

# Investment and Trading Capstone Project

## Stock Price Indicator

### Overview

People need money to obtain what they need to live and thrive, But what do you need to be happy? Also money but a lot of it.

One way to obtain a lot of money is throwing investing, and why invest?

Investing is an effective way to put your money to work and potentially build wealth. Investing allow your money to outpace inflation and increase in value.

You may need a lot of money to retire early, retire with more money, or make a big purchase like buying a house, buying your dream car, or traveling the world, and all of these other words for happiness

Investment firms, hedge funds, and even individuals have been using financial modeling to better understand market behavior and make profitable investments and trades.

### Problem statement

Financial modeling tries to predict the future price of the stocks by learning their historical price over years. So, in this project, I will build a stock price predictor by using a machine learning algorithm that takes daily trading data from the yahoo finance website over a certain date range as input, and outputs projected estimates for a given query date.

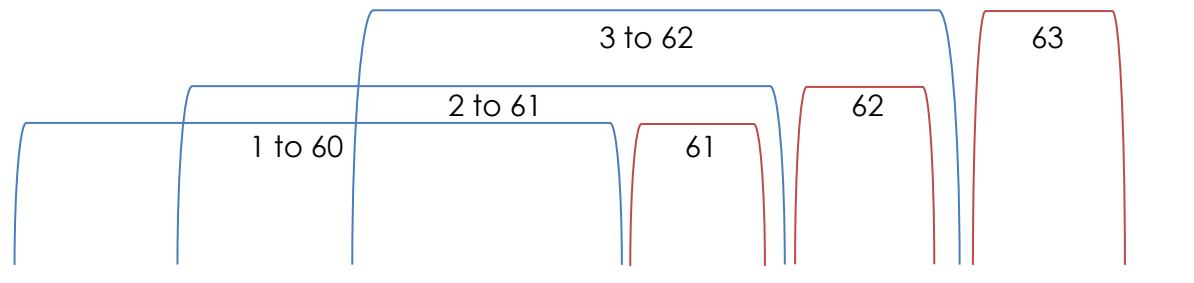
The system will predict the adjusted close price for the stock selected and for different intervals, as well as optimize a stock portfolio that takes the amount of money you intend to invest in the market and make a mix of stocks that maximize the Sharpe Ratio, minimize volatility or maximize return using [Efficient Frontier](#).

### Datasets

The input data will contain the adjusted close price for a big company (mega > 200B and large 10B-200B) in the NASDAQ exchange, and it's about 750 companies used for the portfolio optimization, and I will use FANG stocks for the price predictor.

FB	APPL	NTFX	GOOG	AMZN	..	..
159	155	155	155	155	..	..
160	156	154	153	155	..	..
..	..	..	..	..	..	..
200	190	180	170	160		

I will use 60 day past of adjusted close price to predict the price in day 61

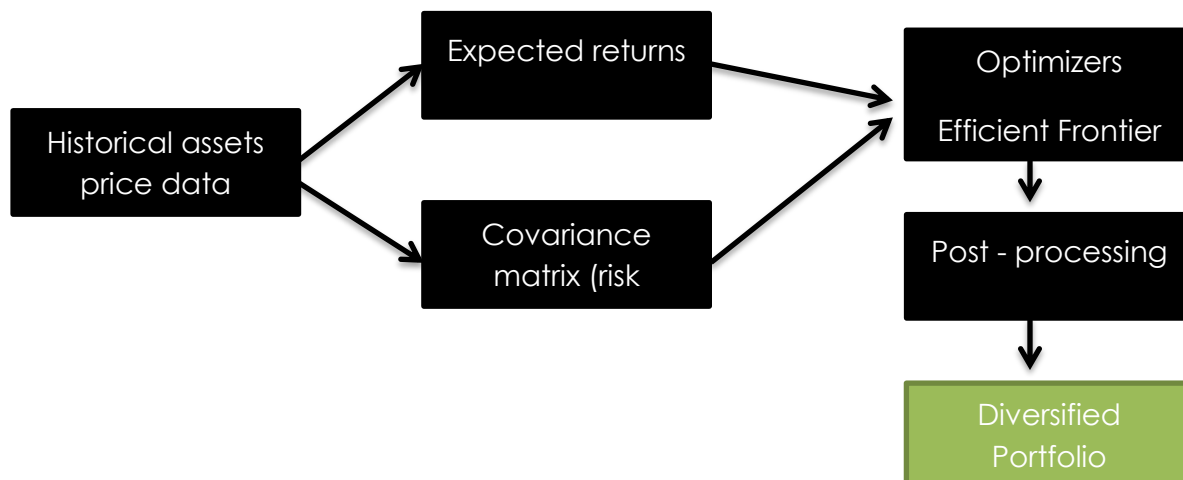


## Algorithms and Techniques

Long Short-Term Memory will be used for the stock price prediction. LSTM is the type of recurrent neural network that is used to learn order dependences in sequence, and because the stock price is ordered by the date and the old price in some way affects the future price, so LSTM will be helpful because it has a memory gate that storing past information.

The processed data feeds to the LSTM model to predict the price for the chosen stock.

In the portfolio optimization, the PyPortfolioOpt library will be used for implementing the portfolio optimization method efficient frontier.



A diversified portfolio will be a pandas data frame that contains the company name and ticker and how many shares will you buy.

## Metrics

Mean square error will be used for evaluating the price prediction model. The smallest the mean square error the better the model performance and the formula is given as follows:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2$$

**MSE** = mean squared error

$n$  = number of data points

$Y_i$  = observed values

$\hat{Y}_i$  = predicted values

For the portfolio optimization backtest will be great but I will not implement it, I will use the [Portfolio Visualizer](#) website to evaluate the method and if the portfolio performs well against S&P 500 Index

## Benchmark

For this project I will use the base model in machine learning for regression (Linear regression) as a benchmark to see how the linear regression model differs from LSTM model and measure the performance by the different value between actual value and the predicted one for each model using mean square error.

## Project Design

This project will be implemented through python notebook, and I will build interface through CLI (command line interface) that lets users specify stock(s) they are interested in and provides predictions at some pre-defined intervals.

- **Set up**
  - lpython notebook
  - Pandas, Numpy, Matplotlib, scikit-learn, keras, PyPortfolioOpt
- **Prepare datasets**

- The adjusted close price obtained from yahoo finance API after getting their tickers from NASDAQ screener.
  - Convert it to pandas data frame
  - Pre-processing
  - Normalize data
  - Split the data to 75:25 for train and test
- **Develop models**
  - Linear regression model with Scikit-learn
  - LSTM model with Keras
  - Compare the two models and evaluate their performance
- **Improve LSTM (play with hyperparameters)**
- **Document and visualize the result**
- **Built a user interface CLI**