Project Report: Statistical Arbitrage

Shubham Patil • 19-03-2020
Eckovation
Machine Learning

Overview

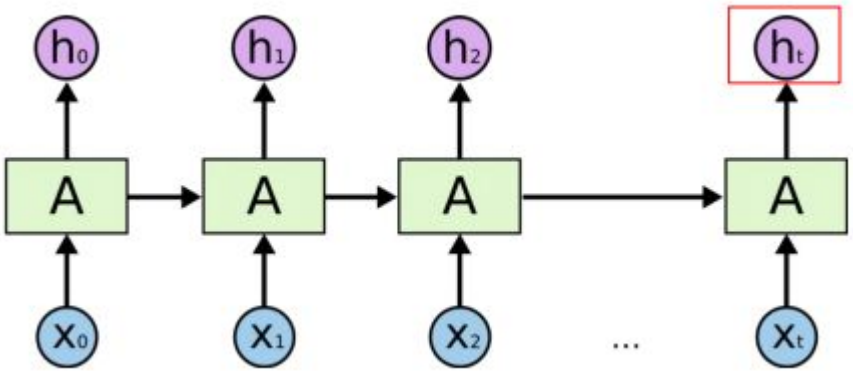
Deep Learning Strategy

- RNN Overview
- Feature and Label Generation
- Model Formation
- Strategy
- Results

Statistical Arbitrage Strategy

- Statistical Arbitrage Overview
- Finding Correlated Pairs
- Stochastic Control
- Parameter Tuning
- Results

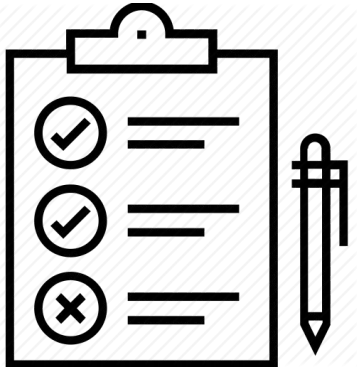
Recurrent Neural Networks (RNN)



What is RNN?

- Family of Neural Network specialized for sequence data.
- ‘Many-to-One’ architecture.
- ‘Vanilla’ vs. Long short-term memory (LSTM)

RNN: Feature and Label Generation



Feature

- Bid/Ask Prices and Spread (10 levels)
- Volumes (10 levels)
- Mean Prices and Volumes
- Accumulated Price and Volume Differences
- Price and Volume Changes
- Order Imbalance Changes
- VWAP

Label Generation

- Mid-Price Movement
- Volume Weighted Average Price (VWAP) Movement
- Settled on classifying VWAP movement over the next time ‘window’

RNN: Model Formation

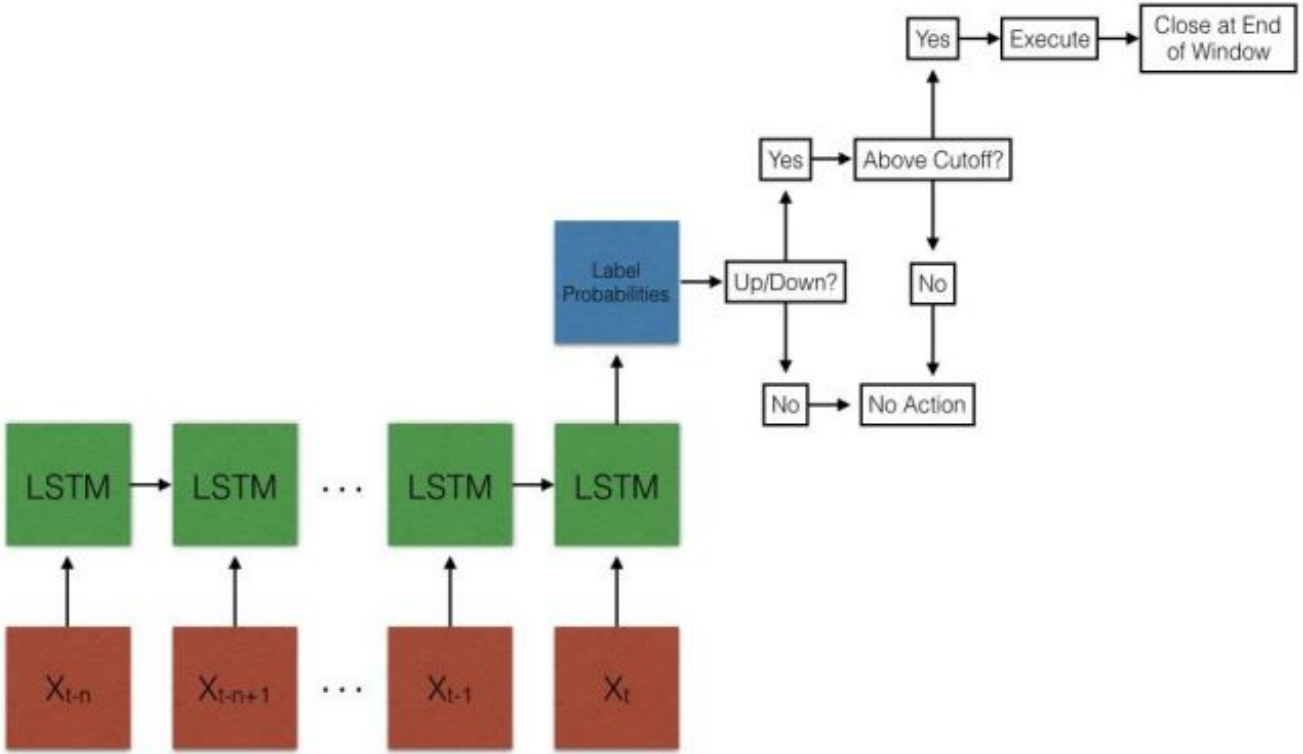
Model:

- Cost Function: Weighted Cross Entropy ○ Helps solve challenge of having an imbalanced dataset
- Output: Softmax Layer
  Outputs a predicted probability for each label
- Unit: LSTM
  Long short-term memory (LSTM) units to model longer term dependencies

Hyperparameter:

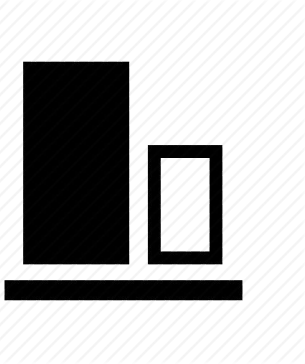
- Number of Units
- Prediction Window for Label
- Trade Probability Cutoff
- Cross Entropy Weights
- Other (e.g. Learning Rate, Dropout)

RNN: Strategy



Statistical Arbitrage Strategy

Baseline model

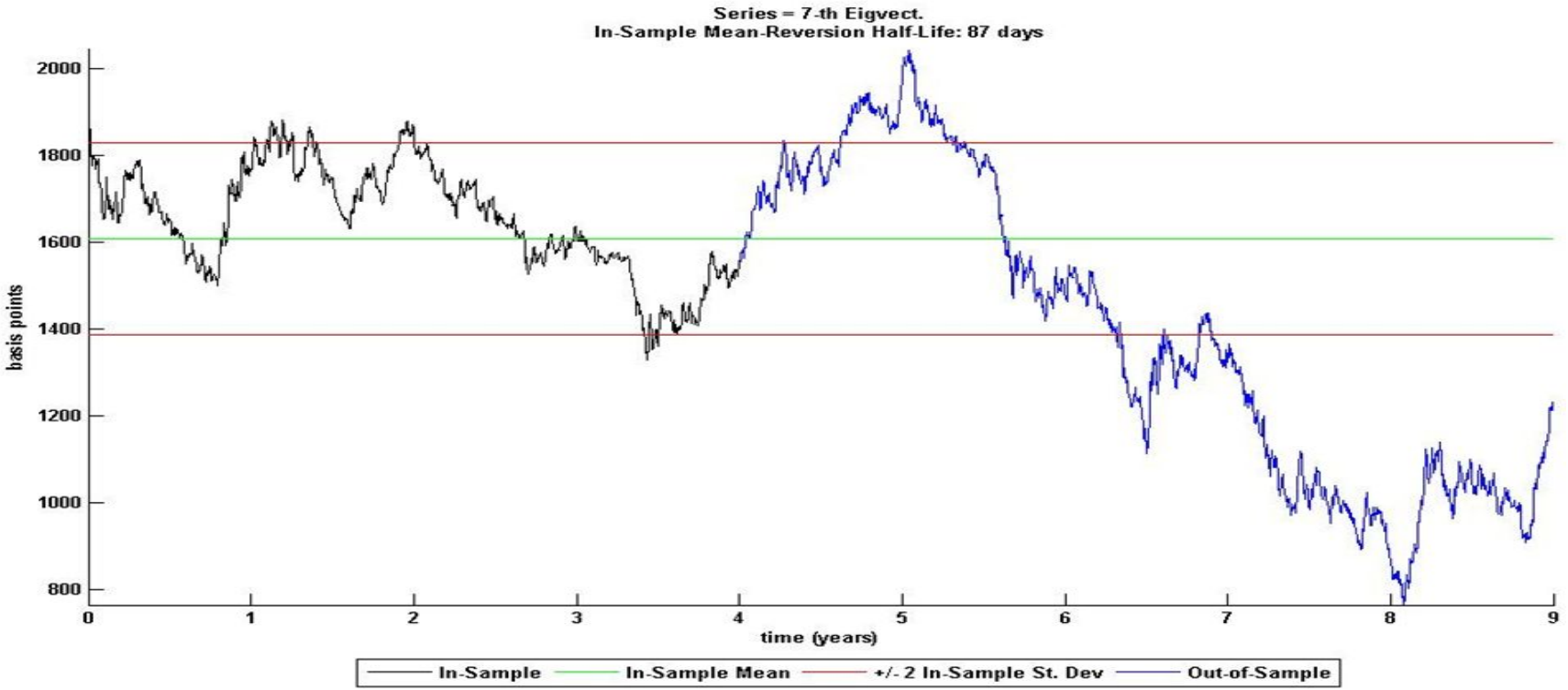


Linearly regress the mid-price returns of a pair of historically correlated stocks.

This type of trading strategy assigns stocks a desirability ranking and then constructs a portfolio to reduce risk as much as possible.

Statistical arbitrage is heavily reliant on computer models and analysis and is known as one of the most rigorous approach to investing.

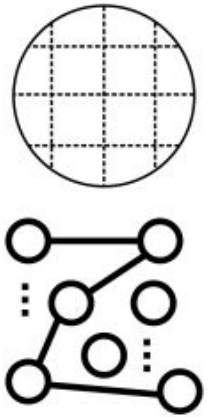
Example of execution process



What's new?

- Identifying most correlated pairs to trade
- Stochastic control to incorporate dynamically optimal thresholds
- Hyperparameter tuning (frequency, training size, leverage, etc.)

Parameter Tuning



Two approaches to parameter tuning:

- Grid search
  - Systematic exploration
  - Enables for sensitivity analysis
  - Inefficient

- Random search
  - Black-box method
  - Explore larger subspace

Validation set

- Evaluated 4 models

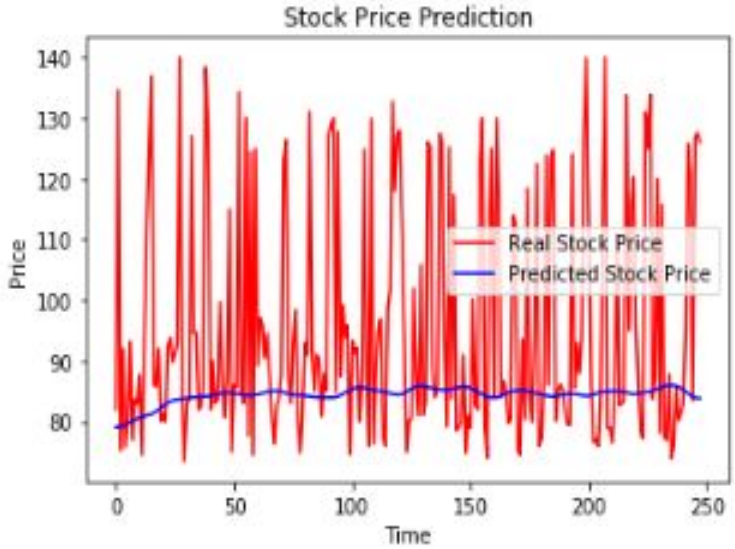
a. Fixed thresholds, pairs picked by performance
b. Fixed thresholds, pairs picked by correlation
c. Stochastic control thresholds, pairs picked by performance
d. Stochastic control thresholds, pairs picked by correlation

- Model

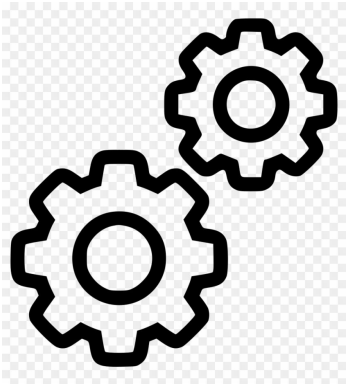
a. performed best on validation set

Test Set Result

In [327]:
```
plt.plot(real_stock_price,color="red",label="Real Stock Price")
plt.plot(predicted_stock_price,color="blue",label="Predicted Stock Price")
plt.title("Stock Price Prediction")
plt.xlabel("Time")
plt.ylabel("Price")
plt.legend()
plt.show()
```



Future Work



To trade based on factors from PCA eigen portfolio and its eigenvalues:

○ take a variable number of eigenvectors, truncate to explain a given percentage of the total variance of the system

Implement a more dynamic strategy

○ Using the correlation from yesterday to decide which pairs to trade today.

○ Or observe the market for a couple of hours and then start trading based on earlier correlation

Thank You.

Shubham Balasaheb Patil
Eckovation
Machine Learning