Forecasting of India's Merchandise Exports: A comparative analysis of Classical Time Series Methods and Deep Learning Method(LSTM in Particluar)

Prepared By:
Ansuman Sharma
Officer Trainee
Indian Economic Service,2022

Supervised By:
Dr. Arup Mitra
Professor,Institute of Economic Growth

Acknowledgement

I would like to express my deep and sincere gratitude to my supervisor Dr. Arup Mitra. His guidance has helped me to carry the research and overcome challenges while working with the paper.He taught me the methodology to carry out the research while i was facing problems in between.I am extremely grateful to what he has offered me.

Abstract

Exports is one of the important factor responsible for the growth of an economy.Forecasting exports accurately can help the government in taking key decisions for the growth of the economy.This paper aims to forecast exports for the future 4 quarters and tries to to compare different modelling techniques to get better forecasting results.Thus in this regard 3 models are used such as ARIMA,VECM and LSTM for forecasting Exports.For examining the accuracy of the forecast Root Mean Square Error(RMSE) is employed.The results show that Deep learning method,LSTM provides a better forecast compared to Time Series Methods VECM and ARIMA.

Keywords: ARIMA,VECM,LSTM,Forecasting,Exports

I Introduction

Export is an important component of the GDP of the country and exports have direct impact on balance of payment,current account deficit and have indirect impact on creation of employment,reduction of poverty and subsequently economic growth of the country.Several factors are responsible for the export performance of the country.It can be both supply and demand side factors. The demand side factors are Gross Domestic Product(GDP) of the trading partner,exchange rate,price of the product,political situation of the trading partner.The supply side factors are wage rate,imports, world prices,domestic productivity.India's export to gdp ratio in 2013-14 was 25.4 percent and came down to 23 percent in 2014-15 and further declined to 18.8 percent in 2017-18 due to faster growth of gdp compared to exports.In 2021-22 India crossed 400 billion dollar mark for the first time and export to gdp crossed 20 percent.

Between 2012-13 and 2016-17 export growth averaged 6.9 percent and between

2016-17 and 2021-22 it averaged 11.5 percent.In the last five years the performance has been driven by merchandise exports.India's share of in global merchandise exports increased from 0.66 percent to 1.55 percent between 2000 to 2020 while China's merchandise export share increased from 3.8 percent to 14.6 percent in the same time period.To compete with China forecasting exports accurately can help in making policy decisions which can help the country to boost exports and increase the share in global merchandise trade. More number of schemes can be introduced to boost exports like PLI scheme.

This paper considers different model to forecast exports taking quarterly time series data and compares different models for better forecasting results.The forecasting is done using 3 models with two models being the classical regression model and one using the deep learning techniques as it is used widely nowadays to forecast time series data.ARIMA(Autoregressive Integrated Moving Average) and VECM(Vector error correction Model) being used to forecast 4 quarters ahead and LSTM(Long short term memory), a recurrent neural network technique being used to forecast 4 quarters ahead. A comparative analysis is being done based on RMSE(root mean square error) metric to find out the best fit on the test data.

II Literature Review

There are several models to forecast time series data. It is done based on the analysis of past historical data as the data in time series are correlated. There are several parametric and non parametric models of time series.The classical time series models like ARIMA,VAR or VECM make some assumptions on the distribution of data.Thats why they are called parametric models.The tree based models like Random forest,XGboost or RNN models like LSTM do not make any assumptions regarding the distribution of data.Thats why they are called non parametric models.Due to this convenience of not not going for any assumptions or not going for residual diagnostics now a days people are shifting to non parametric models if our purpose is forecasting only. For impact analysis of any independent variable we should go for parametric models as the non parametric models are black box.

The advantage of neural networks or RNN over traditional models are when the size of training set increases its performance keeps on increasing due to the back propagation techniques the model gets optimized and it gives accurate results compared to traditional models(Hornik et al. 1989; 1990).Increasing number of hidden layers and number of neurons gives more better results.But in case of less number of data the results of traditional models and neural networks can be at par.

In the regression models we first describe the relation between dependent and independent variable based on some hypothesis and all the classical assumptions should be satisfied to make the model ready for forecasting.Such things are not required for neural networks as the models learns the relationship between the dependent and independent variable through the hidden layers by the technique of back propagation(Gronholdt and Martensen, 2005).Neural Networks have the ability to ignore data that is not important and consider data that is more important.The flexibility of neural network in making both the linear and non linear relationships makes the model more superior to that of traditional models.

Hill and O'Connor (1996) compare time series forecasts based on neural networks with forecasts from traditional statistical time series methods including exponential smoothing, Box-Jenkins and a judgment-based method. According to their results, the neural network model outperforms traditional statistical methods when forecasting quarterly and monthly data which is used in the forecasting competition of Makridakis et al. (1982). Moreover, the neural network model has almost always lower variance of forecasts than those of the traditional models.

Kuo and Reitsch (1996) test the accuracy of forecasts produced by time series, multiple regression and neural network models by using two data sets first of which consists of 56 months measuring 14 variables and the second of which is time series data where the dependent variable is monthly dollar sale volumes of a tuxedo rental firm. Their results prove that neural networks tend to do better than conventional methods in all cases. They find neural networks especially valuable where inputs are highly correlated,some data is missing or the systems are non-linear.

Neural networks are not necessarily the best in all cases. For example, Mileris Boguslauskas (2011) use statistical (discriminant analysis and logistic regression) and artificial intelligence methods (artificial neural networks) in order to classify banks' clients according to their credit risk. They find that neural networks are the second best.

The results of these studies are quite promising for the neural networks, nevertheless most of them are specific to a data set. Therefore, it is not possible to claim that neural networks will always perform better than traditional models. Besides, most studies carried out so far compare traditional econometric models and neural networks; however, they are not necessarily substitutes. It is possible to combine neural networks with regression analysis to generate a much stronger forecasting tool (Kabundi, 2004).

III Modelling Techniques

Since this paper is based on comparative analysis of forecasting results,3 modelling techniques will be discussed here.They are ARIMA,VECM and LSTM(Type of RNN).

1. **ARIMA**

   (a) Autoregressive Process(AR):

   This model is based on the linear relationship of the past values of the variable. The basic assumption is ϵ_t is a purely a random process with zero mean and constant variance.The AR(p) process can be given as:

   $$X_t = c + \alpha_1 X_{t-1} + \alpha_2 X_{t-2} + ... + \alpha_p X_p + \epsilon_{t-p}$$

   where ϵ_t is white noise.

   (b) Moving Average Process(MA):

   This model is based on linear relationship of the current and past disturbances that feed into the current value of time series.The basic assumption is ϵ_t is a purely a random process with zero mean and constant variance.The MA(p) process can be given as:

   $$X_t = c + \alpha_1 \epsilon_{t-1} + \alpha_2 \epsilon_{t-2} + ... + \alpha_p \epsilon_{t-p}$$

   (c) Auto-Regressive Integrated Moving Average Process (ARIMA):

   This model is proposed by Box and Jenkins(1976) which is a set of models that describe the process as a function of its own lags and white noise process.It is a combination of AR and MA process with 'I' representing the differencing required to make the variable stationary.The AR part of ARIMA indicates that the evolving variable of interest is regressed on its own lagged (i.e., prior) values. The MA part indicates that the regression error is actually a linear combination of error terms whose values occurred contemporaneously and at various times in the past.[7] The I (for "integrated") indicates that the data values have been replaced with the difference between their values and the previous values (and this differencing process may have been performed more than once).The model ARIMA(p,d,q) if it is integrated of order 'd' can be given as:

$$X_t = c+\alpha_1 X_{t-1}+\alpha_2 X_{t-2}+....+\alpha_p X_{t-p}+\beta_1\epsilon_{t-1}+\beta_2\epsilon_{t-2}+....+\beta_p\epsilon_{t-q}+\epsilon_t$$

where ϵ_t is white noise.

2. **VECM**

Vector Autoregression(VAR) was introduced by Sims(1980) to analyse joint dynamic behaviour of collection of variables.A VAR system contains a set of m variables, each of which is expressed as a linear function of p lags of itself and of all of the other m – 1 variables, plus an error term.With two variables x and y and p-order VAR can be given by:

$$y_t = \beta_{y0} + \beta_{yy1} y_{t-1} + ... + \beta_{yyp} y_{t-p} + \beta_{yx1} x_{t-1} + ... + \beta_{yxp} x_{t-p} + \epsilon_{yt}$$

$$x_t = \beta_{x0} + \beta_{xy1} y_{t-1} + ... + \beta_{xyp} y_{t-p} + \beta_{xx1} x_{t-1} + ... + \beta_{xxp} x_{t-p} + \epsilon_{xt}$$

where ϵ_{yt} and ϵ_{xt} are uncorrelated white noise disturbances and y_t and x_t are stationary series.

When the two series are conintegrated we do not go for VAR modelling rather we go for VECM modelling. Two or more series are said to be cointegrated if there exists some long term association between the series.Robert Engle and Clive Granger introduced the concept of cointegration in 1987.If two series X_t and Y_t are integrated of order 'd' and there exist coefficients a,b such that aX_t + bY_t is integrated of order less than 'd', then the series X_t, Y_t are cointegrated.

The VECM model of p-order for two variables can be given as:

$$\Delta y_t = \beta_{yo}+\beta_{y1}\Delta y_{t-1}+...+\beta_{yp}\Delta y_{t-p}+\gamma_{y1}\Delta x_{t-1}+...+\gamma_{yp}\Delta x_{t-p}+\lambda_y(y_{t-1}-\alpha_0-\alpha_1 x_{t-1})+\epsilon_{yt}$$

$$\Delta x_t = \beta_{xo}+\beta_{x1}\Delta y_{t-1}+...+\beta_{xp}\Delta y_{t-p}+\gamma_{x1}\Delta x_{t-1}+...+\gamma_{xp}\Delta x_{t-p}+\lambda_x(y_{t-1}-\alpha_0-\alpha_1 x_{t-1})+\epsilon_{xt}$$

where $y_t = \alpha_0 + \alpha_1 x_t$ is the long run cointegrating relationship between two variables and λ_y and λ_x are the error correction parameters that measure how y and x react to deviations from long run equilibrium.

3. **Neural Networks**

Neural Networks are a set of algorithms that tries to recognize the patterns, relationships, and information from the data through the process which is inspired by and works like the human brain.A simple neural network has three components.

   (a) Input layer

   (b) Hidden Layer

   (c) Output Layer

Input Layer:

Also known as Input nodes are the inputs/information is provided to the model to learn and derive conclusions from. Input nodes pass the information to the next layer i.e Hidden layer.

Hidden Layer:

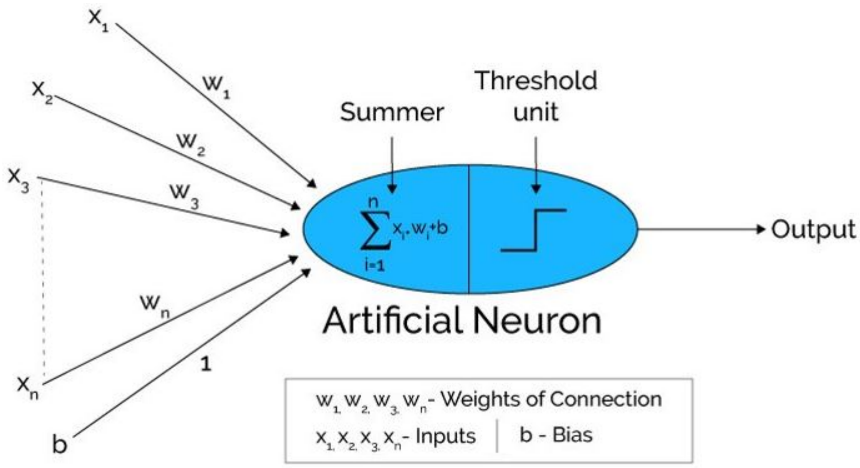
Hidden layer is the set of neurons where all the computations are performed on the input data. There can be any number of hidden layers in a neural network. The simplest network consists of a single hidden layer.There can be multiple hidden layer to improve the accuracy of results.

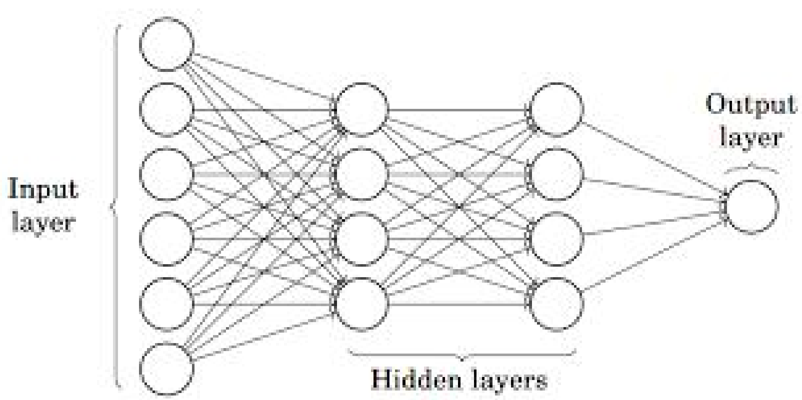
Output layer:

The output layer is the output/conclusions of the model derived from all the computations performed. There can be single or multiple nodes in the output layer.The output can be in continuous or discrete form.

Perceptron:

Perceptron is simple form of neural network which consists of a single layer while multi layer perceptron also known as artificial neural network consists of more than one perceptron grouped together to form multi layer neural network.Below is the illustration of perceptron and neural network with hidden layers.





(a) Input data is passed with some weights attached to it to the hidden layer.

(b) Each hidden layer consists of neurons. All the inputs are connected to each neuron.

(c) After passing on the inputs, all the computation is performed in the hidden layer.The inputs are multiplied by their weights. Weight is the gradient or coefficient of each variable. It shows the strength of the particular input. After assigning the weights, a bias variable is added. Bias is a constant that helps the model to fit in the best way possible.

$$y_1 = w_1x_1 + w_2x_2 + ...w_nx_n + b$$

(d) The activation function is applied to the linear equation y_1. The activation function is a nonlinear transformation that is applied to the input before sending it to the next layer of neurons. The importance of the activation function is to inculcate nonlinearity in the model.

(e) After passing through every hidden layer, we move to the last layer i.e our output layer which gives us the final output.

(f) After getting the predictions from the output layer, the error is calculated i.e the difference between the actual and the predicted output.

(g) The process described above are forward propagation.The weights are updated to minimise the error between actual and predicted output.This process of updating and finding the optimal values of weights or coefficients which helps the model to minimize the error is called Back Propagation.

Gradient Descent:

Gradient Descent is an optimization techniques to update weights until the error(Difference between actual value and predicted value) is minimized.

i. First, the weights are initialized randomly i.e random value of the weight, and intercepts are assigned to the model while forward propagation and the errors are calculated after all the computation.

ii. Then the gradient is calculated i.e derivative of error w.r.t current weights.

iii. Then new weights are calculated using the below formula, where a is the learning rate which is the parameter also known as step size to control the speed or steps of the back propagation. It gives additional control on how fast we want to move on the curve to reach global minima.

$$w_{xnew} = w_x - \alpha * \frac{\partial \epsilon}{\partial w_x}$$

iv. This process of calculating the new weights, then errors from the new weights, and then updating of weights is continued till we reach global minima and loss is minimized.

Activation Function:

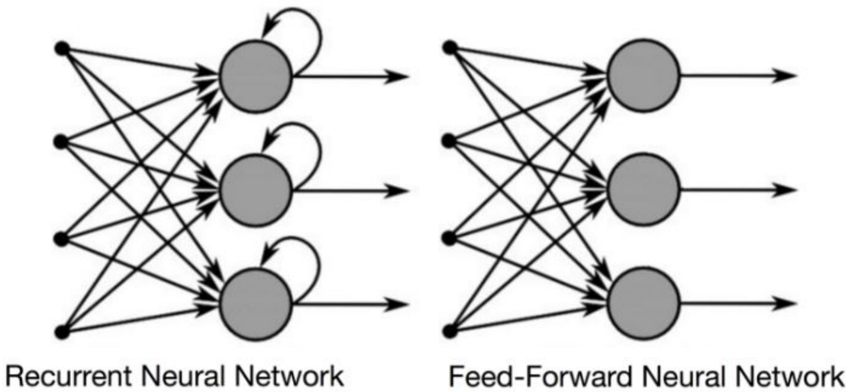
An activation function in a neural network defines how the weighted sum of the input is transformed into an output from a node or nodes in a layer of the network.The choice of activation function has a large impact on the capability and performance of the neural network, and different activation functions may be used in different parts of the model.There can be several Activation functions and Most common ones are Sigmoid Activation function and ReLU activation function.

Recurrent Neural Network(RNN):

RNN works on the principle of saving the output of a particular layer and feeding this back to the input in order to predict the output of the layer.They are preferred algorithm for sequential data like Time Series.

Feed-forward neural networks have no memory of the input they receive and are bad at predicting what's coming next. Because a feed-forward network only considers the current input, it has no notion of order in time. It simply can't remember anything about what happened in the past except its training.

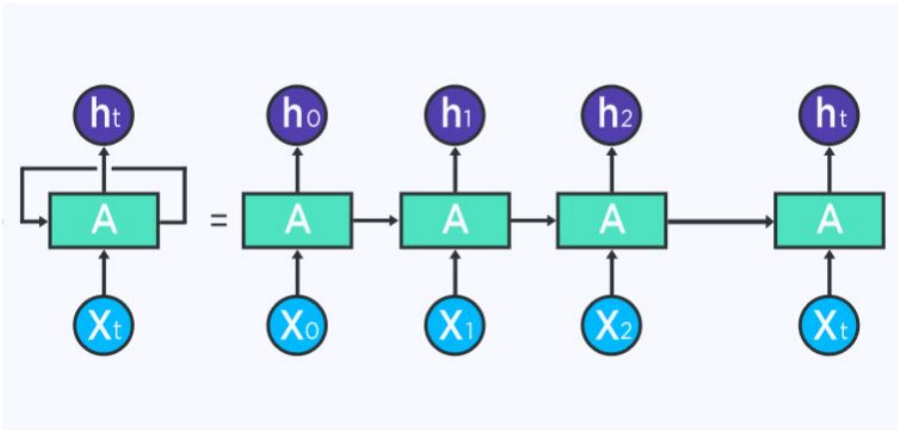
In a RNN the information cycles through a loop. When it makes a decision, it considers the current input and also what it has learned from the inputs it received previously.It produces output, copies that output and loops it back into the network. Below is the illustration.RNN has two inputs: the present and the recent past. This is important because the sequence of data contains crucial



information about what is coming next, which is why a RNN can do things other algorithms can't.

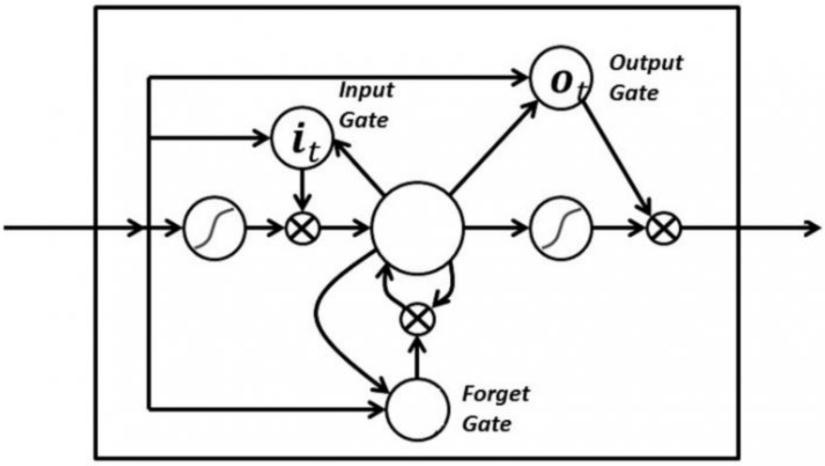
A feed-forward neural network assigns, like all other deep learning algorithms, a weight matrix to its inputs and then produces the output. Note that RNNs apply weights to the current and also to the previous input. Furthermore, a recurrent neural network will also tweak the weights for both through gradient descent and backpropagation through time(BPTT).

RNN as a sequence of neural networks that you train one after another with backpropagation.The image below illustrates an unrolled RNN. On the left, the



RNN is unrolled after the equal sign. Note there is no cycle after the equal sign since the different time steps are visualized and information is passed from one time step to the next. This illustration also shows why a RNN can be seen as a sequence of neural networks. Within BPTT the error is backpropagated from the last to the first timestep, while unrolling all the timesteps. This allows calculating the error for each timestep, which allows updating the weights.

The limitation of RNN is as the gap between relevant information grows they are not capable to learn to connect information.Technically they are not capable of handling long term dependencies.Here comes Long Short Term Memory(LSTM) algorithm in play.LSTM is a special kind of RNN capable of leaning long term dependencies.They were introduced by Hochreiter Schmidhuber (1997) which became popular later.Therefore it is well suited to learn from important experiences that have very long time lags in between.In an LSTM you have three gates: input, forget and output gate. These gates determine whether or not to let new input in (input gate), delete the information because it isn't important (forget gate), or let it impact the output at the current timestep (output gate). Below is an illustration of a RNN with its three gates: The gates in an LSTM



are analog in the form of sigmoids, meaning they range from zero to one. The fact that they are analog enables them to do back propagation.

IV Data Source and Methodology

IV.1 Data Source

The Data used in this study is mainly from RBI from time period April 1996 to March 2020 which has quarterly frequency.The sample has 96 observations and forecasting is being done 4 quarter ahead.Time series data obtained is multivariate in nature having 4 variables such as exports,GDP,Inflation and Real Effective exchange rate(REER).

IV.2 Methodology

Three models ARIMA,VECM and LSTM are being used for forecasting.

1. **ARIMA**

   ARIMA is an univariate analysis of time series.The quarterly data taken is from 1996 April to March 2019 which is 92 observations.4 quarterly observations from April 2019 to March 2020 is kept out of sample for training.These 4 observations is used for calculating Root Mean Square Error of the forecasted value to compare between models.The model with least RMSE is a better fit.

   Steps followed for ARIMA:

   (a) The univariate data exports should first be stationary for further analysis.The stationarity test can be done using Augmented Dickey Fuller Test. If the series is not stationary at levels then we can go first differencing or second differencing or transformation of data.

   (b) Once the data is stationary we plot ACF(autocorrelation function) and PACF(Partial Autocorrelation function) which helps in deciding the order of AR and MA process and finally we get the ARMA structure with least AIC or BIC values.

   (c) After fitting the model we go for residual structures.The error should be white noise,ARMA process is covariance stationary(AR roots lie within unit circle) and ARMA process is invertible(MA roots lie inside the circle).

(d) After residuals diagnostic and if they satisfy the stability conditions then we can go for forecasting.

2. **Vector Error Correction Model**

This is a multivariate analysis of time series.Here we have 92 observations for modelling and 4 observations out of sample to test the model performance.Here we have three more variables GDP,Inflation and REER apart from Exports. VECM is similar to VAR modelling except it contains the error correction term to analyse the short run dynamics when the series are cointegrated. Since the series came out to be cointegrated we will discuss VECM methodology.

(a) First we check for cointegration between the series in multivariate time series and if the series comes out to be cointegrated we go for VECM modelling instead of VAR modelling.

(b) In our analysis the series as per the johanssen cointegration test came out to be cointegrated. So instead of VAR we are going for VECM modelling.

(c) The lag length in VECM modelling is determined from VAR modelling at levels where optimal lag structure is determined using AIC or BIC criteria which is then subtracted from one.

(d) Then we go for residual diagnostics check to check whether the residuals are white noise or not.

(e) After residual diagnostics, if the errors are white noise we can go for forecasting. We can also go for variance decomposition analysis to check for how much of forecast error variance is explained in each variable by other variables.

3. **Long Short Term Memory(Neural Networks)**

Neural Networks paved a new path to the emerging AI industry since decades it has been introduced.Neural Networks do not have memory and to work with sequential data like time series LSTM has been introduced which is a special type of recurrent neural network. They address the memory issue by giving a feedback mechanism that looks back to the previous output and serves as a kind of memory.LSTM, short for Long Short Term Memory, as opposed to RNN, extends it by creating both short-term and long-term memory components to efficiently study and learn sequential data.

(a) The training data of 92 observations is feed into neural network having two stacked LSTM layers with dropout ratio of 0.2

(b) We take dropout ratio to prevent overfitting of data and since there are only 92 observation we take the number of neuron as 20 for first LSTM layer and 10 for second LSTM layer.

(c) The optimal number number of neurons and dropout ratio is found by running the model until we have both training error and validation error to be minimum.

(d) The optimizer taken to update weights so that cost function is minimized is Adam optimizer.

(e) Finally activation function is take as ReLU which is $Max(0, x)$.Activation function decides, whether a neuron should be activated or not by calculating weighted sum and further adding bias with it. The purpose of the activation function is to introduce non-linearity into the output of a neuron.

(f) A neural network without an activation function is essentially just a linear regression model. The activation function does the non-linear transformation to the input making it capable to learn and perform more complex tasks.

(g) After building the model,the model is trained with validation split as 0.1 so that we can check for validation error and we train the model with with different epochs and batch size until we have minimum validation error.

(h) Once the model is trained we can forecast 4 quarter ahead and rmse can be calculated comparing it with out of sample data.

V Results

1. **ARIMA**:

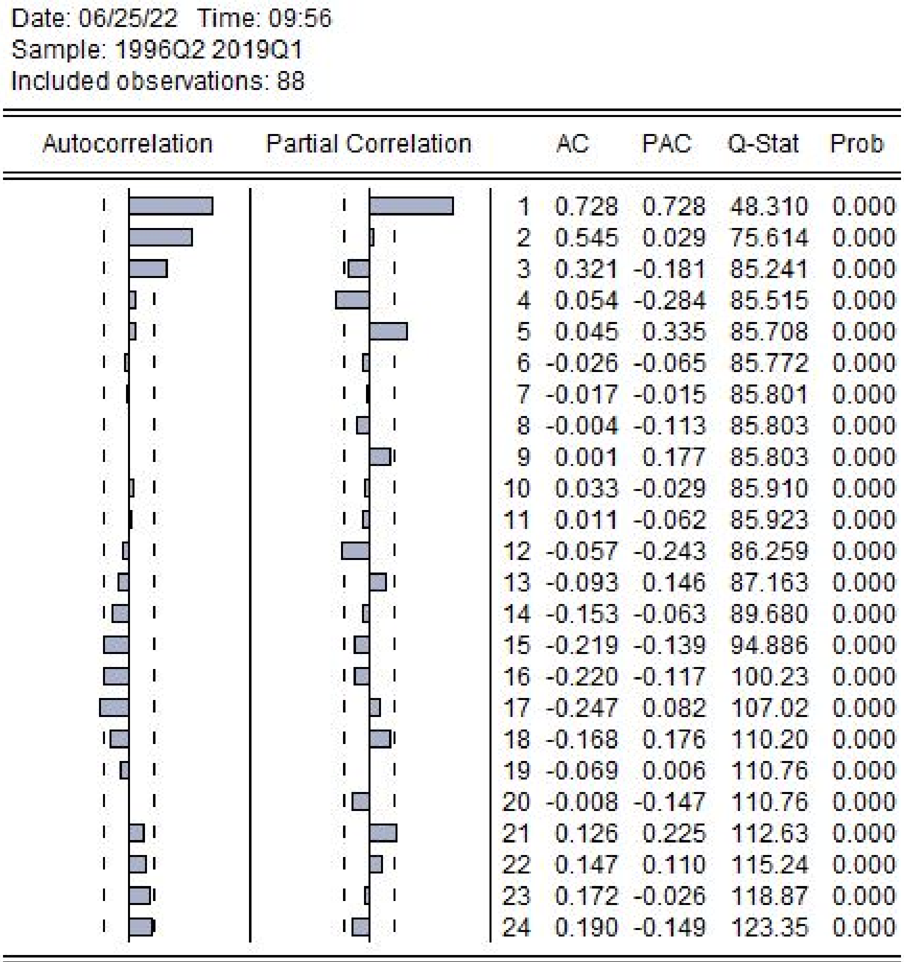
   The Data used in this methodology is from April 1996 to March 2019. These are quarterly data. The exports are non stationary at levels but are stationary at first difference. The ADF results for the exports at first difference is shown below.

Null Hypothesis: DEXPORTS has a unit root
Exogenous: Constant, Linear Trend
Lag Length: 0 (Automatic - based on SIC, maxlag=11)

		t-Statistic	Prob.*
Augmented Dickey-Fuller test statistic		-3.636663	0.0324
Test critical values:	1% level	-4.066981	
	5% level	-3.462292	
	10% level	-3.157475	

*MacKinnon (1996) one-sided p-values.

After the data is stationary we check for ARMA structure from correlogram below shown in the figure which is coming as ARIMA(5,1,3).



Date: 06/25/22 Time: 09:56
Sample: 1996Q2 2019Q1
Included observations: 88

Autocorrelation	Partial Correlation		AC	PAC	Q-Stat	Prob
		1	0.728	0.728	48.310	0.000
		2	0.545	0.029	75.614	0.000
		3	0.321	-0.181	85.241	0.000
		4	0.054	-0.284	85.515	0.000
		5	0.045	0.335	85.708	0.000
		6	-0.026	-0.065	85.772	0.000
		7	-0.017	-0.015	85.801	0.000
		8	-0.004	-0.113	85.803	0.000
		9	0.001	0.177	85.803	0.000
		10	0.033	-0.029	85.910	0.000
		11	0.011	-0.062	85.923	0.000
		12	-0.057	-0.243	86.259	0.000
		13	-0.093	0.146	87.163	0.000
		14	-0.153	-0.063	89.680	0.000
		15	-0.219	-0.139	94.886	0.000
		16	-0.220	-0.117	100.23	0.000
		17	-0.247	0.082	107.02	0.000
		18	-0.168	0.176	110.20	0.000
		19	-0.069	0.006	110.76	0.000
		20	-0.008	-0.147	110.76	0.000
		21	0.126	0.225	112.63	0.000
		22	0.147	0.110	115.24	0.000
		23	0.172	-0.026	118.87	0.000
		24	0.190	-0.149	123.35	0.000

Dependent Variable: DEXPORTS				
Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	33327.60	10922.14	3.051379	0.0032
AR(1)	-0.016055	0.111872	-0.143515	0.8863
AR(2)	0.176913	0.099279	1.781980	0.0789
AR(3)	0.092599	0.112424	0.823655	0.4128
AR(4)	0.037964	0.120920	0.313962	0.7544
AR(5)	-0.007027	0.095412	-0.073652	0.9415
MA(1)	0.938071	0.059963	15.64407	0.0000
MA(2)	0.951169	0.072543	13.11177	0.0000
MA(3)	0.986483	0.048765	20.22932	0.0000
R-squared	0.731274	Mean dependent var		26224.58
Adjusted R-squared	0.702223	S.D. dependent var		38078.63
S.E. of regression	20779.12	Akaike info criterion		22.82338
Sum squared resid	3.20E+10	Schwarz criterion		23.08566
Log likelihood	-938.1701	Hannan-Quinn criter.		22.92875
F-statistic	25.17168	Durbin-Watson stat		2.095748
Prob(F-statistic)	0.000000			
Inverted AR Roots	.62	.13	-.15+.40i	-.15-.40i
	-.47			
Inverted MA Roots	.02-1.00i	.02+1.00i	-.99	

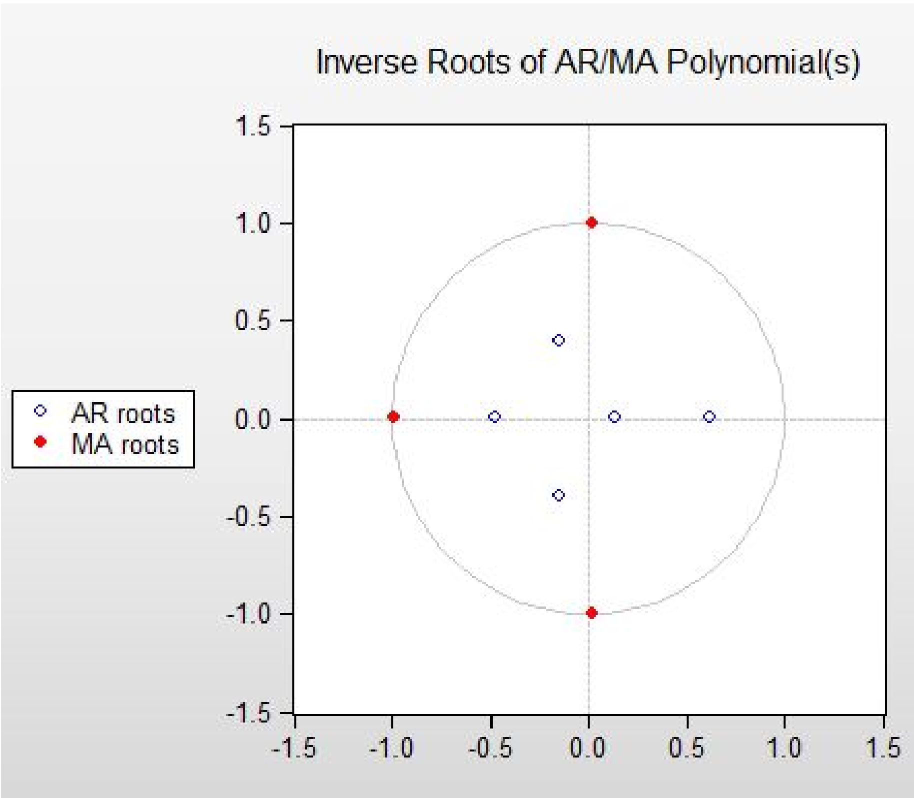
The results of ARIMA(5,1,3) equation is shown in the table above. Some of the coefficients of AR lags are not singnificant.Since this model had Minimum AIC and Maximum Adjusted R Square we considered this structure.

The error in residual diagnostics are coming to be white noise and The AR and MA roots are not lying outside the unit circle.Below are the results attached. The forecasted values for the four quarter ahead from 2019 April to 2020 March is given below in the table:

Values(in Crores)	2019Q2	2019Q3	2019Q4	2020Q1
Forecasted Values	602406.84	620340.01	626894.09	644781.22
Actual Values	562830.4572	550954.023	563063.8411	542483.2108

Q-statistic probabilities adjusted for 8 ARMA terms

Autocorrelation	Partial Correlation		AC	PAC	Q-Stat	Prob
		1	-0.052	-0.052	0.2320	
		2	-0.067	-0.070	0.6204	
		3	-0.065	-0.073	0.9975	
		4	-0.027	-0.041	1.0648	
		5	0.055	0.042	1.3433	
		6	-0.042	-0.046	1.5027	
		7	-0.020	-0.024	1.5413	
		8	-0.086	-0.091	2.2322	
		9	-0.005	-0.022	2.2343	0.135
		10	0.147	0.127	4.3183	0.115
		11	0.055	0.064	4.6195	0.202
		12	-0.138	-0.125	6.5082	0.164



2. **Vector Error Correction Model:**

Here we check for stationarity of other 3 variables along with exports. GDP,Inflation and REER are stationary at first difference.Below are the results of unit root test at first difference:

Variable	ADF t statistic	Critical Value(1 per)	Critical Value(5 per)	Critical Value(10 per)
DExports	-3.63	-4.06	-3.46	-3.15
DGDP	-4.51	-4.06	-3.46	-3.15
DInflation	-2.23	-2.59	-1.94	-1.61
DREER	-3.45	-3.51	-2.89	-2.58

In multivariate time series we check for cointegration to decide whether to go for VAR or VECM. In our data the series came to be cointegrated from the Johanssen cointegration test. So instead of VAR we go for VECM modelling.

Trend assumption: Linear deterministic trend
Series: EXPORTS GDP INFLATION REER
Lags interval (in first differences): 1 to 4

Unrestricted Cointegration Rank Test (Trace)

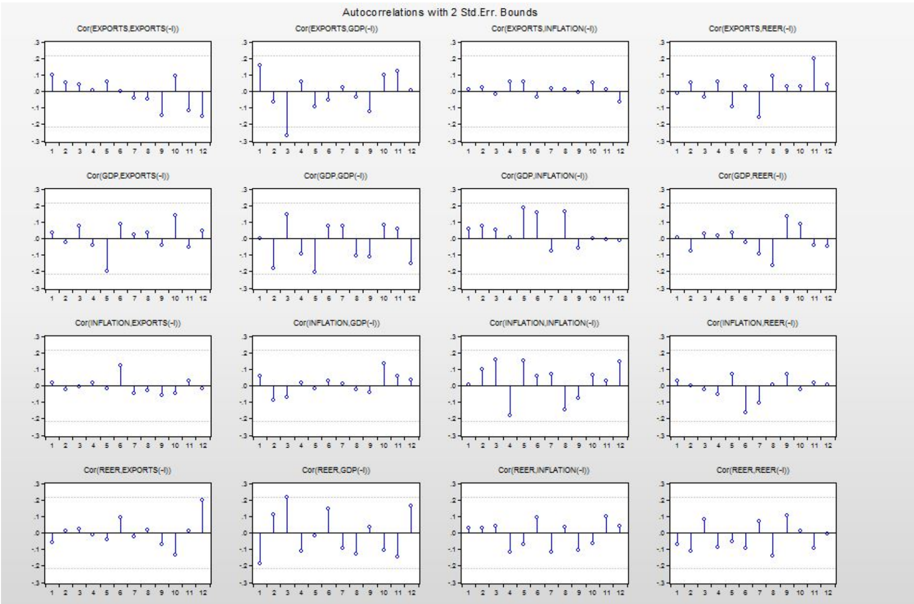
Hypothesized No. of CE(s)	Eigenvalue	Trace Statistic	0.05 Critical Value	Prob.**
None *	0.303525	55.63366	47.85613	0.0079
At most 1	0.145616	24.16375	29.79707	0.1936
At most 2	0.094548	10.47215	15.49471	0.2462
At most 3	0.020829	1.831262	3.841466	0.1760

Trace test indicates 1 cointegrating eqn(s) at the 0.05 level
* denotes rejection of the hypothesis at the 0.05 level
**MacKinnon-Haug-Michelis (1999) p-values

The lag structure for VECM is determined by running VAR at levels and optimal lag length is obtained using AIC criteria. For VECM lag length will be p-1 where p is the optimal lag length obtained from AIC criteria from running VAR at levels.Below are the results:

Sample: 1996Q2 2019Q1 Included observations: 84						
Lag	LogL	LR	FPE	AIC	SC	HQ
0	-2703.383	NA	1.16e+23	64.46151	64.57726	64.50804
1	-2354.305	656.6001	4.18e+19	56.53106	57.10983	56.76372
2	-2338.902	27.50394	4.25e+19	56.54530	57.58708	56.96408
3	-2299.711	66.25226	2.47e+19	55.99312	57.49791	56.59803
4	-2230.886	109.7919	7.10e+18	54.73539	56.70319	55.52643
5	-2195.430	53.18423*	4.56e+18*	54.27214*	56.70296*	55.24931*
6	-2180.226	21.35763	4.79e+18	54.29110	57.18493	55.45440
7	-2167.221	17.03118	5.38e+18	54.36240	57.71924	55.71182
8	-2156.415	13.12133	6.47e+18	54.48607	58.30592	56.02162

The VECM model is built with 4 lags.The speed of adjustment coefficient for the exports as independent variable is negative and statistically significant indicating the system will come to equilibrium although the rest of speed of adjustment is negative but not statistically significant. In the residual diagnostics the error is coming as white noise.The results of residual diagnostics is shown below: As the



model is stable from the results we can go for 4 quarter ahead forecasting.The four quarter ahead forecasting results of VECM is shown below:

Values(in Crores)	2019Q2	2019Q3	2019Q4	2020Q1
Forecasted Values	623415.50	629125.60	642942.40	668077.40
Actual Values	562830.4572	550954.023	563063.8411	542483.2108

Period	S.E.	EXPORTS	GDP	INFLATION	REER
1	19827.16	100.0000	0.000000	0.000000	0.000000
2	25097.50	95.90746	1.685261	0.011616	2.395665
3	30898.93	94.98687	3.149793	0.011192	1.852142
4	35667.24	93.66983	4.765547	0.094996	1.469626
5	41408.95	89.69497	8.292210	0.452252	1.560569
6	45927.01	85.73673	12.09378	0.647548	1.521941
7	50510.41	81.45694	15.85132	0.764417	1.927323
8	54179.39	78.41347	18.75643	0.781024	2.049069
9	58065.76	75.14762	22.04061	0.750631	2.061142
10	61365.15	72.93274	24.29066	0.734868	2.041732

The results from Variance decomposition analysis of exports is shown above.From the above table it can be seen that in short run percentage of forecast error variance explained in exports by exports is very high in proportion and in long run the contribution by GDP increases and forecast error variance explained by other variables is very small even in long run.

3. **LSTM:**

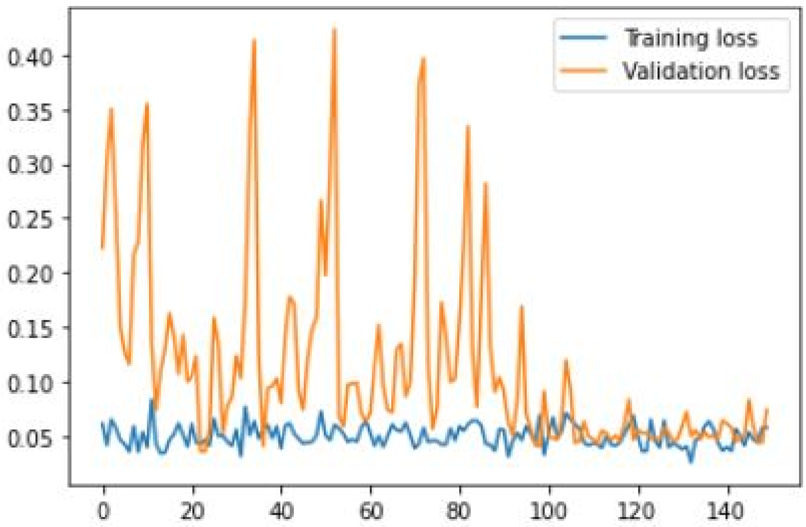
The advantage of neural network in forecasting over classical time series method is we dont need any assumption on the distribution of data. Unlike modeling using regressions, in time series datasets there is a sequence of dependence among the input variables. Recurrent Neural Networks are very powerful in handling the dependency among the input variables. LSTM is a type of Recurrent Neural Network (RNN) that can hold and learn from long sequence of observations. The algorithm developed is a multi-step univariate forecast algorithm. The stacked LSTM model is built with two LSTM layers with 20 and 10 neuron respectively having ReLU activation function and dropout ratio of 0.2 to avoid overfitting.This model doesn't need stationary series or residual diagnostics. Raw Data is feed into the model and after 150 epochs with batch size of 8 the model is trained.

The forecasted output using our Multivariate Data using LSTM is given in the below table:

Values(in Crores)	2019Q2	2019Q3	2019Q4	2020Q1
Forecasted Values	568489.44	568451.13	579357.15	598209.56
Actual Values	562830.4572	550954.023	563063.8411	542483.2108

The training MSE and Validation error is shown in the figure below where we can see the validation error is stable after 120 epochs.

<matplotlib.legend.Legend at 0x7f59faa59090>



Comparision of Models:

The results are shown below in the table:

Model	MAPE	MAE	RMSE
ARIMA	0.124547639	68772.65698	72318.54643
VECM	0.155727212	86057.34198	89352.99015
LSTM	0.043368445	23793.93698	30451.07038

As can be seen from the table above in all the three performance metric Mean Absolute Percentage Error(MAPE), Mean Absolute Error(MAE),Root Mean Square Error(RMSE) LSTM performed better than ARIMA and VECM. Moreover ARIMA performed better than VECM.

VI Conclusion

The outcomes of the undertaken research exhibited efficiencies of the three time series techniques.The forecasting results can be helpful for the government in key policy making to improve exports.According to the outcomes Long Short Term Memory performed better in comparison with Autoregressive Integrated Move Average and Vector error correction Modelling as seen from the values of the performance metric like MAPE,MAE or RMSE. The advantage of extracting non linear effects makes LSTM a better model forecasting as compared to ARIMA or VECM. However ARIMA performed better than VECM. The variance decomposition analysis suggests that in short term forecast the forecast error variance decomposition in exports by exports is large in proportion compared to other variables suggesting ARIMA can be a better fir compared to VECM.

The neural networks produced better predictive accuracy because they are non-linear mapping systems. The neural networks use the time series data to develop an internal representation of the relationship between the variables, and do not make assumptions about the nature of the distribution of the data. Another advantage is that while traditional regression analysis is not adaptive, neural networks readjust their weights as new input data becomes available.

The biggest drawback of the methodology is that artificial neural networks are "black boxes." It is impossible to figure out how relations in their hidden layers are estimated where traditional statistical models are very useful.

In future for better forecasting and in large sample neural networks can always be used to get accurate results for better policy making in any sector like exports,imports, exchange rate etc as due to small small modifications each year neural network is getting superior compared to classical time series models.

VII Limitations

1. This paper has analysed three models with two being parametric models and one non parametric model.LSTM works well with large sample and the data size in this paper is not sufficient for getting a good model using LSTM. If the data size is small statistical models give better results.

2. For better results stacking of models can be done i.e. stacked LSTM-ARIMA models can built for better results.

3. Neural Networks model tend to overfit which might give very less training error but the test error might be large.

4. LSTM like traditional statistical models is getting old.New recent alternative modelling techniques can be used for better results like Mogrifier LSTM, Momentum LSTM and Transformers.

References

[1] Ghassen Chniti, Houda Bakir, and Hédi Zaher. "E-commerce time series forecasting using LSTM neural network and support vector regression". In: *Proceedings of the international conference on big data and Internet of Thing*. 2017, pp. 80–84.

[2] Emmanuel Dave et al. "Forecasting Indonesia exports using a hybrid model ARIMA-LSTM". In: *Procedia Computer Science* 179 (2021), pp. 480–487.

[3] Meftah Elsaraiti and Adel Merabet. "A comparative analysis of the arima and lstm predictive models and their effectiveness for predicting wind speed". In: *Energies* 14.20 (2021), p. 6782.

[4] Steven Elsworth and Stefan Güttel. "Time series forecasting using LSTM networks: A symbolic approach". In: *arXiv preprint arXiv:2003.05672* (2020).

[5] Fuxin Jiang, Gang Xie, and Shouyang Wang. "Forecasting Port Container Throughput with Deep Learning Approach". In: *The 5th International Conference on Computer Science and Application Engineering*. 2021, pp. 1–4.

[6] Harleen Kaur et al. "Predicting Water Availability in Water Bodies under the Influence of Precipitation and Water Management Actions Using VAR/VECM/LSTM". In: *Climate* 9.9 (2021), p. 144.

[7] Diana Mendes et al. "Multivariate forecast for the G7 stock markets: a hybrid VECM-LSTM deep learning model". In: (2021).

[8] Christopher Olah. *LSTM Theory*. `https://colah.github.io/posts/2015-08-Understanding-LSTMs/`. [Online; accessed 01-May-2022].

[9] SUNG HYUN SEO. "Forecasting Korean LNG import price using ARIMAX, VECM, LSTM and hybrid models". In: (2021).

[10] Sima Siami-Namini and Akbar Siami Namin. "Forecasting economics and financial time series: ARIMA vs. LSTM". In: *arXiv preprint arXiv:1803.06386* (2018).

[11] Sima Siami-Namini, Neda Tavakoli, and Akbar Siami Namin. "A comparison of ARIMA and LSTM in forecasting time series". In: *2018 17th IEEE international conference on machine learning and applications (ICMLA)*. IEEE. 2018, pp. 1394–1401.

[12] Yi-Ting Tsai, Yu-Ren Zeng, and Yue-Shan Chang. "Air pollution forecasting using RNN with LSTM". In: *2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech)*. IEEE. 2018, pp. 1074–1079.