Robert Martinez
SID: 861238333
4/7/17
CS171
PS1

Problem Set 1

Problem 1
We will find P(d|tp)

Let:
tp = Positive Test
~tp = Negative Test
d = Does have Disease
~d = Does Not have Disease
where a false positive is denoted by:
(p|~d)
and false negative is:
(~p|d)

P(tp|~d) = 1/100

P(~tp|~d) = 1 – P(tp|~d)
= 1 – 1/100
= 99/100

P(~tp|d) = 2/1000

P(tp|d) = 1 – P(~tp|d)
= 1 – 2/1000
= 998/1000

P(d) = 0.00025%

P(~d) = 1 – P(d)
= 100 – 0.00025
= 99.99975

Using Bayes we get:
P(d|tp) = P(tp|d) * P(d) / (P(tp))

We have already solved for components in the numerator P(tp|d) and P(d) leaving us with P(tp).
But P(tp) can be replaced with [P(tp|d) * P(d) + P(tp|~d) * P(~d)].

Giving us the equation:
P(d|tp) = (P(tp|d) *P(d))/ (P(tp|d) * P(d) + P(tp|~d) * P(~d))
= (998/1000) * (1/4000)) / ((998/1000) * (1/4000)) + ((1/100) * (3999/4000))
= 0.0002495/ (0.0002495 + 0.009975)
= 0.0002495/0.010247
= 0.02434859

= 2.439% probability that you have the disease given a positive test result

Problem 3. [5 pts]

The plot produced by the function plotdata shows that some features have a stronger linear correlation to the output than others. For instance there is a clearly linear relation between Feature6 the average number of rooms per dwelling and the output (median value of a house). In contrast the Dummy variable Feature4 is clearly not correlated to the output as the same value produces different median house prices which is shown as a vertical line in the plot. There are some features that a loosly realted but have a high varience in any line that may be choosen to fit the data. For example Feature 7 the age of proportion of owner-occupied units built prior to 1940, is loosly correlated to the median house prices because the data points do not lie closely to any line that may be fit to the data.

Problem 4c

This plot shows the mean squared error(MSE) on the dataset that the ridge regression learned the weights on (training set) and the data set that has not been shown; namely the testset for varying values of lambda. The training data is over-fit for very small lambda's because it is not penalizing ridge regression and giving full value to its weights which were trained specifically to the training dataset. One could estimate the accuracy of just the Ridge Regression prediction algorithm at this point because lambda has not penalized it. Because the ridge regressor was trained on this data it learned the weights that produce outputs very close to the target values. This is evident in the very small MSE of the training set for lambda's from 10^{-4} to 10^{0}.

In contrast the testing dataset has a significantly greater MSE for lambda's at 10^{-4} to 10^{0}. This is because the training data has never been seen by the regressor and did not have the advantage of training on this data.

As lambda increases we see the significance of our trained weights decrease but not the constant b as it is not effected by the regularization function that is scaled by lambda. This means that at the far right of the graph for very large lambda's greater than 10^{4} we can approximate the offset b by taking difference between the training and testing data.

The best approximation for lambda would be around 42. This is where the testing data error reaches it's lowest point and the error of the training set is still reasonably low. This optimal lambda would represent the correct penalty on the trained weights.