Prediction and Modeling of Complex Systems

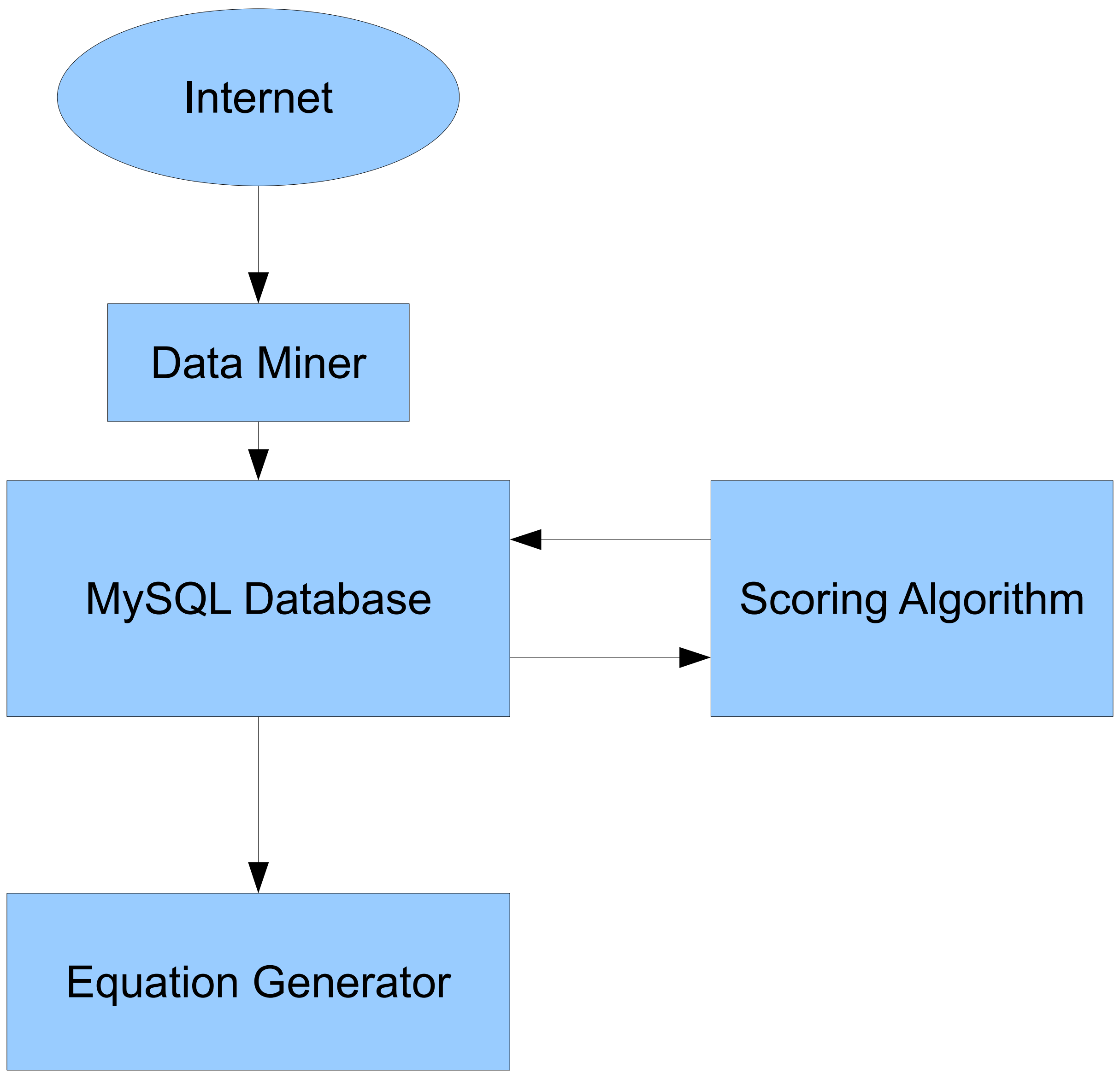
John Sherwood – Computer Systems Lab 2006-2007Period 1

Abstract

The stock market is an immensely complex system made up of millions of interactions between different investors and affected by every action made by thousands of companies. However, economic theory mandates that the actions of most investors are governed by the actions of a few well-informed primary investors and that those other investors primarily follow preexisting trends set by the well-informed investors. Discounting things such as insider trading, the primary catalyst for the well-informed should be news reports, press releases, income reports, etcetera. Therefore, one should be able to predict broad trends across the market and fairly detailed trends for specific stocks by analyzing the available news for companies and their stocks. With the internet as a vast repository of data for almost any purpose conceivable, a program can be constructed to locate and evaluate the available news for various stocks, and based on historical precedent, determine effect of a piece of news on that stock's price over time.

Procedure and Methodology

My program used several modules to attempt to achieve this goal. Using Python, I created two data mining scripts that stored price data for stocks and the available news items for those stocks as well. I then trained my program to generate a quantitative score for the qualitative news data, which I then used in the final part of my program, equation regression. My program assumed a relationship between the generated score of each news item and the price of a stock, exponentially decaying as time passed. The regression algorithm was genetic, refining a pool of random equations through mating and occasionally mutations while discarding the least effective. After testing the equations in timeframes other than those for which they were generated, I refined and updated my mining, scoring, and regression scripts.



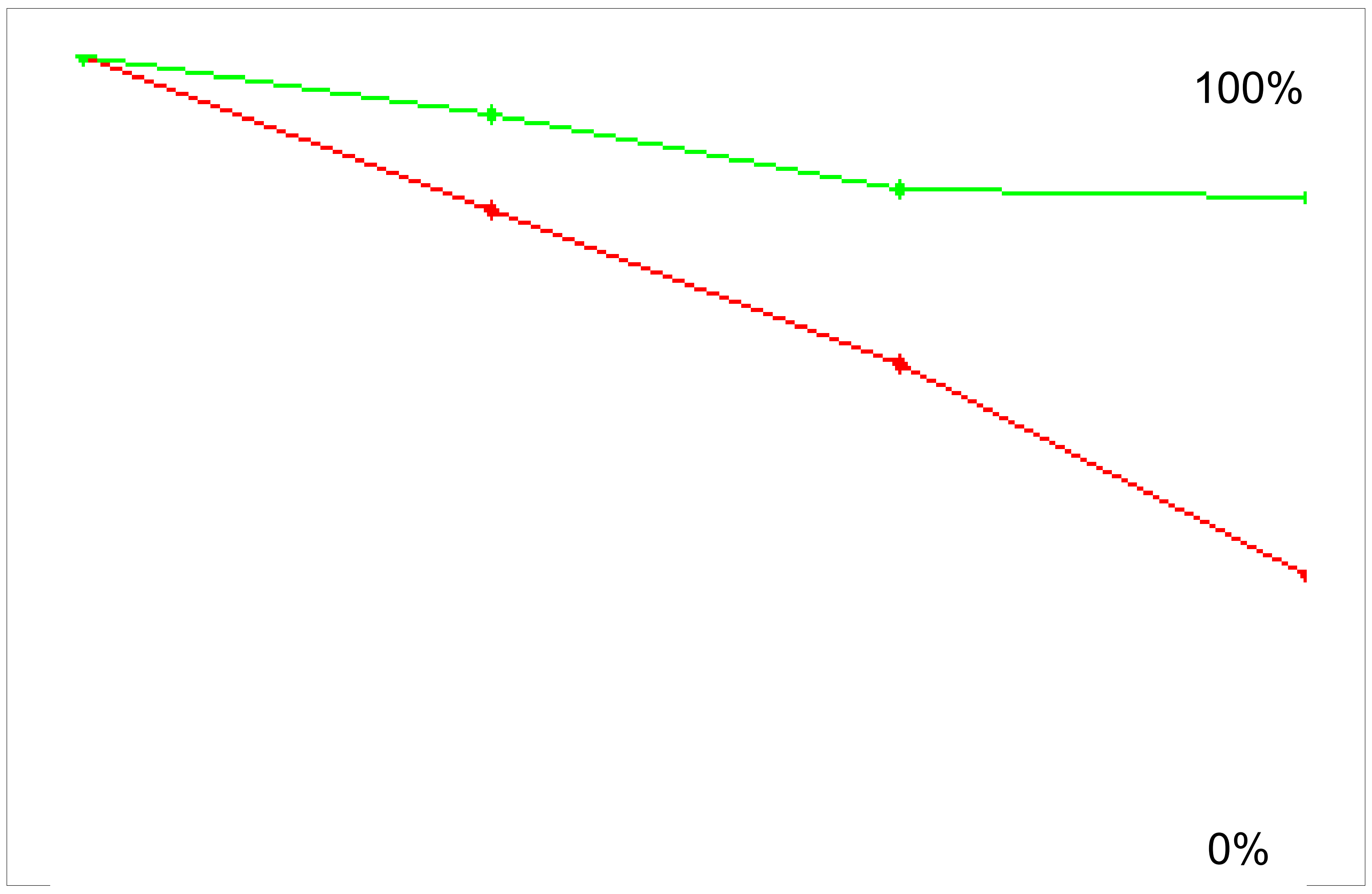
Internet

Data Miner

MySQL Database

Scoring Algorithm

Equation Generator



This graph was generated with a graphing library I developed to easily view the accuracy of my equations

Results

After refining my algorithms and scripts as much as possible, I was able to create equations that were very accurate for the times they were generated for, and fairly accurate for timeframes within the times they were generated for, but had accuracy that steeply declined as the timeframe changed. The graph to the left shows the change in accuracy as timeframe changed. The green line shows the percent accuracy of predictions made within the timeframe the equation was generated for, then shortening the timeframe by one week per point. The red line shows the drop in accuracy by extending the equation's timeframe one week at a time.

Conclusions

Based on the initially accurate equations that quickly grew less accurate as their timeframe changed, this suggests that while my reasoning is sound, some part of the execution is not accurate enough. There are many possible points of failure in the process that may be contributing to this error, such as the possibility that Google! Finance (the site that I use to compile links to news items) is missing news items and thus throwing off predictions, that my scoring algorithm needs to be further refined, or that the form of equation I use to model the effect of a news item on stock prices over time is incorrect. Alternatively, it could be that the effect of a news item on stock prices is determined not only by its content, but also by the time that it is released.

While the existence of multiple feasible reasons for the less than complete success of my project, my research has given me a more than decent standing for future work on this and other related projects.