**Given the features and the target, build a Machine Learning Regression model that can predict the price**

**of the diamond from a given set of features.**

**Note 1: Please divide the data set into 70% for training, 20% for validation and keep the rest 10% for testing.**

**Note 2: Upon observation some of the data is of string type(Eg: Color). Since python or scikit learn works with numbers, it will be required to convert the string type of data into numbers. Clue: Check out Label Encoding using Sci-kit Learn**

**You can use any regression algorithm. Please make sure you answer the following are in your report with the**

**code:**

**• Briefly explain the algorithm.**

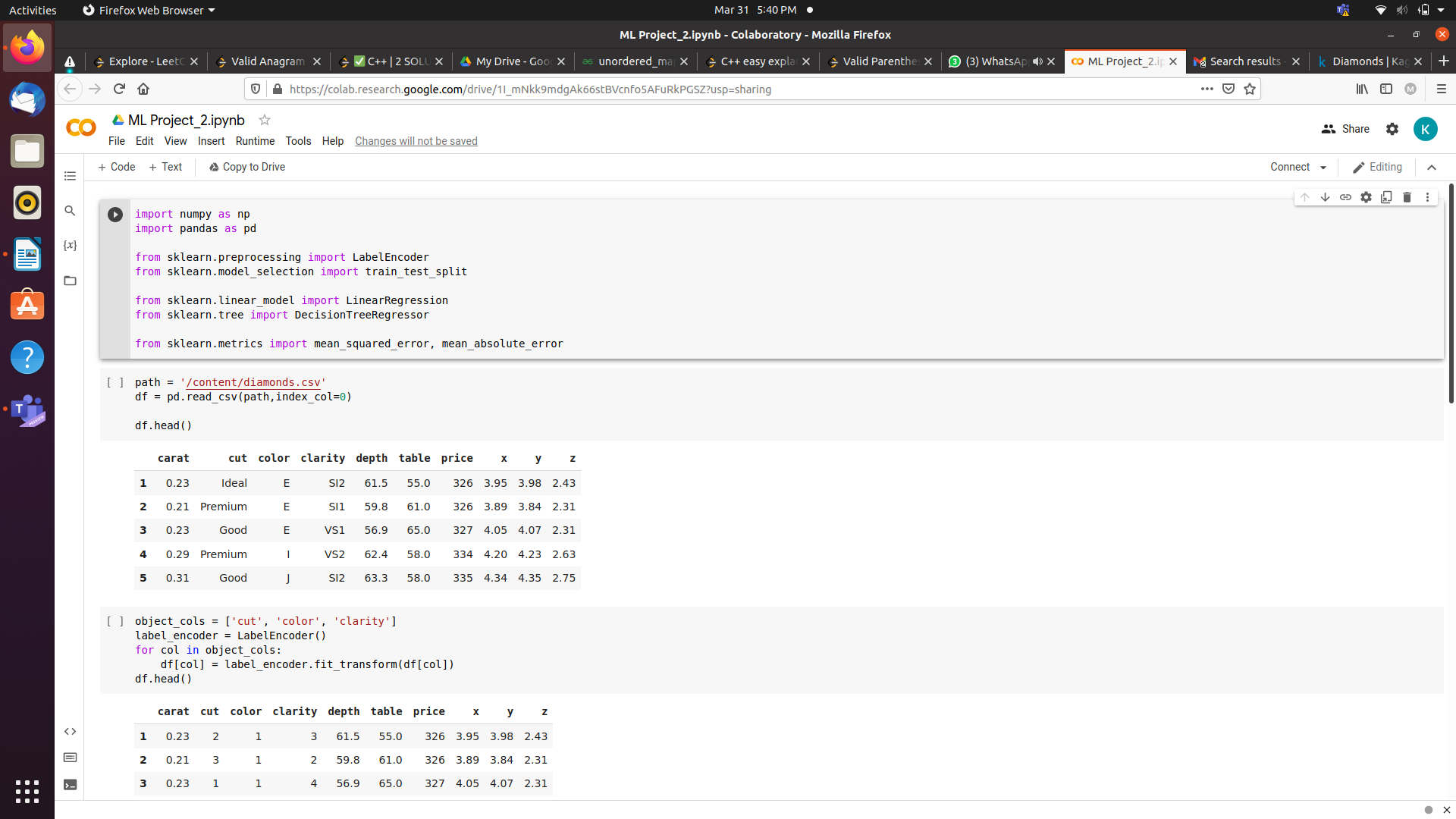
**• What is the Mean Squared Error and Mean Absolute Error obtained?**

**• Observe and note changes in accuracy as you vary parameters. Ex. Number of Nodes in Random Forest**

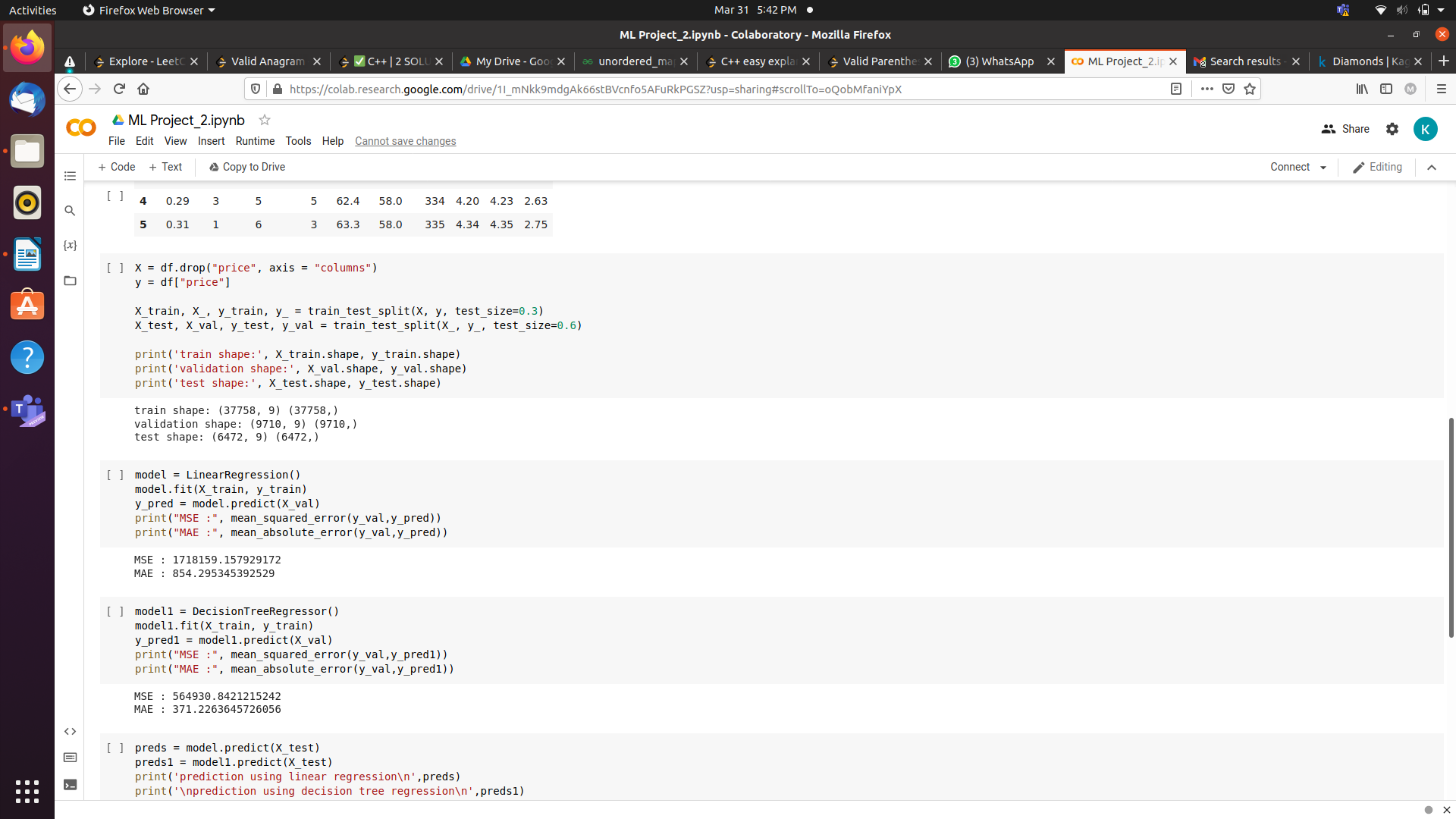
**regressor.**

**CODE:**

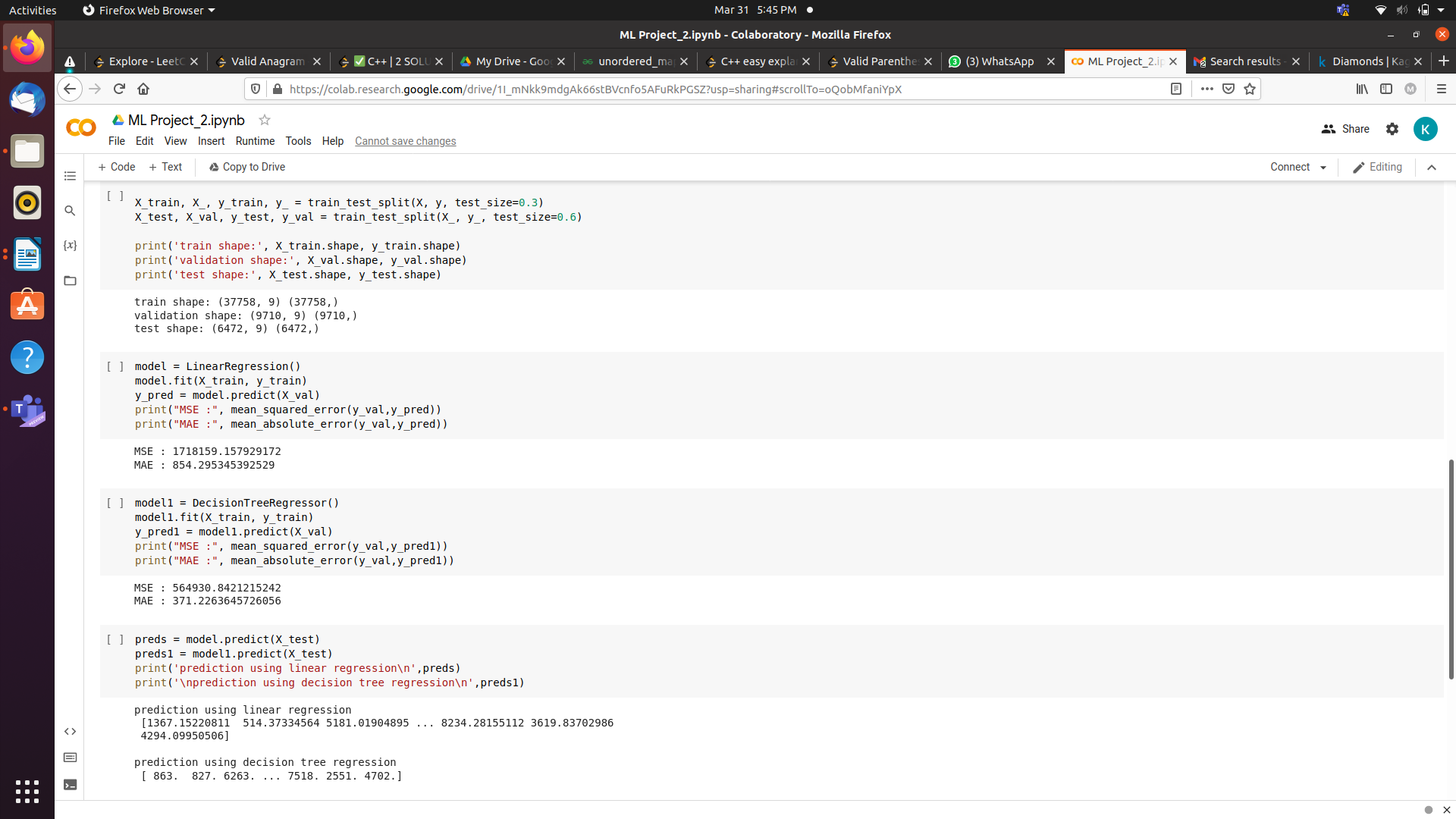
* Converting string type of data (‘cut’,’color’,’clarity’) into numbers using label encoder.



* Division of data set into 70% training ,20% validation,10% testing.



* Comparison of mean squared error and absolute mean error using different models.

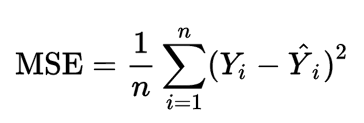


**About the model:**

DECISION TREE REGRESSION:

* A regression tree is basically a decision tree that is used for the task of regression which can be used to predict continuous valued outputs instead of discrete outputs.
* A 1D regression with decision tree.
* The [decision trees](https://scikit-learn.org/stable/modules/tree.html#_blank) is used to fit a sine curve with addition noisy observation. As a result, it learns local linear regressions approximating the sine curve.
* We can see that if the maximum depth of the tree (controlled by the max\_depth parameter) is set too high, the decision trees learn too fine details of the training data and learn from the noise, i.e. they overfit.

**Mean squared error:**

* A measure that tells us how much our predictions deviate from the original target and that’s the entry-point of mean square error.
* 
* **Y** is the actual value and **Y\_hat** is the prediction
* In the Regression Tree algorithm, we do the same thing as the Classification trees. But, we try to reduce the Mean Square Error at each child rather than the entropy.

**Comparison between linear regression and decision tree:**

* Decision trees supports non linearity, where LR supports only linear solutions. When there are large number of features with less data-sets(with low noise), linear regressions may outperform Decision trees/random forests. In general cases, Decision trees will be having better average accuracy

**Results:**

By comparing different models and by tuning MSE is reduced.