Stock Market Prediction

BERNARD CHENG (1002053)
DENNY BAHAR (1001579)
VICTOR TOH (1002090)

Outline

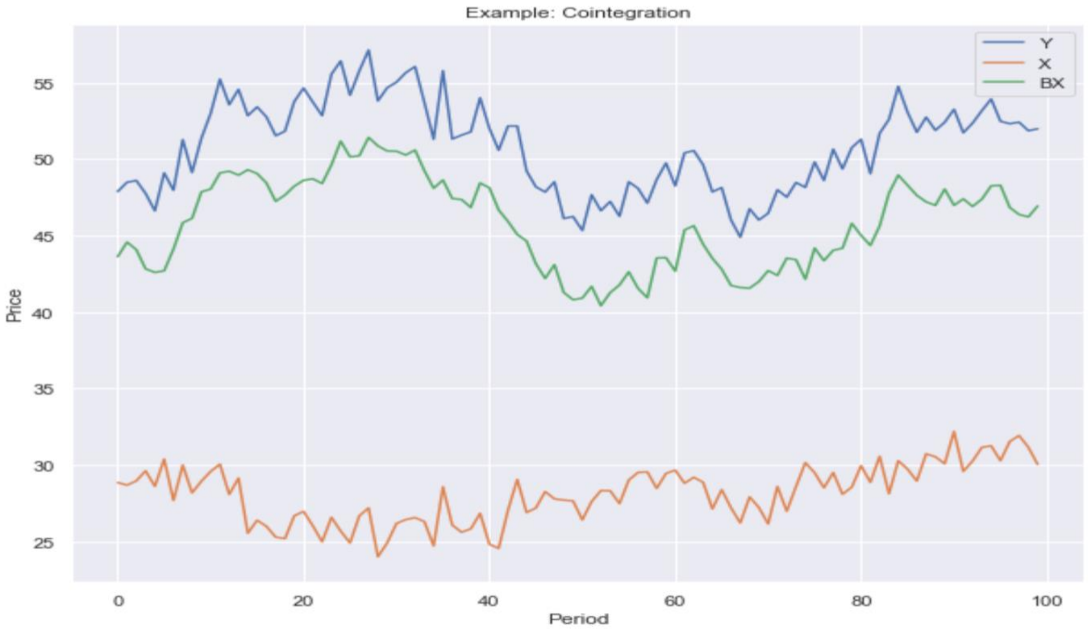
1) Data overview and Methodology
2) Model selection and Benchmark
3) Machine Learning Algorithms
4) Recurrent Neural Network
5) Future Improvements

Project goal

- Examine the possibility to predict stock market based only on the closing price information.
- Examine the effect of including related stocks in the prediction.
- Predict the stock direction (classification) and price change (regression) some period in the future.

Methodology

- Find a pair of related stock by measuring their cointegration.
- Using the cointegration information to improve prediction.
- Difference between cointegrating pair is mean-reverting.



Data description

- Usually related stocks have certain degree of cointegration.
- 12 pairs of financial data taken from various industries (24 markets).
- Data taken from Bloomberg Terminal in CSV format (Open, High, Low, Close, Volume).
- Data period: 1 Jan 1991 - 31 Dec 2017.

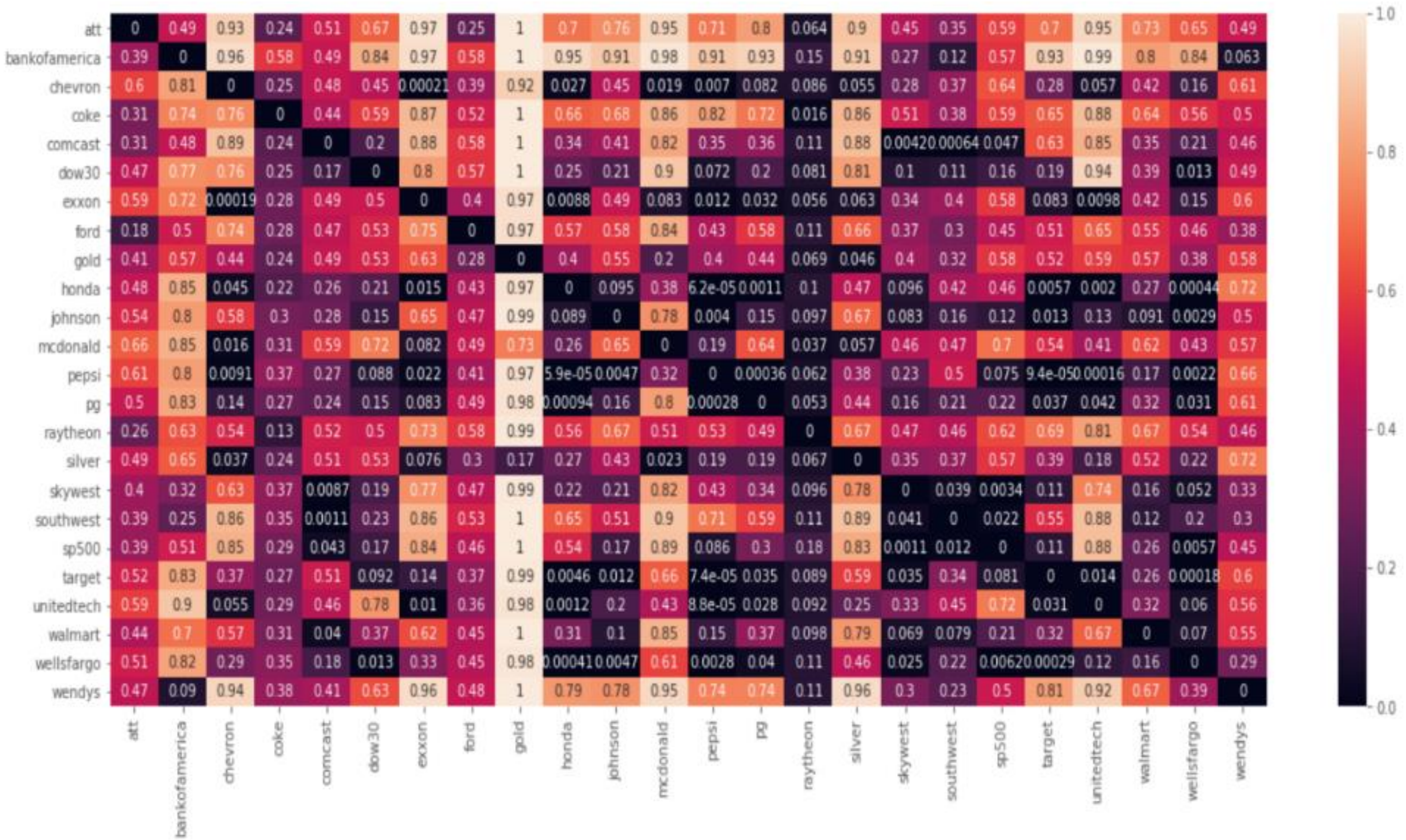
Data description

Pair	Security 1	Security 2	Industry
1	Johnson & Johnson	P&G	Household
2	S&P500	Down Jones Industrial 30	Index
3	Coca Cola	Pepsi	F&B
4	McDonald	Wendy's	F&B
5	Exxon	Chevron	Energy
6	Walmart	Target	Retail
7	Bank of America	Wells Fargo	Financial Service
8	Gold	Silver	Metal
9	United Technologies	Raytheon	Aerospace & Defence
10	Southwest	Skywest	Airline
11	Comcast	AT&T	Telecommunication
12	Honda	Ford	Automobile

Cointegration test

- **Step 1:** Select a pair of stocks.
- **Step 2:** Perform linear regression with 1 stock as the dependent variable and the other as the independent variable.
- **Step 3:** Perform Augment Dickey-Fuller (ADF) test on the residuals. (Ho: Random walk, Ha: Mean-reverting)
- **Step 4:** Select pair with low p-value and fundamentally sound.

Cointegration matrix

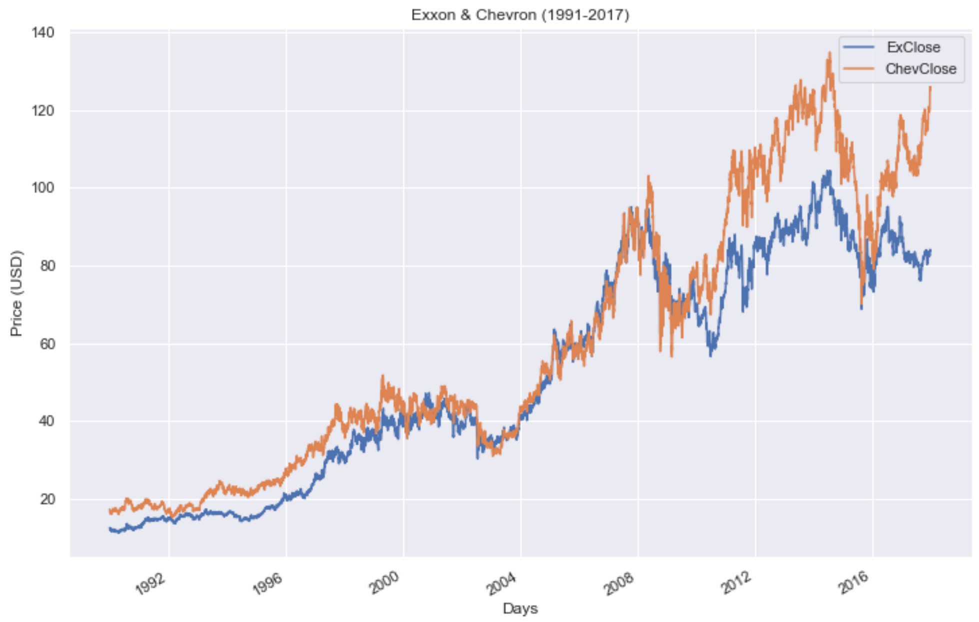


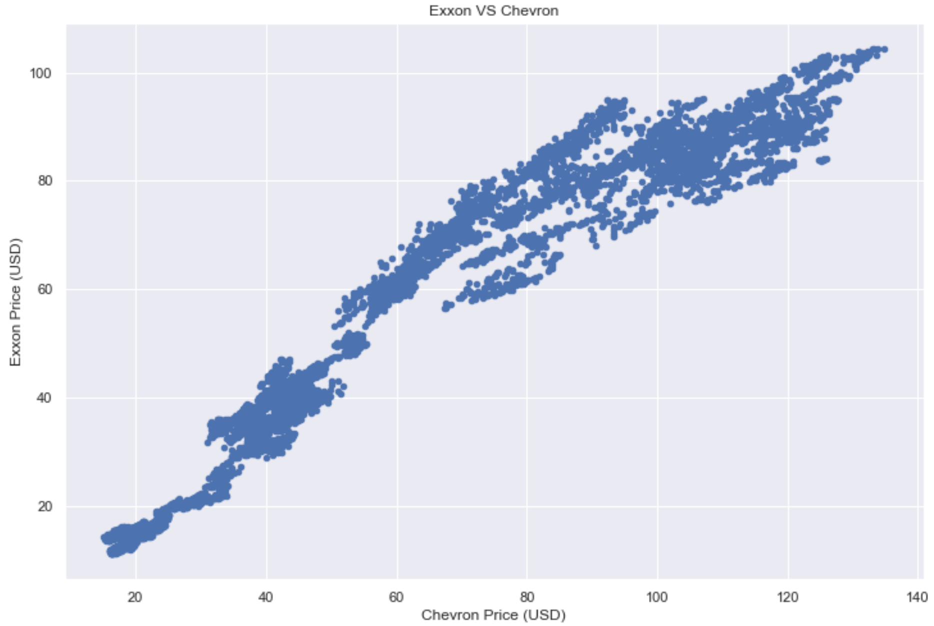
Select cointegrated pair

No.	Pairs	P-value
1	Pepsi ~ Honda	0.000059
2	Target ~ Pepsi	0.000074
3	United Tech ~ Pepsi	0.000088
4	Target ~ Wells Fargo	0.000175
5	Exxon ~ Chevron	0.000187
6	P&G ~ Pepsi	0.000276
7	Wells Fargo ~ Honda	0.000408
8	Comcast ~ Southwest	0.000636
9	P&G ~ Honda	0.000940
10	S&P500 ~ Skywest	0.001051

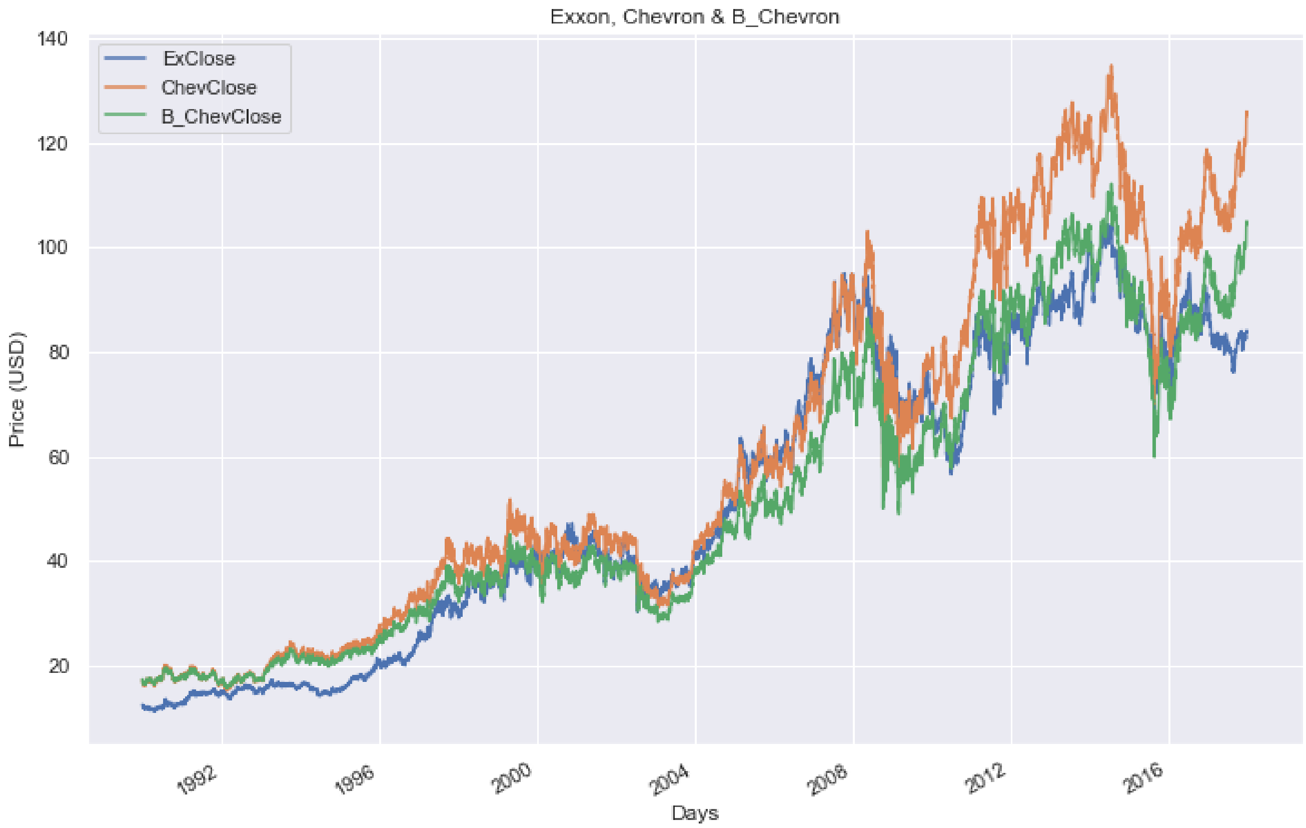
- Exxon-Chevron pair seems to be the pair which makes sense.
- Focus to predict Exxon using its relationship with Chevron to improve prediction.

Exxon-Chevron





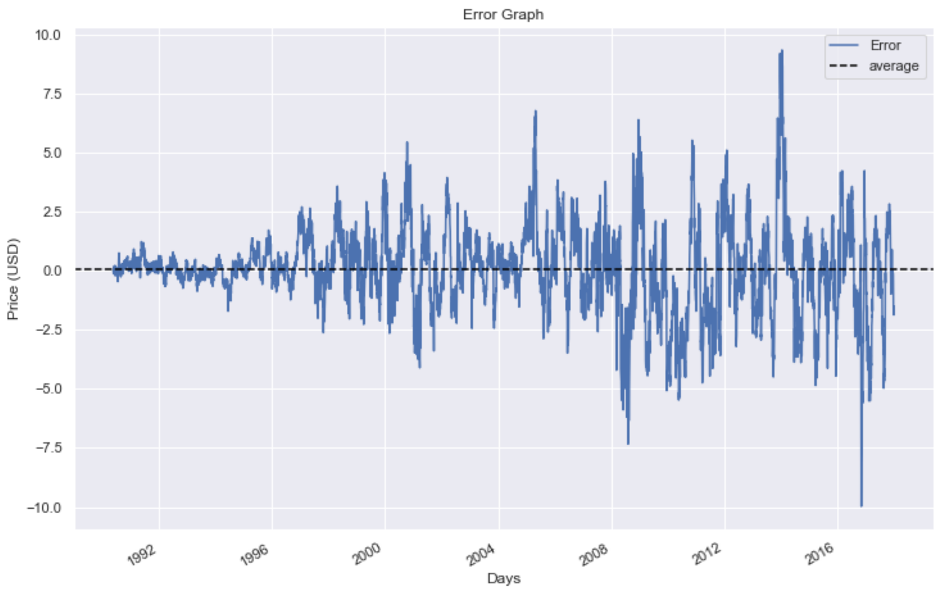
Linearly transformed Chevron



B_ChevClose = lm(ExClose ~ ChevClose)

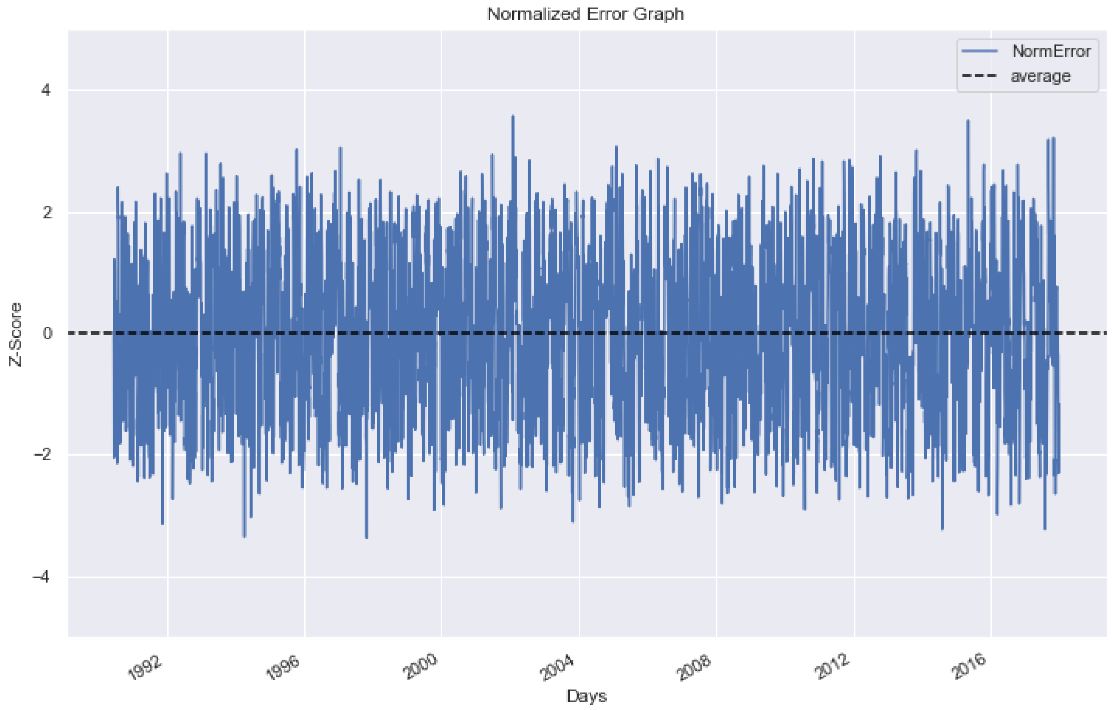
Error plot

- The error between Exxon and B_Chevron is mean-reverting.
- Mean-reverting series can be describe using *Ornstein – Uhlenbeck process*.
- Expected half-life period = 16 days to revert to its mean.



Normalized error

- Normalized using 16-day moving average and 16-day moving standard deviation.



Problem definition

- **Problem 1:** Predict Exxon stock price direction 16 days in the future (classification).
- **Problem 2:** Predict percentage price change of Exxon stock price 16 days in the future (regression).
- Predictor variables:
  - 50-day Exxon momentum
  - 50-day Exxon volatility
  - 50-day Chevron momentum
  - 50-day Chevron volatility
  - 50-day correlation
  - 50-day ADF p-value
  - Normalized error

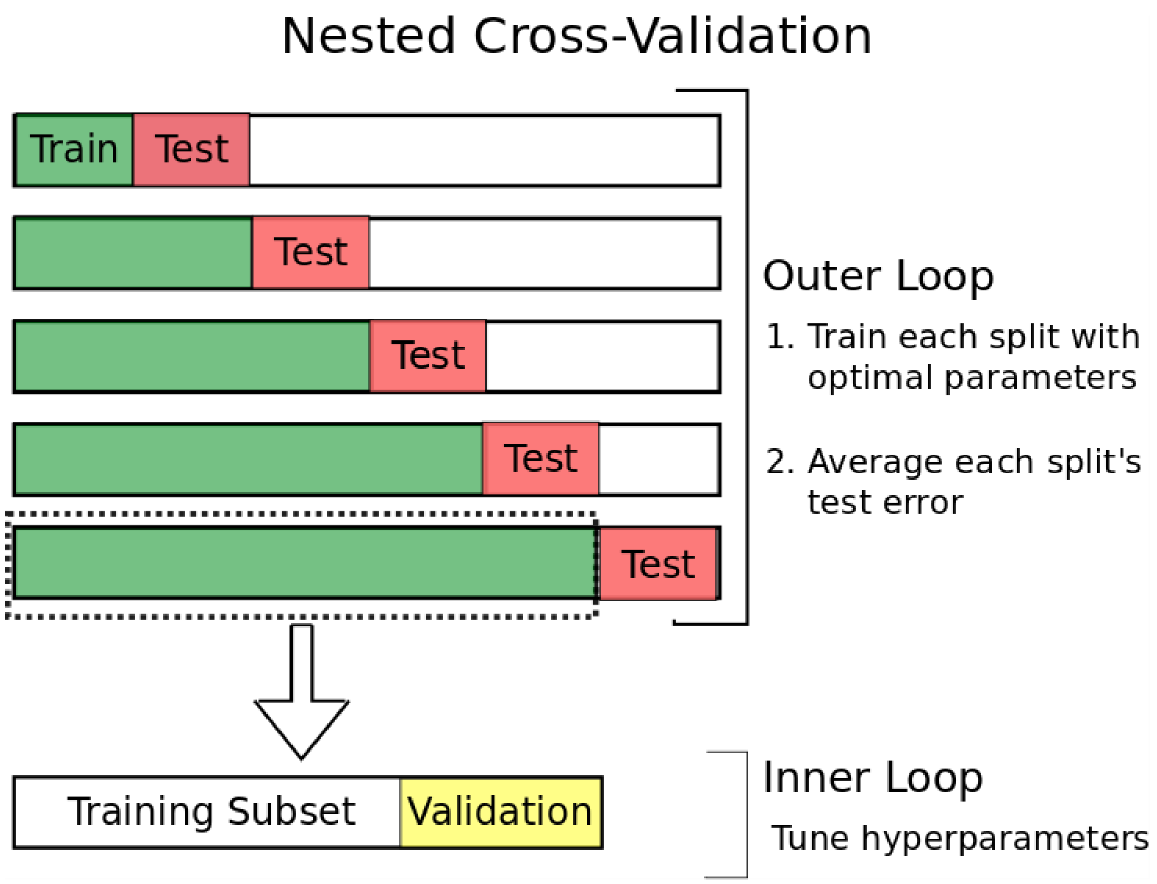
Models Selection

Classification	Regression
Logistic Regression	Linear Regression
Random Forest Algorithm	Random Forest Algorithm
Recurrent Neural Network	Recurrent Neural Network

Model Benchmarks

- Classification Benchmark: Naïve estimate based on market randomness and volatility
- Regression Benchmark: Squared difference between the price change index and the 50-day moving average for Exxon stock.

Problem Type	Evaluation Metric	Benchmark
Classification	Accuracy	50%
Regression	Mean Squared Error (MSE)	0.00197



Model Evaluation

- To account for temporal dependencies
- Removes bias from any arbitrary train-test split in conventional k-folds validation methods
- Provides almost unbiased estimate of the true training and test error
- For practicality reasons, the number of folds used for model evaluation is set at 5.

Machine Learning Algorithms (Classification)

Key Findings:

- Machine learning models manage to beat the benchmark
- Addition of pairwise related features slightly improves the accuracy of the model.

Features	Accuracy	
	Logistic Regression	Random Forest
Exxon variables only	0.568	0.520
All	0.570	0.529
Exxon + ADF, Corr, Momentum	0.570	0.531
Exxon + Corr, Momentum, Normalised	0.563	0.526
Exxon + Volatility, Normalised, Momentum	0.565	0.510

Machine Learning Algorithms (Regression)

Key Findings:

- Results did not beat the benchmark, but remains reasonably close
- For Random Forest, addition of pairwise related features slightly helps to fit the model

Features	Mean of Squared Errors (MSE)	
	Random Forest	Linear Regression
Exxon variables only	0.00259	0.00223
All	0.00251	0.00229
Exxon + ADF, Corr, Momentum	0.00253	0.00227
Exxon + Corr, Momentum, Normalised	0.00250	0.00224
Exxon + Volatility, Normalised, Momentum	0.00247	0.00226

RNN Architecture

- Rationale : To model temoporal aspect as context
- Window size used: 16 days (Expected half-life period for mean-reversion)
- Grid Search to find optimal RNN architecture from evaluation score of nested cross-validation
  - Number of hidden units
  - Number of hidden layers

Problem Type	Input Feature Set	Best Performing Model Architecture
Classification	Exxon Variables	LSTM (128), Dense (2)
	All Variables	LSTM (64), Dense (2)
Regression	Exxon Variables	LSTM (128), Dense (1)
	All Variables	LSTM (64), LSTM (64), Dense (1)

Classification Results: RNN model

Input Feature Set	Best Performing Model Architecture	Mean Test Accuracy	Benchmark
Exxon Variables	LSTM (128), Dense (2)	56.34512493%	50%
All Variables	LSTM (64), Dense (2)	56.39822227%	50%

Key Findings:

- Simpler models tend to result in higher mean test accuracy
- Increasing number of hidden units & hidden layers tends to improve mean test accuracy for training using Exxon variables
- Results suggest that using a simplier model is better when training on more input features

Regression Results: RNN model

Input Feature Set	Best Performing Model Architecture	Mean Test MSE	Benchmark
Exxon Variables	LSTM (128), Dense (1)	0.00222	0.00197
All Variables	LSTM (64), LSTM (64), Dense (1)	0.00233	0.00197

Key Findings:

- Increasing number of training epochs results in marginal improvements in mean test MSE
- Increasing number of hidden units for the LSTM layer tends to improve mean test MSE
- Increasing number of hidden layers improves mean test MSE when using all input features
- Balance between increasing number of hidden layers and hidden units is necessary to get optimal test MSE

Using Exxon Variables

Using all variables

Future Improvements

- Presently, predictors taken into consideration based solely on closing price
- Possibly of incorporating econometrics/other stock financial fundamentals
- Ensemble learning method which considers multiple algorithms

Thank you