Connor Johnson

CS 1675

Assignment 4 Report

1.

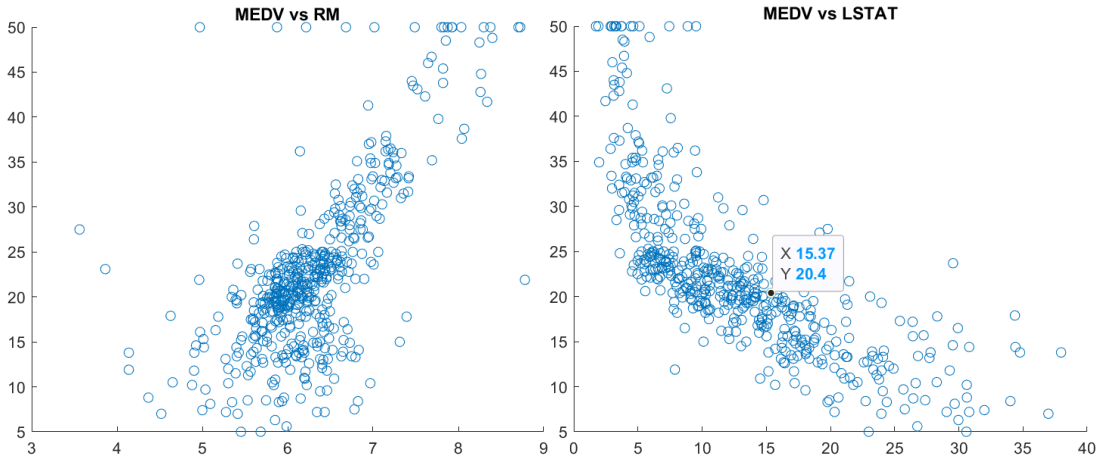
a. CHAS is the only binary attribute in the data.

b.

Attribute	Correlation with MEDV
CRIM	-0.3883
ZN	0.3604
INDUS	-0.4837
CHAS	0.1753
NOX	-0.4273
RM	0.6954
AGE	-0.3770
DIS	0.2499
RAD	-0.3816
TAX	-0.4685
PTRATIO	-0.5078
B	0.3335
LSTAT	-0.7377

RM has the highest positive correlation and LSTAT has the largest negative correlation

c.



RM had the most linear relationship with MEDV. There is a clear positive relationship shown in the scatter plot above. LSTAT had the most nonlinear relationship with MEDV. As seen in the plot, the relationship is more curved and appears to be quadratic.

d. The highest mutual correlation is between the attributes RAD and TAX. They have a correlation of 0.9102.

2.

W = [-0.1011 0.0459 -0.0027 3.0720 -17.2254 3.7113 0.0072 -1.5990 0.3736 -0.0158 -1.0242 0.0097

-0.5860 39.5843]

MSE of train set = 22.0813

MSE of test set = 22.6383

It makes sense that the training set has a lower MSE because this is the set that was used to create model. The model is more likely have less error on the set that was used to create it

3. a.

w = [0.6449 0.7596 0.4709 0.7783 0.5078 1.2809 0.6585 1.3716 0.4268 0.3433 0.7613 1.1852 0.4011]

MSE of train set = 5.6104

MSE of test set = 5.3394

The results of this test are better compared to the linear regression test. The MSE on both the train and test set is significantly lower for gradient scaling.

b. The model didn't work because the values of all the weights went to infinity as the loop progressed.

c. I tried using the methods given but they were not outputting the graphs correctly for some unknown reason.

d. When it was changed to a fixed learning rate, the MSE of the predicted sets was significantly lower, especially when a = 0.01. When changing the number of steps, I found the MSE is inversely related to the number of steps. So, increasing the number of steps lowers the error of the model. Using the learning rate that was give, the MSE was extremely high in the range of 10^55. I changed the learning rate to 0.1/sqrt(i) and this improved the error to similar to how it was in the fixed learning rate cases.