Pattern Recognition And Machine Learning

Project Report

Name	**Rishabh Jain**
Roll number	**B20EE083**
Project Name	**Bitcoin Price Prediction**

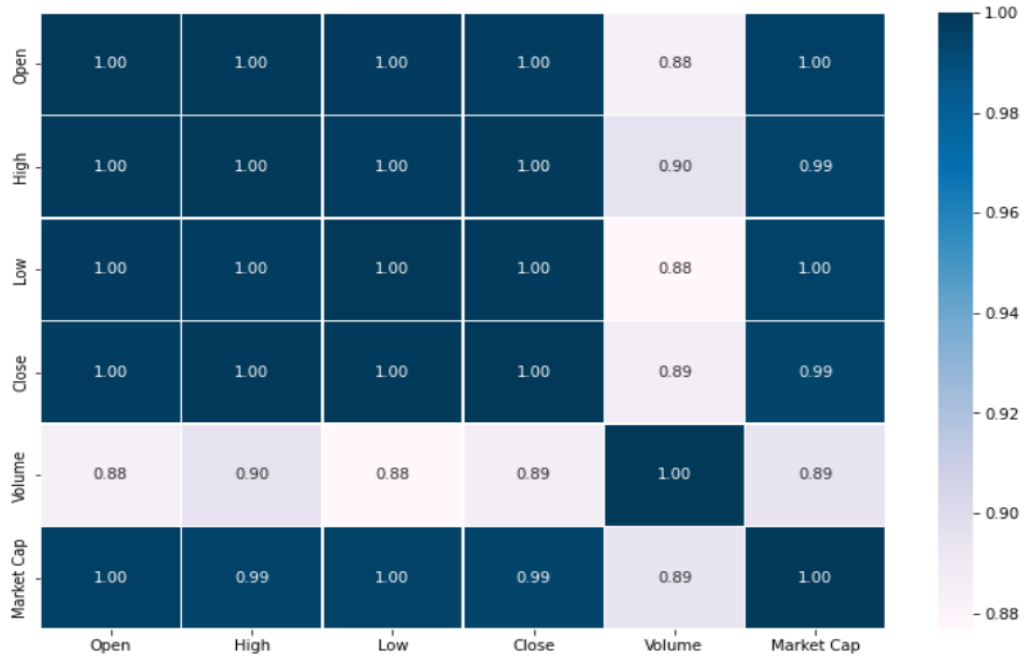
1) **Preprocessing and Visualization→**

➳ **The date column isn't relevant in the supplied dataset, thus we removed it.**

➳ **We eliminated the final few rows since the volume value was not provided.**

➳**The data is in string format in the columns 'Volume' and 'Market Cap.' I fixed that by removing the commas and converting them to integers.**

Correlation Heatmap→



- **If the value is 1**, it is said to be a **positive correlation** between two variables. This means that when one variable increases, the other variable also increases.

2). Regression Models→

◙ Why Regression Models ?

When we want to predict a continuous dependent variable from a set of independent factors, we use regression analysis.

I chose the following Regression model →

① Decision Tree Regressor ②Random Forest Regressor ③

AdaBoostRegressor ④ Linear Regressor ⑤KNeighborsRegressor

① Decision Tree Regressor→ Decision tree regression analyzes an object's characteristics and trains a model in the form of a tree to predict future data and create useful continuous output.

②Random Forest Regressor →A random forest is a meta estimator that employs averaging to increase predicted accuracy and control over-fitting by fitting a number of classification decision trees on various sub-samples of the dataset.

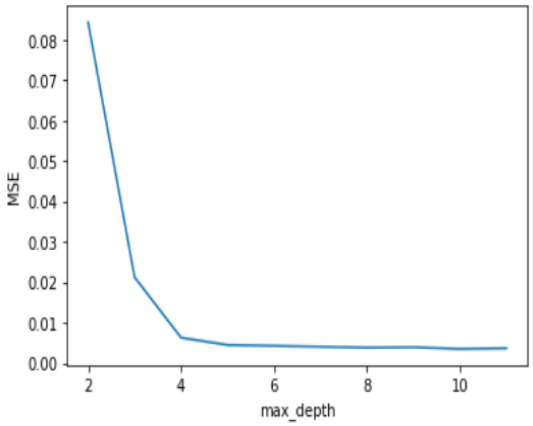
③ AdaBoostRegressor → An AdaBoost regressor is a meta-estimator that starts by fitting a regressor on the original dataset, then fits further copies of the regressor on the same dataset, but with the weights of instances changed based on the current prediction's error.

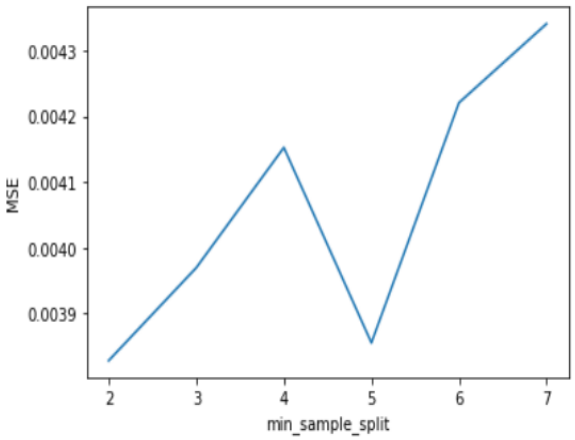
④ Linear Regression → A linear technique to modeling the connection between a scalar response and one or more explanatory factors is known as linear regression (also known as dependent and independent variables).

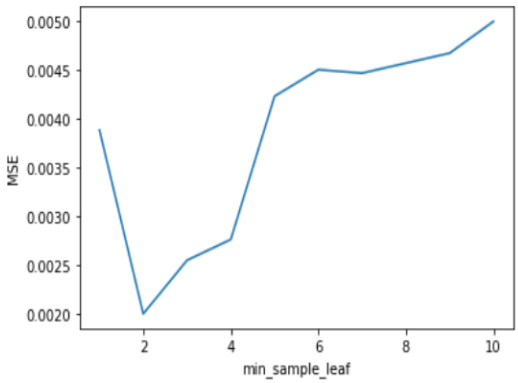
⑤KNeighborsRegressor→ KNN regression is a non-parametric approach that approximates the relationship between independent variables and continuous outcomes by averaging data in the same neighborhood in an understandable manner.

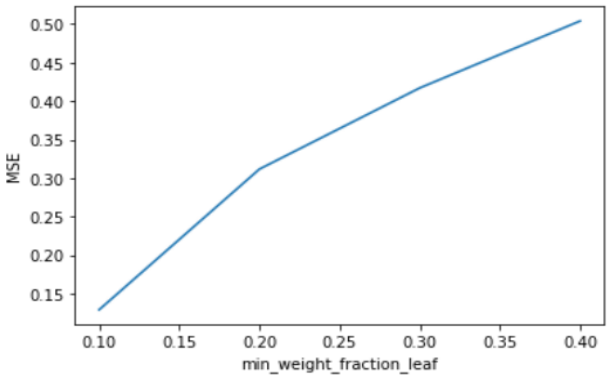
I Varied the following **hyper-parameters** in order to get the better Accuracy →

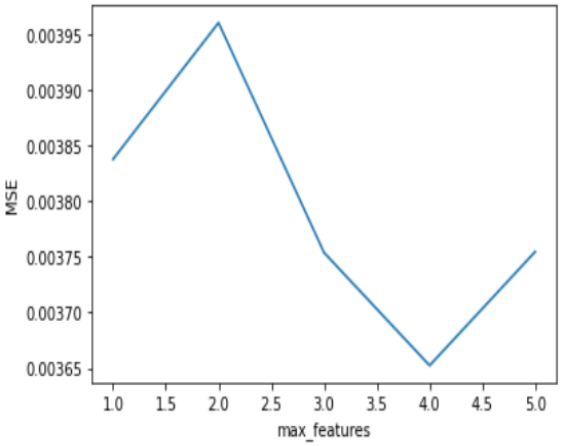
The plot of MSE v/s parameters are shown below.

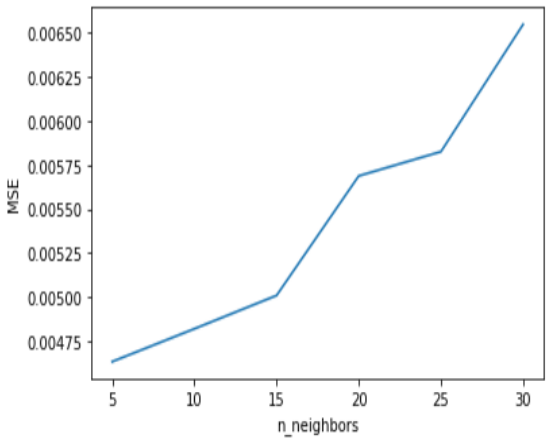

MSE
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
2
4
6
8
10
max_depth

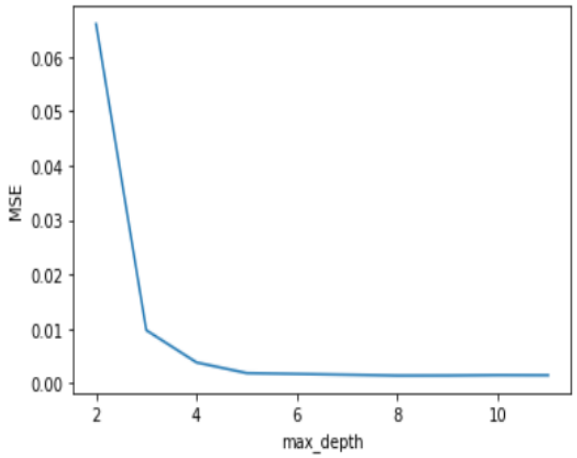

MSE
0.0043
0.0042
0.0041
0.0040
0.0039
2
3
4
5
6
7
min_sample_split

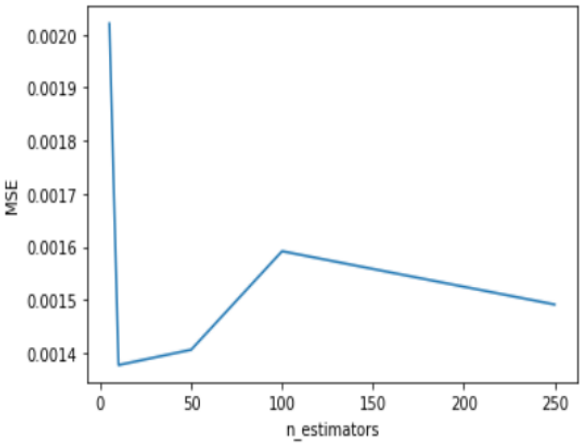

MSE
0.0050
0.0045
0.0040
0.0035
0.0030
0.0025
0.0020
2
4
6
8
10
min_sample_leaf

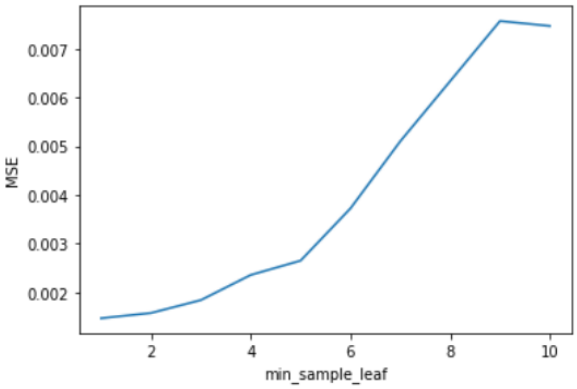

MSE
0.50
0.45
0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.15
0.20
0.25
0.30
0.35
0.40
min_weight_fraction_leaf

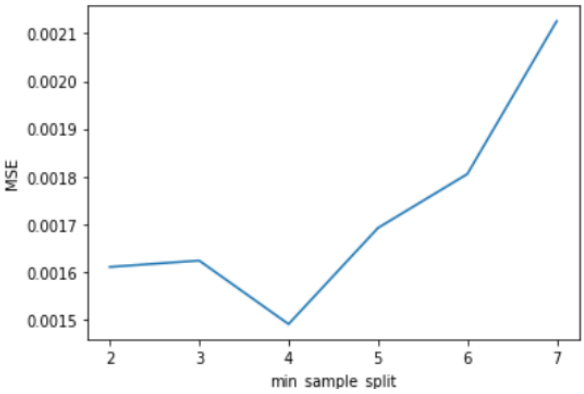

MSE
0.00395
0.00390
0.00385
0.00380
0.00375
0.00370
0.00365
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
max_features

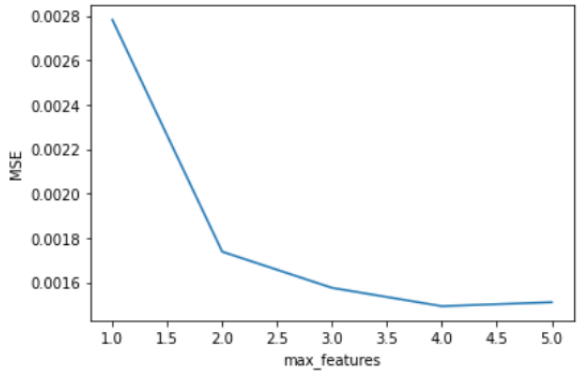

MSE
0.00650
0.00625
0.00600
0.00575
0.00550
0.00525
0.00500
0.00475
5
10
15
20
25
30
n_neighbors

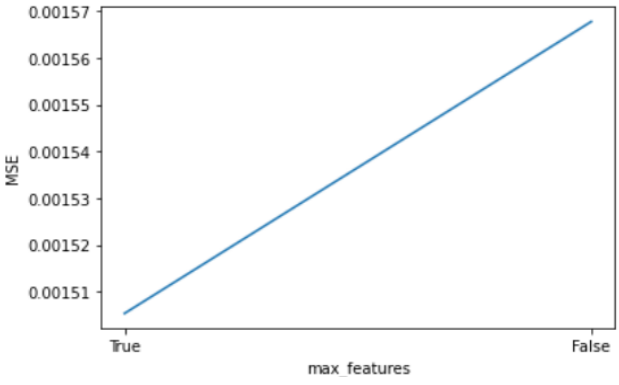

MSE
max_depth

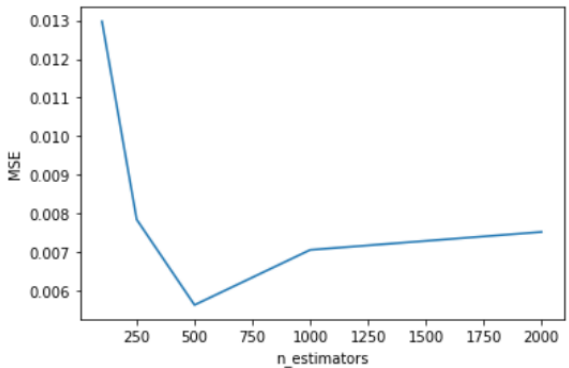

MSE
n_estimators

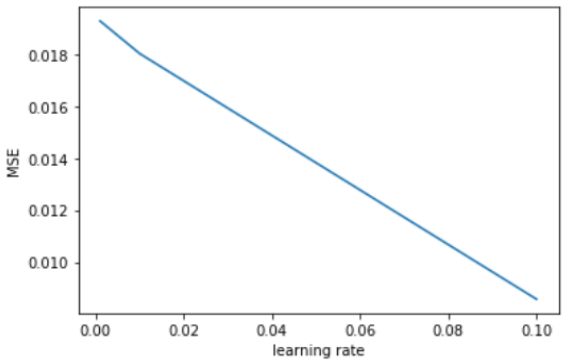

MSE
min_sample_leaf


MSE
min_sample_split


MSE
max_features


MSE
True
False
max_features


MSE
n_estimators


MSE
learning rate

Respective Model and their respective MSE (after Hypertuning)→

Model	MSE
DecisionTreeRegressor	0.103921
RandomForestRegressor	0.00172109
AdaBoostRegressor	0.014899
LinearRegressor	0.000648642
KNeighborsRegressor	0.00393834

Result→ We observed that **LinearRegressor** is performing best for the given dataset.