COMP6229 (2016/17): Machine Learning Coursework

Ronald M. Gualan Saavedra (29021812)
rmgs1u16@soton.ac.uk
MSc. Data Science student

This report summarizes the work done to complete the COMP6229 Coursework [1]. The report is organized in two sections. Section 1 compares the Bayes classifier and ANN (Artificial Neural Networks) in a binary classification problem, emphasizing the capability ANNs have to estimate posterior probability of class membership. Section 2 explores Time series prediction in two challenges: 1) prediction of the chaotic Mackey-Glass time series and 2) prediction of financial time series using a S&P 500 dataset.

1 Neural Network Approximation

In this section, a two-class pattern classification problem is addressed. Each class is Gaussian distributed with distinct means and covariance matrices $\mathcal{N}(\boldsymbol{m_j}, \boldsymbol{C_j})$, where the means and covariance matrices are:

$$\boldsymbol{m_1} = \begin{pmatrix} 0 \\ 3 \end{pmatrix}, \ \boldsymbol{C_1} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, \ \boldsymbol{m_2} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \text{ and } \boldsymbol{C_2} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

The first exercise consists in plotting the decision boundary for which the posterior probability satisfies $P[\omega_1|x] = 0.5$. This was accomplished by using the inherently quadratic discriminant functions (1)-(4) taken from [2]. The result is shown in Figure 1(a). The second part of the exercise, the posterior probability in 3D, is shown in Figure 1(b), where the posterior probability of the second class was included as it allows to easily relate the quadratic decision boundary and the 3D posterior probability.

$$\boldsymbol{x}^t \boldsymbol{W_1} \boldsymbol{x} + \boldsymbol{w_1^t} \boldsymbol{x} + \omega_{10} = \boldsymbol{x}^t \boldsymbol{W_2} \boldsymbol{x} + \boldsymbol{w_2^t} \boldsymbol{x} + \omega_{20}, \tag{1}$$

$$\boldsymbol{W_i} = -\frac{1}{2} C_i^{-1}, \tag{2}$$

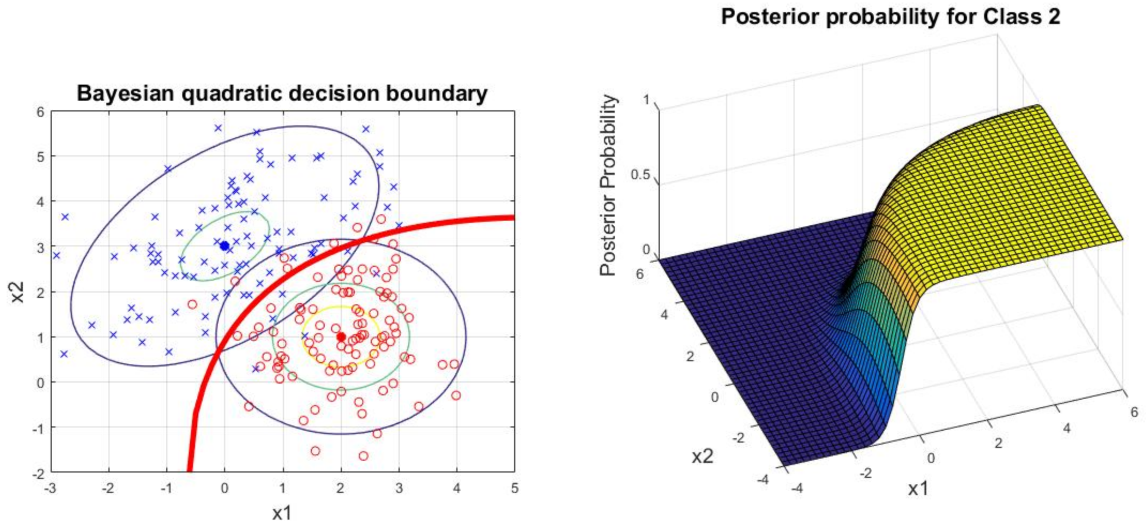
$$\boldsymbol{w_i} = C_i^{-1} \boldsymbol{m_i}, \tag{3}$$

$$\omega_{i0} = -\frac{1}{2} \boldsymbol{m_i^t} C_i^{-1} \boldsymbol{m_i} - \frac{1}{2} \ln |C_i| + \ln P[\omega_i]. \tag{4}$$

For the second exercise, the sampled data from the previous exercise was used to train a neural network for the classification problem. The first neural network uses 20 hidden layers and as can be seen in Figure 2(a) the form of the decision contour is rather complex compared to the Bayesian decision boundary. This is understandable considering the number of hidden neurons and the relatively small training dataset. A second feed-forward neural network with 5 hidden neurons was trained to obtain a less complex decision contour. Figure 2(b) shows that the decision contour of the second neural network is not as complex as that of the first neural network. However, none of them produces a decision contour similar in form to the Bayesian quadratic boundary. Again, this is probably due to the small training dataset (200 records, 100 per class).

Despite the difference in the boundaries produced by these three models, they perform quite similar on the training data, where the Bayes classifier, the neural network #1 (20 hidden neurons) and the neural network #2 (5 hidden neurons) obtain 93.5, 94.0% and 93.5% overall accuracy, respectively. The accuracies showed that the neural networks performed equal or better than the optimal Bayes classifier on the train dataset. However, when testing these models on a different dataset, for instance using 1000 samples per class, the Bayesian classifier obtained a better accuracy than the neural networks. This can be seen on Figure 3, which includes confusion matrices for the models discussed in this section.

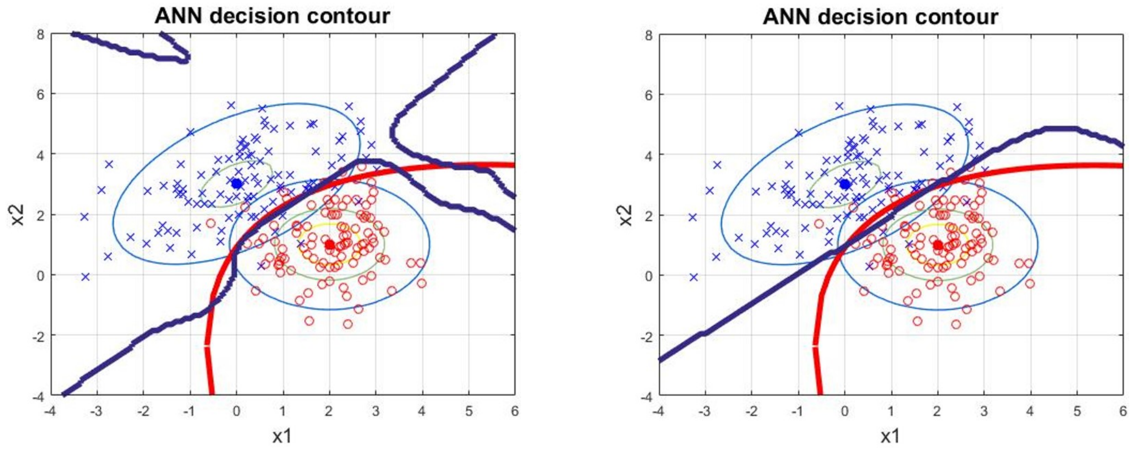
As stated in the specifications of this section [1], artificial neural networks are estimators of posterior probabilities of class membership. However, this was not evidenced in Figure 2. Since a decision contour basically summarizes the shape of the 3D posterior probability function in two dimensions, it was expected that the Bayes decision boundary and the decision contours of the neural networks were alike. This result was not obtained due to the small amount of data used when training the dataset. If more samples are



(a) Quadratic boundary

(b) Posterior probability for Class 2 ($P[\omega_2|x]$)

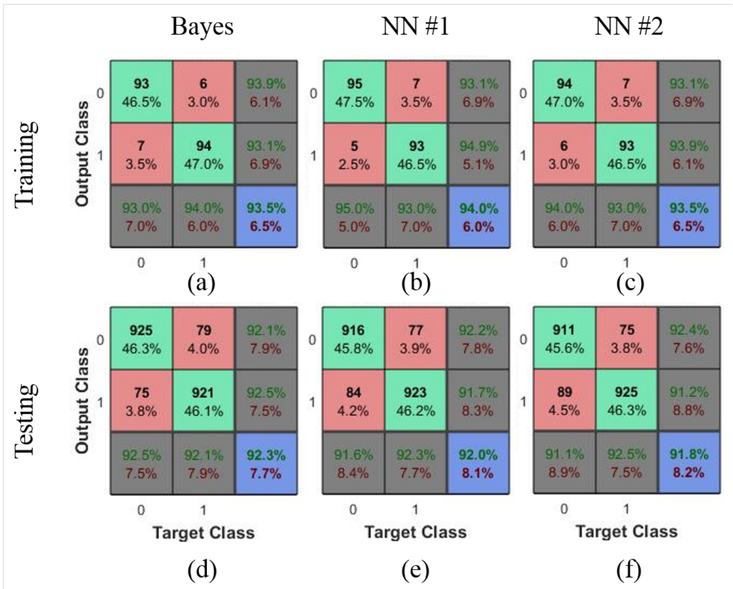
Figure 1: Bayes optimal classifier. The Bayes classifier (thick red line) is accompanied by the 100 random samples for each class (blue and red crosses), its centres (blue and red thick points), and some contour lines that illustrate each distribution.



(a) Neural network #1 with 20 hidden neurons.

(b) Neural network #2 with 5 hidden neurons.

Figure 2: Decision contours of two neural networks: (a) a neural network with 20 hidden neurons, and (b) a neural network with 5 hidden neurons. The decision contours are illustrated as thick blue lines. In addition, the Bayes decision contour and the samples of Figure 1 are included to easy comparison.



(a) (b) (c)

(d) (e) (f)

Figure 3: Confusion matrices for the Bayes classifier, neural network #1 (20 hidden neurons), and neural network #2 (5 hidden layers) for the training (a)-(c), respectively, and testing (d)-(f). For the training, 100 samples per class were used. For the testing 1000 samples were used.

used to train the neural networks, the resulting decision contour is more similar to the Bayes classifier decision boundary. This can be seen in Figure 4, which shows the decision contour of a neural network with 10 hidden layers trained with 1000 samples for each class.

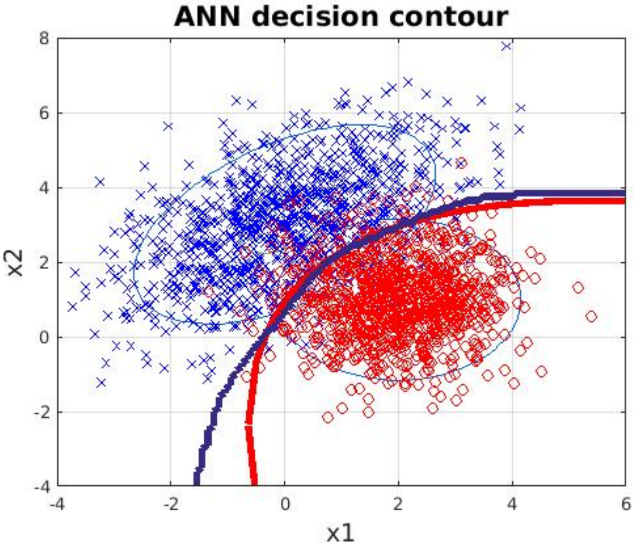


Figure 4: Decision contour of a neural network with 10 hidden neurons trained with 1000 samples per class (thick blue curve). In addition, the Bayes quadratic decision boundary (thick red curve), and the sample points (blue and red crosses, and its corresponding contour lines), are also included for the sake of comparison.

2 Times series prediction

This section describes some experiments regarding Time series prediction, which are formulated as regression problems where an output $s(n)$ is predicted from p samples that occurred in the past. Particularly, Artificial neural networks were used for prediction on two kinds of time series: 1) Chaotic time series, and 2) Financial time series.

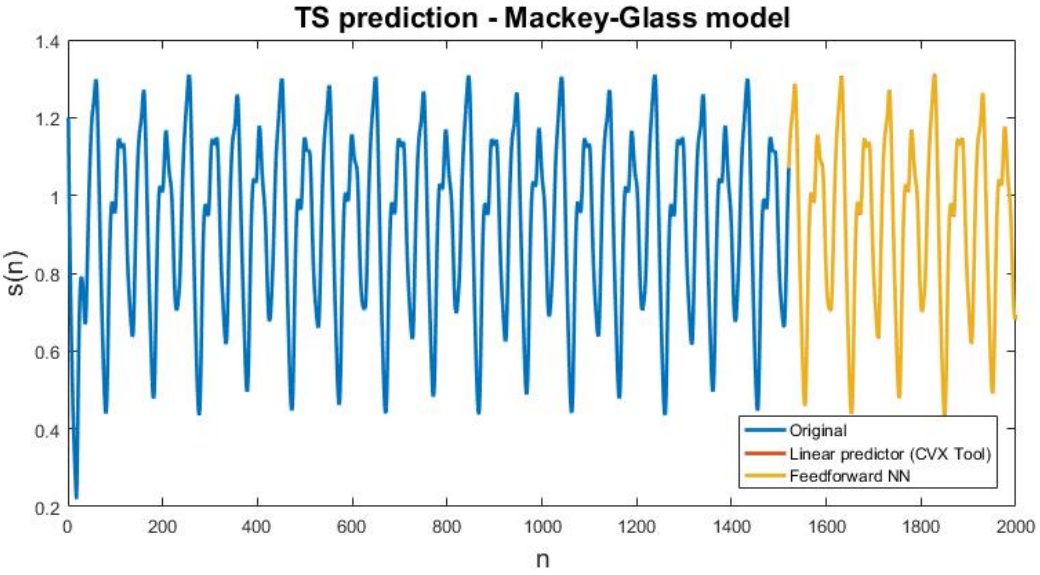


Figure 5: Prediction of the last 480 points of the Mackey-Glass series series in test mode using the CVX linear predictor and a feed-forward neural network.

2.1 Chaotic Time Series

This exercise uses the Mackey-Glass model [3], by means of the Mackey-Glass time series generator from the MathWork's contributed software site [4]. The parameters used for generating time series samples $(a = 0.2,\ b = 0.1,\ \tau = 17,\ x_0 = 1.2,\ \delta_t = 1)$ were extracted from [5], where the author used some machine learning techniques to analyze and predict chaotic time series. Using the mentioned parameters, a dataset of 2000 samples was generated: 1500 samples are for training and 500 for testing.

The dataset mentioned in the previous paragraph is used for training a linear predictor (CVX Tool) and a feed-forward neural network. The task of these models is to produce one step ahead prediction using 20 past steps ($p = 20$) as input parameters. As can be seen in Figure 5, both models performed extremely well. Figure 5 shows that the time series predictions for the test data, i.e. last 480 values, are practically the same. Consequently, the curve of the last plot time series in Figure 5 overlaps the others. The RMSEs for this exercise are $8.55e - 04$ and $9.33e - 06$, for the CVX linear predictor and the feed-forward neural network, respectively. Thus, both models performed very well, but the neural network performed slightly better than the CVX linear predictor in this exercise.

The final part of this section consists in emulate part of the work presented in [6], where Wan claims that it is possible to reproduce the oscillating Mackey-Glass time series using the iterated prediction mode. Wan used a FIR (Finite Impulse Response) network which received no new inputs past the point 500, and accomplished 100 iterated point prediction with remarkably accuracy.

A feed-forward neural network with 20 hidden neurons was used to reproduce this experiment. Wan used 500 points for training and 100 for the iterated prediction. First, the experiment in [6] is reproduced using the same amount of training data, iterated points, and the same Mackey-Glass time series parameters (i.e. $a = 0.2,\ b = 0.1,\ \tau = 30, x_0 = 0.9,\ \delta = 6$). The results can be seen in Figure 6. The RMSE for the predicted iterated points was 0.0469. In conclusion, the iterated prediction of 100 point of the Mackey-Glass time series performed by Wan in [6], was successfully reproduced in this work.

After reproducing Wan's experiment, a more complex experiment aiming to evaluate the prediction of sustained oscillations in the iterated mode is executed. Here, the neural network trained with the test dataset of 1500 samples of the beginning of this section, is used to predict the next 8500 points of the Mackey-Glass time series. Thus, the total size of the dataset is 10000 points. Figure 7(a) shows the whole dataset of the experiment (as an illustration of the overall performance), and the error of the iterated prediction, which also seems to have a chaotic behaviour. Figure 7(a) and (b), show the first and the last part of the prediction in a more understandable window size of 1000 elements. Figure 7(c) demonstrates that although the phase of the iterated prediction is slightly different than the original Mackey-Glass time series after 7500 iterated predictions, it still reproduced the general shape of the chaotic time series. The RMSE for the 8500 iterated predictions is 0.0933, which is low considering that these are iterated predictions, i.e. predictions without any type of real data correction feedback.

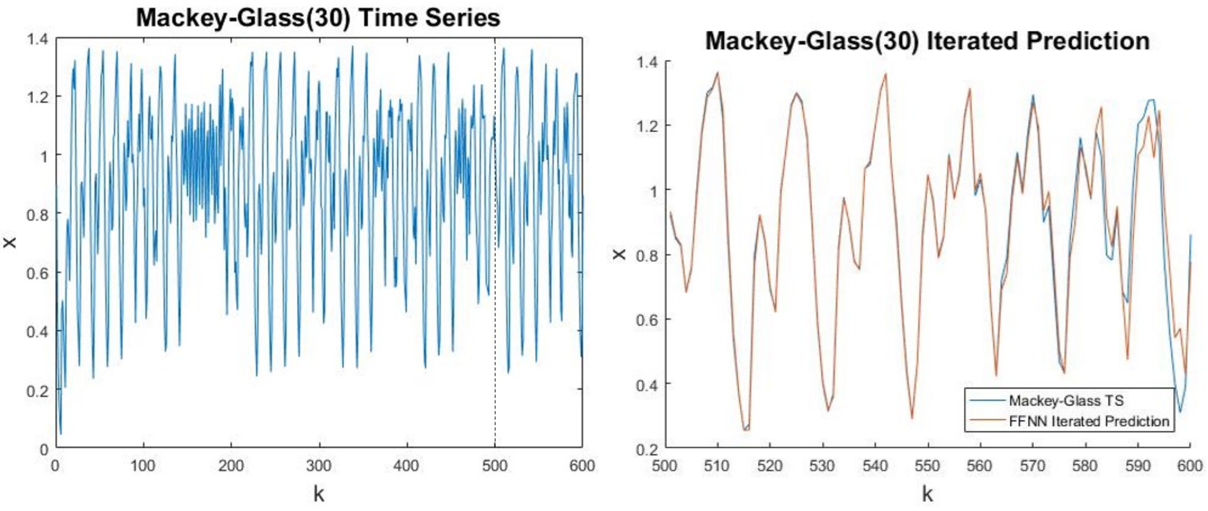


Figure 6: (a) Mackey-Glass(30) series with 600 points, and (b) Iterated prediction of the last 100 points of the original time series (a) using a feed-forward neural network with 20 hidden neurons.

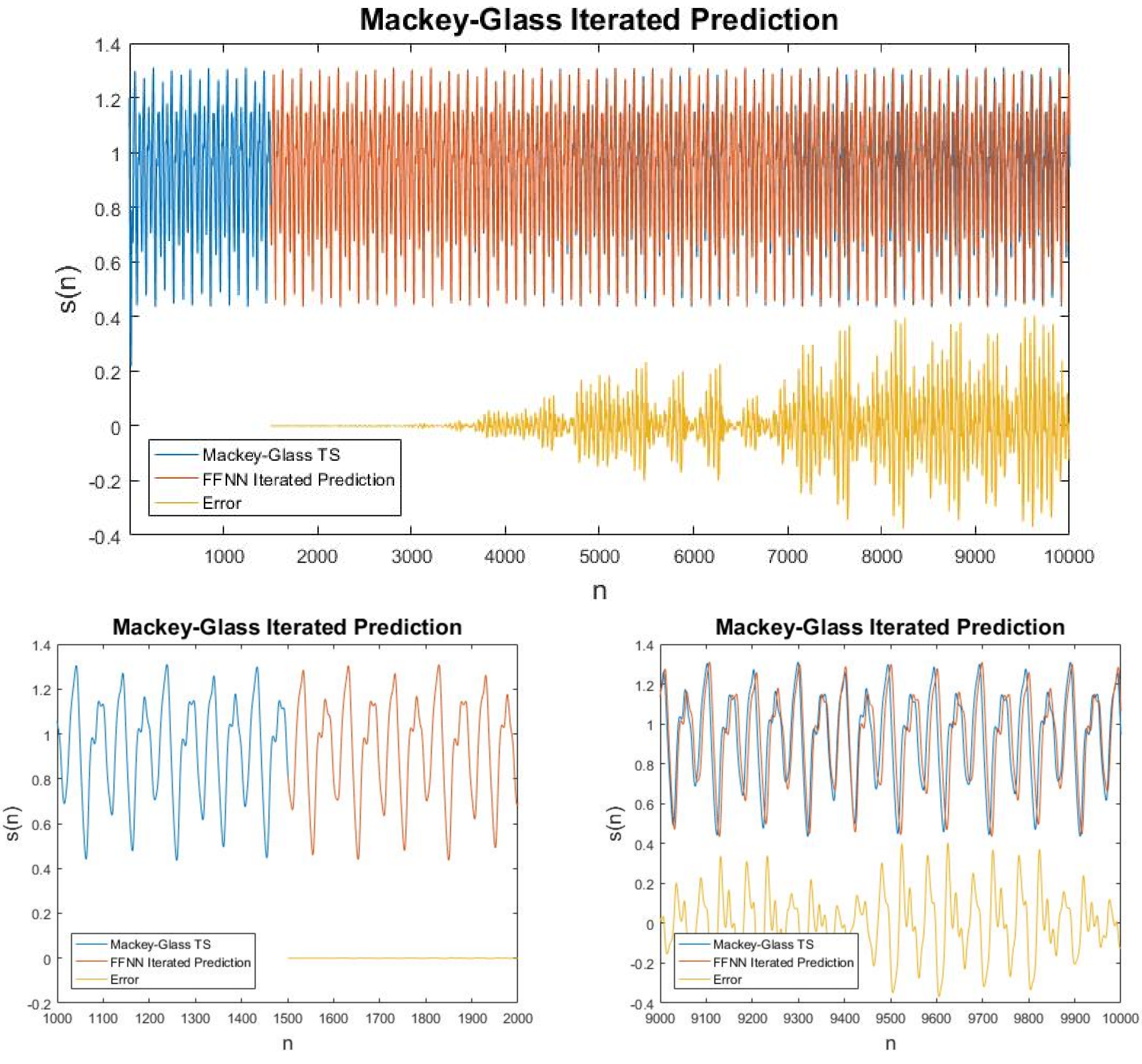


Figure 7: (a) Mackey-Glass(30) series with 10000 points (blue), the iterated prediction (last 8500 points, orange), and the error of the iterated prediction (yellow). (b) A window of 1000 points showing the first 500 iterated predictions. (c) A window of 1000 points showing the last 1000 iterated predictions.

Table 1: Statistics of the predictions of the *Close price* using (a) 10 neural networks based on the past p *Close price* values, and (b) 10 neural networks based on the past p *Close price* values and the past p values of *Volume Traded.*

Experiment	Target	Mean	St. Dev.	Min	Max	RMSE
1	2212.23	2202.10	5.39	2195.40	2210.70	11.38
2	2212.23	2212.60	10.78	2193.10	2224.10	10.23

2.2 Financial Time Series

In this section, the dataset used is the historical data of S&P 500 stock index, which contains 5 years of daily trade data taken from 2011-12-09 to 2016-12-09 [7]. The task is to predict the *Close price* of the S&P 500 index for the last day of the dataset, i.e. 2016-12-09, using one step ahead prediction from 20 past days. This hypothetical experiment has two assumptions: 1) the period of the available data is from 2011-12-09 to 2011-12-08, 2) if the predicted Close price for 2011-12-09 is accurate, it could be potentially exploited to make some money.

Since the aim of this section is to accomplish the task described in the previous paragraph as accurately as possible, in the first experiment 10 feed-forward neural network (with 10 hidden neurons each) were trained using the historical data. This small set of neural networks served to assess the variation of the predictions. This is necessary because each network produces slightly different outputs thanks to its randomly initialized parameters. As can be seen in Figure 8(a) the prediction range is not accurate. The real Close price (USD 2212.23) is barely included in the range of predicted values, which is USD 2200 ± 10.78 ($\overline{y_p} \pm 2\sigma$). The RMSE of the outputs versus the real value is 11.38 (See Table 1). Figure 8(a) also shows that predictions are inaccurate in the training dataset too. This is illustrated through the use of a confidence interval (1.96σ). In conclusion, the obtained results are not good enough to support the decision to invest in the stock market.

The lack of accuracy evidenced in this exercise might be due to the nature of the dataset. Under certain conditions, neural networks are capable of approximate any uniformly continuous distribution [6]. This was evidenced in Figure 5. However, Stock market datasets contain sudden changes of direction, that are not characteristic of uniformly continuous distributions.

In a second experiment, the *Volume Traded* is used as an additional input for the regression problem to try and improve the *Close price* prediction. In this case the input matrix has $2p$ columns. This kind of network is also known as Nonlinear Autoregressive model with eXogenous input (NARX) [5,8]. A NARX neural network can mathematically be represented by $y(n+1) = f[\boldsymbol{y}(n); \boldsymbol{u}(n)]$ where the vectors $\boldsymbol{y}(n)$ and $\boldsymbol{u}(n)$ denote the output and input of regressors, respectively [5,8].

For this experiment, 10 feed-forward neural networks with 10 hidden neurons were used as NARX networks, where the *Volume traded* is the eXogenous input. The results are presented in Figure 8(b). In this experiment the mean of the predictions is close to the target value (See Table 1), however the variance is bigger than that of the previous experiment. The effect in the variance is also seen in the training predictions. Thus, by using past values of *Volume Traded* the mean prediction was better than the mean prediction using only *Close prices*, but its variation was larger.

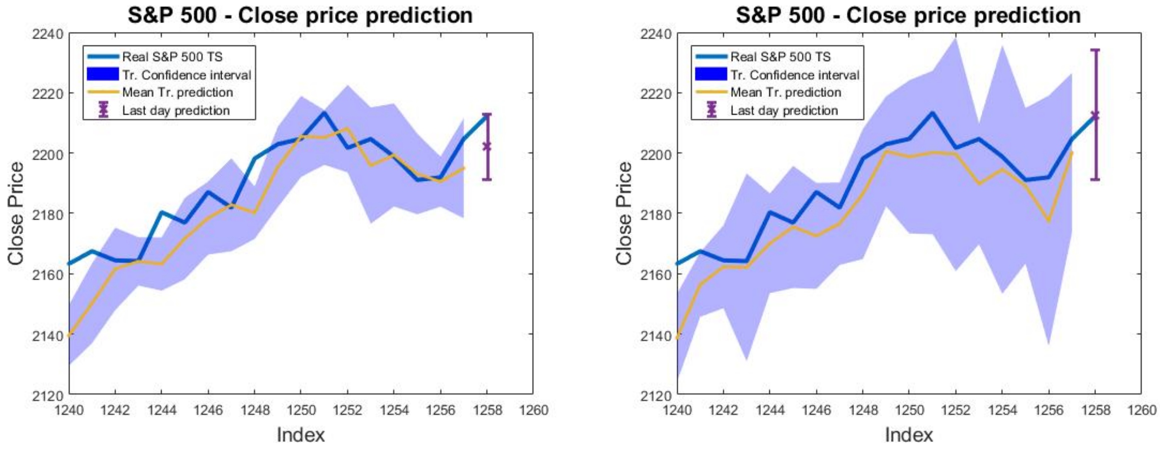


Figure 8: (a) *Close Price* prediction for 2016-12-09 using a feed-forward neural network feed with the last p values. (b) *Close Price* prediction for 2016-12-09 using a feed-forward neural network as a NARX network using as extra input the past p values of *Volume Trade.* The confidence interval uses 1.96 z-score.

References

[1] Mahesan Niranjan, "COMP3206/6229 (2016/17): Machine Learning Assignment," Nov. 2016.

[2] P. E. Hart, D. G. Stork, and R. O. Duda, "Pattern classification," *John Willey & Sons*, 2001.

[3] M. C. Mackey, L. Glass, and others, "Oscillation and chaos in physiological control systems," *Science*, vol. 197, no. 4300, pp. 287–289, 1977.

[4] Marco Cococcioni, "Mackey-Glass time series generator - File Exchange - MATLAB Central," June 2009.

[5] M. Ardalani-Farsa and S. Zolfaghari, "Chaotic time series prediction with residual analysis method using hybrid Elman–NARX neural networks," *Neurocomputing*, vol. 73, no. 13, pp. 2540–2553, 2010.

[6] E. A. Wan, "Modeling nonlinear dynamics with neural networks: Examples in time series prediction," in *PROCEEDINGS-SPIE THE INTERNATIONAL SOCIETY FOR OPTICAL ENGINEERING*, pp. 327–327, Citeseer, 1993.

[7] Yahoo Finance, "S&P 500 Dataset - Yahoo Finance," Dec. 2016.

[8] J. M. P. Menezes and G. A. Barreto, "Long-term time series prediction with the NARX network: an empirical evaluation," *Neurocomputing*, vol. 71, no. 16, pp. 3335–3343, 2008.