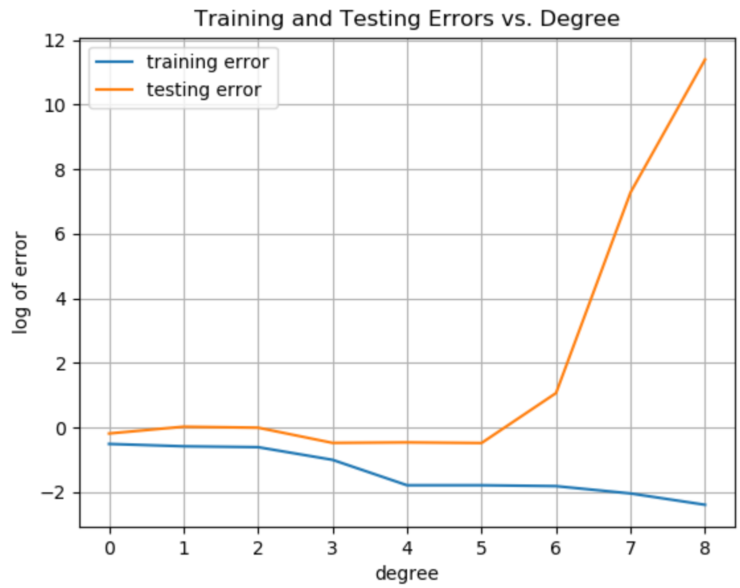
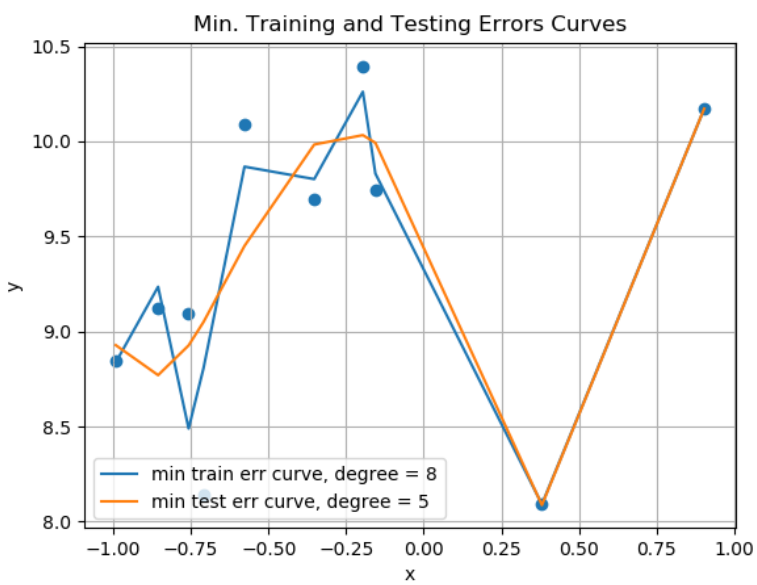
Christopher Tsai

CS 349 HW #4 Free Response Questions

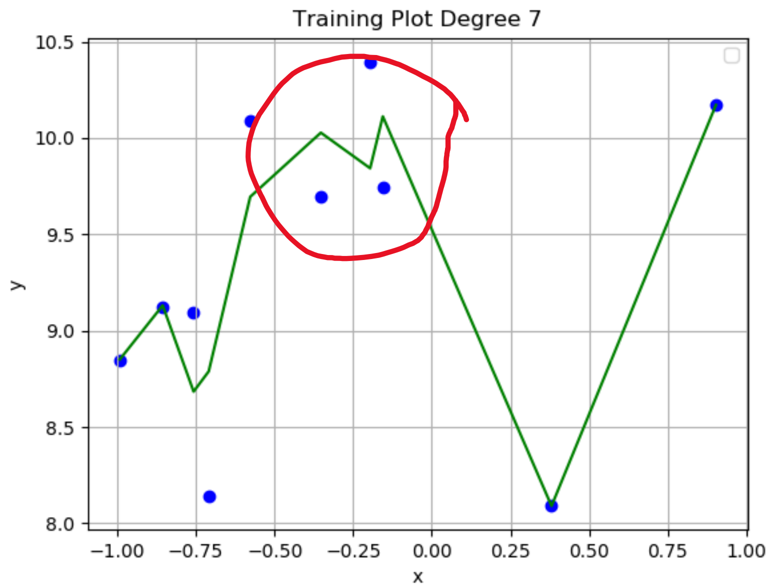
1.

a. 

b. 

2. As seen from the first figure, training error decreases as the degree increases but testing error first decreases until the degree is near its midpoint, and then increases erratically.

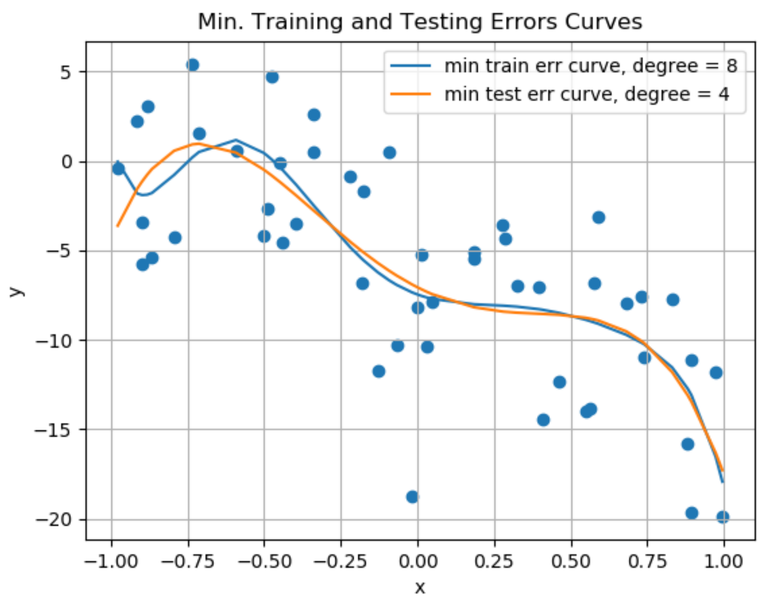
   In regard to the training data, increasing the degree before it reaches its "natural" degree of 4 decreases error because the curve is able to have higher variance (more "hills and valleys") at higher degrees. The reason the error still decreases after the degree of 4 is that it starts to overfit. For this particular run, overfitting starts to be seen clearly at the degree of 7:



You can see that at this degree, small "bumps" start to form where they shouldn't (see the bump at around x = -0.2 and y = 10).

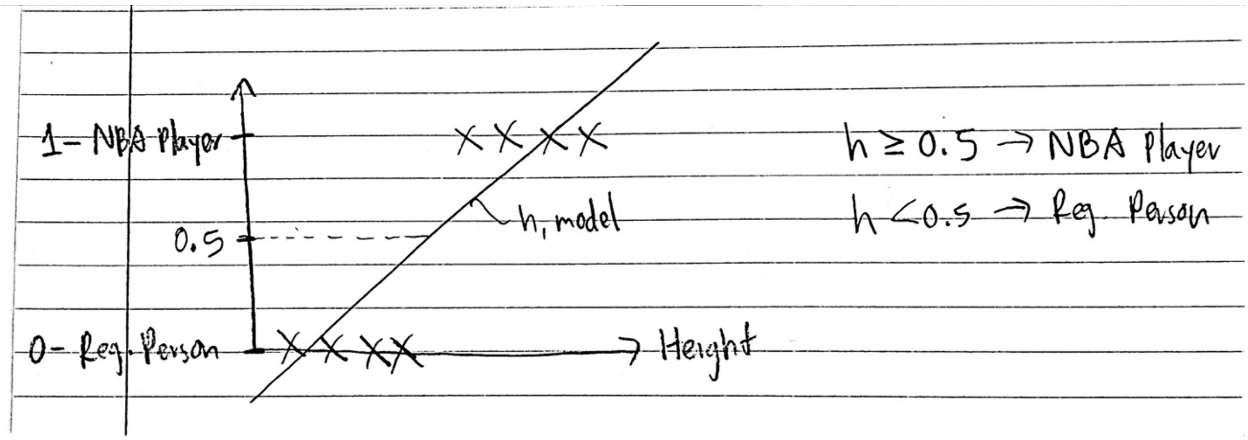
In regard to the testing data, the error decreases until reaching degree 4 and 5 (they basically have the same error) because this is the "natural" degree of the data. It starts increasing erratically afterwards because the model curve is only fit to 10 data points but there are 90 testing data points. The curves with higher degrees have "bumps" that are penalized by a high number of points. Moreover, testing data more often than not spans the entire [-1, 1] range for x but the curve model doesn't always do that.

3.

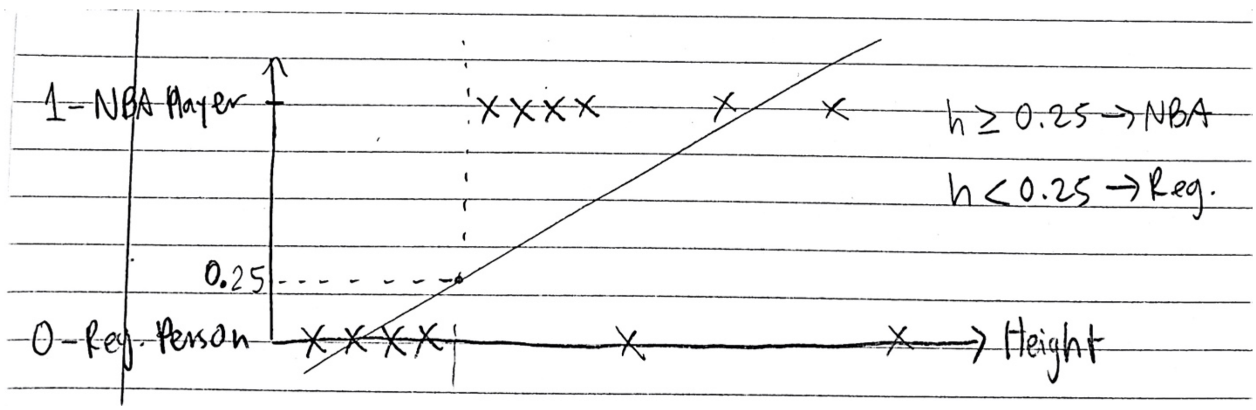


4. Increasing the number of training samples caused my testing error to not erratically increase after degree 4. Now, both training and testing error generally decrease as degree increases. Like before, the lowest training error occurred with the overfit degree of 8 and the lowest testing error occurred at the “natural” degree of 4. This likely shows that the test data is unaffected by overfitting. In other words, the program can find what degree best describes a given test data.
5. Linear regression can be used for classification, but in limited cases and with limited effectiveness. An example use of this is a case where we have one attribute with a

continuous value (let's say height) and a binary output class (let's say NBA player or regular person). Here, we need to establish a threshold at which the hypothesis h will change value, let's say 0.5:



The problem with this (vs. doing linear regression) is that this threshold value is not robust. For another group of data, threshold might be much lower or much higher. Here it's 0.25:



Logistic regression is a better alternative for this scenario.