**

**support vector machines** are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model by creating a hyper plane that assigns new examples to one category or the other . in support vector regression (SVR). The model produced by support vector classification (as described above) depends only on a subset of the training data, because the cost function for building the model does not care about training points that lie beyond the margin.

In pattern recognition, the ***k*-nearest neighbors algorithm** (***k*-NN**) is a non-parametric method used for classification and regression. In both cases, the input consists of the *k* closest training examples in the feature space. The output depends on whether *k*-NN is used for classification or regression:

in case of the knn regression the output is the property value for the object. This value is the average of the values of its *k* nearest neighbors.a useful technique can be used to assign weight to the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones

A hybrid model that combines KNN regression,Linear regression,Support Vector Regression, Random Forest regression model

used for predicting stock prices can forecast better price accuracy.

The paper is structured as follows. Section I describes various machine learning model. Section II

Section 2 presents the most well-known methods for building ensembles .Section 3 contains Proposed methodology.

## **2 Ensembles of Regression Models**

1) Bagging is a ''bootstrap" ensemble method that creates indivrated by randomly drawing, with replacement, N examples - where N is the size of the original set; many of the original examples may be repeated in the resulting training set while others may be left out. After the construction of several regression models, averaging the

predictions of each regression model performs the final prediction . Instability (responsiveness to changes in the training data) is a prerequisite for bagging to be effective.

2)Another method that uses different subset of training data with a single data mining method is the boosting approach . Boosting is similar in overall structure to bagging, except that it keeps track of the performance of the learning algorithm

and concentrates on instances that have not been correctly learned. Instead of choosing the t training instances randomly using a uniform distribution, it chooses the training instances in such a manner as to favor the instances that have not been accurately learned. After several cycles, the prediction is performed by taking a weighted average of the predictions of each regression model, with the weights being proportional to each regression model's performance on its training set. Additive

Regression is a practical version of the boosting approach

3)Another approach for building ensembles of regression models is to use a variety of learning algorithms on all of the training data and combine their predictions. When multiple regression models are combined using averaging methodology, we expect to obtain good results based on the belief that the majority of experts are more likely to be correct in their decision when they are close in their opinions Stacked generalization or Stacking, is a more sophisticated approach for combining predictions of different learning algorithms. Stacking combines multiple regression models to induce a higher-level regression model with improved performance. In detail, the original data set constitutes the level zero data and all the

base regression models run at this level. The level one data are the outputs of the base regression models. A learning algorithm is then used to determine how the iduals for its ensemble by training each regression model on a random redistribution of the training set. Each regression model's training set is generated by randomly drawing, with replacement, N examples - where N is the size of the original set; many of the accurately learned. After several cycles, the prediction is performed by taking a weighted average of the predictions of each regression model, with the weights being proportional to each regression model's performance on its training set.

# **Proposed Methodology**

Stacking is concerned with combining multiple

classifiers generated by different learning algorithms

L1 , ... , LN on a single dataset S, which is composed by

a feature vector s i = (x i , y i ).

• The stacking process can be broken into two phases:

1. Generate a set of base-level classifiers C1 , ... , CN

• Where C i = L i (S)

2. Train a meta-level classifier to combine base level classifier

The training set for the meta-level classifier is generatedthrough a leave-one-out cross validation process.

∀i = 1, ... n and

• The learned classifiers are then used to generate

predictions for

• The meta-level dataset consists of examples of the form where the features are the predictions of the base-level classifiers and the class is the correct class of the example in hand.

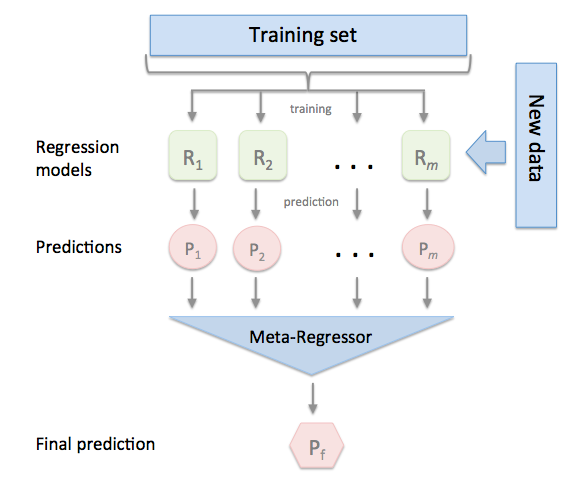


Figure 1: System Architecture

**ARCHITECTURE OVERVIEW:**

The prediction system has a two tier architecture. The top tier is dedicated to preparing the data sets from multiple information sources to make them ready for the predication tasks in the next tier. It is composed of two major parts. The first part is data prepossessing . In this process we process the data by adding more feature and removing unnecessary feature and removing the bad data and also the absence of the data . The second part is the data alignment. The second tier is dedicated to the market volatility analysis and prediction through the model integration and training, which uses multiple kernel learning methodology to train the model It consists of three tasks: First, we build one regression model per source. Second, we train the model with the same data sets ,then we and find the accuracy by cross validation of the result and finally we perform the comparison between the those result of different model to find the best prediction model

In this paper, we use the multi-kernel learning method .

**A. Data Prepossessing:**

In the real world, many data sets are very messy. Most stock price/volume data is pretty clean, rarely with missing data, but many datasets will have a lot of missing data. filter out other unimportant feature from the feature because not all the feature will be included into the final feature list .The reason behind it is the unnecessary feature and those value which has no relation with the stock market prediction will reduce the accuracy of the prediction

**B. Feature Extraction:**

We only work with the valuable feature , created some new valuable feature through manipulation.

Right feature can help to get more accurate results.

**C .Training The ML Model:**

Now comes the training and testing. The way this works is we take, 75% of our data, and use this to train the machine learning classifier. Then we take the remaining 25% of our data, and test the classifier. Since this is our sample data, we should have the features and known labels. Thus, if we test on the last 25% of our data, we can get a sort of accuracy and reliability,

**C . Finding The Best Method**

we estimate the best method by evaluating the accuracy of the Ml method using cross validation

and comparing them to find the best method for this specfic kind of datasets.

**C . Predict The Feuture Value**

we use the best selected method to predict the future closing price of the stock market

# **Experiment Setup:**

In this experiment we use the following softwere

**1.python**

**2.Numpy**

**3.Scikit-learn**

**4.Pandas**

**5.Matplotlib**

**Python** is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991,

**NumPy** is a library for the Python programming language, adding support for large, multi- dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

**Scikit-learn** (formerly scikits.learn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy.

**Pandas** is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license

**Matplotlib** is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits

**Datasets:**

We'll use the data of the stock price for Alphabet (previously Google).We collect the data from the Quandl. (**Quandl** is a platform for financial, economic, and alternative data that serves investment professionals).Quandl's sources include open data from providers such as the UN, Worldbank.Google.Yahoo and central banks core financial data.

# **Result Analysis**

ARIMA: In figure 2 where horizontal axis represents month and vertical axis represents monthly sales. By analyzing the figure 3, it can be concluded that ARIMA model has successfully covered most linear component of data set but seasonal component hasn’t cover successfully.

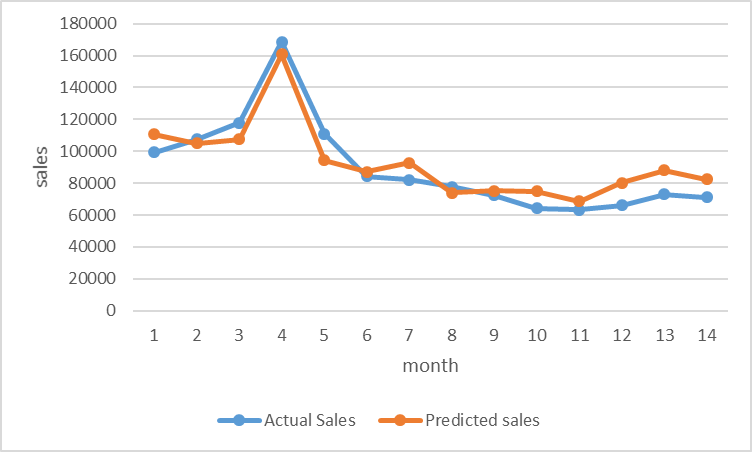


Figure 2: Actual vs ARIMA prediction

Fb-Prophet: In figure 3 where horizontal axis represents month and vertical axis represents monthly sales. By analyzing the figure 3, it can be concluded that Fb-Prophet has successfully covered linear and on-linear component.

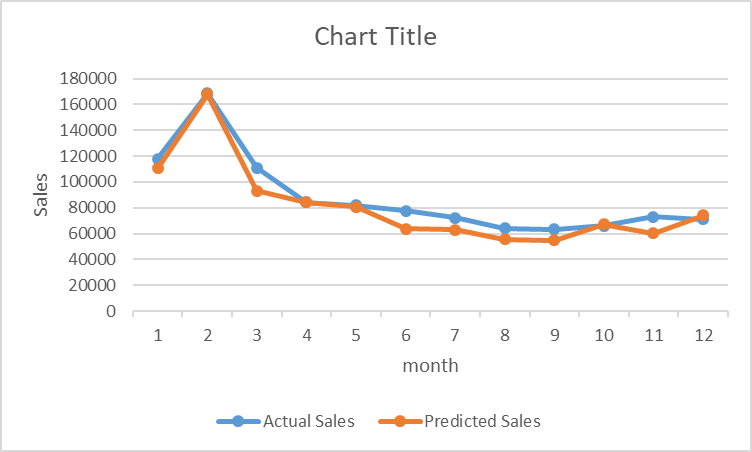


Figure 3: Actual vs Fb-Prophet prediction

|  |  |  |  |
| --- | --- | --- | --- |
| Performance measure | RMSE | MAE | MAPE |
| ARIMA | 10153 | 9033.428571 | 10.844% |
| Fb-Prophet | 19036 | 12861.250000 | 12.623% |

Table: Performance Measure of ARIM and Fb-Prophet

Here, We get better RMSE value for ARIMA as tested dataset has large trendy component and less seasonal component as AIRMA works better for trendy data.

**Conclusions**

It is known that if we are only concerned for the best possible correlation coefficient,

it might be difficult or impossible to find a single regression model that performs as well as a good ensemble of regression models. In this study, we built an ensemble of

regression models using four different learning methods: the Linear Regression and

the KNN regresssion ,Support Vector regression,Random Forest regression

# **reference**

1.C.L. Blake, C.J. Merz, UCI Repository of machine learning databases. Irvine, CA: University of California, Department of Information and Computer Science (1998). [http://www.ics.uci.edu/~mleam/MLRepositorv.htm

2.L. Breiman, Bagging Predictors. Machine Learning, 24(3) (1996) 123-140.

3.L. Breiman, Stacked Regression. Machine Learning, 24 (1996):49-64.

4.T.G. Dietterich, Ensemble methods in machine learning. In Kittler, J., Roli, F., eds.:

Multiple Classifier Systems. LNCS Vol. 1857, Springer (2001) 1-15

5.N. Duffy, D. Helmbold, Boosting Methods for Regression, Machine Learning, 47, (2002)

153-200.

6. J. Fox, Applied Regression Analysis, Linear Models, and Related Methods, ISBN:

080394540X, Sage Pubns (1997).

7.J. Friedman, Stochastic Gradient Boosting, Computational Statistics and Data Analysis

38 (2002) 367-378.

8. Y. Grandvalet, Bagging Equalizes Influence, Machine Learning, Volume 55(3) (2004)

251-270.

9. N.L. Hjort, G. Claeskens, Frequentist Model Average Estimators, Journal of the

American Statistical Association, 98 (2003) 879-899.

10.Y. Morimoto, H. Ishii, S. Morishita, Efficient Construction of Regression Trees with

Range and Region Splitting, Machine Learning, 45(3) (2001) 235-259.

11.D. Opitz, R. Maclin, Popular Ensemble Methods: An Empirical Study, Artificial

Intelligence Research, 11 (1999): 169-198, Morgan Kaufmann.

12.C. Perhch, F. J. Provost, J. S.

Simonoff,Tree Induction vs. Logistic Regression: A

Learning-Curve Analysis. Journal of

MachineLearning Research 4 (2003) 211-255

13.Y. Wang, I. H. Witten, Induction of model trees for predicting continuous classes, In

Proc.of the Poster Papers of the European Conference on ML, (1997) 128-137.

14.I. Witten, E. Frank, DataMining:Practical Machine Learning Tools and Techniques with

Java Implementations, Morgan Kaufmann, San Mateo (2000)