**Team Members Period**

Sihao Huang 10

Lorenzo Schneiderman

**Team Name**

LongXShortY

**Project Title**

Neural Network-based stock trading system

**I. Project Idea**

Implement a backpropagating neural network that is able to predict market movement and generate live actionable data for S&P 500 component stocks.

**II. Critical Features**

* Retrieve live market data from financial data providers, and obtain fundamentals for the requested stock (P/E, EPS, Market Cap, etc)
* Create an adaptive structural function (neural network) that projects price movement
* Use backpropagation to train the neural network for each individual stock
* Display prediction confidence interval, predicted price change, and stock information in a graphical interface

**III. To be added later Features**

* Provide a pool user-configurable of technical analysis tools as ANN convolutions
* Create a time-sensitive recurrent neural network to allow time-series prediction
* Visual the neural network on a graphical interface to provide insight into its learning capabilities

**IV. Development stages:**

1) Implement a functional neuron. Be able to obtain financial data for individual stocks from an offline database for training.

2) Using the class for the neuron, create a network of interconnected neurons. Connect

to a live datafeed and retrieve pricing data.

3) Implement backpropagation in the neural network, allowing it to find correlations in the input field, which consists of a moving window of historical pricing data and a set of fundamentals. Connect to a financial data provider which supplies fundamentals for given stock. (minimally viable product)

4) Create a simple graphical user interface where the user can look up a S&P 500 component stock for its current price and fundamental data, train the neural network according to its data, and obtain projected pricing.

5) Add a set of technical analysis tools to provide additional input data (e.g. implied volatility) or price data convolutions (e.g. moving average, Aroon, Stochastic oscillator)

6) Expand on the graphical user interface to provide a color-graded map of the neural network and allow configuration of its internal parameters.

7) Add delay elements to the neural network to allow for time sensitivity.

**V. Diagrams of features:**