Gold Price Prediction Using Machine Learning

Mohit Agarwal K (19BEC073)
Dept. of Electronics and Communication Engineering
Institute of Technology,Nirma University
Ahmedabd,India
19bec073@nirmauni.ac.in

Himanshu Nakrani (19BEC075)
Dept. of Electronics and Communication Engineering
Institute of Technology,Nirma University
Ahmedabad, India
19bec075@nirmauni.ac.in

***Abstract*—gold is among the most precious metal on the earth. predicting gold price is very useful for investing and buying gold. in this paper gold price prediction using regression is discussed. different regressor like linear, random forest, SVR, extra tree, bayesian, decision tree, etc, has been aplied. among those extra tree regressor gives the highly accurate model.**

***Index Terms*—bayesian ridge, decision tree, elastic net, extra tree, gradient boosting, huber, KNeighbors, random forest, regression, SVM, XGBoost**

I. INTRODUCTION

Investing in gold has evolved throughout time, whether through traditional techniques such as purchasing jewellery or modern ones such as purchasing gold coins and bars (which are already available at scheduled banks) or investing in a Gold Exchange traded fund (Gold ETF). A Gold ETF is an exchange-traded fund (ETF) that tracks the price of gold on the market. Each unit of the Gold ETF is equal to one gramme of gold, and it is backed by genuine gold of the highest quality. Due to its rising value and wide range of applications, gold is another asset that many investors are considering as a viable investment option.

Due to their pessimistic views about the situation in the established foreign currency markets and capital markets, investors' appetite for gold as a defensive asset is increasing. Gold is also known as "the asset of last resort," meaning that it is the asset that investors turn to when the developed world's financial markets fail to deliver satisfactory returns.

This rising gold price, along with the volatility and price declines in other areas such as capital markets and real estate markets, has drawn an increasing number of investors to gold as a viable investment. However, the price of gold has recently seen tremendous volatility, making gold investments riskier.

II. LITERATURE REVIEW

Diamond Price Prediction using Machine Learning.[1] from this paper we got to know about different regressiom algorithms and we applied it in our paper. Gold Price Data — Kaggle[2] we have taken dataset from this refrerence and we also studied other people's work shared there and then we compared it with our proposed methods.

by reviewing some paper and kaggle notebooks we found that majority people are using random forest as regressor model to predict gold-price because it gives high accuracy. our propoced method is to apply extra tree regression, at it gives very more accuracy than random forest also. and we have proved it further in this paper.

III. BACKGROUND

in this section we have discussed the different regression model we used in this paper.

A. *Linear regression*

Linear Regression is a supervised learning-based machine learning technique. It performs the function of regression. Regression is a method of modeling where a target value using independent predictors. This method is frequently used to forecast and determine the link between cause and effect factors. The amount of independent variables and the type of relationship between the independent and dependent variables are the primary differences between regression techniques. A useful way for evaluating which factors impact a topic of interest is regression analysis. The regression procedure allows you to accurately establish which elements are most important, which factors may be ignored, and how those factors interact. Linear regression uses following equations to predict the value:

single variable:$y = mx + c$ .where, x and y are independent variable and dependent variable respectively.m is the slope and c is the bias.

Multiple variable:$y = m_1x_1+m_2x_2+m_3x_3+\ldots+m_nx_n+c$ .where, $x_1, x_2, x_3 ... and x_n and y_1, y_2, y_3 ... and y_n$ are independent variable and dependent variable respectively. m is the slope and c is the bias.

B. *Gradient boosting regression:*

A prominent boosting technique is gradient boosting. Each predictor in gradient boosting corrects the mistake of its predecessor. Unlike Adaboost, the training instance weights are not adjusted; instead, each predictor is trained using the predecessor's residual errors as labels. CART is the foundation learner in a method called Gradient Boosted Trees (Classification and Regression Trees). Shrinkage is a crucial characteristic to consider while using this approach.After each tree in the ensemble is multiplied by the learning rate (eta), which varies from 0 to 1, the prediction of each tree in the ensemble is decreased. There is a trade-off between eta and

the number of estimators; in order to achieve a specific model performance, a lower learning rate must be compensated by a higher number of estimators. Predictions may now be made because all trees have been taught. Each tree predicts a label, with the formula providing the final prediction.

$$y(pred) = y1 + (eta * r1) + (eta * r2) + ... + (eta * rN)$$

GradientBoostingRegressor is the scikit-learn class for gradient boosting regression.GradientBoostingClassifier is a classification algorithm that uses a similar approach.

C. *Bayesian Ridge regression (Bayesian linear regression)*

By expressing linear regression using probability distributions rather than point estimates, Bayesian regression permits a natural mechanism to survive limited or poorly distributed data. Instead of being estimated as a single value, the output or response 'y' is supposed to be chosen from a probability distribution.To create a completely probabilistic model mathematically, the answer y is considered to be Gaussian distributed around $X_w X_a s$ follows.

$$p(y|X, w,) = N(y|Xw,)$$

Bayesian Ridge regression, which calculates a probabilistic model of the regression issue, is one of the most effective types of Bayesian regression. The antecedent for the coefficient w is given as follows by spherical Gaussian:

$$p(w|) = N(w|0, 1Ip)$$

Bayesian Ridge Regression is the name of the generated model, which can be found in scikit-learn as sklearn.linear model. Bayesian Ridge Regression is performed using the BeyesianRidge module.

D. *Support Vector Machine Regression*

With a few minor modifications, the Support Vector Regression (SVR) employs the same concepts as the SVM for classification. n the case of regression, a margin of tolerance (epsilon) is provided as an estimated estimate to the SVM that the issue would have already requested. Aside from that, there is a more difficult reason: the algorithm is more sophisticated, thus it must be taken into account. The underlying principle, however, remains the same: decrease error by customising the hyperplane to maximise the margin while keeping in mind the constraints. SVR is characterised as an optimization problem in which the flattest tube that comprises the bulk of the training instances is determined after first generating a convex-insensitive loss function to be minimised. As a consequence, a multi objective function is formed by combining the loss function with the geometrical characteristics of the tube. The most influential cases that determine the shape of the tube in a supervised-learning environment are the support vectors, and the training and test data are assumed to be independent and identically distributed (iid), obtained from the same fixed but unknown probability data distribution function.

Kernel: Convert a lower-dimensional data set to a higher-dimensional data set with this function. This increase in dimension is required when we are unable to locate a separating hyperplane in a certain dimension and must shift to a higher dimension.

Boundary line: A decision boundary line can be thought of as a demarcation line with positive examples on one side and negative examples on the other (for simplicity). Aside from Hyper Plane, SVM has two lines that form a margin. This line's examples can be classified as either positive or negative. Within or beyond the boundary lines, the support vectors may be found.

Hyperplane: Hyperplane is a term used to describe a plane that is in SVM, this is the line that separates the data classes. In SVR, we will refer to it as a line that will help us forecast a continuous value or a target value.

Support Vector: Here are the data points that are closest to the boundary. The distance between the two places is small to non-existent. In SVR, support vectors are positions outside the -tube. The lower the value of the, the more points outside the tube there are, and hence the more support vectors.

E. *KNN regression*

One of the most widely used machine learning methods is K-Nearest Neighbors. it can be used for both classification and regression problems. The KNN algorithm predicts the values of new data points based on 'feature similarity.' This means that a value is assigned to the new point based on how similar it is to the points in the training set. Steps involved in K-Nearest Neighbors Algorithm regression algorithm is given below:

F. *huber regression*

Huber regression is a sort of robust regression that considers the probability of outliers in a dataset and gives them less weight than other examples. The Huber Regressor optimizes the squared loss for the samples where $|(y - X'w)/sigma| < epsilon$ and the absolute loss for the samples where $|(y - X'w)/sigma| > epsilon$, where w and sigma are parameters to be optimized.

- find out the distance between new data-point to training data-points. We can use any distance measure like Euclidian, Manhattan (for continuous) and Hamming distance (for categorical).
- find out appropriate K value and then select nearest k data-points. We can use 'elbow method' to determine value of K.
- find average of these K data-points. It will be our final prediction.

G. *elastic net regression*

The strategy combines the lasso and ridge regression methods by learning from their flaws to better statistical model regularisation. The penalties from both the lasso and ridge approaches are used to regularise regression models in elastic net linear

regression. The goal of Elastic Net is to reduce the following loss function:

$$L_{enet}(\hat{\beta}) = \frac{\sum_{i=1}^{n}(y_i - x_i'\hat{\beta})^2}{2n} + \lambda(\frac{1-\alpha}{2}\sum_{j=1}^{m}\hat{\beta}_j^2 + \alpha\sum_{j=1}^{m}|\hat{\beta}_j|),$$

The elastic net approach overcomes lasso's limitations, as when high-dimensional data needs just a few samples. The elastic net approach allows "n" variables to be included until saturation is reached. If the variables are highly correlated groups, lasso will usually pick one from each group and ignore the others.

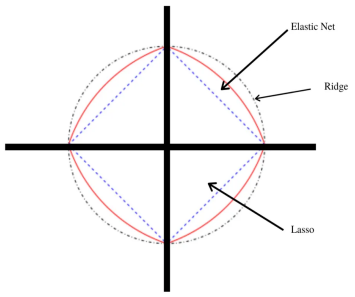

Fig. 1. stastistical model of elastic net

H. Xgboost (Extreme Gradient Boosting) regression

XGBoost is a boosting method that belongs to a group of techniques called boosting algorithms and It is based on the gradient boosting (GBM) framework. Boosting works on the principle of an ensemble. The main idea behind boosting algorithms is to build a weak model, make conclusions about the importance of particular features and parameters, and then use those conclusions to build a new, stronger model that capitalizes on the previous model's misclassification mistake and attempts to reduce it.

I. Regression using neural network

The advantage of Artificial Neural Networks for Regression over Linear Regression is that linear regression can only learn a linear relationship between features and targets and so cannot learn complex non-linear relationships. Other techniques are required to learn the complex non-linear relationship between the features and the objective. Artificial Neural Networks are one of these techniques. Because each layer contains an activation function, Artificial Neural Networks can learn the complex link between features and targets. Artificial Neural Networks are one of the deep learning algorithms that simulate the workings of neurons in the human brain. The neuron is the basic unit of the brain; human nervous system has roughly 86 billion neurons connected by 1014-1015 synapses. Each neuron gets a signal from the synapses and processes it before producing an output. To develop a neural network, this concept was taken from the brain. Each neuron adds biases, applies an activation function, and outputs the results of a dot product between the inputs and weights. A neural layer is formed when a big number of neurons are grouped together to produce a large number of outputs. Finally, a neural network is formed by combining multiple layers. The Input layer, Hidden layers, and Output layer make up Artificial Neural Networks. The number of hidden layers is not limited to one. Each layer is made up of a certain number of neurons. Each of the neurons in each layer will be assigned an Activation Function. The activation function is the one in charge of adding non-linearity into the relationship.

J. Decision tree

One of the most widely used and useful models for supervised learning is the Decision Tree. It may be used to solve both regression and classification problems, albeit the latter is more commonly used. There are three types of nodes in this tree-structured classifier. The Root Node is the first node in the graph, and it represents the complete sample. It may be further divided into nodes. The characteristics of a data collection are represented by the interior nodes, while the decision rules are represented by the branches.Finally, the outcome is represented by the Leaf Nodes. This method is quite beneficial for solving decision-making issues.

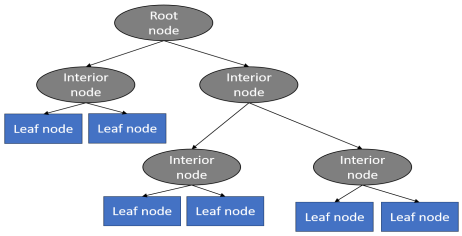

Fig. 2. decision tree algorithm

The core algorithm for building decision trees called ID3 by J. R. Quinlan which employs a top-down, greedy search through the space of possible branches with no backtracking. The ID3 algorithm can be used to construct a decision tree for regression by replacing Information Gain with Standard Deviation Reduction. A decision tree is constructed from the top-downdown, starting with a root node, and includes splitting the data into subsets containing instances with comparable values (homogenous). The homogeneity of a numerical sample is calculated using standard deviation. The standard deviation of a numerical sample that is perfectly homogenous is 0.

Problems with Decision Trees Decision trees are very sensitive to the data used to train them. If the training data is modified, the resultant decision tree and, as a result, the predictions may differ significantly. Also, because they can't go back after a split, decision trees are computationally costly to train, have a high risk of over-fitting, and tend to locate local optima. To address these flaws, we turn to Random Forest, which demonstrates the capability of merging many decision trees into a single model.

K. *Random forest regression*

Random Forest Regression is a supervised learning approach for regression that use the ensemble learning method. The ensemble learning approach combines predictions from several machine learning algorithms to get a more accurate prediction than a single model.

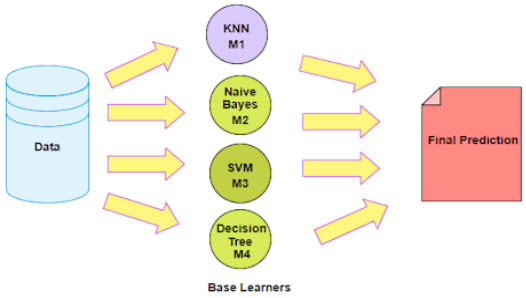

Fig. 3. random forest algorithm

The structure of a Random Forest is seen in the diagram above. the trees run in parallel with no interaction amongst them. During training, a Random Forest constructs many decision trees and outputs the mean of the classes as the forecast of all the trees.

Random Forest Regression is a strong and precise model. It generally works well on a wide range of situations, including those with non-linear connections. However, there are several drawbacks: there is no interpretability, over-fitting is a possibility, and we must pick the amount of trees to include in the model.

Boosting is a set of algorithms that use weighted averages to make weak learners into stronger learners. Random forest is a bagging technique and not a boosting technique. Bagging (Bootstrap) is the process of randomly selecting a small selection of data from a larger dataset. The trees in random forests are run in parallel. There is no interaction between these trees while building the trees.Random forest regression works by training a large number of decision trees and then calculating the mean prediction (regression) of the individual trees.

Feature and Advantages of Random Forest :

- It's one of the most precise learning algorithms available. It generates a very accurate classifiers and regressors for various data sets.
- It can handle tens of thousands of input variables without deleting any of them.
- It runs efficiently on large databases.

Disadvantages of Random Forest :

- For some datasets with noisy classification/regression tasks, random forests have been found to over-fit.
- Random forests are biassed in favour of qualities with more levels when data includes categorical variables with differing number of levels. As a result, random forest variable significance ratings are unreliable for this sort of data.

L. *extra tree(extremely randomized tree) regression*

The Extra Tree Regression (ETR) approach is a developed approach derived originally from Random Forest (RF) model and suggested by Geurts et al. Extra Trees is also an ensemble machine learning algorithm that combines the predictions from many decision trees. It can often perform as well as or better than the random forest method. The Extra Trees approach uses the training dataset to generate a huge number of unpruned decision trees. In the case of regression, predictions are formed by averaging the prediction of the decision trees, whereas, in the case of classification, majority voting is used. There are two major differences between extra tree regression(ETR) and random forest. First, Random Forest chooses the best possible split while Extra Trees chooses it randomly. Once the split points are chosen, the two algorithms determine which of the subsets of features is the best. As a result, Extra Trees incorporates randomization while maintaining optimization. Second, to avoid bias, it cultivates the trees using the whole learning samples.

Dataset We used the gold price dataset available on the kaggle website. This dataset has 6 features and 2290 data rows. In this dataset price of gold and various stocks are given between 1/2/2008 and 5/16/2018. dataset is described in figure below.

	SPX	GLD	USO	SLV	EUR/USD
count	2290.000000	2290.000000	2290.000000	2290.000000	2290.000000
mean	1654.315776	122.732875	31.842221	20.084997	1.283653
std	519.111540	23.283346	19.523517	7.092566	0.131547
min	676.530029	70.000000	7.960000	8.850000	1.039047
25%	1239.874969	109.725000	14.380000	15.570000	1.171313
50%	1551.434998	120.580002	33.869999	17.268500	1.303297
75%	2073.010070	132.840004	37.827501	22.882500	1.369971
max	2872.870117	184.589996	117.480003	47.259998	1.598798

	Date	SPX	GLD	USO	SLV	EUR/USD
0	1/2/2008	1447.160034	84.860001	78.470001	15.180	1.471692
1	1/3/2008	1447.160034	85.570000	78.370003	15.285	1.474491
2	1/4/2008	1411.630005	85.129997	77.309998	15.167	1.475492
3	1/7/2008	1416.180054	84.769997	75.500000	15.053	1.468299
4	1/8/2008	1390.189941	86.779999	76.059998	15.590	1.557099

IV. Implementation

in this section we discussed the implementation details as well as the dataset.

R Squared Error: It is also known as the coefficient of determination. This measure indicates how well a model matches a certain dataset. It shows how near the regression line is to the actual data values. The R squared value ranges from

0 to 1, with 0 indicating that the model does not match the data and 1 indicating that the model fits the dataset perfectly.

Simple Linear Regression We applied the simple linear regression between gold price column and one of the stock prices. We got the highest R squared error of 0.7253 when we train the model using the ' SLV ' stock and gold price. In this case In the all other cases we got very poor accuracy. R squared error in different cases is shown in below table.

	feature	R squared error
linear regression univariate	EUR/USD	-0.000110034977
	SPX	0.00063761742367
	USO	0.03899298335418
	SLV	0.72536077323585
	features	R squared error
linear regression 2 variable	uso, eur/usd	0.1121540076161
	spx, eur/usd	0.00075917113598
	slv, eur/usd	0.84425060293417
	features	R squared error
linear regression multivariate	all	0.86578865658692

Fig. 4. R2 score of different linear regression models

Multiple Linear Regression: We trained our model using gold price and all the other four features. We got the R squared error of 0.865788 in this case. Which is very high compared to simple linear regression.

Gradient Boosting Regression: We have got highest of 0.982293283239958 R2 score for the value of $n_estimators$.

	n_estimators	R squared error
Gradient boosting	2	0.30416560186679
	20	0.93559568110726
	200	0.98058829120976
	2000	0.98253104166802
	20000	0.98229328323996
	200000	0.98242223686657

Fig. 5. R2 scores for gradient boosting algorithm

Bayesian Ridge Regression: We changed n_iter and $fit_intercept$ hyper parameters and we got the highest R2 score of 0.865158917255618.

	n_iter	fit_intercept	R squared error
Bayesian ridge	10	TRUE	0.86515891725562
	100	TRUE	0.86515891725562
	1000	TRUE	0.86515891725562
	10000	TRUE	0.86515891725562
	10	FALSE	0.83604411417321
	100	FALSE	0.83604411417321
	1000	FALSE	0.83604411417321
	10000	FALSE	0.83604411417321

Fig. 6. R2 scores for Bayesian ridge algorithm

KNN Regression: We got the highest R2 score of 0.9440157559112641 for the value of K equals to 2. plot of k value versus R2 score is ploted below.

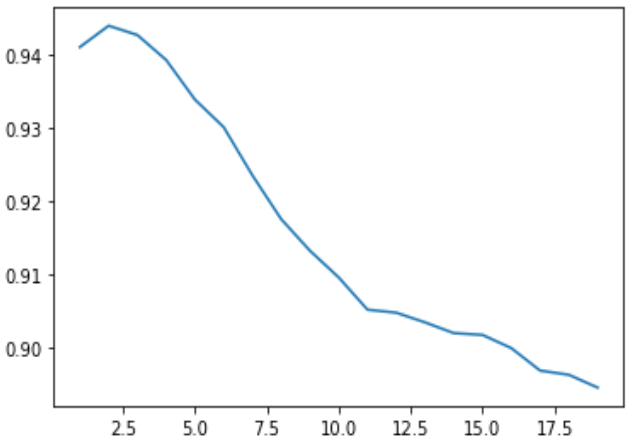

Fig. 7. R2 scores for KNeighbors algorithm

Huber Regression: We changed epsilon hyper parameter and got the highest R2 score of 0.8657885345363798.

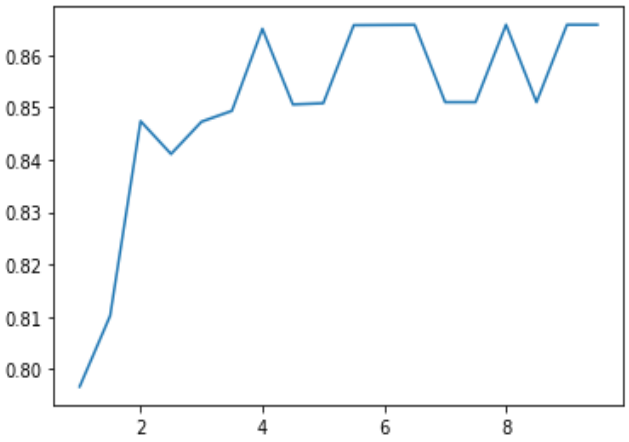

Fig. 8. R2 scores for huber algorithm

Elastic Net Regression: We got 0.8651716512783918 R2 score in this case. For hyper-parameter values, alpha=1, $l1_ratio = 0.01$.

Xgboost Regression For $n_estimators$ value of 1000 we got the good R2 score of 0.9843944737423515.

Regression Using Neural Network: Using neural network we got the descent R2 score of 0.807807137379559 for hidden layer size of 1000.

Decision Tree: In this case we got R2 score of 0.985419029891625, which is quite high compared to previous all moodels.

Random Forest Regression: In random forest we got the highest R2 score of 0.9903789234145185, which is highest after we got in extra tree regression.

	alpha	l1_ratio	R squared error
elastic net	0.1	0.01	0.86431134818797
		0.1	0.864287017987114
		1	0.864120477752759
	1	0.01	0.865171651278392
		0.1	0.865120803870714
		1	0.864493717275619
	10	0.01	0.848807824210922
		0.1	0.850685774297665
		1	0.863683658505521
	100	0.01	0.493266106424458
		0.1	0.490665552673287
		1	0.445722976091722
	1000	0.01	0.083885215928862
		0.1	0.026698628773212
		1	0.00027812015431

Fig. 9. R2 scores for Elastic net algorithm

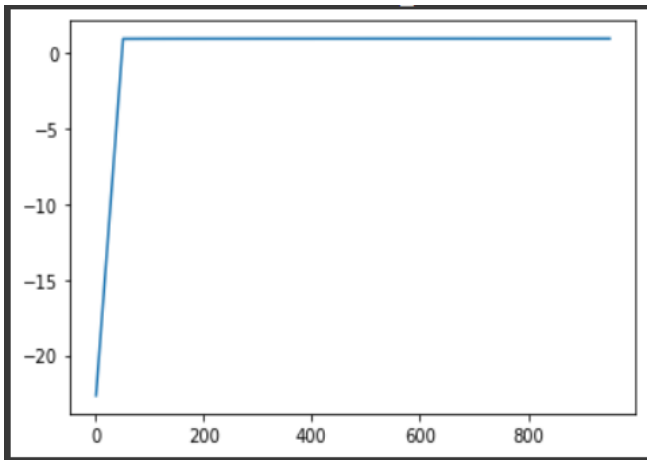

Fig. 10. R2 scores for XGBoost algorithm

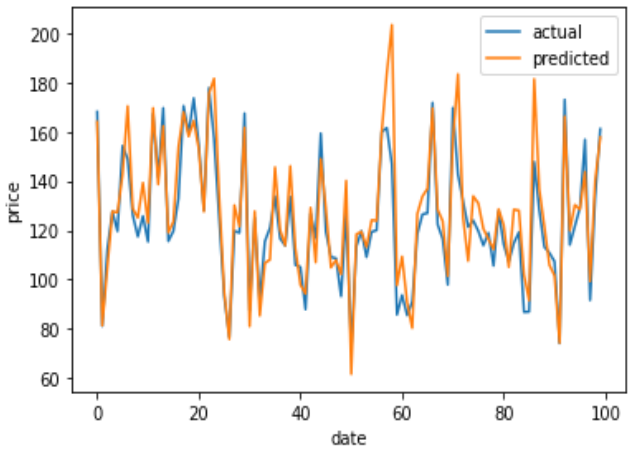

Fig. 11. original value vs predicted value plot neural network algorithm

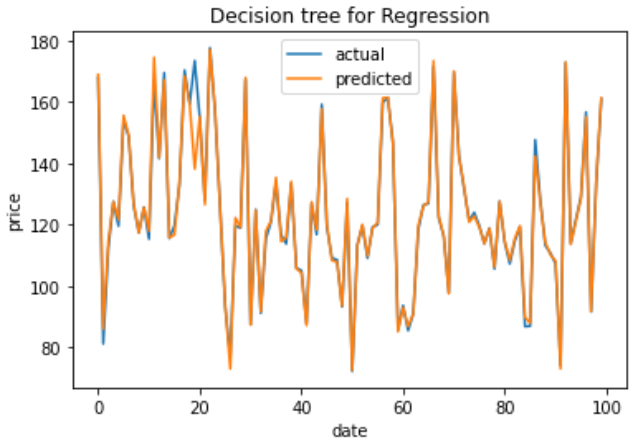

Fig. 12. original value vs predicted value plot decision tree algorithm

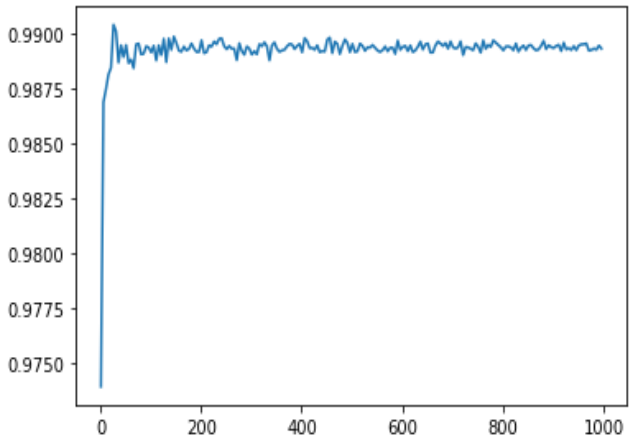

Fig. 13. R2 scores for random forest algorithm

Extra Tree Regression: We got the highest R2 score among all the previous implemented algorithms. We got the highest R2 score of 0.9932789254370804 for value of hyperparamerter $n_estimator$ equals to 451. by image 2 we can see that predicted values is almost overlapping the test values.

V. Conclusion

in this paper we implemented different regression models on gold price dataset. we changed different hyperparameters of every regressor and tried to find optimal fit. we found that using extra tree(extremely randomized tree) regressor we can get the very high R2 score of upto 0.9932.

Acknowledgment

This paper was performed under the guidance of proff. Manish Patel and Prof. Ruchi Gajjar of Electronics and Communication Department,Institute of Technology Nirma University. The authors of the paper are thankful to the university and the professors for the guidance in completion of the term paper as a part of the teaching and learning process.

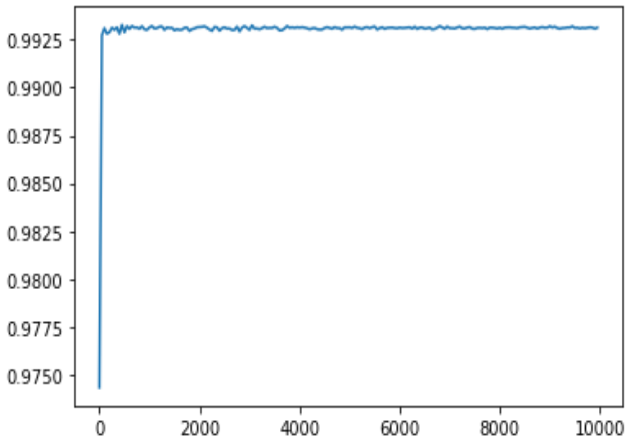

Fig. 14. R2 scores for extra tree algorithm

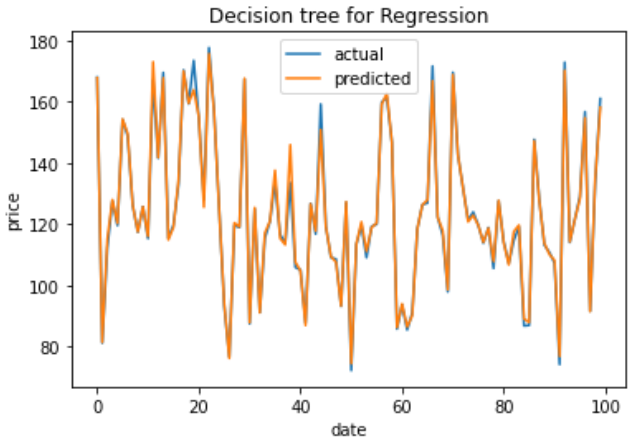

Fig. 15. original value vs predicted value plot extra tree algorithm

REFERENCES

[1] Harshvadan Mihir, Soham Jani, and Manish I. Patel, Ruchi Gajjar. Department of Electronics and Communication Engineering Institute of Technology, Nirma University. Diamond Price Prediction using Machine Learning. 2021 2nd International Conference On Communication, Compution And industry 4.0 DOI: 10.1109/C2I45156.2021.9689412

[2] DEBDATTA CHATTERJEE. "Gold Price Data." Gold Price Data — Kaggle, www.kaggle.com, https://www.kaggle.com/datasets/altruistdelhite04/gold-price-data. Accessed 6 Apr. 2022.

[3] Manjula Ka, Karthikeyan .P, University of Calicut. Gold Price Prediction using Ensemble based Machine Learning Techniques DOI: 10.1109/I-COEI.2019.8862557

[4] Dr. Abhay Kumar Agarwal, Swati Kumari. Department of Computer Science and Engineering, Kamla Nehru Institute of Technology. International Journal of Trend in Scientific Research and Development (IJTSRD)

[5] Z. Ismail, A. Yahya and A. Shabri. Department of Mathematics, Faculty of Science, Department of Basic Education, Faculty of Education,University Technology Malaysia. Forecasting Gold Prices Using Multiple Linear Regression Method. American Journal of Applied Sciences 6 (8): 1509-1514, 2009 ISSN 1546-9239

[6] Selvanathan, E.A., 1991. A note on the accuracy of business economists gold price's forecast. Aust. J. Manage., 16: 91-95. http://www.agsm.edu.au/eajm /9106/pdf/selvanathan.pdf

[7] Mrs. B. Kishori 1, V. Preethi, "Gold Price forecasting using ARIMA Model", International Journal of Research, 2018.