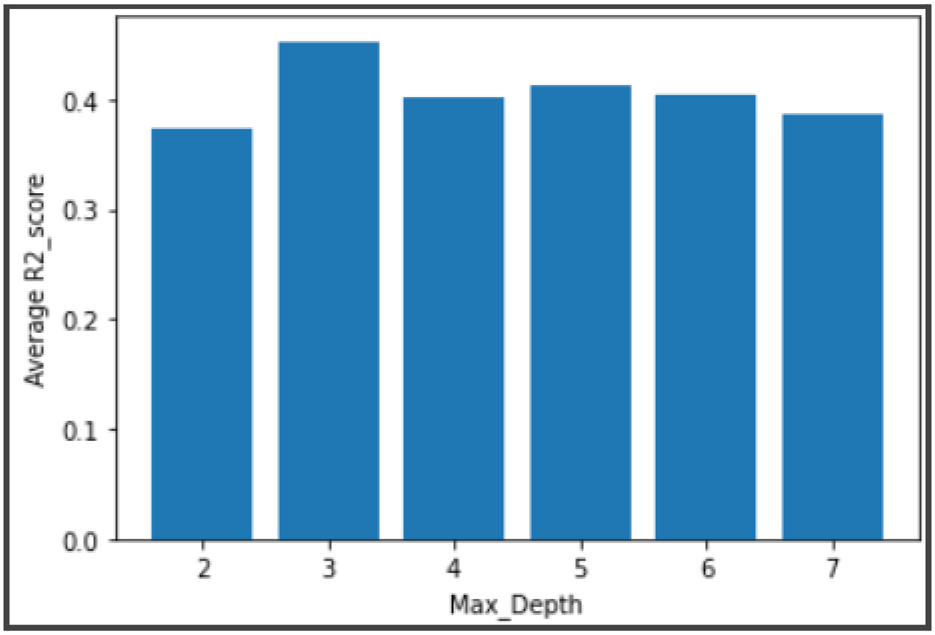
LAB 3 REPORT

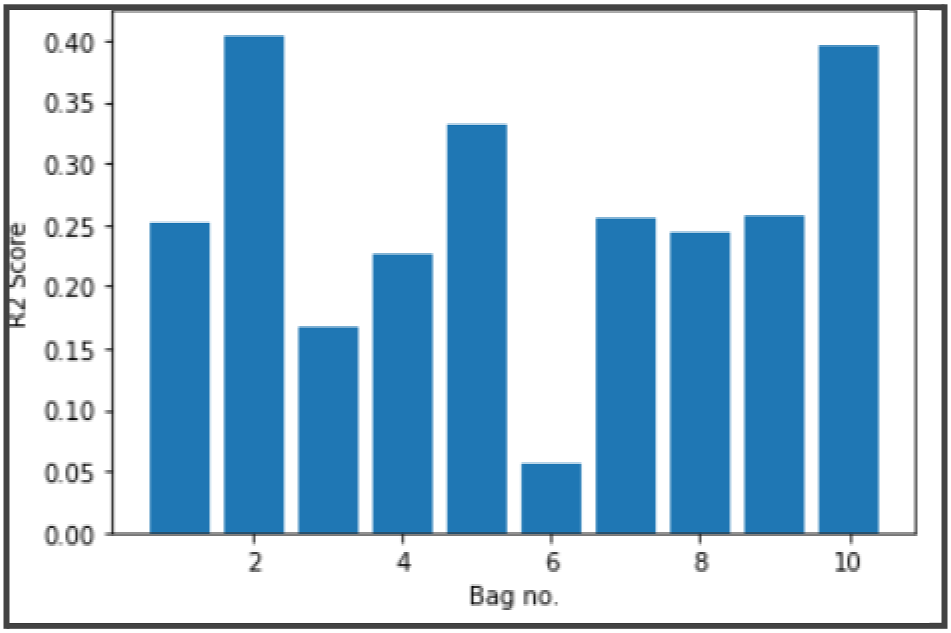
[redacted]

Q1)

- I had pre-processed data that includes dropping rows with NaN values , encoding data and normalizing the data .
- For the first part , I divided the data into a training and testing set in ratio of (75:25) , then trained a decision tree regressor on the training set and calculated the score on the testing set.
- The score achieved on testing data was around 83% to 90%.
- For the second part , the data set was divided into 5 parts for cross validation.
- For calculating cross-validation score average r2_score was used.
- Cross-validation was performed for the decision tree regressor for different values of max_depth to find the best max_depth for the decision tree regressor.
- For the third part , a graph of average r2_score vs max_depth was plotted.



- For bagging , I had created 10 bags (or sub-data) from the complete dataset with the help of data.sample() function. Then, for each bag a decision tree regressor was trained. There performance was calculated by calculating there R2 score on the testing set and plotted on a graph.

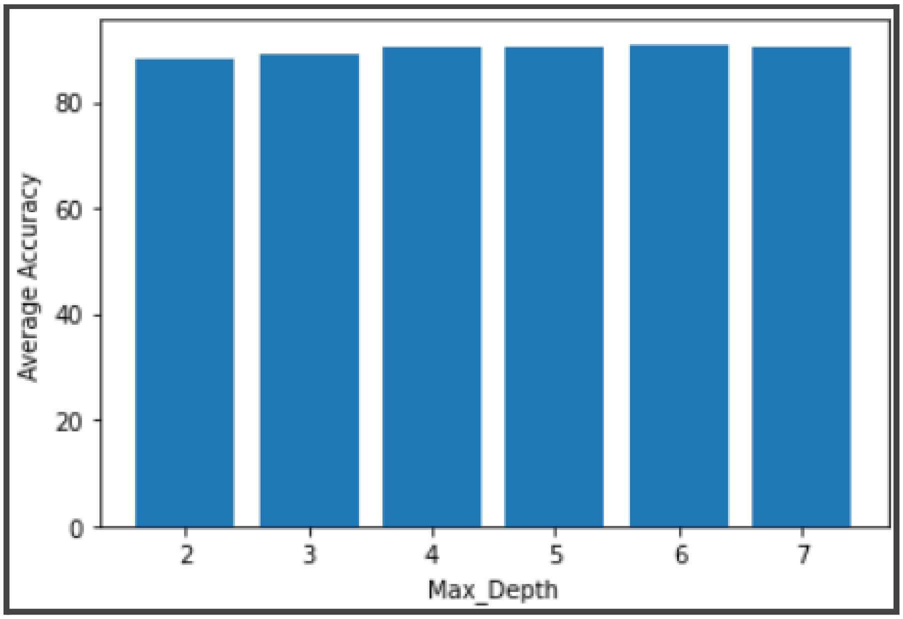


- Also the average performance of all decision tree regressor was calculated by taking the average of there individual R2 score.
- For the 9th Part RandomForestRegressor was made with the help of sklearn.
- The Random Forest Regressor was trained on training data.
- Its mean square error and mean absolute error was calculated on the predicted price on the testing set and its actual price of the testing set.
- For the 10th Part AdaBoost Regressor was made with the help of sklearn.
- The AdaBoost Regressor was trained on training data and its mean square error and mean absolute error was calculated on the predicted price on the testing set and its actual price of the testing set.

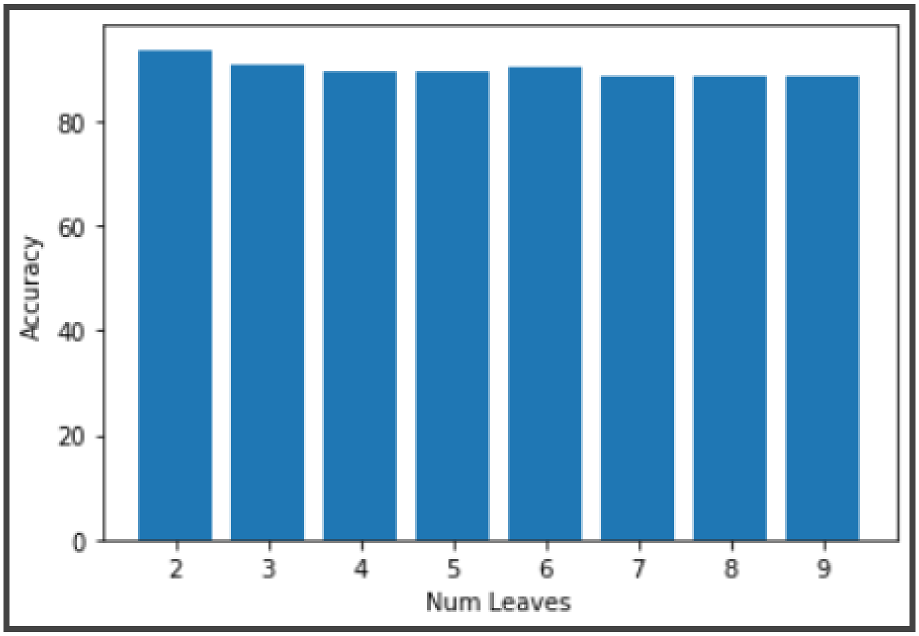
Q2)

- I had pre-processed data that includes dropping rows with NaN values and encoding data .
- The dataset was divided into training sets and testing sets in the ratio 75:25.
- For the 1st part a decision tree classifier was trained on the training set and its accuracy was calculated for its prediction on the testing set.
- The accuracy achieved was around 84% to 93 %.

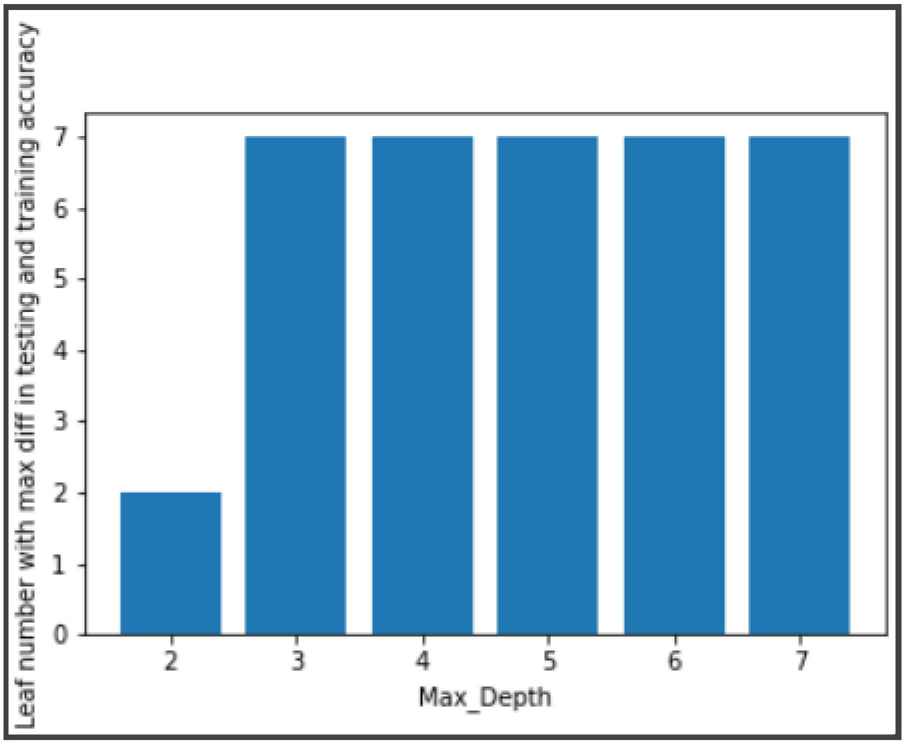
- For the next part , the dataset was divided into 5 disjoint sub-data sets for cross-validation. The average accuracy was used for the cross validation score.
- Cross validation was performed for different values of max_depth to find the best max_depth for the decision tree classifier.
- For the 3rd part a graph was plotted for average accuracy versus max_depth.



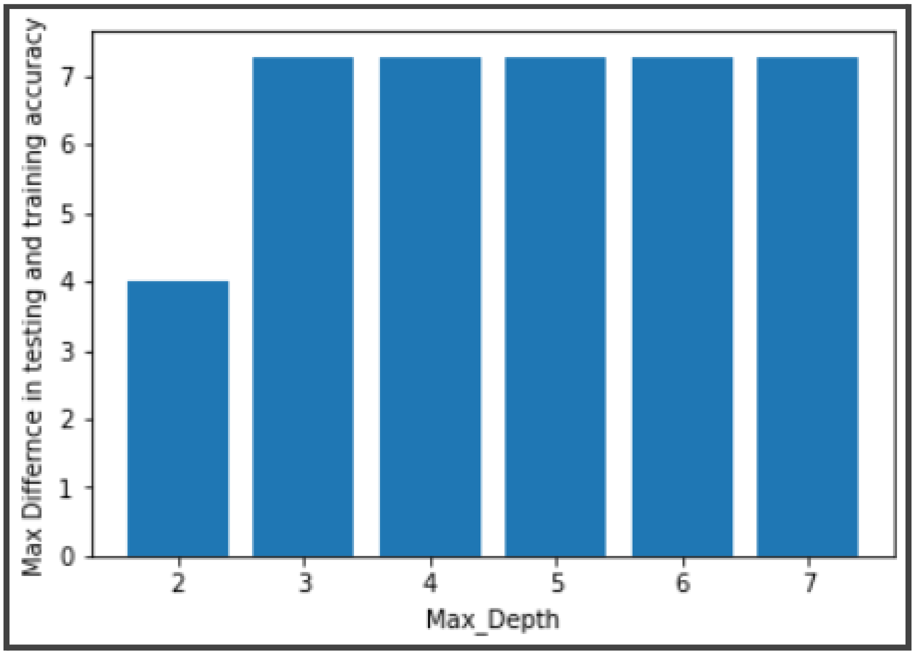
- For the 4th part , a XGBoost model was implemented with the help of sklearn with subsample=0.7 and max_depth=4.
- For the next part, the XGBoost model was trained on the training set and its accuracy was calculated for its prediction on the testing set.
- For the 6th part , LightGBM was implemented with the help of sklearn with max_depth =3 .
- LightGBM was trained on a training set for different values of num_leaves and each time its accuracy was calculated on a testing set.
- A graph of accuracy versus value of num_leaves was plotted.



- For the next part, LightGBM was trained for different values of max_depth.
- For each value of max_depth different values of num_leaves for lightGBM were tried and the num_leave value for which difference between testing accuracy and training accuracy was maximum for the max_depth was noted.
- A graph was plotted between num leave value (for which difference between testing and training accuracy was maximum) and max_depth.



- Another graph was plotted between max difference in testing and training accuracy and max depth.


Max Differnce in testing and training accuracy
Max_Depth