International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2016): 79.57 | Impact Factor (2017): 7.296

A Comprehensive Study of Macroeconomic Forecasting Models based on Neural Network

Aashi Agarwal[1], Shalini Gupta[2]

[1]Textile Engineering, IIT Delhi

[2]Chemical Engineering, IIT Delhi

Abstract: ***Economic policies are made after forecasting economic conditions. Currently, we are using linear models that demand large amounts of data and are prone to model dependence. In this paper, we have compared and contrasted four different neural networks to overcome these shortcomings and forecast civilian unemployment accurately*.**

Keywords: Unemployment, neural networks, encoder decoder

1. Introduction

Current methods of macroeconomic forecasting are prone to problems like high data demands, high sensitivity, etc. We have used machine learning and neural network structures to overcome these shortcomings. The encoder decoder model performed best, outperforming the Survey of Professional Forecasters(SPF). Our models are robust to model sensitivity and can account for novel data if needed.

1.1 Existing Forecasting Approaches

Currently, there are two ways in which economic forecasting is done: structural and non structural.

Structural methods use theoretical economics to form the basis of forecasting models. These are good approaches in case one wants to predict conditional forecasts. Neural networks can be used as approximations of these models, however, they can never fully replace them.

Non structural methods of economic forecasting use the properties of data and find a best fit statistical model for them instead of relying on some economic theory for the same. They are employed primarily for unconditional forecasting as they are partial towards predictive accuracy over causal inference. Our neural network models are more suited as a replacement of these.

1.2 Vector Autoregression (VAR)

VAR models demand lesser assumptions regarding the underlying data generating process, can be easily estimated using OLS/GLS methods and allow the use of multiple series for analysis. However, these are inherently linear models. Thus, accounting for non linearities is a difficult and complex task. Another shortcoming of these models is their sensitivity to network architecture. This forces the researcher to make strong assumptions regarding the underlying data generating process. This presents a conflict of interests, since the researchers choose these models explicitly to avoid making such assumptions.

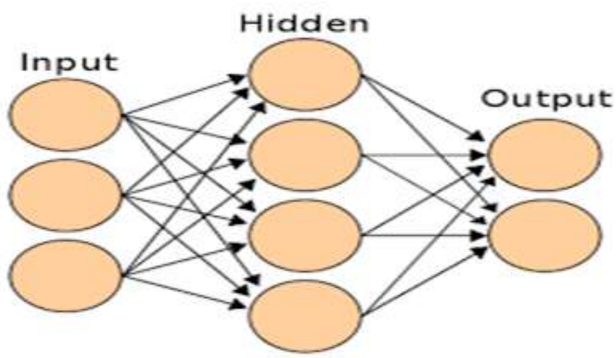
1.3 Consensus Forecasting

This method uses a number of predictions from various models to find an aggregate prediction. This helps to ensure better accuracy of prediction as there can be a few models which give very optimistic or very pessimistic results. This model provides an aggregate of all possible values. This is done to ensure that in case there is some deviation in the prediction due to noise, it gets rectified by using predictions from different models. This model has other benefits as well. It can be used as a benchmark to check the accuracy of new models. If more accurate models are added to this model, its accuracy is improved further. The only limitation associated with this is that we cannot be very sure about the accuracy of the individual models.

2. Methods

Nowadays, neural networks are used extensively in forecasting methods as they provide more accurate results than traditional methods.

A neural network looks like this:



It has an input layer, hidden layer(s) and output layer. The input variables xis are mapped to intermediate hidden layers hj, which are further mapped to output variables yks as follows:

$$hj = f\,(\sum wji*xi + bj)$$

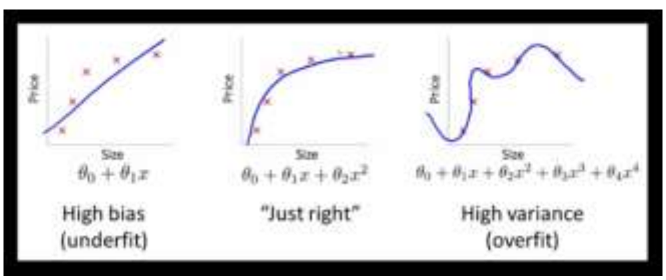
$$yk = g\,(\sum vkj*hj + vk0)$$

where f and g are activation functions of hidden and output layers.

wji and vkj are the "weight parameters" as they give the weightage of every xi and hj in the hidden and output layers respectively. bj and vk0 are called offset parameters. Now we train our model by minimizing errors and hence obtaining optimum values.

But, there are some difficulties which affect the efficiency and effectiveness of neural networks. The most important one is to handle the problem of overfitting.

Overfitting: It occurs when the model fits the training data very accurately (hence giving a complex model) but gives huge errors when tested with new data (test data). This is because while trying to fit well with the training data, it accommodates the fluctuations due to noise, and therefore, is not able to give accurate results for new data. Hence such a model is not good.

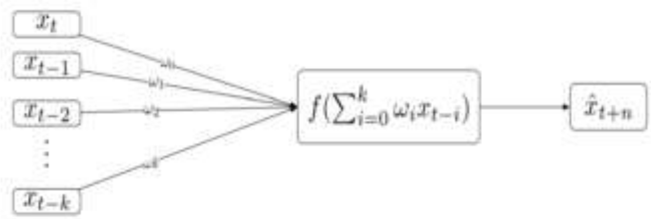


This picture explains the problem of overfitting. Underfit model gives high errors for both training as well as test cases, but overfit model gives very less errors for training set, but high errors for test cases. Hence, we need to have an optimum fit ("just right").

This problem is generally solved by using regularization technique. Some training data is used as validation data to determine the best possible value of regularization parameter, which gives minimum errors for validation data as well.

Apart from overfitting, a huge computational expense is also a problem associated with neural networks. Usually, back propagation algorithm is used to train the data, using gradient descent method to reach optimum values of model parameters, by minimizing the cost function.

The basic architecture upon which deep learning models are built is the perceptron. A perceptron has three distinct sets of nodes: (1) a set of nodes representing model inputs, (2) a set of computational nodes, and (3) a set of nodes representing model outputs.

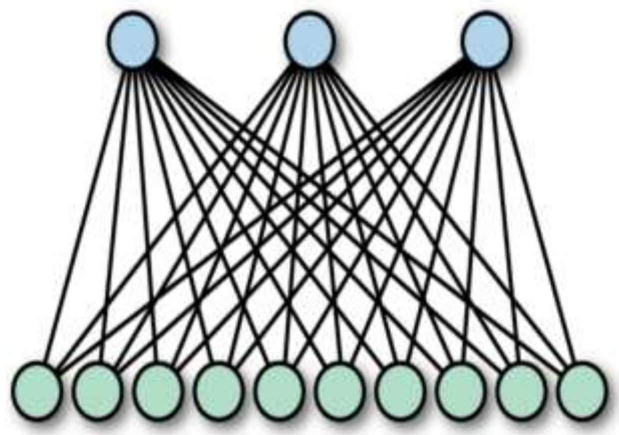


This is a visual representation of perceptron.

2.1 Architecture

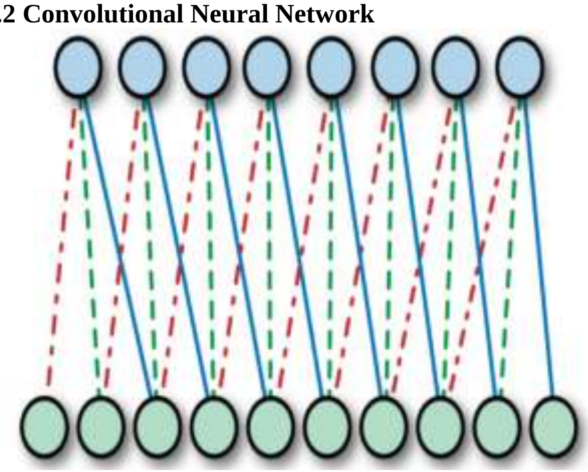
Four main types of architecture have been presented in this paper which are commonly used in deep learning. These are used depending on the type of data whose model is to be predicted.

2.1.1 Fully Connected Architecture



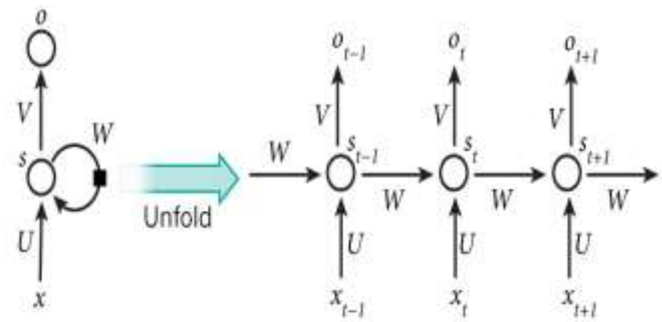
The first architecture is fully connected architecture. In this, inputs to each node are the outputs from the previous one. The node connecting the inputs to the outputs from the last computational layer performs element-wise addition between the layer outputs and the inputs. The actual model contains more number of layers.

2.1.2 Convolutional Neural Network



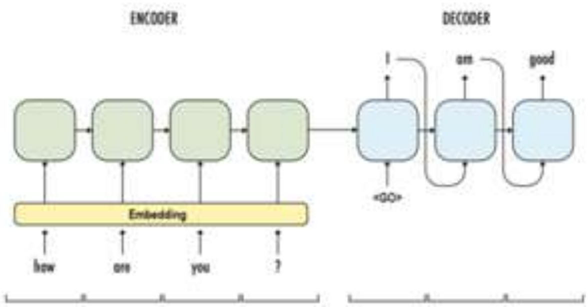
The second architecture i.e. convolutional neural network moves only in one direction, i.e. it is feed forward. In this model, every layer is connected only to a few computational nodes, i.e. each node receives information from selected layers. It is specially used for image recognition. It identifies which patterns maximize predictive accuracy and searches for those patterns in the data.

2.1.3 Recurrent Neural Network



This is the third type of architecture which is specially used for language translation. It takes as input not only the output from the previous node, but all the previous ones in that sequence. It is important because it provides an insight of long term data along with the elements of the recent data. It consists of many layers and has the following properties: each layer receives the long term understanding of the sequence till that point and also the fresh output generated by the previous element itself.

2.1.4 Encoder- Decoder Network



This is an extension of the above architecture. In this we use the inputs from the entire sequence as well as also consider the predictions made previously by the model. This is very useful in tasks like language translation as it keeps a check if the final prediction is grammarly correct or not based on the previous predictions made by it. The first section which considers current responses is known as encoder and the second section which considers both the current responses as well as the previous predictions is known as decoder.

3. Literature Survey

We have used these models to predict civilian unemployment rate. We have used the database from US Bureau of Labour and Statistics that checks the percentage of labour force currently unemployed in US. We chose to predict unemployment because it is a meaningful indicator and does not undergo revision.

0, 3, 6, 9 and 12 forecast horizons have been used. The models were trained on all these forecast horizons thus yielding 20 model variants. The input is the 36 monthly values of UNRATE, and the first and second order differences in UNRATE.

4. Approach

Model is trained on the UNRATE data from 1963 to 1996. The validation dataset comprises of every tenth observation in this data. The data from 1997 to 2014 is used as test data. We have also introduced stochasticity to the training by inclusion of optimization as a result of which there is variation in the weights and consequently forecasts of the trained models. Thus, we trained each model 30 times to accommodate the expected model performance and variance in performance across repeated runs. In addition, we have done regularisation by ignoring the output of randomly selected nodes to avoid overdependence on model structure.

5. Results

Consider Table 1. The performance of each model is presented in terms of Mean Absolute error which is result of measuring the difference between two continuous variables. DARM(Directed Autoregressive Model) and SPF are used as benchmark models. Our models outperform DARM in every case except at the 9th and 12th forecast horizon where DARM outperforms Convolutional and LSTM model.
Table 1: Performance metrics for models

The encoder decoder models outperforms all others at every instance and shows an 89% reduction in error as compared to SPF model. The performance of Fully Connected and Long Short Term Memory models is similar whereas convolution is a little behind them.

At 3 month forecast horizon, FC and CONV models perform better than SPF models. LSTM model lags slightly compared to them. Encoder decoder model is the best performer. At 6 month forecast horizon all three models give comparatively good predictions however, the SPF outperforms FC, LSTM and CONV models. Encoder decoder model is still better than all of them.

SPF demonstrates outcome variations due to set of similar but heterogeneous models whereas our neural networks are repeatedly trained by less heterogeneous models. Thus, neural networks show less variation across repeated trainings compared to SPF models.

5.1 Ensemble Models

To produce ensemble estimates of unemployment we have used a simple perceptron model. We have measured ensemble predictions of each neural network individually and combined prediction of all four models as well. The Table shown below gives performance metrics and ratio of ensemble MAE to benchmark model MAE that tells us how much better the model is performing.

Table 2: Mean Absolute Error for Ensembled models and aggregate SPF forec

Horizon	Metric	Fully Connected	Convolutional	LSTM	Encoder Decoder	Combined	DARM
0 month	MAE	4.3	7.2	4.8	4.1	9.4	11.7
	MAE/DARM	36.3	61.3	41.1	35.2	80.1	
	MAE/SPF	42.3	71.6	48.0	41.1	93.6	
3 month	MAE	21.5	22.4	24.2	18.4	21.5	32.8
	MAE/DARM	65.6	68.4	73.6	56.0	65.6	
	MAE/SPF	93.4	97.3	104.8	[illegible]	[illegible]	
6 month	MAE	41.4	43.8	40.1	30.1	41.6	49.3
	MAE/DARM	83.9	88.9	81.3	61.1	84.3	
	MAE/SPF	116.1	123.0	112.4	84.5	116.6	
9 month	MAE	54.8	73.1	66.4	45.9	65.0	65.8
	MAE/DARM	83.3	111.1	101.0	69.8	98.8	
	MAE/SPF	108.9	145.2	132.0	91.2	129.1	
12 month	MAE	77.9	108.3	90.0	61.8	82.5	90.7
	MAE/DARM	85.8	119.3	99.1	68.1	91.0	
	MAE/SPF	123.6	171.8	142.7	[illegible]	131.0	

All metrics presented in basis points.

Similar to our previous observations, encoder decoder models aain outperforms SPF models at all forecast horizons. FC, CONV and LSTM models remain competitive with SPF models at 3 and 6 month horizons but fall behind SPF in third and fourth quarter horizons. They all perform better than DARM with the exception of LSTM and and CONV ensembles.

5.2 Encoder Decoder Model Performance

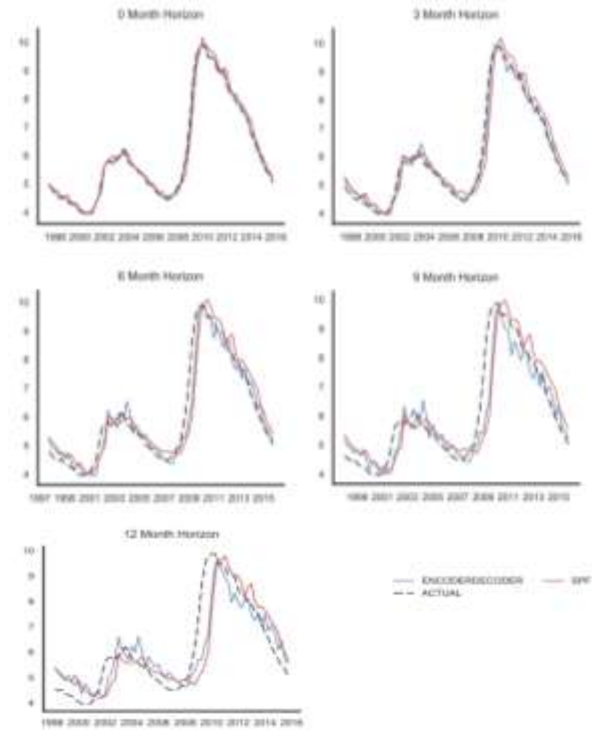
The figure below shows the predictions of encoder decoder model at every quarter from Q1 1997 to Q4 2004. We have also provided the SPF forecasts and actual values. The encoder decoder predictions are better compared to SPF forecasts.

An interesting point to note is the location of inflection points during 2007-08. There was a financial crisis at this time. The encoder decoder model is able to reverse course quicker compared to the SPF models implying its responsiveness and ability to adjust to major changes in the data.

Table 3 provides the inflection points of minimum and maximum unemployment for 2007-08 financial crisis and recovery. The encoder decoder model responds to recession and recovery faster than SPF models.

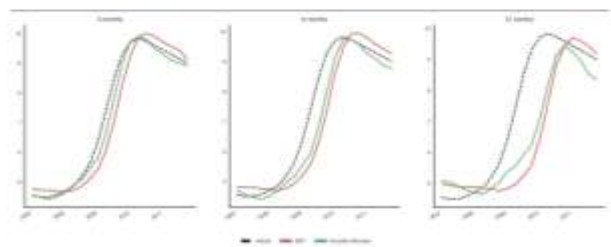
Table 3: Inflection points for SPF and encoder-decoder forecasts near the 2007 recession

	SPF	Encoder Decoder
Unemployment Nadir (Q1 2007)		
3 Month Horizon Model	Q3 2007	Q1 2007
6 Month Horizon Model	Q3 2007	Q2 2007
9 Month Horizon Model	Q2 2008	Q3 2007
12 Month Horizon Model	Q3 2008	Q1 2008
Unemployment Apex: Q1 2010		
3 Month Horizon Model	Q1 2010	Q4 2009
6 Month Horizon Model	Q2 2010	Q4 2009
9 Month Horizon Model	Q3 2010	Q1 2010
12 Month Horizon Model	Q4 2010	Q2 2010



Each facet represented here shows the predictions of the SPF (red line) and best-performing encoder-decoder model (blue line). The black line indicates the actual, observed level of unemployment.

Figure: UNRATE Forecasts by Encoder Decoder Model at different horizons



This is a representation of forecasts of unemployment over 2007-2009 recession/recovery. The vertical axis represents the level of unemployment. The horizontal axis represents time (in quarters). The lines in the figure portray (smoothed) unemployment forecasts of the encoder-decoder model (green) along with the SPF model predictions (red). The remaining line (black) indicates the observed level of unemployment.

Figure: Smoothed forecasts of SPF and encoder-decoder model at three different quarter horizons

6. Conclusion

The models presented provide near accurate results for forecasting unemployment. They all represent individual characteristics which are useful for different situations. Mostly, all of them are accurate with respect to near term forecasting. Encoder decoder network provides additional advantage as it also adds value based on the previous experience of the model. Hence, it provides added accuracy in certain cases.

Neural networks have been proved to be very useful models despite the limitations associated with them. The major limitation is the huge computational requirement and the time required by them to converge in case of large data. Other problems include overfitting, dependance of accuracy on architecture of model when we have less training data and difficulty in training non stationary data.

7. Future Scope

1) To further improve the performance of these models, data related to other relevant parameters should also be included (which would increase the complexity of the model) like labour flows, technological advancements, probability of a catastrophic event etc. These additional indicators will provide more information regarding the model and hence give better results.
2) Better architectures should be implemented to get more accurate results. This paper has been limited to a few simple architecture models. New techniques can be used to develop better models.

References

[1] Lucas, Robert E (1976). “Econometric policy evaluation: A critique”. In: CarnegieRochester conference series on public policy. Vol. 1. Elsevier, pp. 19–46.
[2] Pescatori, Andrea, Saeed Zaman, et al. (2011). “Macroeconomic models, forecasting, and policymaking”. Economic Commentary 19.20, p. 1.
[3] Durland, J. M., & McCurdy, T. H. (1994). Duration-dependent transitions in a Markov model of US GNP growth. Journal of Business & Economic Statistics, 12(3), 279-288. https://doi.org/10.1080/07350015.1994.10524543
[4] Faust, J., & Wright, J. H. (2013). Forecasting inflation. Handbook of economic forecasting, 2(Part A), 3-56. https://doi.org/10.1016/b978-0-444-62731-5.00023-3
[5] “Human Decisions and Machine Predictions,” with Jon Kleinberg, Himabindu Lakkaraju, Jure Leskovec, and Jens Ludwig
[6] “Targeting Poverty by Predicting Poverty: Using Machine Learning in Targeted Transfer Program, with Melissa Adelman, Jonathan Glidden, Paul Niehaus, and Jack Willis
[7] Stockton, D., 2012. Review of the Monetary Policy Committee’s Forecasting Capability, presented to the Court of the Bank of England http://www.bankofengland.co.uk/publications/Documents/news/2012/cr3stockton.pdf Varian, H.R., 2014. Big data: New tricks for econometrics. The Journal of Economic Perspectives, 28(2), pp.3-27
[8] Zarnowitz, V. and Braun, P., 1993. Twenty-two years of the NBER-ASA quarterly economic outlook surveys: aspects and comparisons of forecasting performance. In Business cycles, Indicators and Forecasting (pp. 11-94). University of Chicago Press.

Author Profile

Aashi Agarwal, Senior Undergraduate, Textile Engineering, IIT Delhi

Shalini Gupta, Senior Undergraduate, Chemical Engineering, IIT Delhi