Josh Cordova
Math 189Z
Homework 3

"Hidden Markov Model for Stock Trading"

This article describes the use of HMM to predict stock prices. To apply the HM model to stocks the authors used four algorithms to calibrate the probabilities, the state sequence, and the rest of the parameters once the first two are set. After the model was calibrated, the authors set up a training window, and used the assumption that a Markov Model is in control to predict the probability of the performance of the S&P 500 over that training period. We then scan backwards looking for a period with a similar probability. Using the difference in observation probability, the model then calculates the estimated change in closing price.

To evaluate the reliability of this model, they used several measures of R^2 over time, as well as the four standard error estimators. They found that it overall beat out the historical average model, and on practical testing they find that it outperforms both the historical average model and the rather simple "Buy and Hold" strategy.

"Gene finding and the Hidden Markov models"

This article describes the algorithms that go into sequencing genes. It first describes an algorithm familiar to anyone who took Bio 52, the Open Reading Frame Algorithm, along with the related significance test. The next step in gene modeling involves HMMs, using the same methods as the stock paper. One difference is that instead of (Open, High, Low, Close), they use the gene sequences. The structure is hard to determine, as not all nucleotides are involved in the coding regions. Due to this, it's hard to accurately construct the matrices involved in the model. However, through the Viterbi algorithm (also used in the previous paper), we can estimate the values in the model. Once trained, the model is extremely effective.

"Random Forests vs Neural Networks"

This article details the use cases for the two very powerful predictive learning algorithms. While the random forest method is only effective for tabular data, neural networks are much more versatile. However, the data we plan to work with will be tabular; versatility is not an advantage for us. Moving on to ease of use, random forests require less data processing and setup. This is a practical advantage over neural networks when used in production code and will yield more consistent results without much tuning. However, we will use neural networks in our project for two reasons. With enough tuning you may end up with significantly better results using NN, and this is our project and we want to learn as much about neural nets as we do COVID risk factors.