**ML:**

LINEAR REGRESSION

Dependent Variable: mpg (miles per gallon)

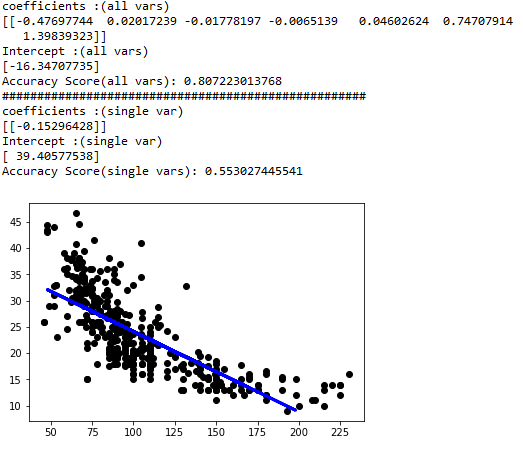
Independent Variables: cylinders (X1), displacement(X2), horsepower(X3), weight(X4), acceleration(X5), model year(X6), origin(X7)

Data Correction: Since there were six missing values from horsepower I replaced those values with mean values of horsepower using rest of the data.

Implementation:

I have used the sklearn package from python to implement regression. First I used all of the independent variables to build a model and performed accuracy check on the test data. Then to get a best-fit line I chose a single variable, horsepower, and built a second model of linear regression. Obviously the accuracy of the first one outperformed the second.

Result Screenshots:

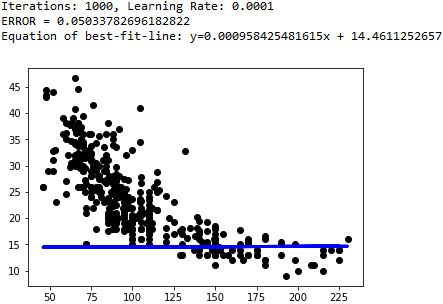


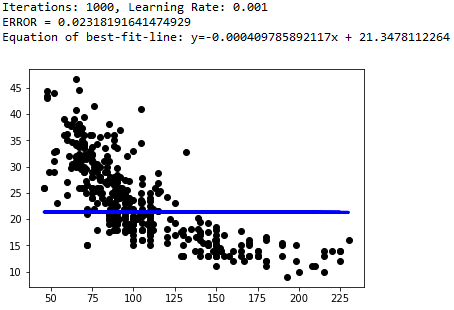
GARDIENT DESCENT

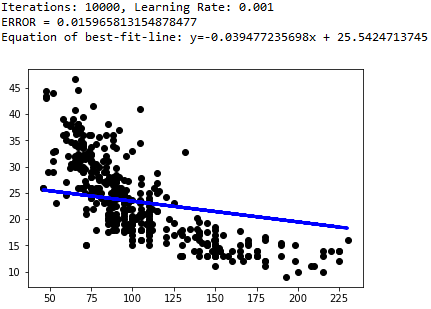
Implementation:

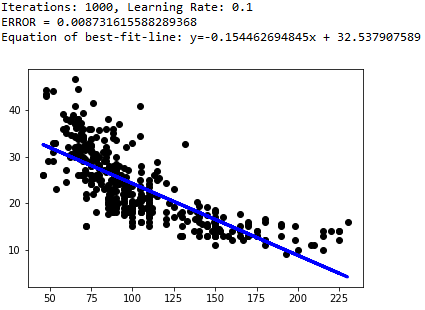
I implemented a single variable gradient descent algorithm that is one y and one x. In order to get the best results I also implemented feature scaling (reduced all variables to a scale of 0 to 1). I chose “horsepower” again to test the algorithm and tried various values for iterations and learning rate (alpha).

Result Screenshots:









KNN

Implementation: Simply used the KNeighborsClassifier from sklearn library to classify haberman dataset.

Result: Accuracy = 0.7096774193548387