HKUST CSE FYP 2017-18, TEAM R04

OPTIMAL INVESTMENT STRATEGY USING SCALABLE MACHINE LEARNING AND DATA ANALYTICS FOR SMALL-CAP STOCKS

MACHINE LEARNING AND FINANCE

MACHINE LEARNING AS A SOLUTION

MACHINE LEARNING OFFERS SOLUTIONS TO SOME OF THE MOST IMPORTANT CHALLENGES FACED BY THE BANKING SECTOR TODAY.

Customer Segmentation

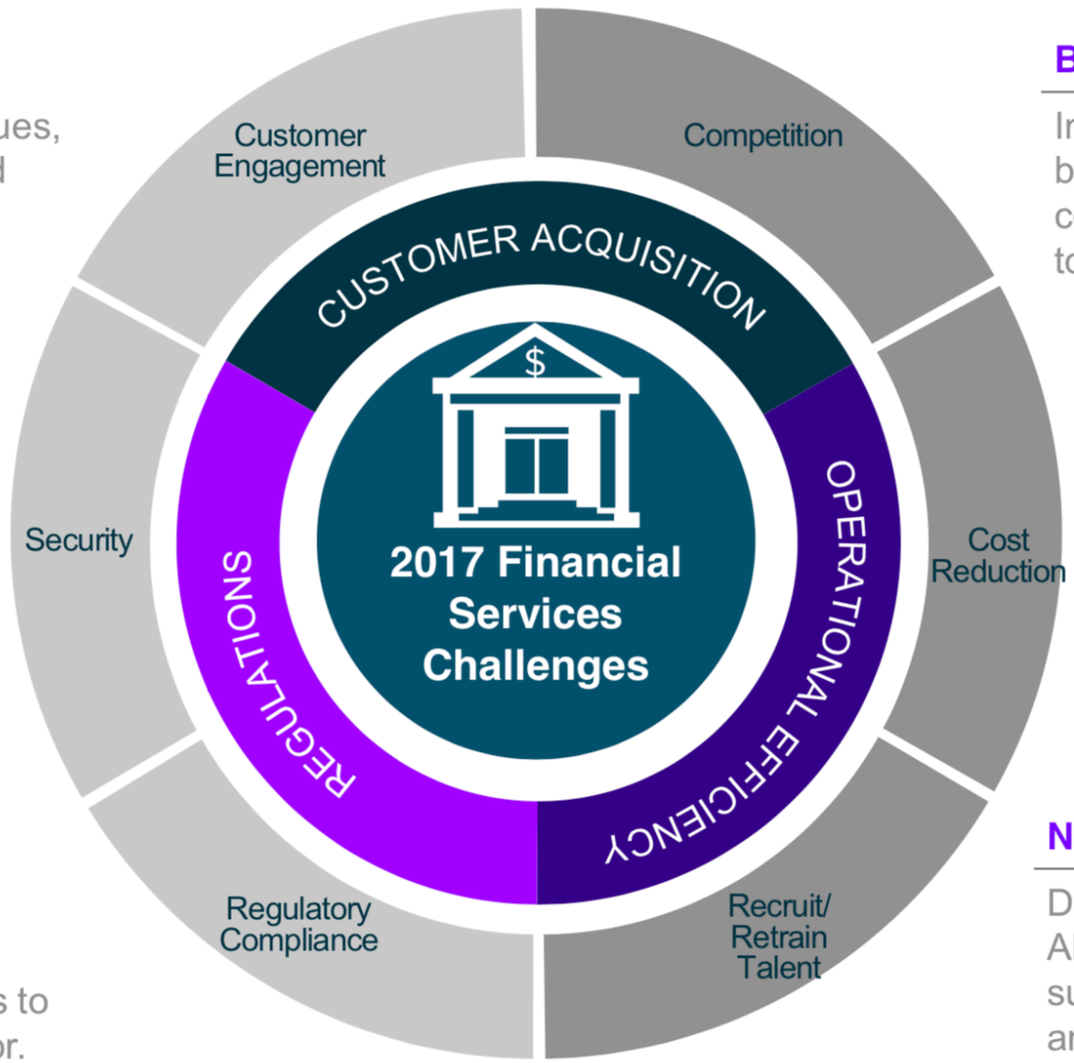
Through unsupervised learning techniques, banks can segment their customers and offer a personalised, targeted product offering.

Fraud & AML Detection

Machine Learning offers significantly improved fraud, AML (Anti-Money Laundering) and credit risk detection possibilities.

Compliance

Compliance through automated reports, stress testing solutions, and behavioral analysis of emails and phone recordings to determine suspicious employee behavior.



Big Data & Agility

Investment in Machine Learning offers banks the speed and agility they need to compete with tech-savvy Fintech firms and to make use of Big Data.

Cognitive Automation

Combined with Robotics, Machine Learning offers the ultimate automation potential with many back office risk, finance and regulatory reporting processes contenders for automation.

Natural Language Processing

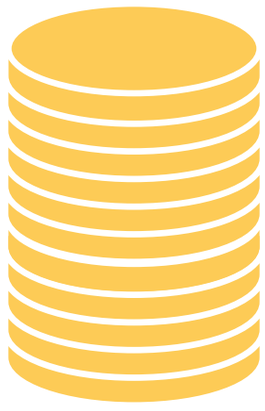
Digital skills are in short supply in FS. Algorithms can evaluate CVs of successful employees and search for and identify online candidates with similar traits and experience.

SMALL-CAP

< US$ 2B

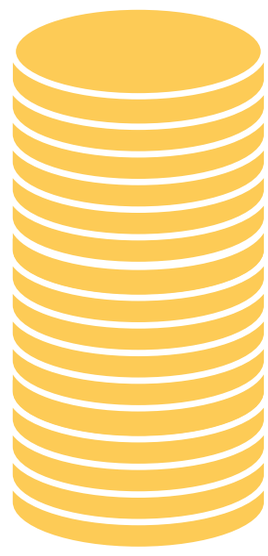
MID-CAP

US$ 2B - US$10B



LARGE-CAP

> US$ 10B



Market Capitalisation = Market value of a company's outstanding shares

SMALL CAPITALISATION STOCKS

- Higher risk and volatility
- Potentially higher returns
- Of most interest to Retail Investors
- Institutional Investors not very active
- Listed on NASDAQ for at least 15 years

TARGET SEGMENT: RETAIL INVESTORS

- Lack sophistication and expert knowledge
- Access to lower quality research and resources
- Look for:
  - higher returns for lower risk
  - diversified portfolio in a smaller investment

THE SMALL-CAP MARKET

- Little analyst coverage
- Less financial information published
- Market inefficiencies

MACHINE LEARNING MODELS FOR PREDICTION

+

PORTFOLIO ALLOCATION USING PREDICTIONS

+

WEB APPLICATION FOR USER INTERACTION

OBJECTIVES

- Experiment with different machine learning algorithms for stock price forecasting
- Use time series predictions to allocate stocks within risk threshold of user
- Develop a web application that allows users to specify parameters and track portfolio over time

DATASOURCES

- Python scraper for ticker symbols of NASDAQ small-cap stocks from Zacks Stock Screener Tool
- Cleaned for inconsistencies in preferred stocks' symbols
- Extraction of historical stock prices using AlphaVantage API
- Filtered to obtain prices between Oct 2001 and Feb 2018

PRICE PREDICTION MODEL

LEVERAGES MACHINE LEARNING TO PREDICT STOCK PRICES FOR A MONTH AHEAD

Price Prediction Model

PROBLEMS SOLVED BY ML

1

Classification

2

Regression

PROBLEM WE ARE SOLVING

1

Classification

2

Regression

MACHINE LEARNING FOR STOCK PRICES

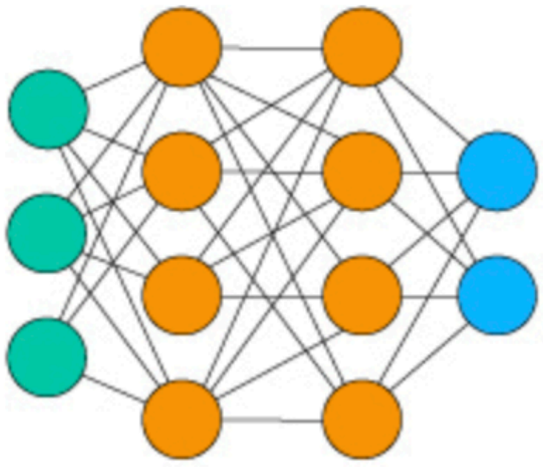
- Time series: a long list of decimal values (Stock prices)
- Features and targets?

5.9732, 5.9732, 5.9001,
5.9732, 6.0406, 5.9001,
6.2541, 6.0743, 6.0743,
5.8664, 5.8327, ...

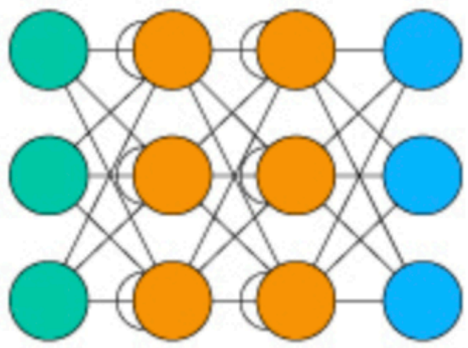
FEATURE 1	FEATURE 2	.....	FEATURE M	TARGET VARIABLE
5.9732	5.9001	.....	6.0406	6.2541
5.9001	6.0406	.....	5.9001	5.8327
.....	.....	.....	.....	.....
.....	.....	.....	.....	.....

MACHINE LEARNING ALGORITHM – LONG SHORT-TERM MEMORY

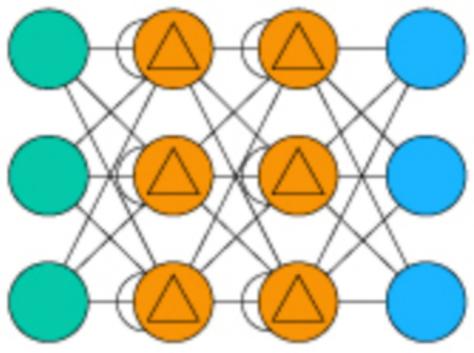
- RNN (Recurrent Neural Network): class of Artificial Neural Network that allows units to form a directed graph
- LSTM: type of RNN that can model long temporal sequences



Multi Layer Perceptron



Recurrent Neural Network



LSTM Recurrent Neural Network

MACHINE LEARNING ALGORITHM – LONG SHORT-TERM MEMORY

- Critical parameter to decide: sequence length for machine learning to create dataset

M = sequence length

FEATURE 1	FEATURE 2	.....	FEATURE M	TARGET
5.9732	5.9001	.....	6.0406	6.2541
5.9001	6.0406	.....	5.9001	5.8327
.....	.....	.....	.....	.....
.....	.....	.....	.....	.....

MACHINE LEARNING ALGORITHM – LONG SHORT-TERM MEMORY

- Multiple Strategies of choosing sequence length
- Strategy 1:
  - Fix sequence length for all stocks. e.g.: 10
    - May not give best results
- Strategy 2:
  - Optimise sequence length based on test RMSE
    - Unclear hypothesis space, exhaustive search expensive

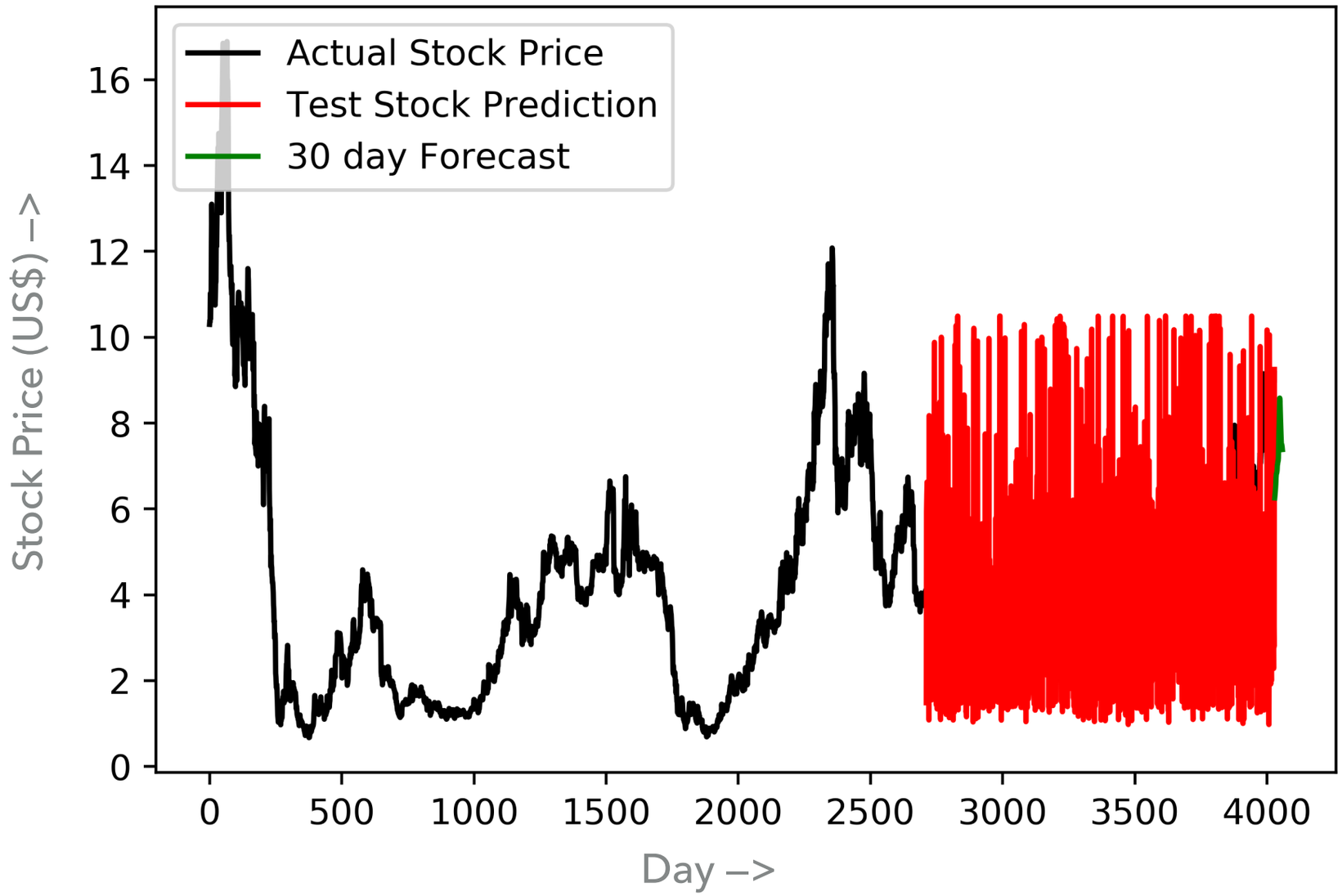
MACHINE LEARNING ALGORITHM – LONG SHORT-TERM MEMORY

- Take sequence length as 7
- Need 30-day forecast
- Divide the time series into 70/30 for training/testing
- Train using Root Mean Square Error as loss function
- Create dataset from time series as follows:

Features (Input)		Target (Output)
$p_t, p_{t+1}.....p_{t+6}$	→	p_{t+36}
$p_{t+1}, p_{t+2}.....p_{t+7}$	→	p_{t+37}

p_t : stock price on day ‘t’

MACHINE LEARNING ALGORITHM – LONG SHORT-TERM MEMORY

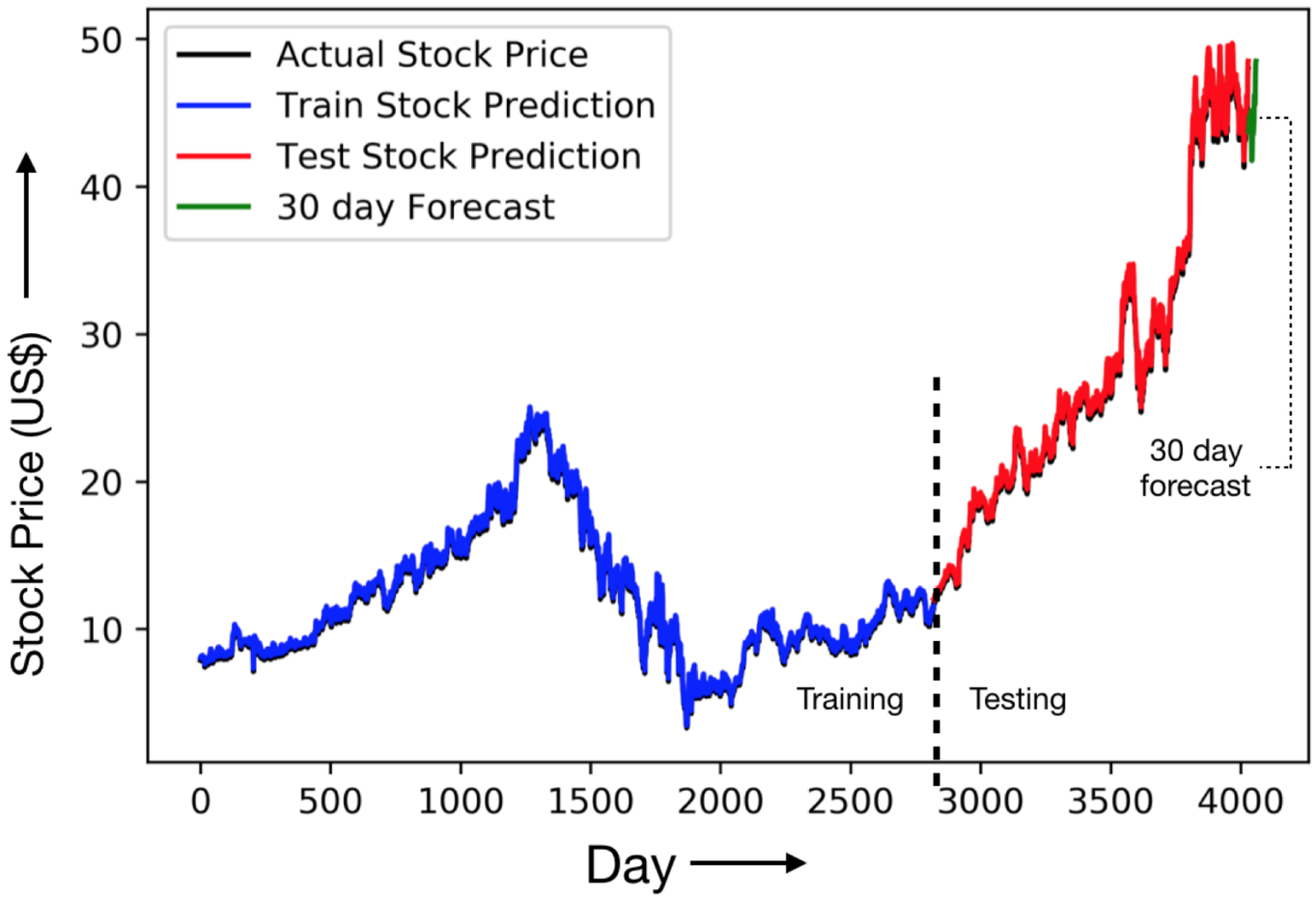


- Unable to generalise on testing data
- Unreliable forecast

MACHINE LEARNING ALGORITHM – LINEAR REGRESSION

- Simpler model
  - Fewer parameters
- $StockPrice_t = \beta_1 * StockPrice_{t-30} + \beta_2 * StockPrice_{t-60} + \beta_0$
- Train using R^2 loss as loss function

MACHINE LEARNING ALGORITHM – LINEAR REGRESSION



- Performs well on testing data
  - Follows general trend unlike previous case
- 30-day forecast reliable

ASSET ALLOCATION MODEL

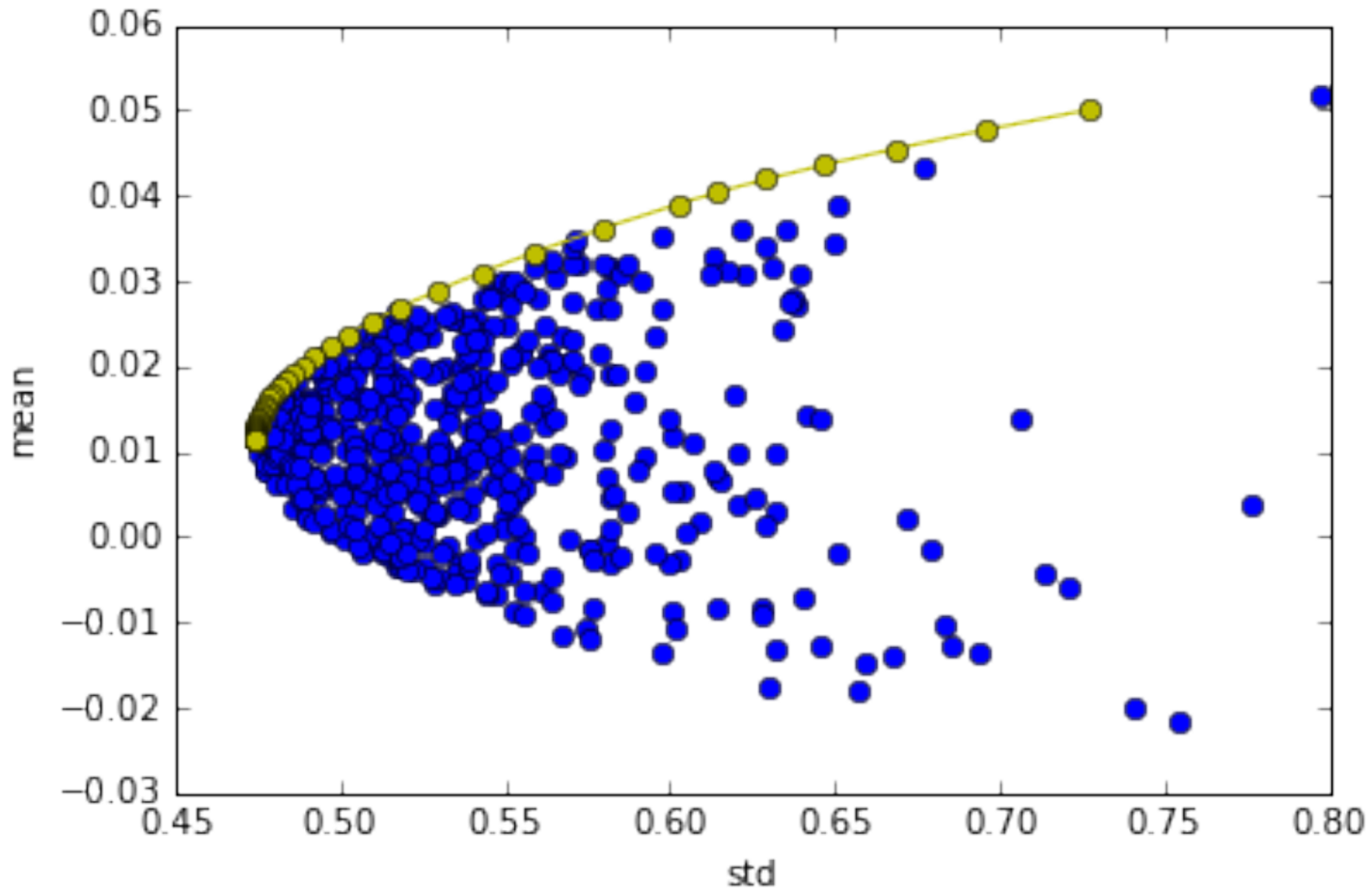
USES PREDICTIONS TO FIND OPTIMAL SET OF STOCKS WITH THE RATIOS TO INVEST IN

Asset Allocation Model

MEAN VARIANCE OPTIMISATION

- Proposed by Henry Markowitz in 1952
- Weighted average of individual stocks
  - $R_w = w_1R_1 + w_2R_2 + \ldots + w_nR_n$
  - (R: return, n: number of stocks)
- Use covariance matrix to minimise mean variance

MEAN VARIANCE OPTIMISATION



Markowitz Bullet

ALLOCATOR SCRIPT DESIGN

- User input: number of stocks, volatility threshold
- Modular design offers flexibility
- Sorting parameters
  - Minimise risk (SD)
  - Maximise return (E[R])
  - Maximise risk efficiency (E[R]/SD)

Stock	E[R]	SD	E[R]/ SD
A	5%	1.2%	4.16
E	7%	2.2%	3.18
C	10%	4%	2.5
D	2%	0.8%	2.5
B	8%	4.5%	1.77

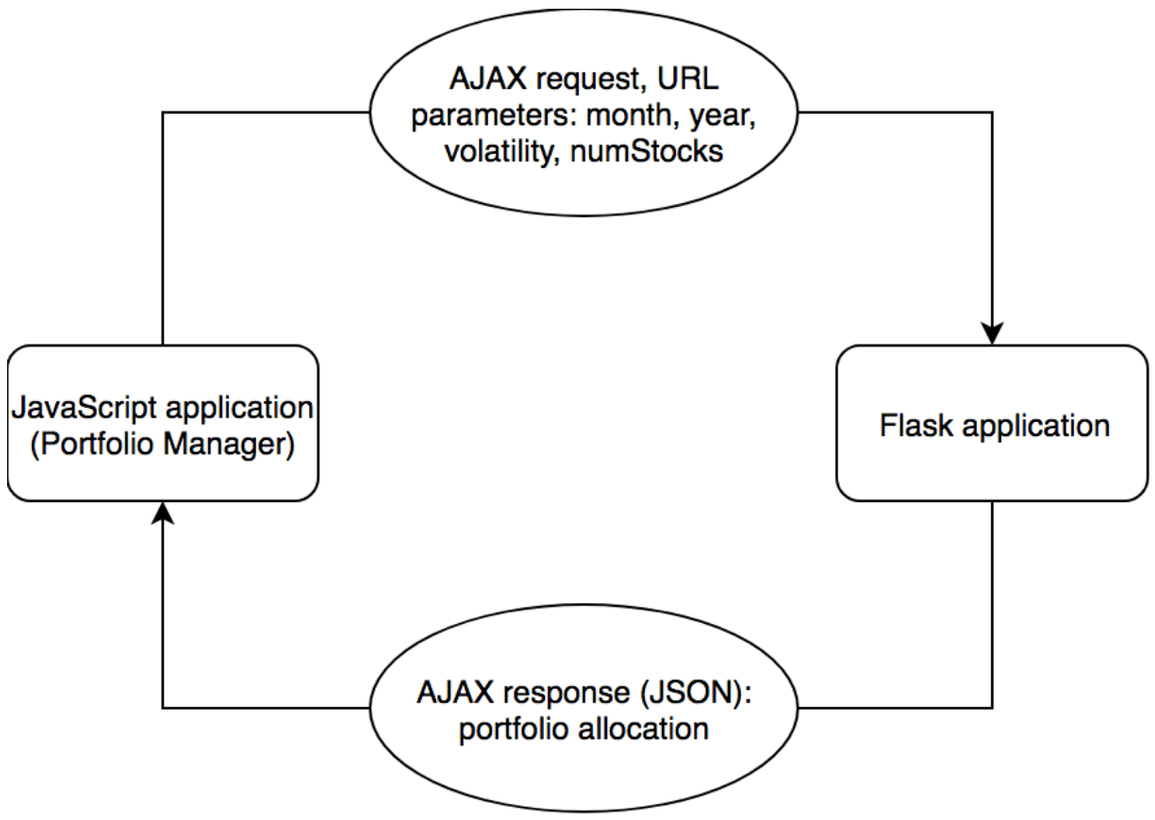
ALLOCATOR SCRIPT IMPLEMENTATION

1 User provides input through web application

2 Processing input to obtain parameters

3 Covariance Matrix constructed and Convex Optimisation done using cvxopt library

4 Results returned to JavaScript application



WEB APPLICATION

INTERACTIVE USER INTERFACE FOR MANAGING, TRACKING CHANGES TO PORTFOLIO

Web Application

FRAMEWORKS AND TOOLS

Component	Purpose
HTML5, CSS	Styling web pages
Bootstrap	Styling components of
AngularJS	Backend application logic
D3.js	Render charts and graphs using SVG components
jQuery	Application logic for front-end components' behaviours
Flask	Develop front-to-back end applications in Python, used for running allocation script
Firebase	Services like Authentication, NoSQL user database

SERVICES OFFERED

1. Authentication using social network APIs - Google, Facebook

2. **Stocks Analyser**
   Graphical representation of historical prices and predicted price for upcoming month for all stocks

3. **Portfolio Manager**
   View current portfolio constituents, ratios and growth.
   Optimise portfolio using custom parameters.

4. **Portfolio Growth Analyser**
   Evaluate growth over time
   Compare growth with that of benchmarks

DEMO

TESTING AND EVALUATION

PRICE PREDICTION MODEL TESTING

1 Debugging and testing

2 **Loss function** (during model training)**:**

- RMSE (Root Mean Square Error) - LSTM
- R^2 loss - Linear and Multiple Linear Regression

$$RMSE = \sqrt[2]{\frac{\sum(y_{true} - y_{predicted})^2}{n}}$$

$$R^2 = 1 - \frac{\sum(y_{true} - y_{predicted})^2}{\sum(y_{true} - \frac{\sum_{i=0}^{n} y_{true}}{n})^2}$$

PRICE PREDICTION MODEL EVALUATION

1 Portfolio Growth Analyser feature of Web Application

2 **Multiple Linear Regression** gave best, most consistent results across all stocks

ASSET ALLOCATION MODEL TESTING

1 White box testing - Pylint for syntax and coding errors

Black box testing -

2 CPU usage, memory, context switching statistics to check for memory leaks in convex optimisation component

3 Manual checks for formats, validation of value ranges

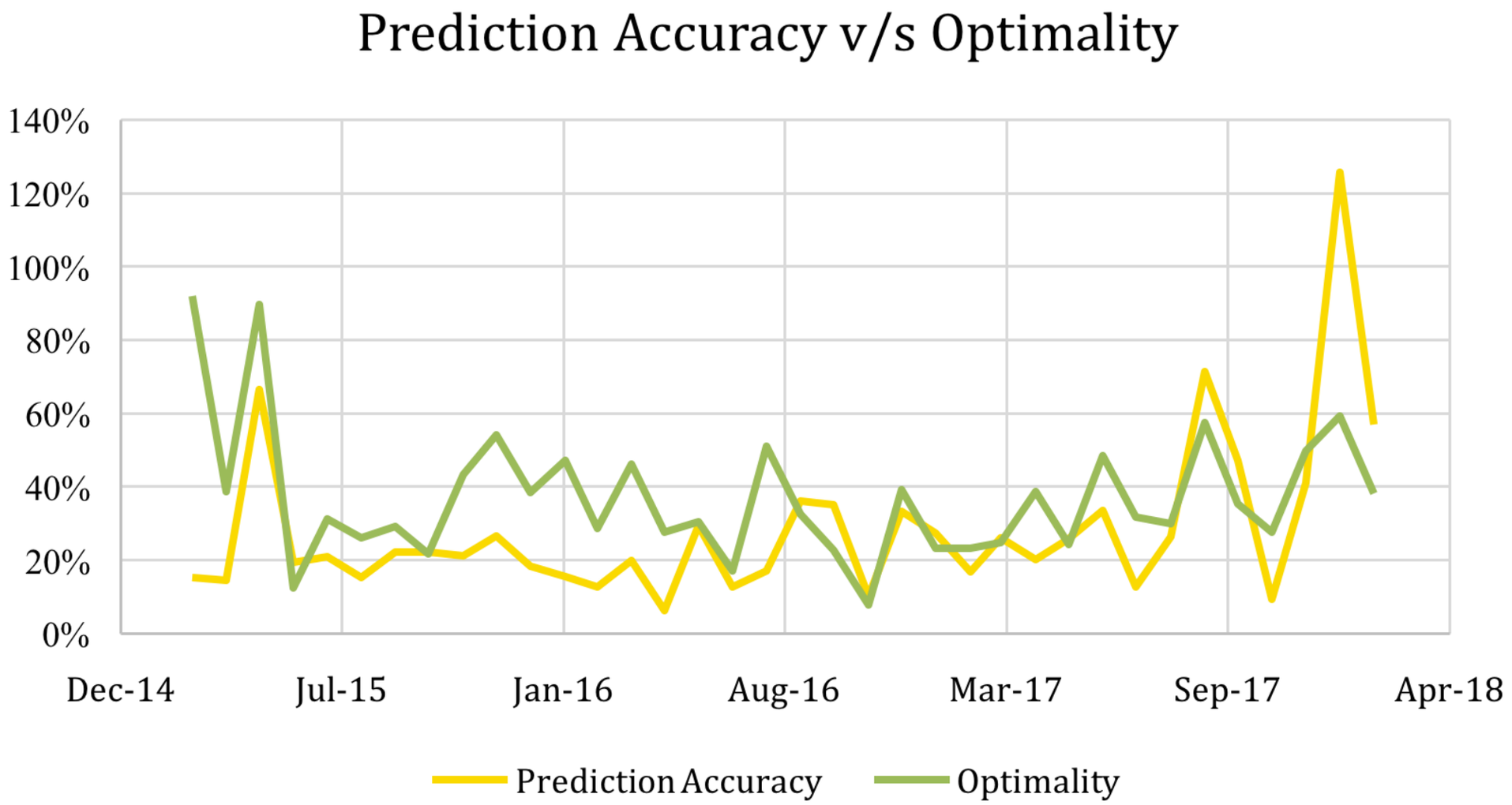
ASSET ALLOCATION MODEL EVALUATION

1 Beat benchmarks in **35 out of 36** simulated months

2 $$Optimality = \frac{(actual\ portfolio\ growth)}{(optimal\ portfolio\ growth)} \times 100$$

3 $$Prediction\ Accuracy = \frac{(actual\ portfolio\ growth)}{(predicted\ portfolio\ growth)} \times 100$$

ASSET ALLOCATION MODEL EVALUATION



WEB APPLICATION EVALUATION

Usability Testing	Average Rating
Usability of Login Page	4.2 / 5.0
Usability of Services Page	4.7 / 5.0
Usability of Stocks Explorer Page	4.1 / 5.0
Usability of Portfolio Manager Page	4.4 / 5.0
Usability of Portfolio Growth Analyser Page	4.4 / 5.0

DISCUSSION AND CONCLUSION

CHALLENGES FACED

1. Data collection and preprocessing for consistency
2. Accurate prediction of stocks prices over time
3. Adaptation of portfolio allocation theories for price prediction models generated using machine learning techniques
4. Integration of Flask application into web application

FINAL THOUGHTS

- Expectation that LSTM would perform better than multiple linear regression.
    - Overfitting
        - Limitation of resources, computation power, time
- No inclusion of transaction fees in calculation of portfolio growth
    - Real life limitations beyond scope of our project

FURTHER AREAS OF EXPANSION/IMPROVEMENT

- Try more machine learning algorithms
- Incorporate other portfolio theories
- Improve current algorithm to increase prediction accuracy
- Inclusion of non-financial data like tweets, weather data, Google Trends results.

THANK YOU!

QUESTIONS?