Stock Market Prediction - Tesla

Daniel Karcz Andrew, A20357971, Student, IIT , Chandana Ravindra Prasad A20406271, Student IIT

IIT, Illinois Institute of Technology, 3300 S Federal St, Chicago, IL 60616

Abstract – If stock price prediction was an easy problem to solve, no stock traders would be rich. For this project, we classify "stock traders" into two groups: day trading (high frequency trading) and institutional investing (institutional pensions investors, personal investors, etc). This project proposal seeks to compete with institutional investors, based on stock trade transactions, macroeconomics, and fundamental analysis within the one week to one month time frame.

Existing stock market prediction methods incorporate the supply and demand of stocks via predictive indicators, and big data, at the nanosecond level. In this project, using daily stock transactions, energy prices, macroeconomics, and fundamental analysis of Tesla; we introduce reinforcement learning methods to predict buy, sell, or hold trade signals, in order to compute the most profitable instance to trade or hold TESLA stock. This project shines light on using a blend between Reinforcement Learning and fundamental analysis trading behavior to identify the optimal price point to buy, sell, or hold Tesla stock.

I. OVERVIEW

A - Research Goal

The goal of this group project is to generate trading buy, sell, or hold signals, in order to compute the most profitable instance to buy, sell, or hold TESLA stock.

B - What we seek to address

The Securities Act of 1933: Rule 156 states that portrayals of past growth, to convey an impression that future growth results will be achieved by an actual or hypothetical investment is misleading. Therefore, even though this project is not an investment opportunity and not held to SEC regulations, we will embrace the mentality that past stock trade transactions data alone will not predict future stock prices. However, we aim to prove that the combination of stock trade transaction analysis, economic, and company data will predict buy, sell, or hold prices of Tesla stock more often than not.

This project seeks to address if a few highly statistically significant features can predict future buy, sell, or hold prices for Tesla stock within the one week to one month time frame.

C - Methodology

The methodology used will be Reinforcement Learning. The Reinforcement Learning method used will implement a decision policy system based upon a reward structure utilized over the training data. Learning to increase the amount of reward and decrease the amount of penalty.

This is accomplished via searching through trial and error, initialize and test; and then using memory to remember what worked best for the last trade, and start the next trade from there.

D - Metrics

Law of Numbers: If our model has a positive expectancy of 25% profit per trade; based off last 5 years of daily stock trades (training data), and our model produces 25 trades per week, it is far better off than our model which generates only 5 trades per week. Therefore, we will execute trades as often as possible, when the Reinforcement Learning model is in our favor to buy and sell. The model just needs to have more profitable trades then loser trades, in order success to be achieved.

II. OUTLINE

E - References

A Multiagent Approach to Q-Learning for Daily Stock Trading:
Jae Won Lee, Jonghun Park, Member, IEEE, Jangmin O, Jongwoo Lee, and Euyseok Hong

Adaptive stock trading with dynamic asset allocation using reinforcement learning:
Jangmin O a,, Jongwoo Lee b , Jae Won Lee c , Byoung-Tak Zhang a*

Trading financial indices with reinforcement learning agents:
Parag C. Pendharkar, Patrick Cusatis*

F - Data Sources with description

1) Source: Google Finance -> Tesla stock daily prices

Features = (Date, Open, High, Low, Close, Adj Close, Volume)

Dimensions = 2000 X 7

2) Source: Macro Trend Data -> Crude Oil daily prices

Features = (Date, Price)

Dimensions 2000 * 2

3) Source: Twitter -> Elon Musk tweets

Features = (ID, Date, Text)

Dimensions 1000 * 3

4) Source: Macro Trend Data -> Electricity Prices

Features = (Date, Price)

Dimensions 2000 * 2

Work in Progress Data

5) Source: FRED -> GDP

Features = (Date “Quarterly”, Amount)

Debating is useful after rolling-up time series data to various frequencies. Then performing analysis to get to stationary; in order to do prediction.

6) Source: FRED -> Interest Rates

Features = (Date “Quarterly”, Amount)

Debating is useful after rolling-up time series data to various frequencies. Then performing analysis to get to stationary; in order to do prediction.

7) Source: Bureau Labor Statistics -> Income and Earnings

Features = (Date “Quarterly”, Amount)

Debating is useful after rolling-up time series data to various frequencies. Then performing analysis to get to stationary; in order to do prediction.

G - Data Processing

For GDP, Interest Rates, and Income & Earnings data; we will roll-up time series data to various frequencies, and then perform analysis to find a way to get stationary data in order to do stock predictions. This is currently being researched via quantstart.com and quantopian.com.

H- Data Stylizing Facts

ARIMA: See section I -”Model Selection”

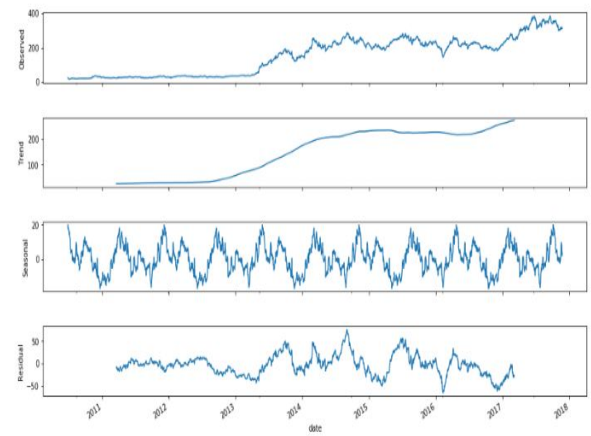
I - Model selection

Understanding correlation and stationarity: ARIMA (AutoRegressive Integrated Moving Average) models.

Autoregressive examines the correlation of todays price with individual previous prices and describes that relationship. We use rolling mean to smooth out the noise.

The idea of stationarity - the data is stationary if mean and variance are the same overtime. Tesla’s stock has been increasing over time particularly since 2013 and is not stationary.

It is a requirement for proper statistical analysis - there are a few ways to transform the data and make it stationary. We do this by finding trends in the data and any seasonality observed. We remove these aspects of the data for modelling and then make predictions and once the prediction is done, we add the seasonality and trend data back into the predicted results.



We can see our observed stock prices on top, the overall trend in the price second, any yearly repetition third, and finally our residual which is whatever variation is still left over and needs explaining

ref : https://towardsdatascience.com/tesla-stock-price-prediction-f16a702f67d7

When the graph is split, it breaks down as trend, seasonal and residual. Adding up these 3 will result in obtaining the observed graph. Out of these 3, residual is all that is left to predict. We can hope that the residual graph shows some form of stationarity. To improve the stationarity, we can also try using first difference - taking the current price and subtracting the one immediately before and conduct a test called Dickey-Fuller Test.

Keeping all these factors in mind, we can come up with the parameters needed to train an ARIMA model for our data. We plan on training the model on a subset of data and then test the model on the rest.

Creating the agent that decides among buy, sell or hold: We use Markov Decision Process and Q learning to achieve this.

Q(s,a) ~ r + y* Q' (s' , a')

s : Agent's current state

a : Current optimal action

y : Discount factor

r : Immediate reward from Q -> Q'

s' : Next optimal state

a' : Optimal action in the next state

J - Evaluation:

Since twitter seems to be a source of news for a lot of people, and since the CEO of TESLA - Elon Musk is the face of the company, the sentiments expressed in the content that he posts on social media will influence the way people think about the company which in turn affects the stock values. So, we consider the sentiments from the twitter account of Elon Musk and evaluate the sentiments using VADER (Valence Aware Dictionary and Sentiment Reasoner).

It is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. We use this to get the compound polarity score which acts as a single measure of polarity (combines positive, negative, and neutral polarity scores).

In order to evaluate the model, here are a few things we focus on:

- Avoid overfitting
- Avoid Overtrading
- Considering the effects of non-stationary processes / regime changes
- Lookahead (time travel illusion)

We use mean squared error to test and evaluate the accuracy of the model.

K - Software packages and tools

R STUDIO CRAN:

Tensor Flow

Library(ReinforcementLearning)

Markov Decision Processes

Q-Learning