Online non-linear prediction of financial time-series patterns

Mr Joel da Costa
Supervisor: Prof. Tim Gebbie

University of Cape Town
MSc. Advanced Analytics (STA5004W) Research Proposal

This is a technical document, which serves as an accompaniment to the project proposal at:

The purposes of this document are twofold: firstly, to provide a more detailed overview and explanation of the project implementation, and secondly, to provide an updated state of affairs for the project.

Current Situation

- Finalisation of proposal documents and plans
- Learning Julia

Future Situation

- All initial proposal documents to be finalised
- FFT Data Interpolation library complete

Figures and Tables

To be updated during the course of the project

Milestones and Project Deliverables

Date	Item	Deliverable	Dependency
31/03	1 Literature Review & Learn Julia Basics	Literature Review	None
21/04	2 FFT Data Interpolation	FFT Interpolation Library	None
15/05	3 Synthetic Data Generation	Synthetic Data Collection	None
31/05	4 Data preprocessing on surrogate data	Processing Library	Collection of surrogate data set
21/06	5 Dimensional Reduction	Auto Encoder Class	3 and/or 4
10/07	6 Finalise Data Collection	Final Dataset	None
05/08	7 Offline Neural Network	Training Library, Model, Validation Test Outputs	2 - 6
31/08	8 Online Neural Network	Process Library & Model	7
30/09	9 Backtesting Module	Process Library & Statistical Test Results	8
31/10	10 Testing on Extended Datasets	Models and Statistical Test Results	9
30/11	11 Dissertation Write up and Revisions	Final Hand In	10

Requirements Specification

Hardware

A macbook pro will be use for the crux of the development, with the following specs:

- 2,8 GHz Intel Core i7
- 16 GB 2133 MHz LPDDR3
- SSD

Software & Packages

Julia will be the primary programming language used to develop the project, though Python and R will also be used interchangeably for Exploratory Data Analysis (EDA), and as needed.

The list of external libraries within these languages will be added here as the project progresses.

Data Science Workflow

1. **FFT Interpolation**

   Process Input
   Daily OHLCV data for a single stock, with missing data for some days. This is a 6xN data matrix, for N days.

   Process Output
   Full data set of OHLCV data for a single stock, also 6xN

   Technique
   FFT Interpolation - pre existing libraries can be used for this

   Validation
   Spot checking can be done to make sure the values have been interpolated correctly

2. **Data Preprocessing**

   Process Input
   Full data set from step 1 output

   Process Output
   A KxN data matrix, for N days and K features

   Technique
   The data will use a custom preprocessing technique, used to to layer different time slices of data - daily, weekly and monthly. The actual values will be the log feature fluctuations (i.e. log(yt) - log(yt-1)) of the OHLCV data for the relevant time slice. Thus, if daily, weekly and monthly sampling is done, K will be 15 (OHLCV log feature fluctuation values for each slice)

   Validation
   Code unit testing and spot checking can be done to verify the preprocessing was done correctly

3. **Dimension Reduction**

   Model Input
   K dimension vector

   Model Output
   L dimension vector

Technique
An unsupervised AutoEncoder will be used to perform feature selection/dimensional reduction

Validation
Initially this will be kept simple, and compared to PCA values to verify correctness. A more complex auto encoder model may then be used to increase performance

4. **Offline Neural Network**

Model Input
L dimension vector

Model Output
n*5 dimension vector - the next n feature fluctuations for each of the OHLCV figures

Technique
Standard neural network, trained using stochastic gradient descent

Validation
Walk forward validation can be used with an expanding window, as this will be best suited to the Online usage of the model below

5. **Online Neural Network**

Process Input
Auto encoder and trained neural network

Process Output
Online deep network

Technique
The online network will be a deep network made up of the auto encoder and trained neural network from previous steps. The network input will be a new sample of the preprocessed data as from step 2, and the output will be the n-predicted feature fluctuations, as per step 4. After the prediction is made, the network will perform an online update using gradient descent.

Validation
Backtesting module, using methodologies as detailed by Bailey et al [1][2].

Research Workflow

Each of the above Data Science steps will be done be achieved by following the process steps detailed below.

1. Problem Definition

   The problem, and solution are defined here. In context of the above steps, there would be clear definitions of the input data to be used, output data to be expected, and the details of the technique used to achieve this.

2. Data Processing

   At this step the data is prepared to be in an appropriate input format, as defined in step 1.

   Initial data collection may be done in Java in order to interface with APIs, though after that it is expected this step will be performed in Julia.

3. Data Exploration

   Data exploration will be conducted in order to make sure that the understanding of the data is correct - summary statistics may be used to this end (e.g. in the case of synthetic data sets, there will be clear expectations here). Visualisation will also be used as a technique to check this. Either of these may be done in R or Python as considered most convenient for the task at hand.

4. Baseline Modeling

   In the case where it is a modelling task, a baseline model will be used as a reference point going forward. For the AutoEncoder, this will be a simple configuration that could be compared to PCA, or for the neural network it could be a single layer configuration to measure improvements against.

5. Secondary Modeling

   This stage is the more experimental stage of modelling. It is expected that numerous configurations for a model will be considered before arriving at the final model. In line with this, the modeling may go through it's own iterative machine learning process of parameter exploration, training and validation.

6. Testing

   In some instances, more generalised statistical testing will be performed, e.g.

statistical arbitrage tests.

7. Process Preparation

   Once the above steps have been completed, the current stage will be prepared as an input to the next one - e.g. a consumable model, or library which performs a task.

Data

Data will be collected from Bloomberg and Thomson-Reuters through the available API's, most likely using JAVA. The full data requirements and specifications, as set out in the project proposal, are below:

- A surrogate dataset of one or two stocks will be used to start development
- Thereafter a fuller set of data will need to be collected, from Bloomberg and Thomson-Reuters
- Bloomberg OHLCV daily data for 20 years will be considered in the stock combinations as detailed in the table below
- Thomson Reuters Tick History intraday data consisting of top-of-book and transaction updates for the same stocks as listed in the table below
    - Data will be processed to create 5 minute and 10 minute bars from the intraday data as well as volume time bars to be used as an input to the online learning algorithm
- Synthetic data cases (Eg. Monte Carlo simulations) will also be considered in order to discuss issues encountered with in-sample versus out-of-sample backtesting
    - Examples of such data cases would be where stocks are all increasing/decreasing over time, or both for a combinations of stocks.
- As detailed in table 1, there are 5 primary stocks, each of which will be considered in various Stock, Equity Index and Bond Index pairs, as well as by themselves. E.g. For AGL, the following will be considered: AGL, AGL and BHP, AGL and ALSI40 and AGL and ALBI
- The Daily TRI (Total Return Index) OHLCV for 20 years will be considered for all pairs, and the Intraday data will be considered for the Single and Stock pairs)

Stock	**Stock Pair**	**Equity Index**	**Bond Index**
AGL	BHP	ALSI40	ALBI
SBK	SNL	ALSI40	ALBI
SHF	RCH	ALSI40	ALBI
WHL	SHP	ALSI40	ALBI
MTN	VOD	ALSI40	ALBI