

Assignment 3

Project Machine Learning using Bitcoin's data has been going smoothly. Since last update I was able to finish the normalization of the data, on the feature close, to make the gradient descent process run much quicker. The problem now is that because the close is between the range of (0-1) and the dates (x axis) runs into the thousands, there is again a large scale difference. I don't want to change the x axis as I think the date is important to leave alone, but this does affect the minimize cost function. Before it had a hard time with theta 1, which is the slope, and now its having a hard time with theta 0, which is the bias. The new predictive model is actually more accurate than the first model, having a cost of only .0063719 compared the original .00752. However the caveat is that the model is much flatter with a smaller slope than the first one. I am worried that this model might be fitting the current data too well and can't generalize and make good predictions in the future such as now. I will come back to this when I do my final comparisons.

The next thing I wanted to accomplish was find a way to incorporate the moving averages that I calculated earlier into my prediction model. I think the predictive model having require both moving averages would not be user intuitive, I think an easier solution is to having something that compares the crossing of the two. What I mean is if the shorter term moving average (10 SMA) is above the slower (22 SMA) then I would find the difference between the two. A positive value indicating that the 10 is above the 22, and a negative value meaning the 22 is above the 10. The values would also have carry a magnitude, meaning that if the 10 is much higher than the 22, the new 'crossover' value would retain this information. I created a new 'crossover' value as I think it encapsulate more of the information I originally wanted to convey in a much easier input for the user. The file is called 'crossover.csv.'

The next steps are to add the new crossover value into the gradient descent algorithm, and also see if adding trading volume makes a difference in the accuracy of the predictive model. I also had an idea of playing with a lambda value (regularized logistic regression) to see if it make the predictive model any better. However since I have not implemented the multiple linear regression yet I am unable to tell if this regularized logistic regression is needed. If I only use 3 features (natural log high, crossover, and trading volume) it might be essential to regularize the algorithm.

The last thing I want to potentially add is I want to see if I can graph this predictive model, I don't know if it will be entirely possible as if there are 3 features that would require a 4 dimensional space and I don't know how to graph anything beyond 2d. I saw there are ways to graph a 3d using a specific plugin for matplotlib but I will have to experiment further and see if this is feasible.

Overall the project is nearly complete and this iteration is just fixing up and polishing the ground work laid in the first assignment. I feel satisfied with the progress completed as I was able to improve and practice my machine learning skills in a project outside of the machine learning course. I've included additional images of the work done.