|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ASSIGNMENT COVERSHEET** | | | | UTS LOGO | | | |
| **UTS: ENGINEERING & INFORMATION TECHNOLOGY** | | | | | | | |
|  | **NAME OF STUDENT(s) (PRINT CLEARLY)**  Joel Cappelli  *SURNAME FIRST NAME* | | | | | | **STUDENT ID(s).**  12137384 |
| **STUDENT EMAIL**  joel.cappell@gmail.com | | | | **STUDENT CONTACT NUMBER**  0422 384 087 | | | |
| **NAME OF TUTOR** | | | **TUTORIAL GROUP** | | | **DUE DATE**  16 June 2016 | |
| **ASSESSMENT ITEM NUMBER/ TITLE**  Hung Nguyen  49275 NEURAL NETWORKS AND FUZZY LOGIC Major Project | | | | | | | |
| 🗆 I confirm that I have read, understood and followed the guidelines for assignment submission and presentation on page 2 of this cover sheet.  🗆 I confirm that I have read, understood and followed the advice in my Subject Outline about assessment requirements.  🗆 I understand that if this assignment is submitted after the due date it may incur a penalty for lateness unless I have previously had an extension of time approved and have attached the written confirmation of this extension.  **Declaration of Originality**: The work contained in this assignment, other than that specifically attributed to another source, is that of the author(s) and has not been previously submitted for assessment. I understand that, should this declaration be found to be false, disciplinary action could be taken and penalties imposed in accordance with University policy and rules. In the statement below, I have indicated the extent to which I have collaborated with others, whom I have named.  **Statement of Collaboration**:    15 June 2016  **Signature of Student(s) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**Date**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | | | | | | | |
| - -- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - | | | | | | | |
| **ASSIGNMENT RECEIPT** | | | | To be completed by the student if a receipt is required | | | |
| **SUBJECT NAME/NUMBER** | | **NAME OF TUTOR**  49275 NEURAL NETWORKS AND FUZZY LOGIC | | | | | |
| **SIGNATURE OF TUTOR** | | | | | **RECEIVED DATE**  15 June 2016 | | |

# **49275 NEURAL NETWORKS AND FUZZY LOGIC**

## Major Project

## The use of data mining and neural networks for forecasting stock market returns

### Joel Cappelli

### 12137384 15 June 2016

## Project Overview

This project examines the performance of different forecast models of the **S&P**/**ASX 200 XJO** Australian Stock Exchange Index, end of day movement. The **S&P**/**ASX 200 XJO** is recognised as the institutional investable benchmark in Australia. The index covers approximately 80% of Australian equity market capitalisation and measures the performance of the 200 largest index-eligible stocks listed by float-adjusted market capitalization. The index was launched in April 2000.

The forecasting techniques examined were feedforward multi-layer perceptron networks (MLP), generalised regression neural network (GRNN), probabilistic neural network (PNN) and ordinary least squares (OLS). Both prediction of the level of end of day index return and index direction movement is considered. Performance of the models was evaluated by using classification rate, model metrics and trading simulations.

Results show that the PNN was the dominant predictor of the direction of next day index movement. This was proven by the highest classification rate on out-of-sample data and best performance in trading simulations. Interestingly, performance of the OLS linear model of level prediction showed very promising results close to that of the PNN. This brings into question the usefulness of non-linear models such as Neural Networks for intra-day prediction.

The methodology of this project follows that of [1].

## Literature review

There have been a total of 412 articles published between 1994 and 2015 focusing on the use of Artificial Neural Networks in the field of finance [3]. Applications in shares/bonds and bankruptcy and financial distress prediction are the most prominent [4]. Many studies in stock forecasting/trading have aimed at specially predicting the price levels of indices or stock. However, some recent studies have suggested that trading strategies guided by forecasts on the direction of price change may be more effective and may lead to higher profits [2], [1]. Forecasting period is typically one month ahead for papers authored pre-2000 and most feature New York Stock Exchange Indices (NYSE).

Most implementations of Neural Networks in this domain use feedforward multi-layer networks with gradient descent backpropagation learning techniques. Hybrid learning and self-organising architecture methods are becoming more popular as computing technology has improved and open-source implementations are readily available.

## Project aim and approach

The project approach is as follows:

* Select relevant input variables from a variety of sources such as intermarket indices, exchange rates, interest rates and commodities.
* Build next-day Neural Network forecasting models of return level and direction of the **S&P**/**ASX 200 XJO**
* Compare the performance of a variety of Network architectures through trading simulations

1. usage of the intermarket influences from the major
2. global stock market indices to forecast the trading signals of the AORD. Recent studies have
3. shown that the intermarket influences enhance the prediction accuracy
4. Therefore, this study
5. also uses the current day’s relative returns of the Close prices of the above markets to identify the
6. next day’s trading signals of the AORD.

## System design

**Input variables**

Global market indices

Treasury and/or government bonds

Dividend yield of ASX200

FX Rates, USD/Yen, USD/JPY, AUD/USD

Commodities

**Data mining and feature selection**

Information gain analysis

Compute log-diff of each input variable

Standardise inputs

Max-min outputs

Direction network

(+/-)

Stock return network

(x %)

Apply trading rules

16 years of daily data from Bloomberg, Quandl and Reserve Bank of Australia (RBA), 60%/20%/20% training/test/validation split, 10 years for training and 3 years for out-of-sample validation and test.

Log-difference of variables were provided to the network

Intra-day prediction of predict the level and to classify the sign of the movement of ASX200

In general, large-scale deterministic components, such astrends and seasonal variations, should be eliminated from the inputs since the network will attempt to learn the trend and use it in the prediction (Nelson et al., 1999; Pantazopoulos et al., 1998). Therefore, the data collected in this study, excluding DIV, T1, SP, DY, and ER, were seasonally adjusted allowing the networks to concentrate on the important details necessary for an accurate prediction (the source and definition of all the variables are given in the Appendix)

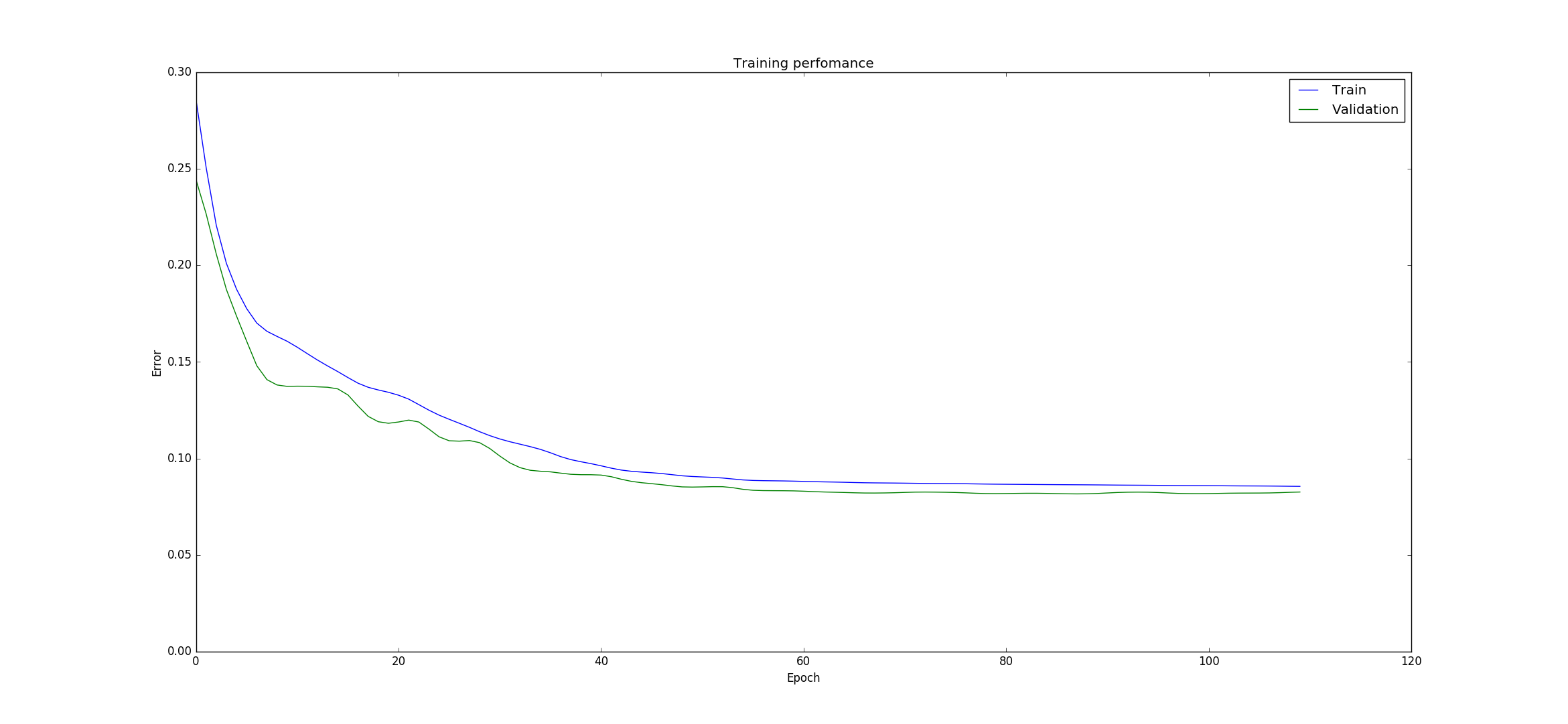
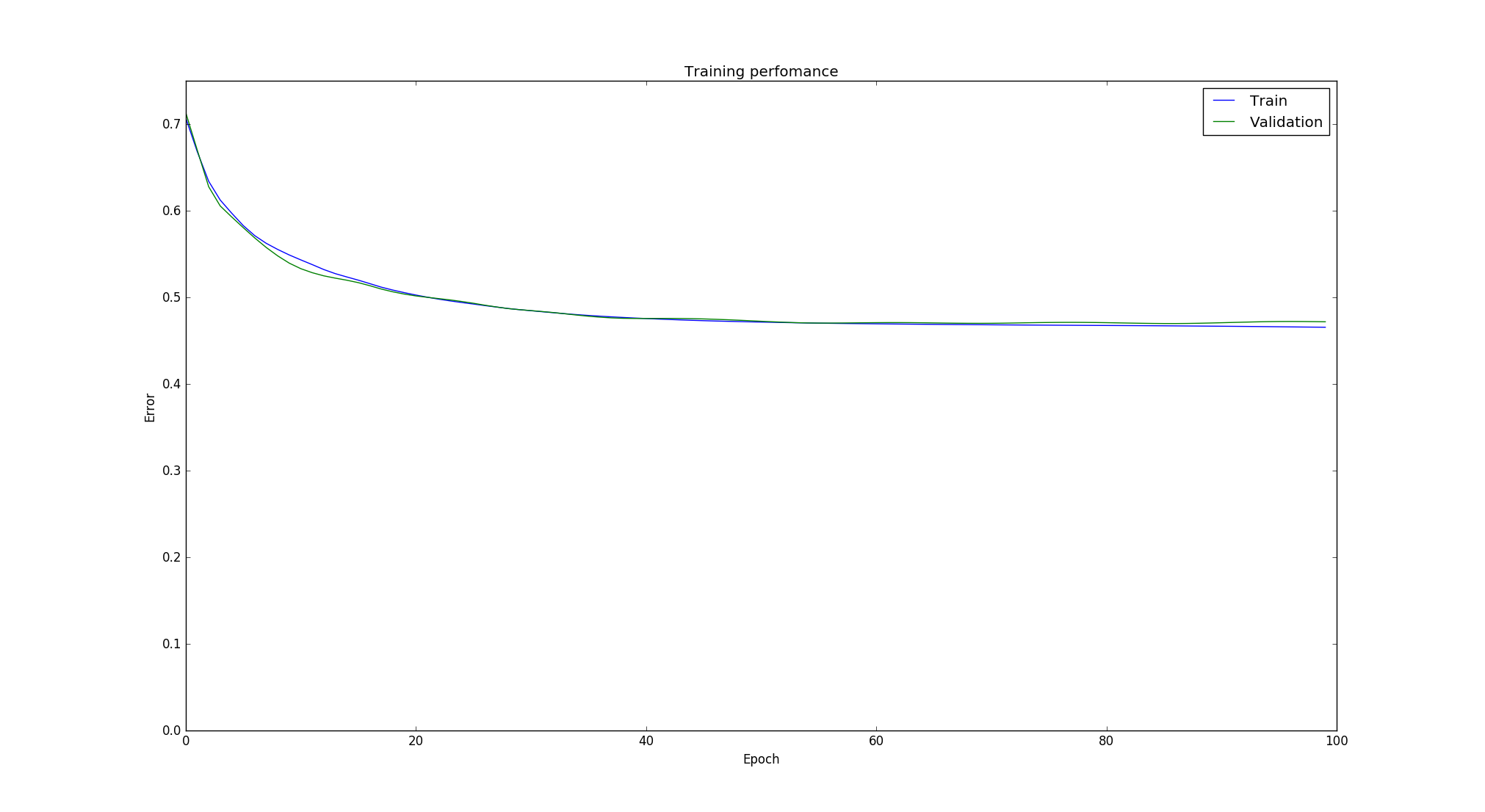
* S&P 500 (U.S.)
* S&P/TSX Financials Sector Index (Canada)
* Shanghai Composite Index (China)
* S&P/ASX 200 Daily Dividends
* Gold (Commodity)
* WTI Crude Oil (Commodity)
* AUD/USD (Currency)
* USD/JPY (Currency)
* Australian 90 Day Bank Bill
* Australian Government 10 Year Bonds
* United States 10 Year Treasury Note

## Implementation

* Python 2.7
* Utilises open-source libraries, such as NeuPy and scikit-learn.
* The input data is first cleaned and aligned based on a specified *lag* before being fed into the neural networks.
* Used conjugate gradient optimisation for network learning
* Runtime performance is very dependent on the time horizons and number of inputs we select.
* Optimal network architecture with bias at each hidden node

input layer<11>, tanh<22>,output layer<1>

Network Training



## Results and discussion

[ALGORITHM] ConjugateGradient

[ARCHITECTURE] Tanh(10) > Tanh(29) > Output(1)

Network options

Verbose:

[OPTION] verbose = True

BaseNetwork:

[OPTION] epoch\_end\_signal = None

[OPTION] show\_epoch = 10

[OPTION] shuffle\_data = False

[OPTION] step = 0.1

[OPTION] train\_end\_signal = None

ConstructableNetwork:

[OPTION] error = rmse

GradientDescent:

[OPTION] addons = ['LinearSearch']

ConjugateGradient:

[OPTION] update\_function = fletcher\_reeves

LinearSearch:

[OPTION] maxiter = 10

[OPTION] search\_method = golden

[OPTION] tol = 0.1

Test MLP Results - Intra Day Returns Performance

Mean err = -1.80020861245e-05

RMSE = 0.00770790148919

Test MLP Level Est - Classification Results

Feedforward\_MLP\_extLEVEL: Guessed 1001 out of 1670 = 59% correct

Feedforward\_MLP\_extLEVEL: Strategy PNL = 22440.326, BuyHold PNL = 5499.034

GRNN:

[OPTION] std = 1.8

Test GRNN Results - Intra Day Returns Performance

Mean err = -0.000357764985329

RMSE = 0.00794325610411

Test GRNN Level Est - Classification Results

GRNN\_Network: Guessed 1015 out of 1670 = 60% correct

GRNN\_Network: Strategy PNL = 22875.036, BuyHold PNL = 5499.034

Test OLS Results - Intra Day Returns Performance

Mean err = -0.000216915219182

RMSE = 0.00763313987633

Test OLS Level Est - Classification Results

OLS\_LevelPredictor\_ext: Guessed 1037 out of 1670 = 62% correct

OLS\_LevelPredictor\_ext: Strategy PNL = 24570.958, BuyHold PNL = 5499.034

[ALGORITHM] ConjugateGradient

[ARCHITECTURE] Tanh(11) > Tanh(27) > Output(1)

Network options

Verbose:

[OPTION] verbose = True

BaseNetwork:

[OPTION] epoch\_end\_signal = None

[OPTION] show\_epoch = 5

[OPTION] shuffle\_data = False

[OPTION] step = 0.1

[OPTION] train\_end\_signal = None

ConstructableNetwork:

[OPTION] error = rmse

GradientDescent:

[OPTION] addons = ['LinearSearch']

ConjugateGradient:

[OPTION] update\_function = fletcher\_reeves

LinearSearch:

[OPTION] maxiter = 10

[OPTION] search\_method = golden

[OPTION] tol = 0.1

[THEANO] Initializing Theano variables and functions.

[THEANO] Initialization finished sucessfully. It took 0.47 seconds

[WARN] There is no data to plot

Test MLP Direction Est - Classification Results

Feedforward\_MLP\_extDIRECTION\_SINGLE\_VEC: Guessed 1033 out of 1670 = 61% correct

Feedforward\_MLP\_extDIRECTION\_SINGLE\_VEC: Strategy PNL = 24193.394, BuyHold PNL = 5499.034

[ALGORITHM] PNN

PNN:

[OPTION] std = 1.25

Test PNN Est - Classification Results

PNN\_Network: Std dev 1.25: Guessed 1022 out of 1670 = 61% correct

PNN\_Network: Strategy PNL = 23626.884, BuyHold PNL = 5499.034

A trading simulation was developed to further compare our results to a typical buy-and-hold strategy.

We compared two hypothetical portfolios, one representing the ASX200 and another investing based on the buy/sell signals generated by our neural network.

Results look promising but further testing is required!

## Conclusion and future work

Results we obtained using RMSE and accuracy percentage were promising, and broadly inline with what was achieved in the paper.

There does appear to be inefficiencies in the market we can exploit.

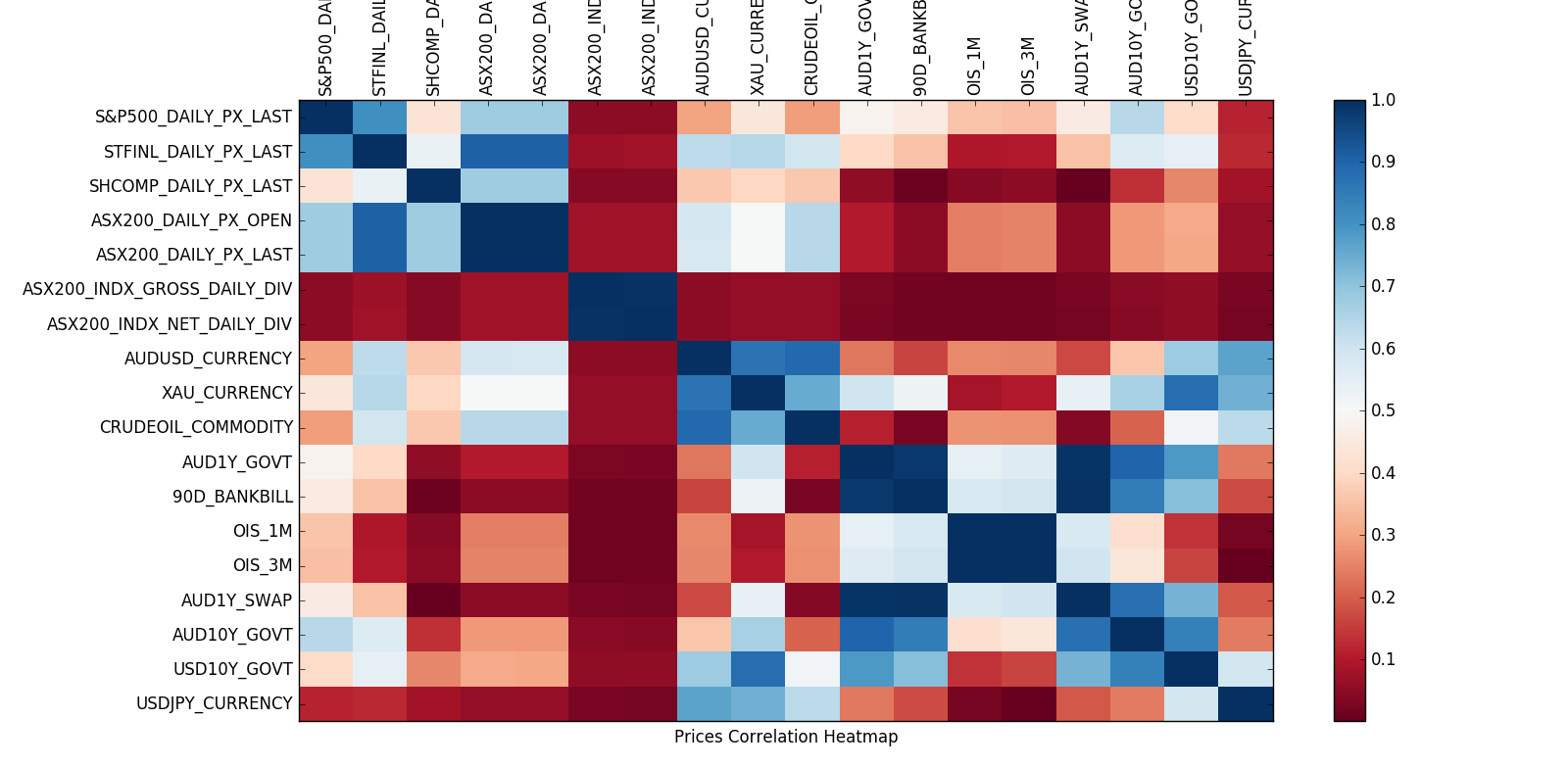
The trading simulation showed better than expected results, and could be enhanced to account for dividends, taxes and transaction costs to produce a more realistic P&L.

Further work can be done to optimise the input selection and neural network parameters.

We can also look at applying more rigorous testing against different time frames and market conditions to ensure our results are consistent and not just a statistical anomaly.

# Appendix

## Predictor Correlation Heat map



## OLS Regression

OLS Regression Results

===========================================================================

Dep. Variable: y R-squared: 0.213

Model: OLS Adj. R-squared: 0.209

Method: Least Squares F-statistic: 61.19

Date: Sun, 05 Jun 2016 Prob (F-statistic): 6.51e-121

Time: 23:41:09 Log-Likelihood: 2522.1

No. Observations: 2505 AIC: -5020.

Df Residuals: 2493 BIC: -4950.

Df Model: 11

Covariance Type: nonrobust

===========================================================================

coef std err t P>|t| [95.0% Conf. Int.]

---------------------------------------------------------------------------const 0.1626 0.002 91.649 0.000 0.159 0.166

x1 0.0328 0.002 14.280 0.000 0.028 0.037

x2 0.0097 0.002 4.728 0.000 0.006 0.014

x3 -0.0065 0.002 -3.780 0.000 -0.010 -0.003

x4 -0.0033 0.002 -1.610 0.108 -0.007 0.001

x5 0.0076 0.002 3.878 0.000 0.004 0.011

x6 0.0005 0.002 0.276 0.782 -0.003 0.004

x7 0.0007 0.002 0.402 0.688 -0.003 0.004

x8 0.0023 0.002 1.233 0.218 -0.001 0.006

x9 0.0032 0.002 1.793 0.073 -0.000 0.007

x10 0.0078 0.002 4.104 0.000 0.004 0.011

x11 0.0018 0.002 0.828 0.408 -0.003 0.006

===========================================================================

Omnibus: 375.637 Durbin-Watson: 2.272

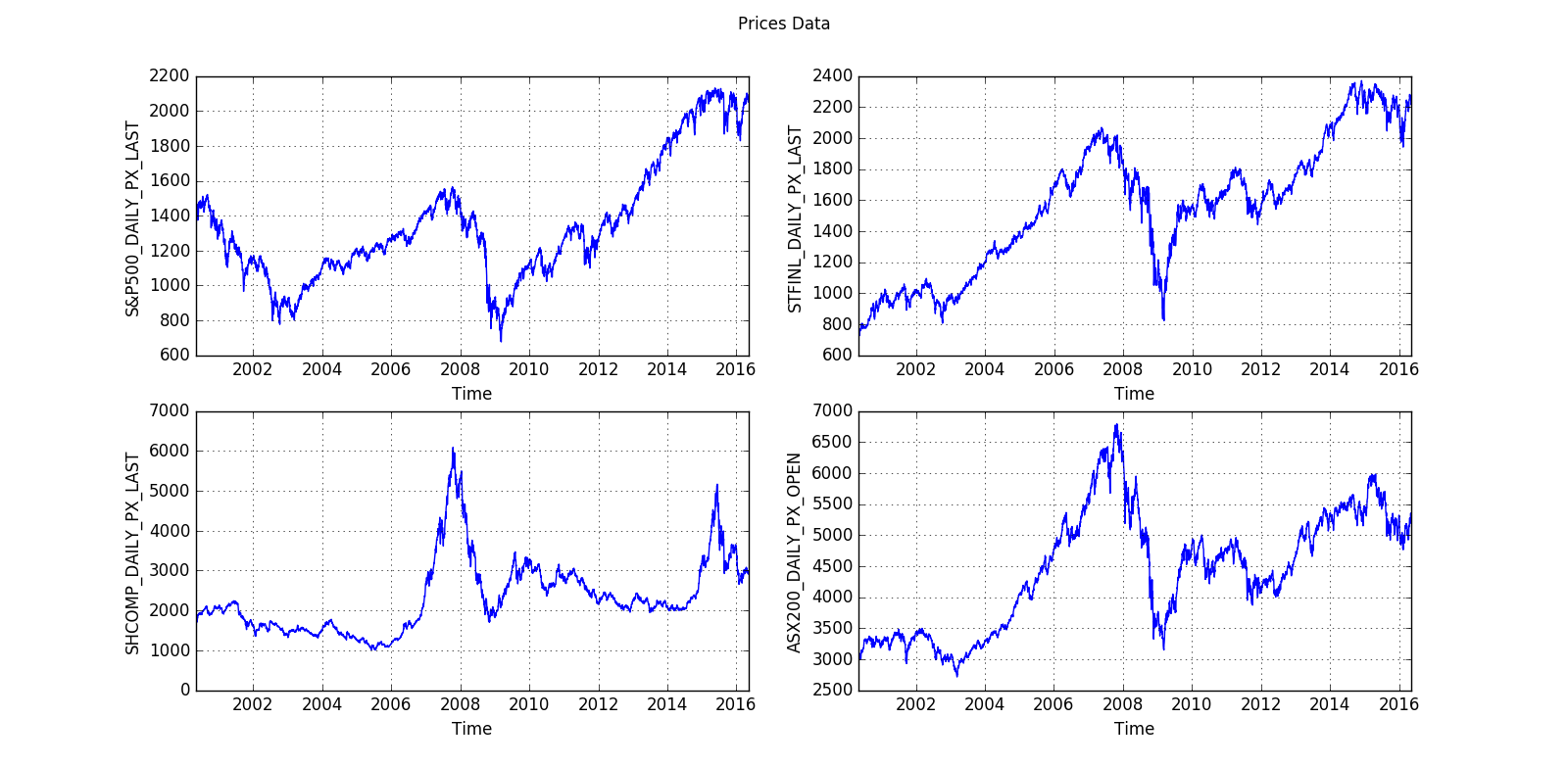
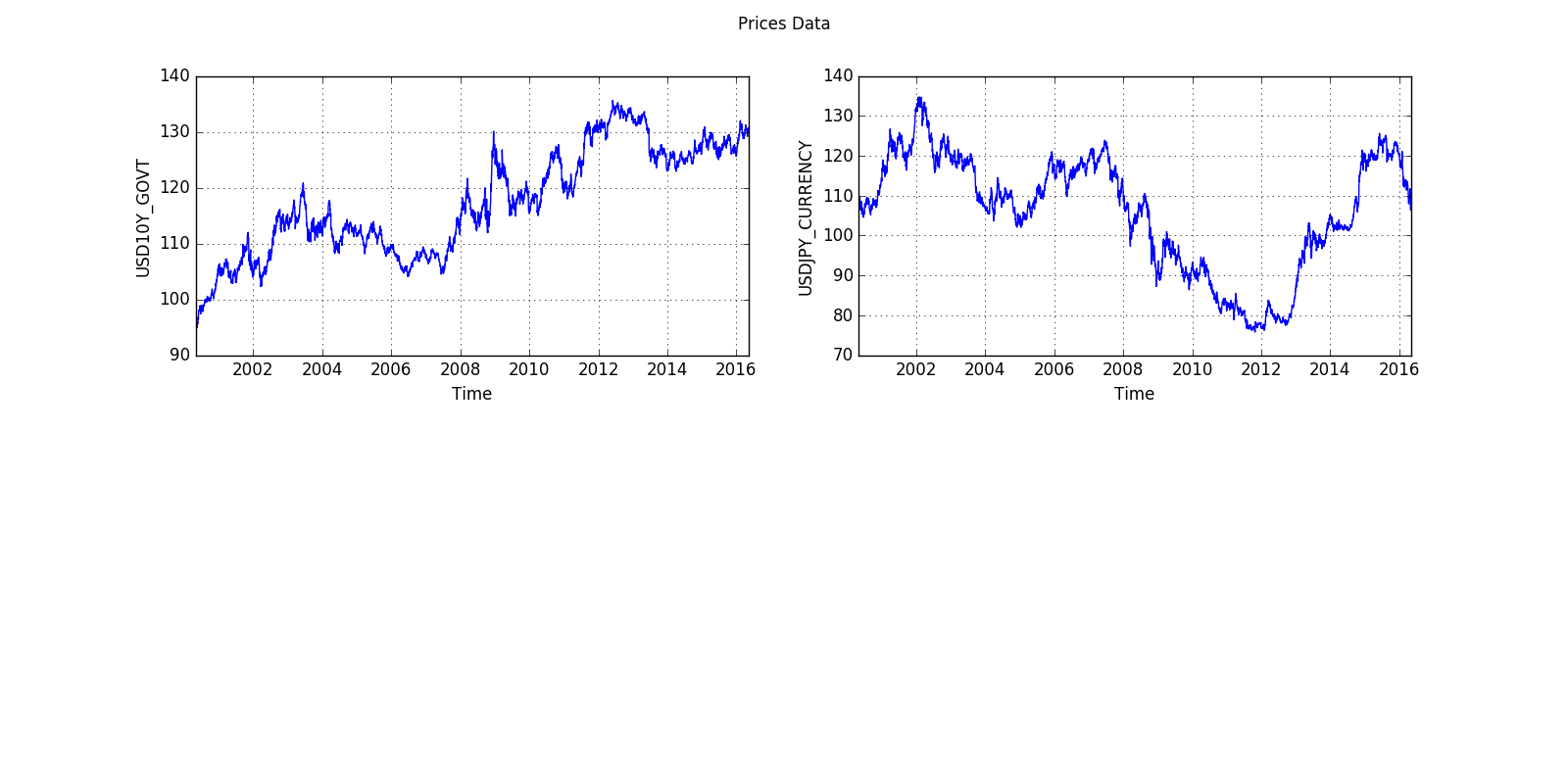
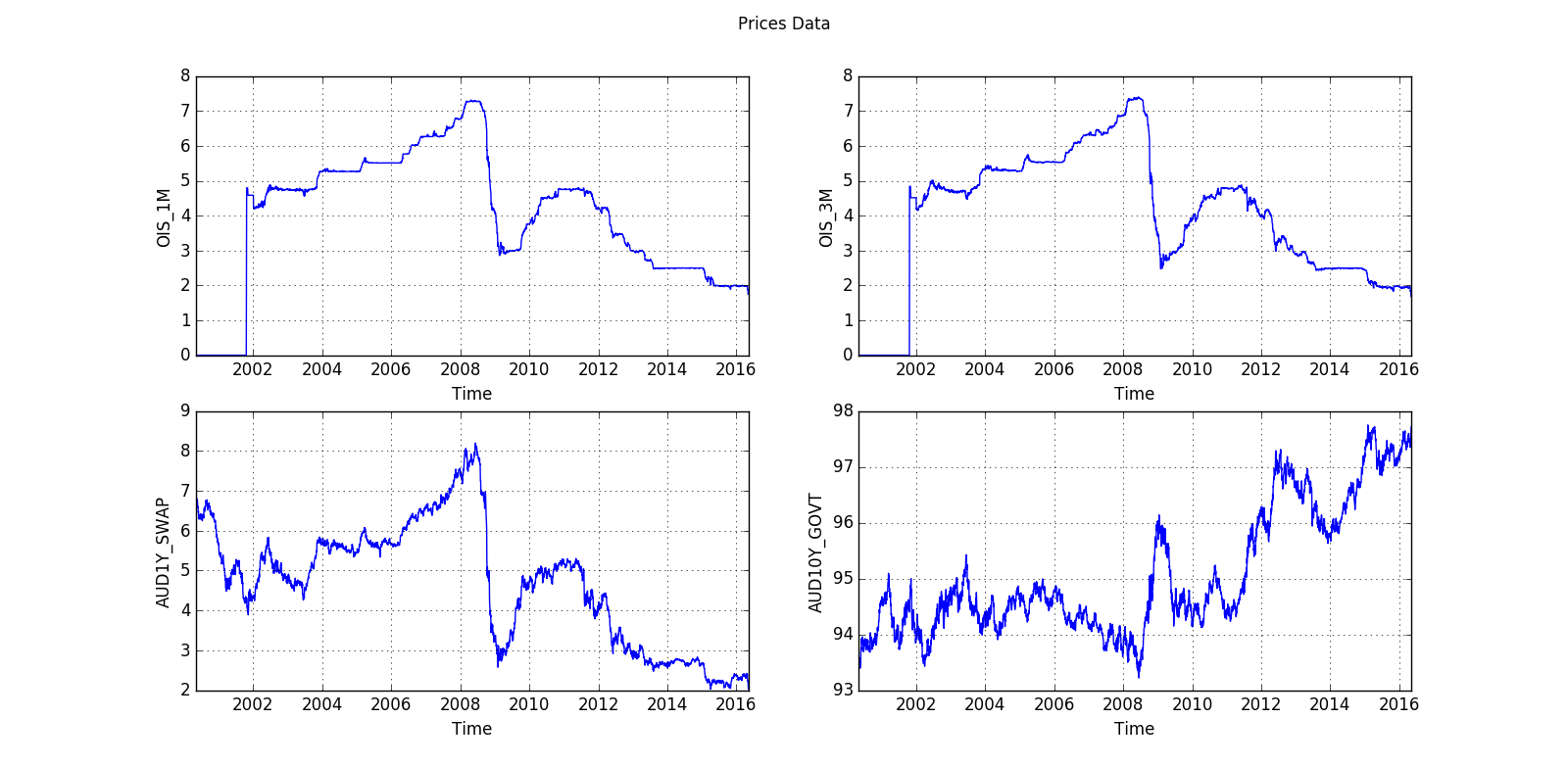
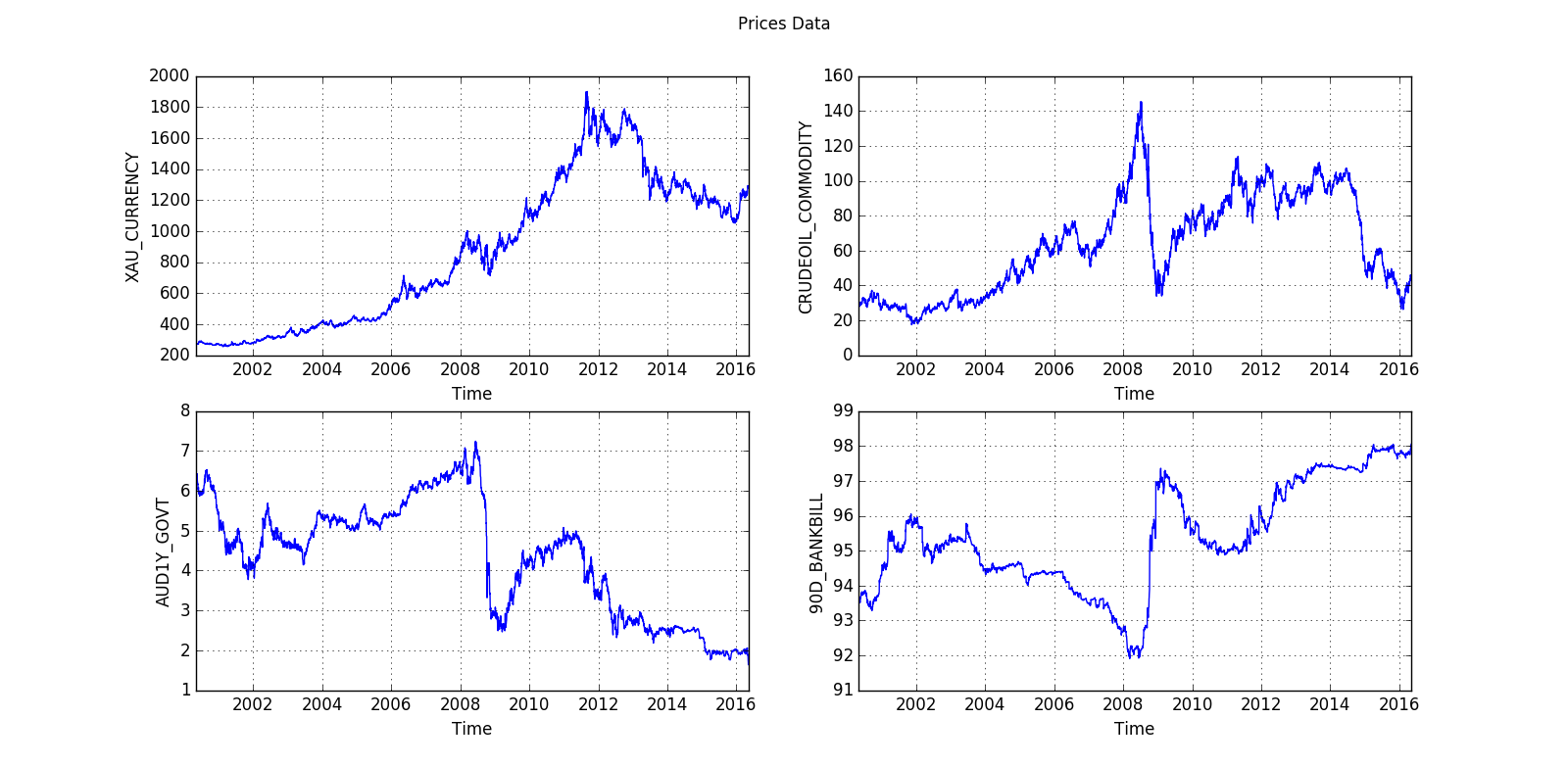
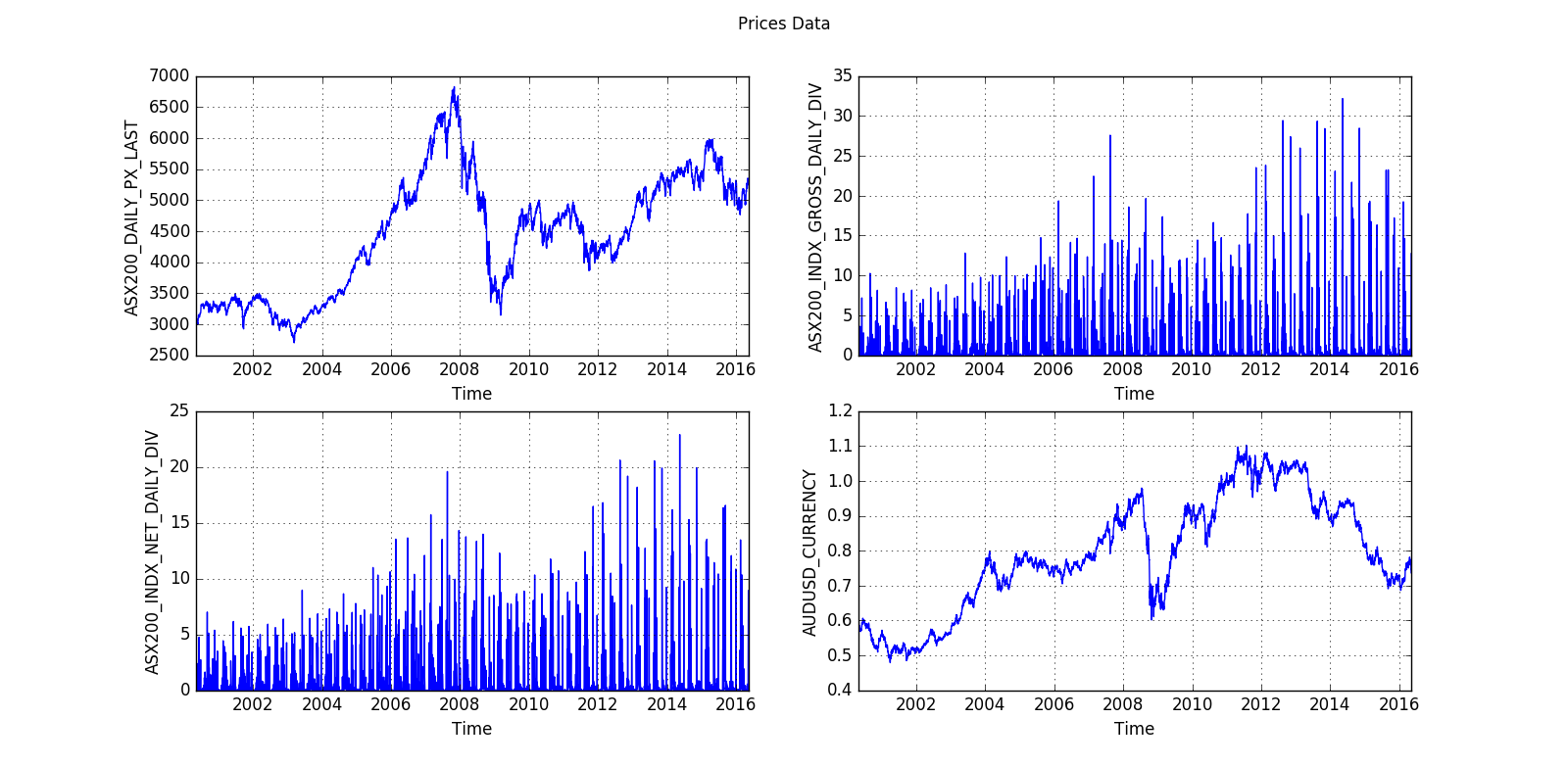
Prob(Omnibus): 0.000 Jarque-Bera (JB): 5256.201

Skew: -0.180 Prob(JB): 0.00

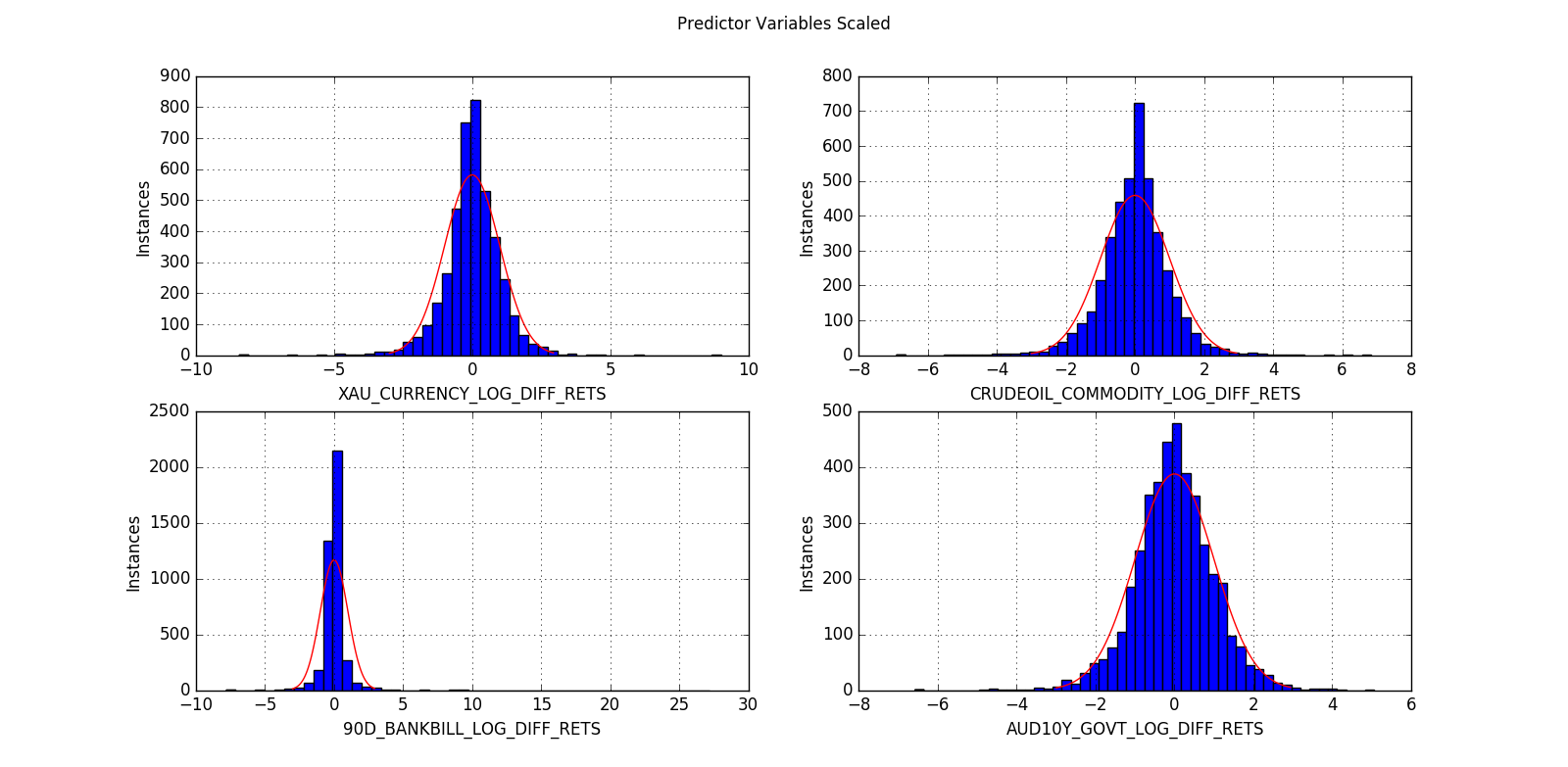
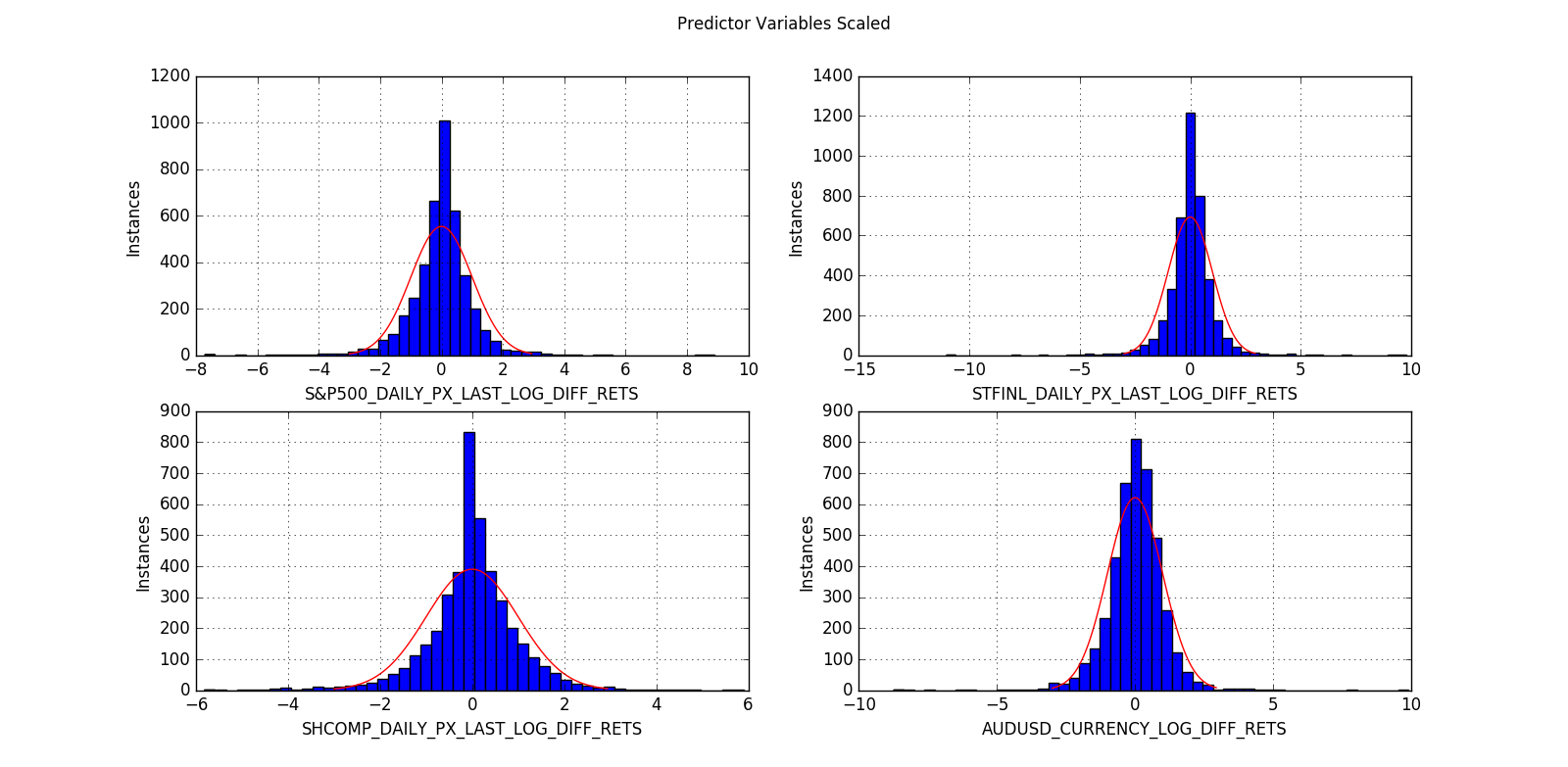
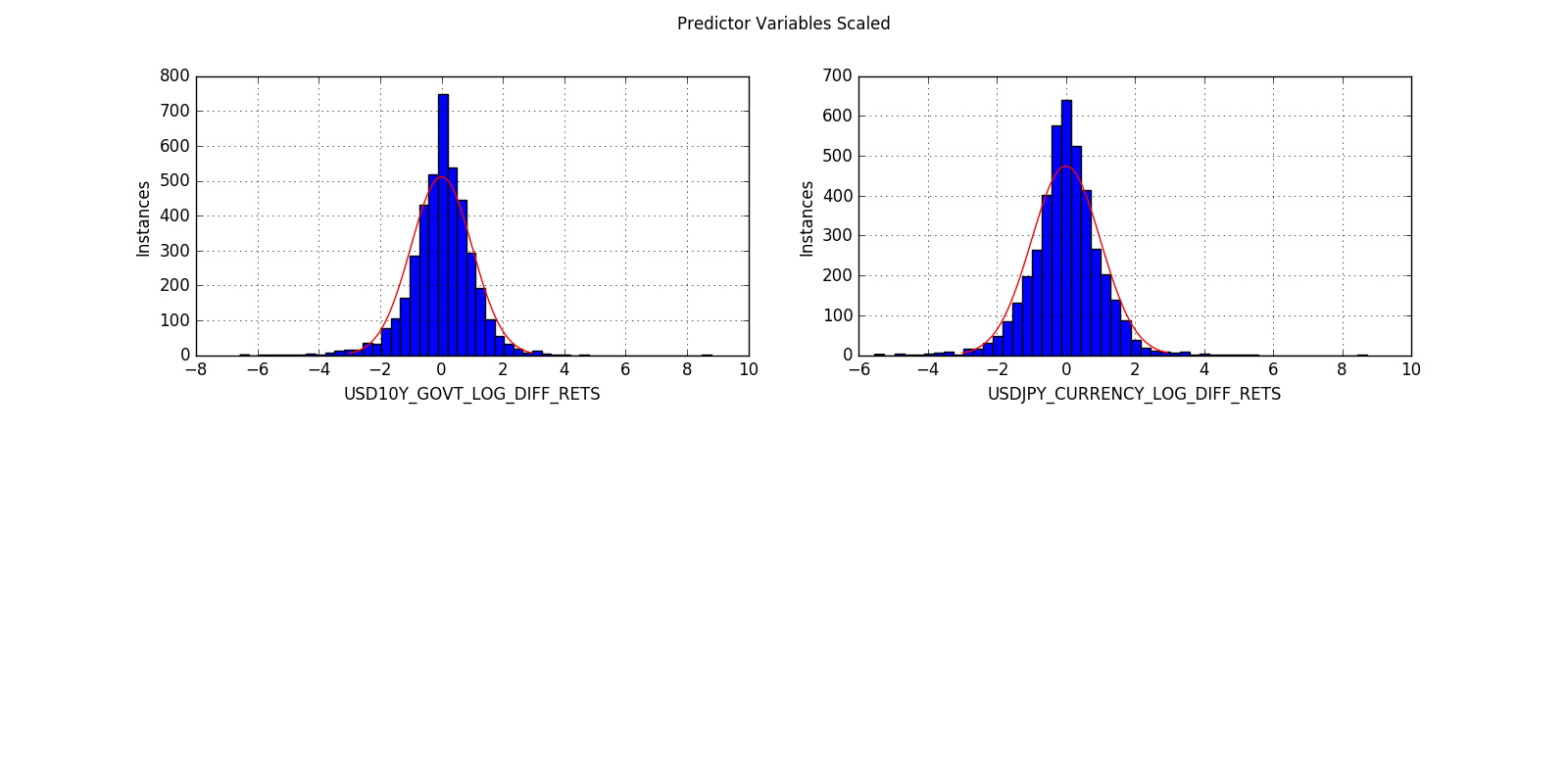
Kurtosis: 10.087 Cond. No. 2.69

===========================================================================

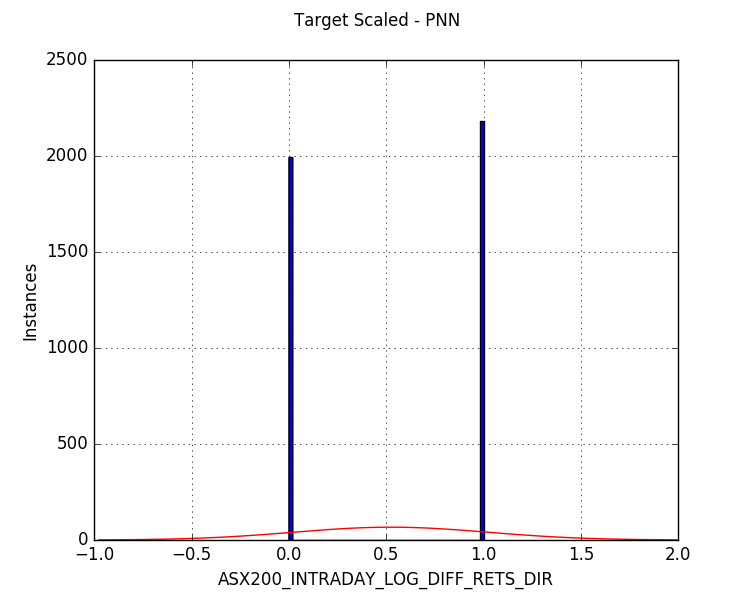
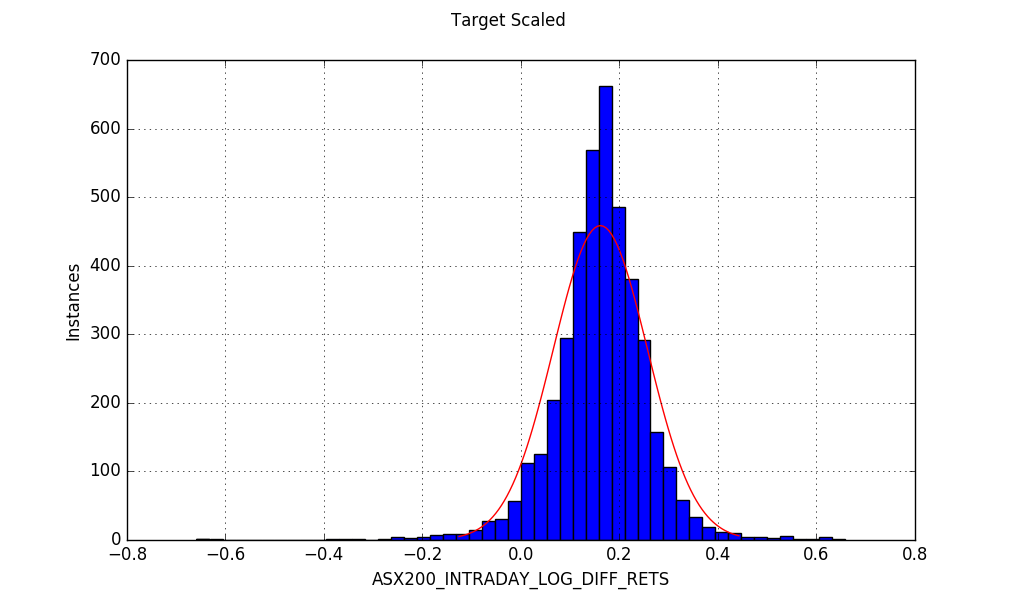
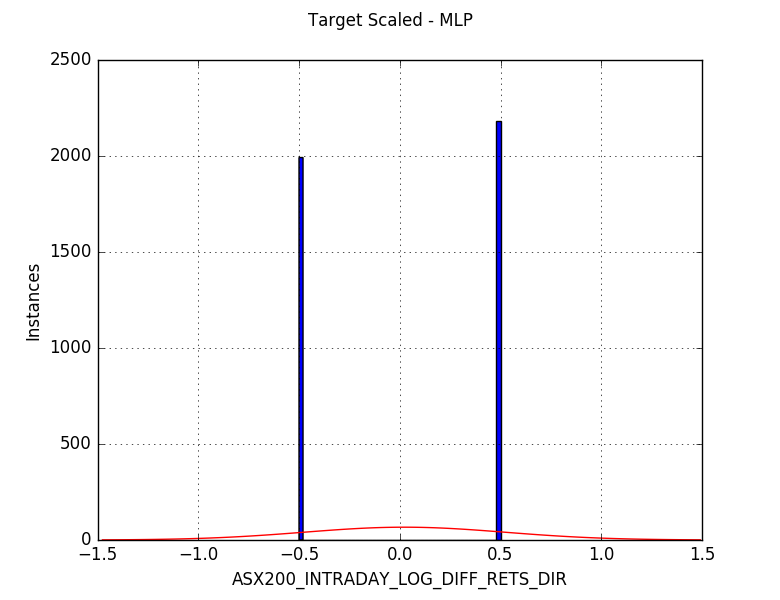
## Prices Data

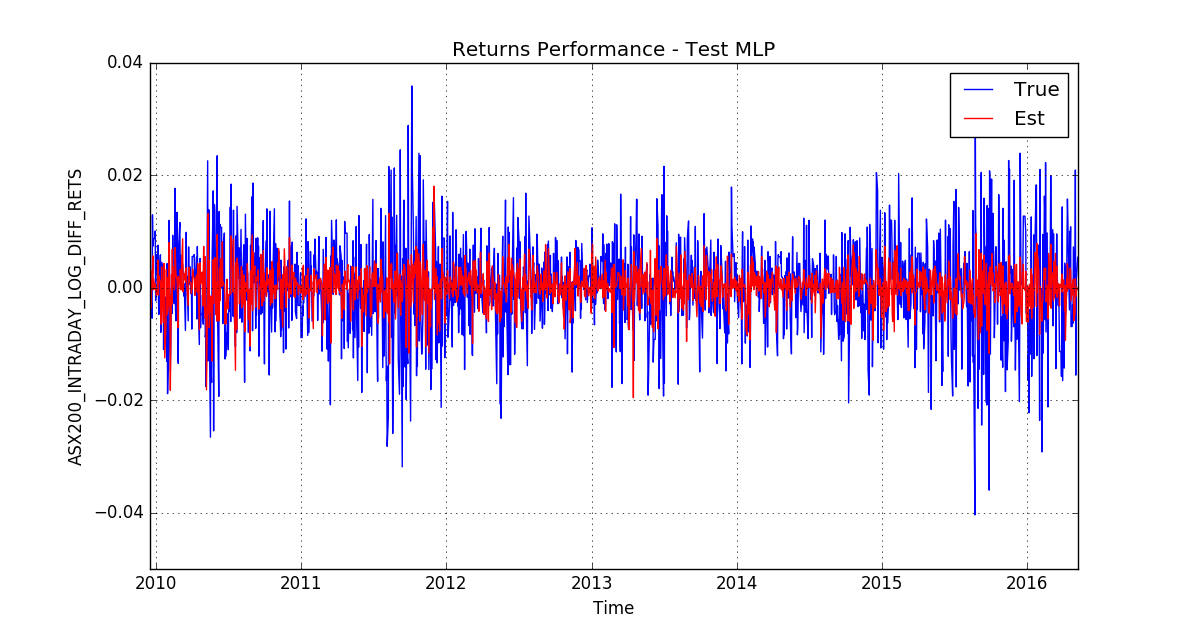
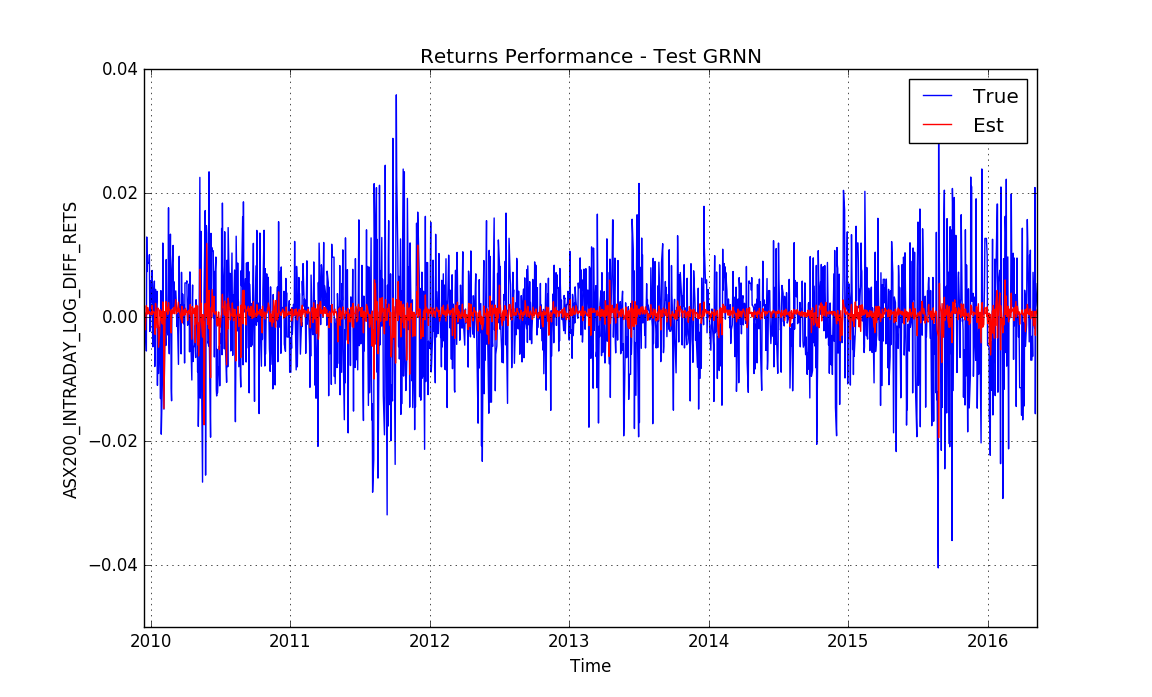
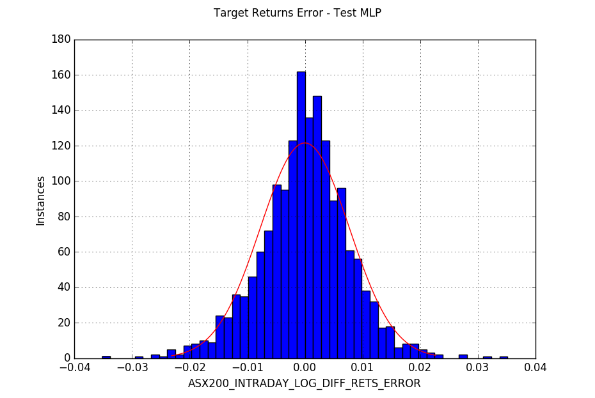
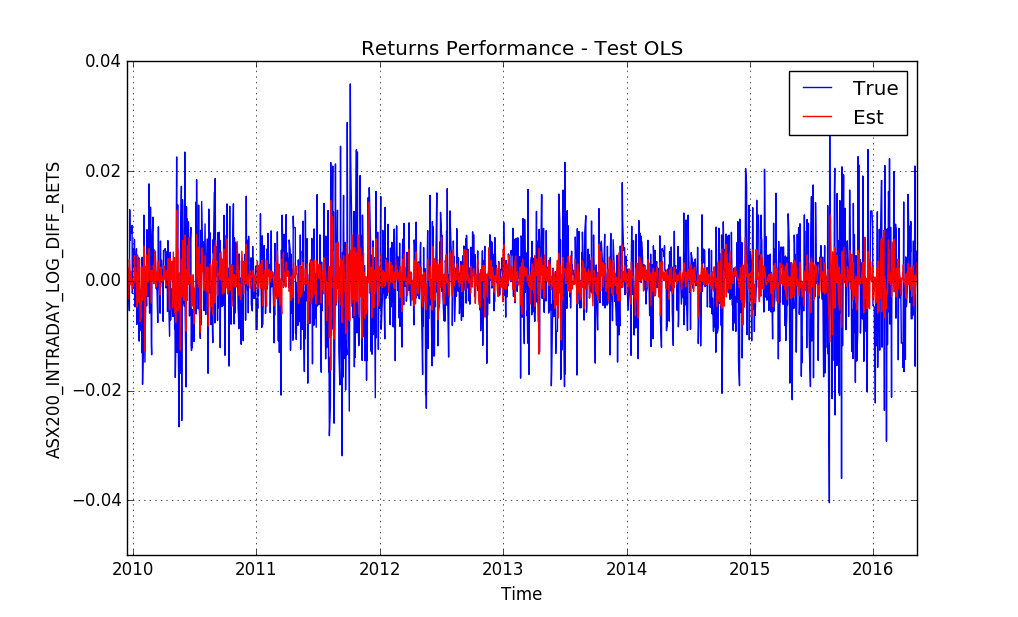
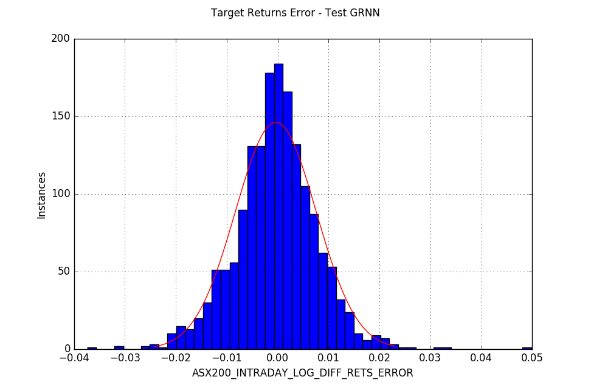
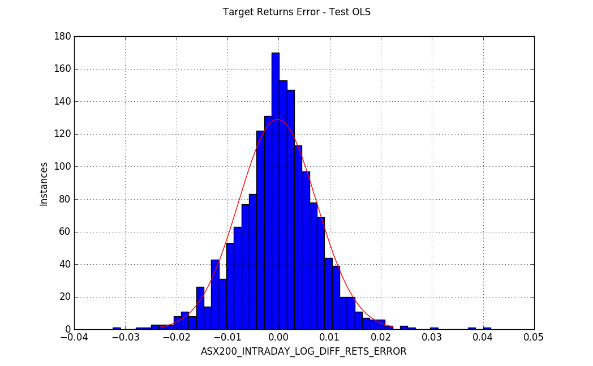


## Predictor Variables Standardised

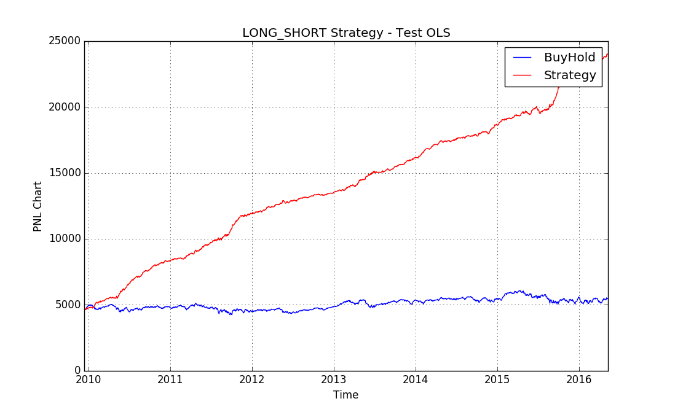
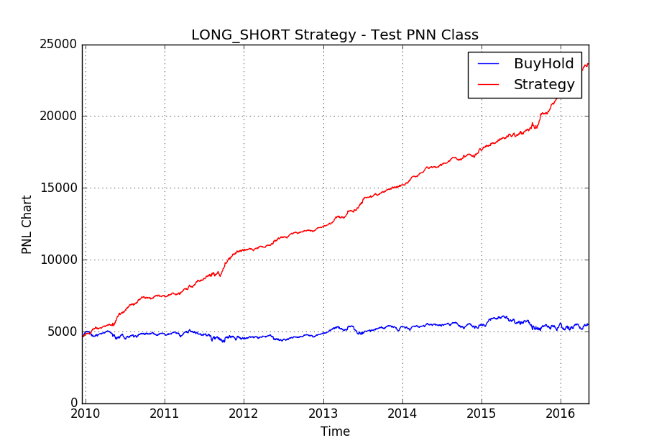
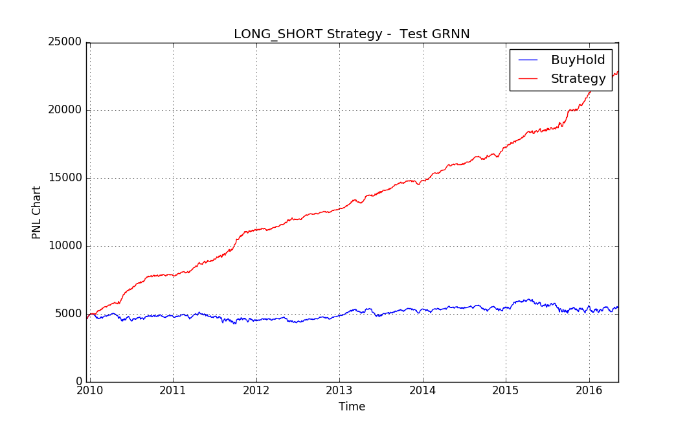
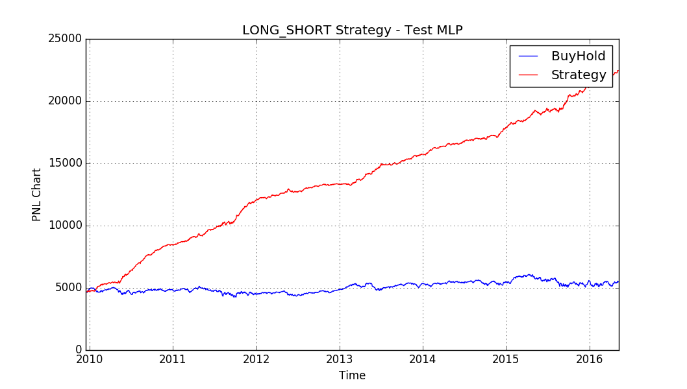
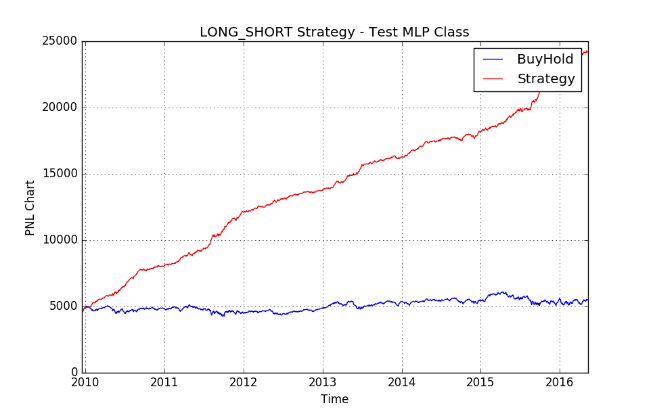


## Targets Scaled



## Trading Simulation Results



## Daily data summary

16 years of daily data was compiled. From 9th May 2000 to 9th May 2016.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id.** | **Data label** | **Source** | **Lag (days)** | **Description** |
| *x1* | *'S&P500\_DAILY\_PX\_LAST'* | Bloomberg | -1 | S&P500 Index close price |
| *x2* | *'STFINL\_DAILY\_PX\_LAST'* | Bloomberg | -1 | Canadian Financials Index close price |
| *x3* | *'SHCOMP\_DAILY\_PX\_LAST'* | Bloomberg | -1 | Shanghai Composite close price |
|  | *'ASX200\_DAILY\_PX\_OPEN'* | Bloomberg | 0 | ASX200 open price (not include dividends) |
| *y* | *'ASX200\_DAILY\_PX\_LAST'* | Bloomberg | 0 | ASX200 close price (not include dividends) |
| *x6* | *'ASX200\_INDX\_GROSS\_DAILY\_DIV'* | Bloomberg | 0 | Gross dividends ASX200 |
| *X4* | *'AUDUSD\_CURRENCY'* | Bloomberg | -1 | AUD USD Exchange rate |
| *X5* | *'XAU\_CURRENCY'* | Bloomberg | -1 | Yuan AUD Exchange rate |
| *X7* | *'CRUDEOIL\_COMMODITY'* | Bloomberg | -1 | Crude oil price |
| *X9* | *'AUD1Y\_GOVT'* | Bloomberg | -1 | 1Y Australian Gov. Bond |
| *X8* | *'90D\_BANKBILL'* | Bloomberg | -1 | 90 Day Bank Bill |
| *x12* | *'OIS\_1M'* | Bloomberg | -1 | Overnight Index Swap Rate 1 Month Tenor |
| *x13* | *'OIS\_3M'* | Bloomberg | -1 | Overnight Index Swap Rate 3 Month Tenor |
| *x14* | *'AUD1Y\_SWAP'* | Bloomberg | -1 | Swap Rate 1 Y Maturity |
| *x15* | *'AUD10Y\_GOVT'* | Bloomberg | -1 | 10Y Australian Gov. Bond |
| *x10* | *'USD10Y\_GOVT'* | Bloomberg | -1 | 10Y American Gov. Bond |
| *x11* | *'USDJPY\_CURRENCY'* | Bloomberg | -1 | USD Yen Exchange rate |

## References

1. David Enke, Suraphan Thawornwong (2005), The use of data mining and neural networks for forecasting stock market returns; Expert Systems with Applications 29 (2005) 927–940
2. Mark T. Leunga, Hazem Daoukb, An-Sing Chenc (2000) Forecasting stock indices: a comparison of classification and level estimation models; http://papers.ssrn.com/sol3/papers.cfm?abstract\_id=200429
3. Michal Tka, Robert Verner (2016) Artificial neural networks in business: Two decades of research; Applied Soft Computing 38 (2016) 788–804
4. Rodolfo C. Cavalcante, Rodrigo C. Brasileiro b , Victor L.F. Souza b , Jarley P. Nobrega, Adriano L.I. Oliveira (2016) Computational Intelligence and Financial Markets: A Survey and Future Directions; Expert Systems With Applications 55 (2016) 194–211
5. C. D. Tilakaratne, S. A. Morris, M. A. Mammadov, C. P. Hurst; Predicting Stock Market Index Trading Signals Using Neural Networks; Centre for Informatics and Applied Optimization School of Information Technology and Mathematical Sciences University of Ballarat, PO Box 663, Ballarat, Victoria, Australia