Homework 4

Adam Karl

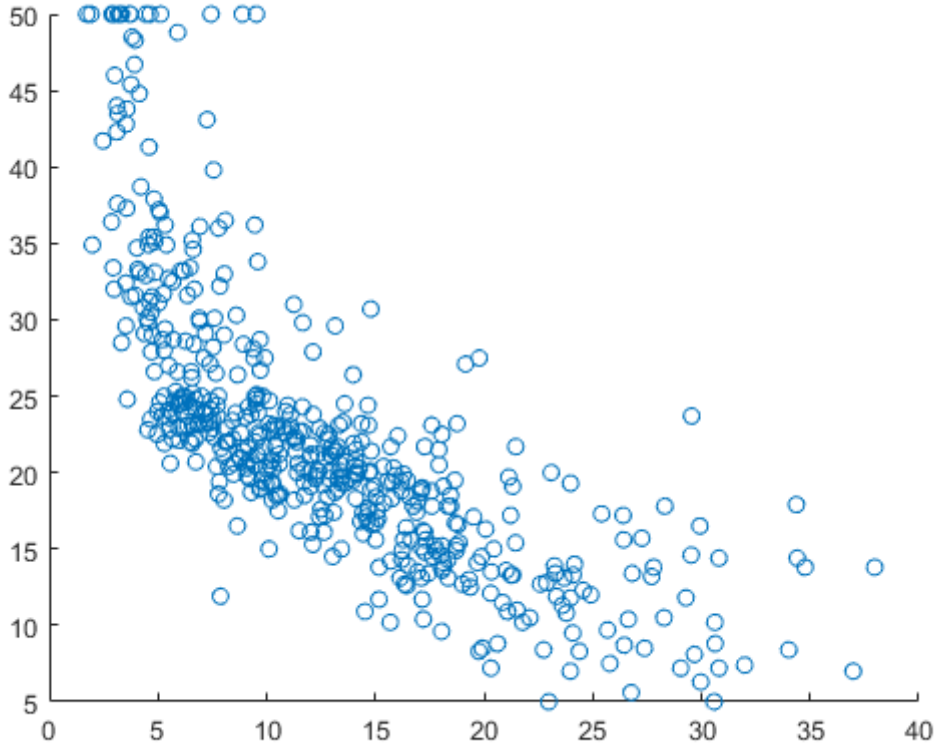
February 25, 2021

1 Exploratory Data Analysis

a. There is only one binary attribute, named CHAS. It is 1 if the tract touches the Charles River, or 0 otherwise.
b.

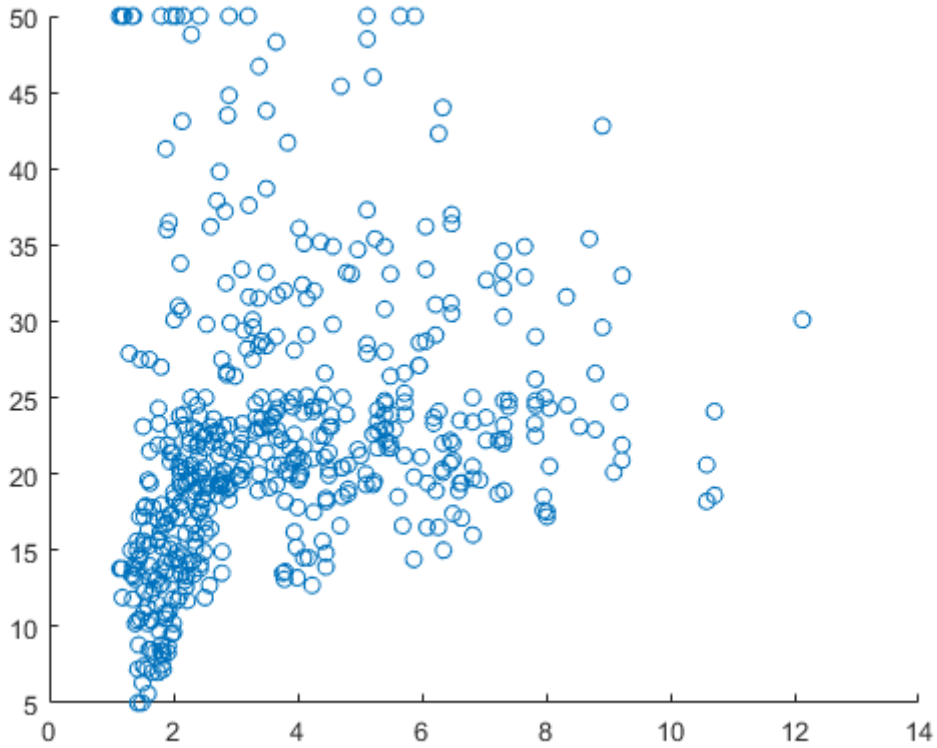
- CRIM: -0.3883
- ZN: 0.3604
- INDUS: -0.4837
- CHAS: 0.1753
- NOX: -0.4273
- RM: 0.6954
- AGE: -0.3770
- DIS: 0.2499
- RAD: -0.3816
- TAX: -0.4685
- PTRATIO: -0.5078
- B: 0.3335
- LSTAT: -0.7377

The highest positive correlation is between RM (average number of homes in a dwelling) and average home value with 0.6954. The highest negative correlation is between LSTAT (percentage of lower status population) and home value with -0.7377.
c. In my opinion, the scatter plot of XXX vsLSTAT looks the most linear, and shows an obvious negative correlation. However, there does appear to be a slight curve to the trend.

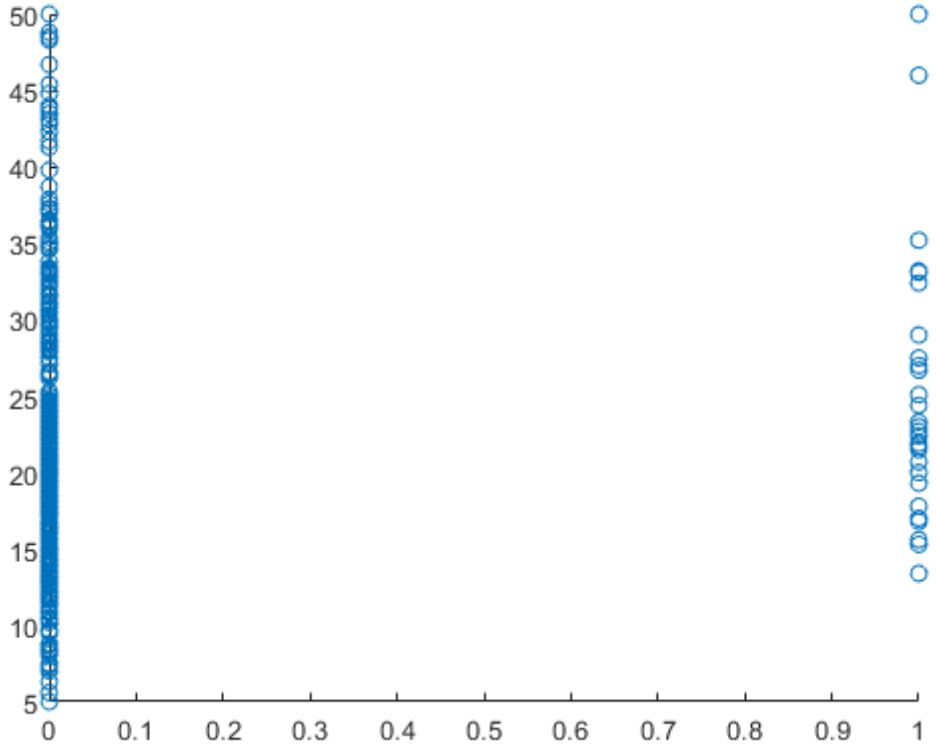


LSTAT vs MEDV

Depending on interpretation, I'm between DIS vs MEDV and CHAS vs MEDV for the most nonlinear graph. While DIS vs MEDV has the vaguest of positive correlations, practically no information or correlation can be gleaned from CHAS (a binary attribute) vs MEDV.



DIS vs MEDV



CHAS vs MEDV

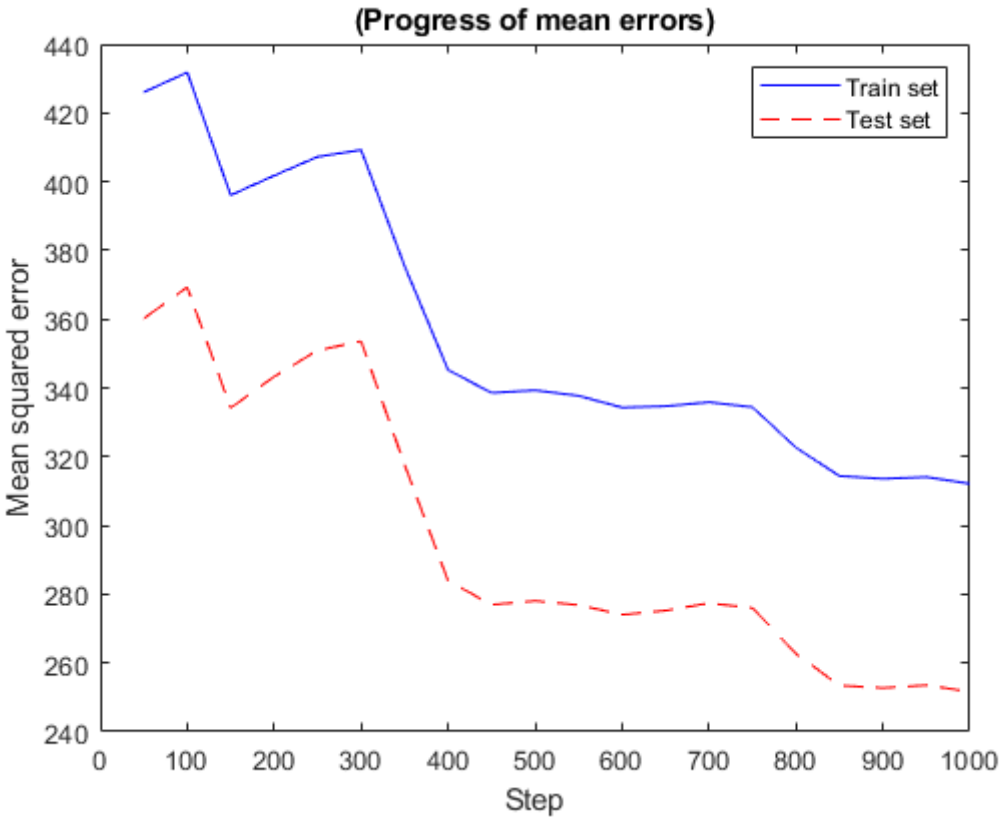
d. The highest correlation is between RAD (index of accessibility to radial highways) and TAX (full-value property-tax rate per $10,000) with a correlation of 0.9102.

2 Linear Regression

d. Applying the training set model to both the training and testing data, I got a mean squared error of 24.4759 for the training data and 24.2922 for the testing data. It's a bit odd that the model fits the test data better than the data the model is based on, but part of this is likely because the test data has a smaller sample size.

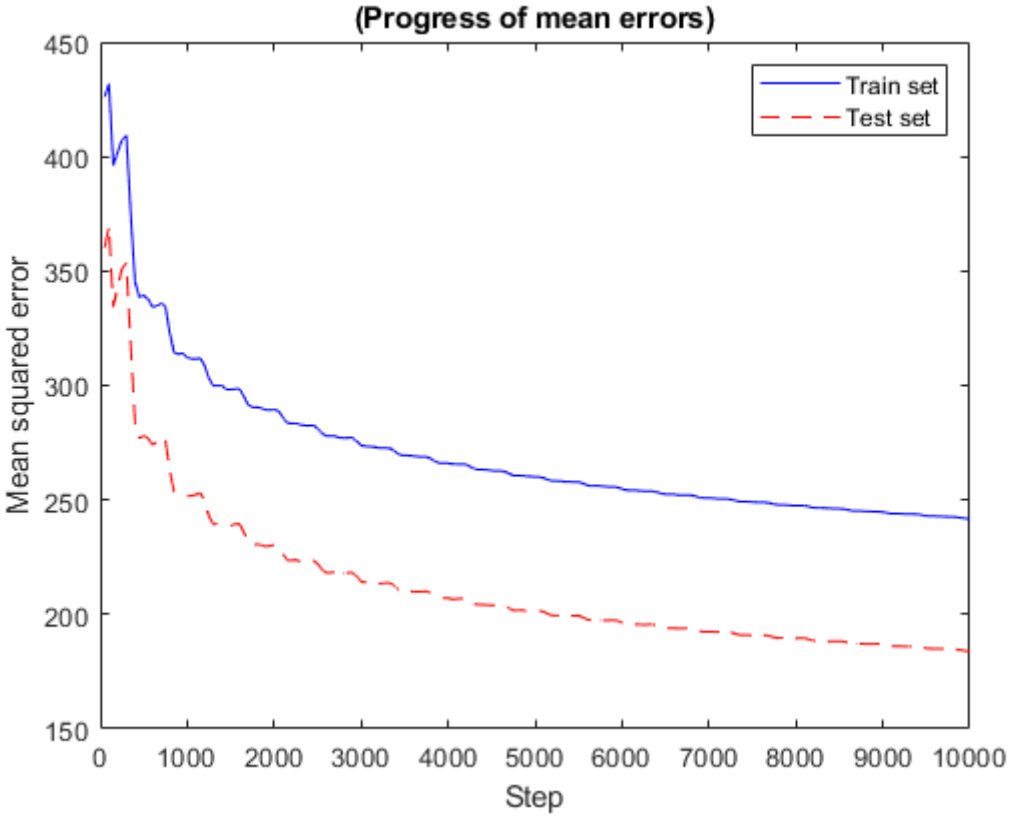
3 Online (stochastic) gradient descent

a. Applying the online model after 1000 iterations to both the training and testing data, I got a mean squared error of 312.0710 for the training data and 251.5408 for the testing data. The model still fits the test data better than the training data, but the errors are much higher for both compared to the linear regression offline model.
b. Feeding in the non-normalized data, the model seems to break since my function returns NaN for all of the final values. It would appear this method only works with normalized data.
c.



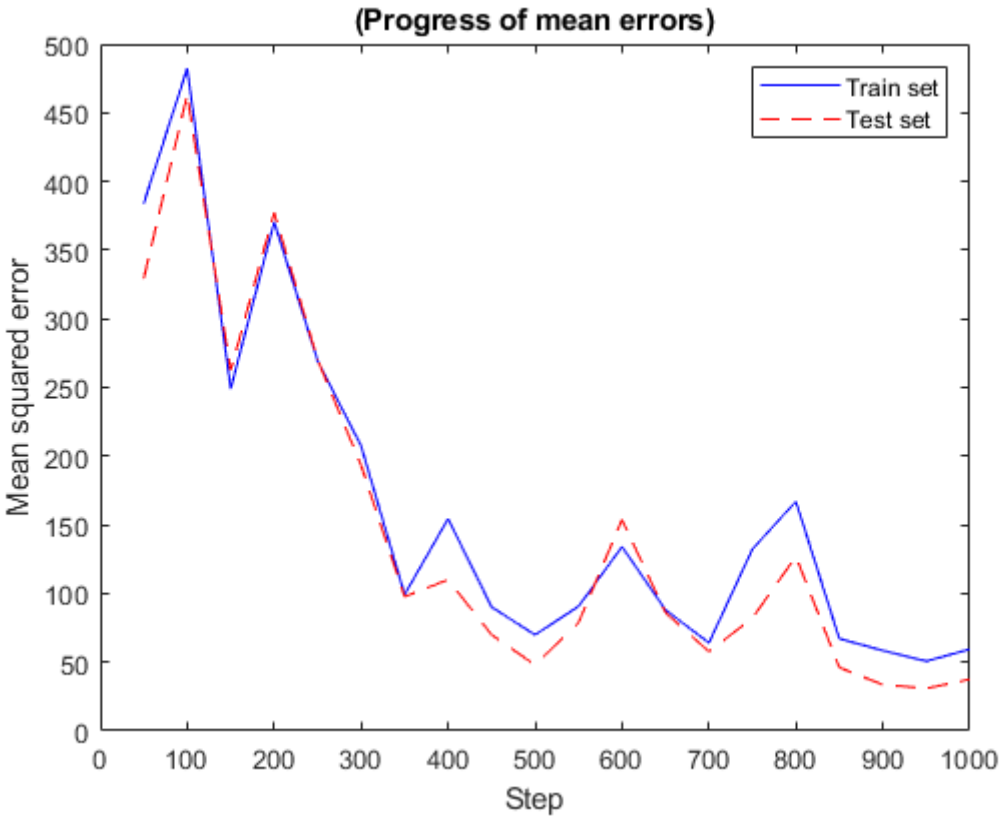
CHAS vs MEDV

d. I first tried adding additional steps. Adding steps keeps decreasing the error, but at a decreasing rate of return.



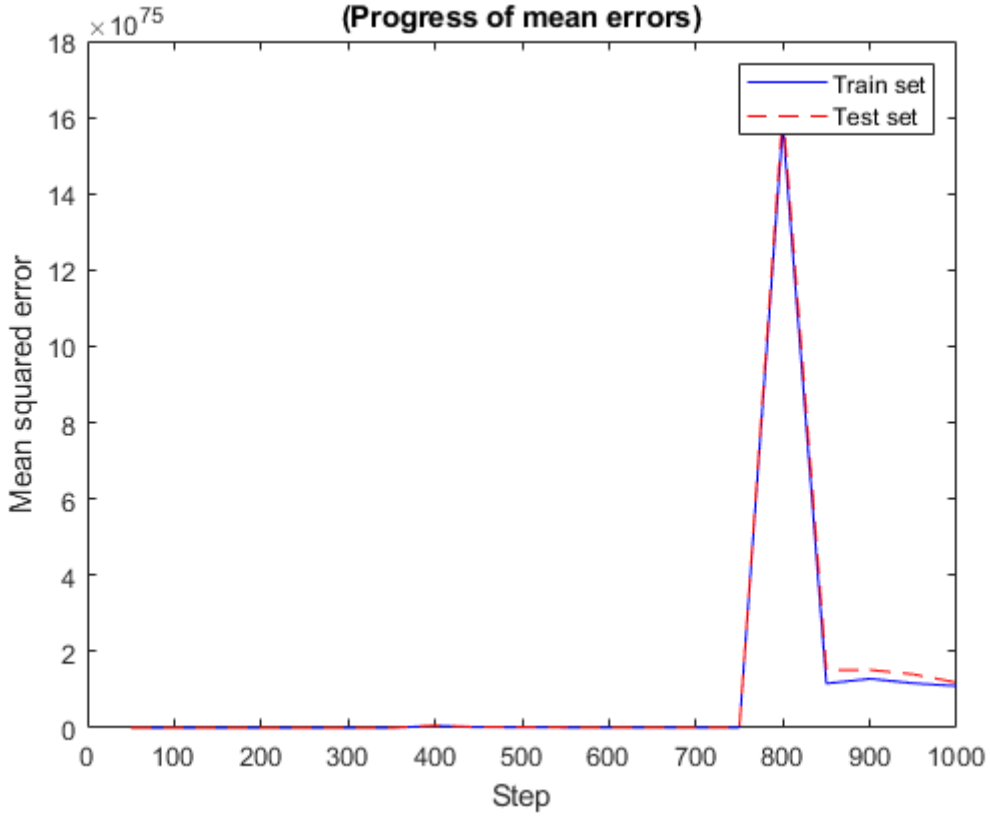
10,000 steps

Then I created a set learning rate. This does allow the error to come down significantly, at the cost of a more unstable error after many iterations.



rate = 0.05

Trying 2/sqrt(n) for the learning rate, on the other hand, makes the error blow up out of control.



rate = 2/sqrt(n)