Emma Peterson
edp2117
AI 4117
Homework 3

Problem 1: Regression

Part 1: Linear Regression with one feature

1.2 Gradient Descent

b. The mean square error of my model on the training set is 0.00364981533271.

1.4 Testing model and making a prediction

a. Using my model, the predicted height for a 4.5 year old girl would be 1.04616421223 meters.
b. The mean square error for my model on the test set of 20 girls is 0.00640555075731. Although it is still very small, this is a little larger than the training mean square error, which is to be expected.

Part 2: Linear Regression with multiple features

2.1 Data preparation

a. The mean and standard deviation of the features are:
    i. age: μ = 5.21050632911, σ = 1.91161892642
    ii. weight: μ = 18.3066043038, σ = 6.15654350334

2.3 Plotting risk function for different learning rates

b. An alpha value that is too large causes the gradient descent function to take “steps” towards the minimum that are too large, so it steps past the min and converges too quickly. Alpha values that are too small take more than 50 iterations to converge, so they have a slower convergence rate and do not appear to converge on the graph.
c. An alpha value of 0.05 appears to be the best. The beta values beta0, beta1, and beta2 calculated by gradient descent for that alpha are: [1.0340008333529496, 0.09851497615965048, 0.03144281332632269].
d. Using the weights from part c, the predicted height for a 5 year old girl weighing 20 kilos is 1.03180096524 meters.

2.4 Normal equation

a. The β vector calculated with the normal equation is [1.09646081, 0.12943908, 0.00141145]. It is slightly different from the one obtained with gradient descent. The values for beta0 and beta1 are slightly higher, while the value for beta2 is slightly lower.

b. Using the weights from part a, the predicted height for a 5 year old girl weighing 20 kilos is 1.08259528 meters.
c. The results are very similar, showing that both methods are accurate (or at least similarly inaccurate) predictors.

Problem 2: Classification with Support Vector Machines

3. Using Support Vector Machines with different kernels to build a classifier
    a. The best parameters and their accuracies (there was a small amount of variation in these values depending on the run of the program)

Kernel	C	Gamma	Degree	Best Reported Accuracy
linear	0.01	--	--	0.6167
RBF	100.0	1.0	--	0.9567
polynomial	100.0	--	4	0.7000

Notes:

My linear kernel is putting the entire dataset inside the fat margin of the decision boundary, which explains why the accuracy is about 50% and why no line is visible in the graph.

The program iterates through different lists of parameters to find the best one, which means it can be pretty slow at times!

PDF's of figures are each in separate files, they should be in the zipped submitted folder.