UNITED NATIONS MILLENIUM DEVELOPMENT GOALS

Fagun Vasavada (fav150030), Madhulata Hanumantha (mxh151130), Preetha Thoguluva (pxt151830), Shobhika Panda (sxp150031), Yogesh Darji (ypd150030)

Problem Statement

Since its founding in 1944, the World Bank has been gathering data to help it alleviate poverty by focusing on foreign investment, international trade, and capital investment. The data collected is an indicator of the progress of social goals like achieve universal primary education, ensure environmental sustainability, promote gender equality and empower women etc. The data is available for the years 1972 to 2007 on over 1200 macroeconomic indicators in 214 countries around the world. The World Bank provides these data to the public through their data portal in a raw format.

The project focuses only the indicators that are relevant to the Millennium Development Goals listed below.

- Achieve universal primary education
- Ensure environmental sustainability
- Improve maternal health
- Reduce child mortality
- Combat HIV/AIDS
- Promote gender equality and empower women
- Eradicate extreme poverty and hunger
- Combat malaria and other diseases

There are a set of indicators from the World Bank dataset that represent our progress towards these goals. The goal of the project is identify these goals from the dataset, gather the data from the World Bank's online portal, come up with an approach to aggregate data and handle missing data, carefully collect and rigorously analyze the past observations of the time series from the World Bank to develop an appropriate model which describes the inherent structure of the series, predict future values for the series using this model. and predict the value of the indicators for the years 2008 to 2012. This would enable the United Nations to foresee the indicator values and take measures to ensure better progress.

Technical Solution

Understanding Time Series Data

A time series is a sequential set of data points, measured typically over successive times. It is mathematically defined as a set of vectors x(t), t = 0,1,2,... where t represents the time elapsed. The variable x(t) is treated as a random variable. The measurements taken during an event in a time series are arranged in a proper chronological order.

A time series containing records of a single variable is termed as univariate. But if records of more than one variable are considered, it is termed as multivariate.

Since our data is value of the World Bank indicators collected over time, our data is univariate time series data and discrete since it is observed at discrete points of time unlike continuous time series data where data is observed continuously.

A time series in general is supposed to be affected by four main components, which can be separated from the observed data. These components are: Trend, Cyclical, Seasonal and Irregular components.

The general tendency of a time series to increase, decrease or stagnate over a long period of time is termed as Secular Trend or simply **Trend.**

Seasonal variations in a time series are fluctuations within a year during the season. The important factors causing seasonal variations are: climate and weather conditions, customs, traditional habits, etc.

Since we are dealing with data dealing with social goals, we analyzed and found that there no seasonal variations in the data. We could only find a trend.

Analyzing the World Bank Data

For each of the development goals, the World Bank had a list of statistics, for example, percentage of girls

attending primary education, percentage of boys attending primary education represented the data towards the goal to achieve universal primary education. We identified these statistics and aggregated them for each year in the time series data for better representation of the trend in the data.

Model Parsimony

While building a proper time series model we have to consider the principle of parsimony. According to this principle, always the model with smallest possible number of parameters is to be selected so as to provide an adequate representation of the underlying time series data.

Following this principle, we chose the data where it was not very sparse. For the missing data, we filled values of the series mean.

Fitting a Model

A successful time series forecasting depends on an appropriate model fitting. A lot of efforts have been done by researchers over many years for the development of efficient models to improve the forecasting accuracy. We studied these models and identified a couple of them for our usecase.

Autoregressive Integrated Moving Average (ARIMA)

The basic assumption made to implement this model is that the considered time series is linear and follows a particular known statistical distribution, such as the normal distribution. The popularity of the ARIMA model is mainly due to its flexibility to represent several varieties of time series with simplicity. **[6, 21, 23]**

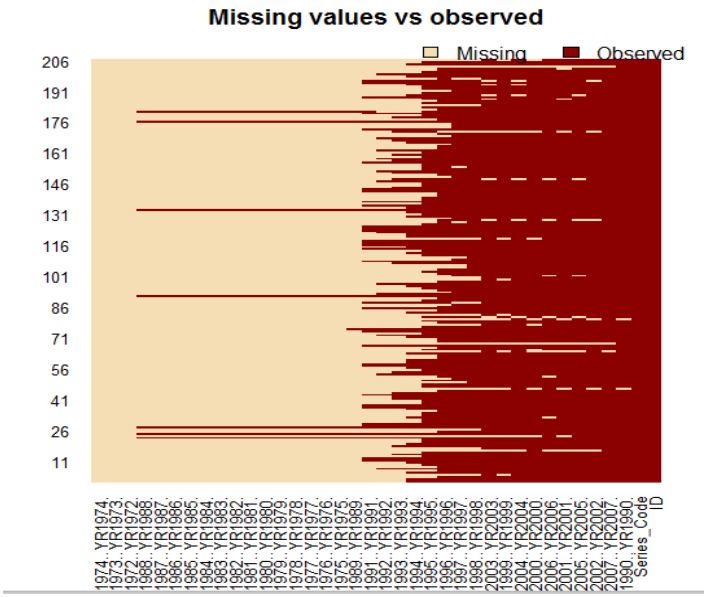
However, the severe limitation of these models is the pre-assumed linear form of the associated time series which becomes inadequate in many practical situations.

Artificial neural networks (ANNs)

These models have attracted increasing attentions in the domain of time series forecasting. The excellent feature of ANNs, when applied to time series forecasting problems is their inherent capability of non-linear modeling, without any presumption about the statistical distribution followed by the observations. The appropriate model is adaptively formed based on the given data. Due to this reason, ANNs are data-driven and self-adaptive by nature.

Implementation

The time-series data was made available was in a raw format. The first step in the process was to get a clean workable dataset, to feed to the prediction model. We analyzed data by plotting the time series data & analyze the health of the data. Identifying the important features vector was crucial part of the process. Thus, getting a semi-structured data enabled us to gain better efficiency while running the prediction models, with low RMSE values (Feature Vectors: Aforementioned Millennium Development Goals).



When fitting a model to a set of time series data, the following procedure provides a useful general approach:

1. Plot the data. Identify any unusual observations.
2. Transform the data to stabilize the variance. Handle NULL/NA values to reduce distortions in the final output.
3. Generate stable data by selecting the data set with maximum information. Need be replace the few N/A values with row means.

4. Generate the appropriate dataset in the format expected by the model to be used(e.g. handle special datatypes like Dates etc..).
5. Try the chosen model(s), and use training dataset.
6. Forecast the values for the requisite time-period & test dataset.
7. Plot these values against the original data set to visually see how the prediction models performed & to validate the results.

After analyzing various platforms to run or prediction models, we chose R platform to implement our models. R provides a large library of regression & prediction models to work with, which allowed us to evaluate our hypothesis on multiple high credibility learning models.

Following are the ***models*** we leveraged to present out findings:

ARIMA :

- In statistics and econometrics, and in time series analysis, an autoregressive integrated moving average (ARIMA) model is a generalization of an autoregressive moving average (ARMA) model. These models are fitted to time series data either to better understand the data or to predict future points in the series (i.e. forecasting).
- The auto.arima() function in R uses a variation of the Hyndman and Khandakar algorithm which combines unit root tests, minimization of the maximum-likelyhood algorithm to implement ARIMA mode.

Gradient Boosting Model :

- XGBoost is short for "Extreme Gradient Boosting". XGBoost is used for supervised learning problems, where we use the training data (with multiple features) $\mathbf{X}_i$ o predict a target variable $\mathbf{y}_i$.
- The *forecastxgb* package in R aims to provide time series modelling and forecasting functions that combine the machine learning approach of *xgboost* with the convenient handling of time series data.

ETS Model:

- Exponential Smoothing is a technique to make forecasts by using a weighted mean of past values, wherein more recent values are given higher weights
- While running the models on the huge dataset, we realized that some of the models were not preforming very well due to the nature of data being non-stationary. The ETS models with seasonality or non-damped trend help to account for such variations. Thus for seasonal nature of datasets, there are a large number of restrictions on the ARIMA parameters which can be relaxed by using ETS model with the forcast() function in R.

Neural Networks Model for TS:

- These models have attracted increasing attentions in the domain of time series forecasting. The excellent feature of ANNs, when applied to time series forecasting problems is their inherent capability of non-linear modeling, without any presumption about the statistical distribution followed by the observations. The appropriate model is adaptively formed based on the given data. Due to this reason, ANNs are data-driven and self-adaptive by nature. We have tried to run these cross-validation models & presented the RMSE achieved by this model.

Finally, we tried to represent the results of the models as graphs, which allowed us to view the progression of the forecast with respect to the historic data collected.

To analyze the effectiveness of our work, we generated RMSE values for each of the selected feature vectors & compared it with the available data. We presented these findings by ploting the results in R platform.

Conclusion

After several iterations of optimizing the models and re-structuring dataset, we present our findings and analysis of the various models that we implemented in the next section. We have validated the predictions by calculating the RMSE plots for each Indicator, with the goal to measure the accuracy.

2.1 – Model1 - Algeria

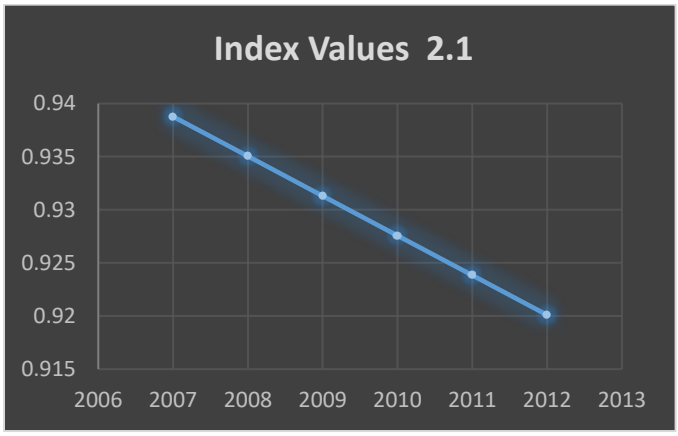


Figure 1: ARIMA regression for 2.1

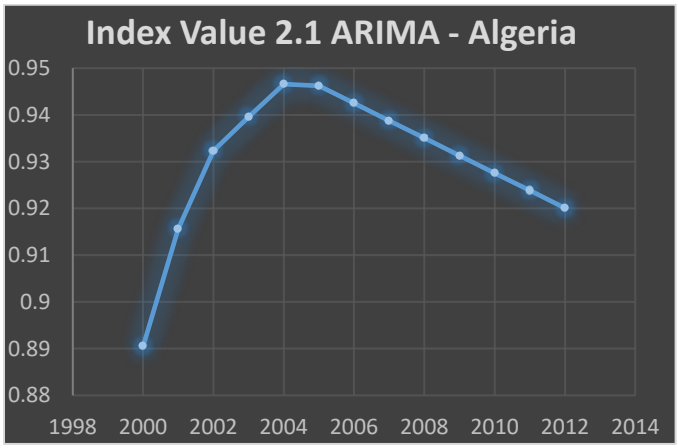


Figure 2: ARIMA for 2.1, Algeria

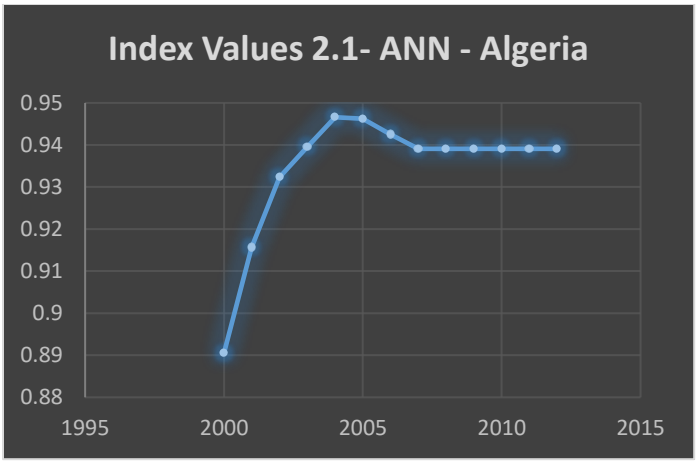


Figure 3: ANN for 2.1, Algeria

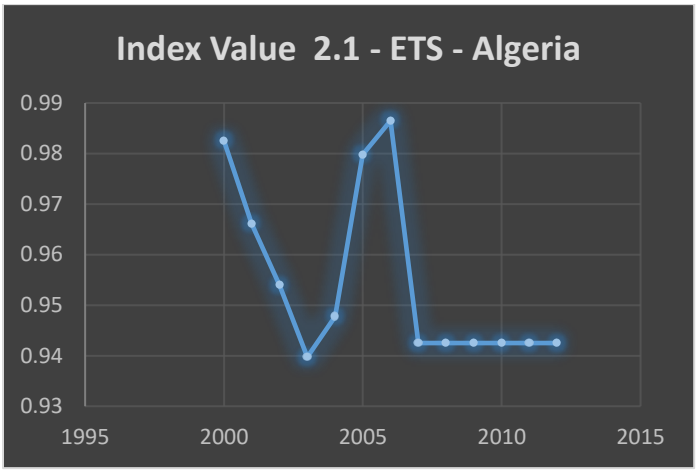


Figure 4: ETS for 2.1, Algeria

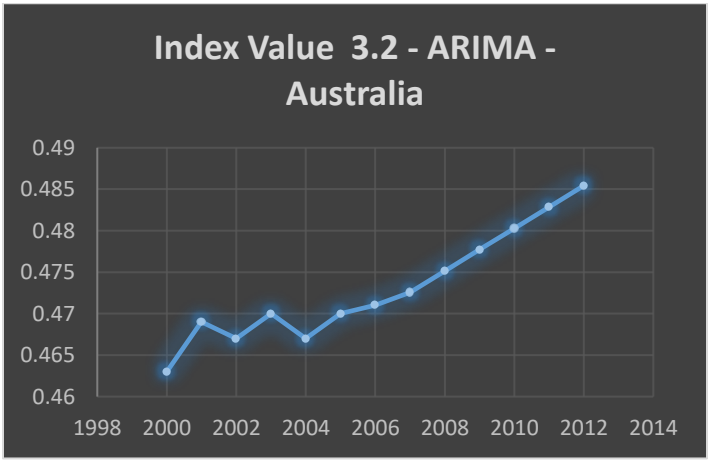


Figure 5: ARIMA for 3.2, Australia

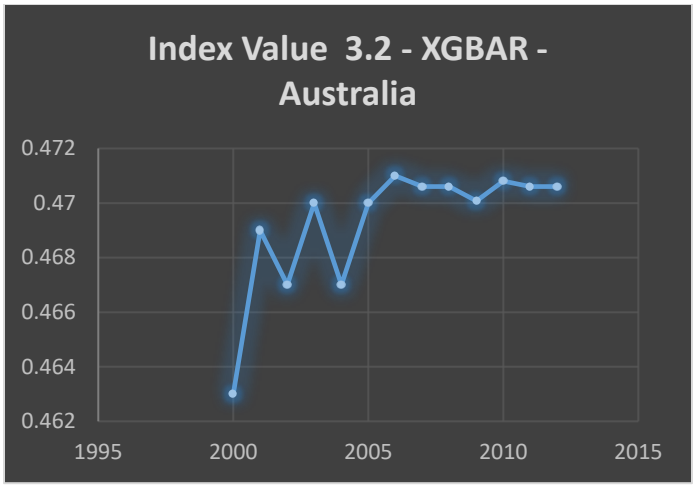


Figure 6: XGBAR for 3.2, Australia

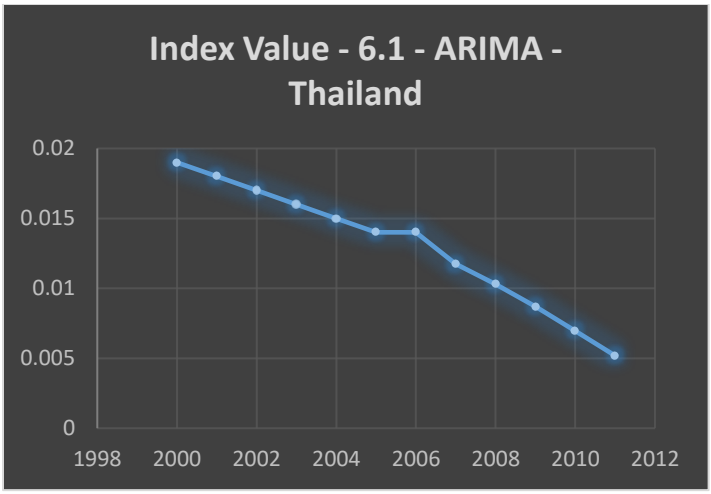


Figure 9: ARIMA for 6.1, Thailand

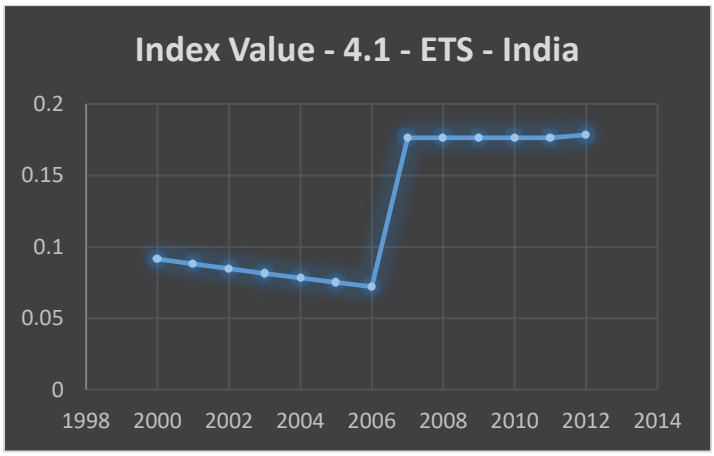


Figure 7: ETS for 4.1, India

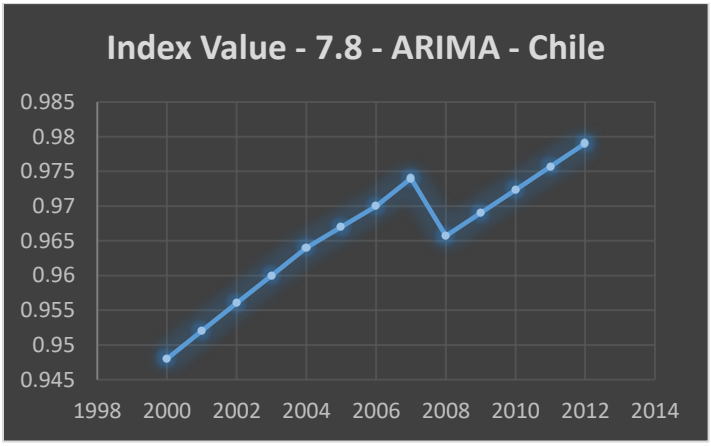


Figure 10: ARIMA for 7.8, Chile

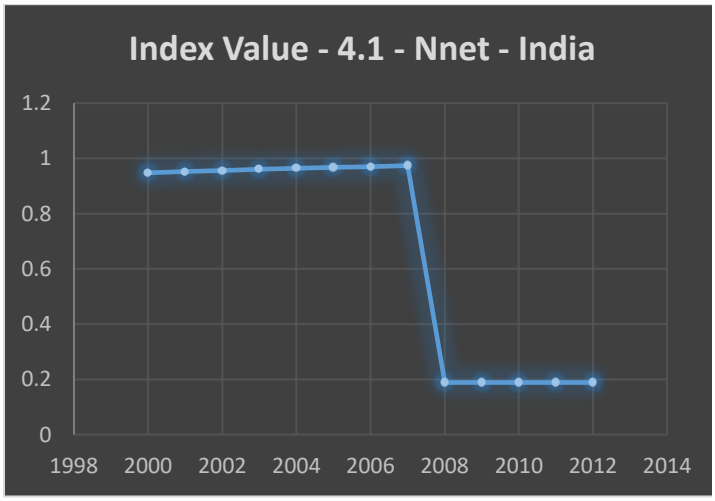


Figure 8: NNet for 4.1, India

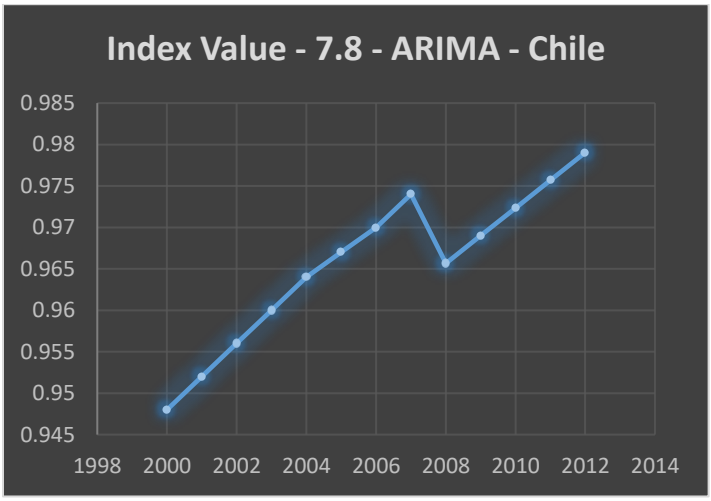


Figure 11: ARIMA for 7.8, Chile

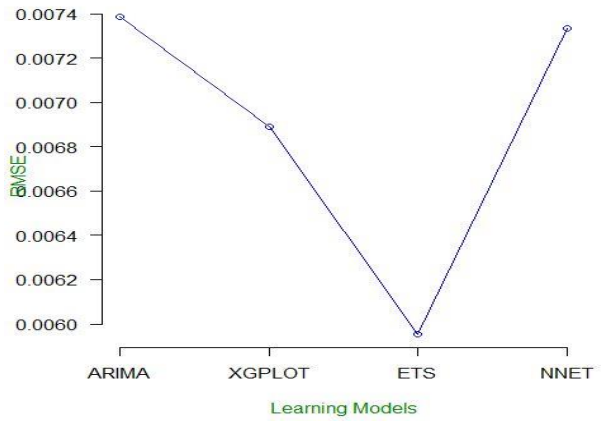


Figure 12: RMSE values for various models – Indicator 2.1

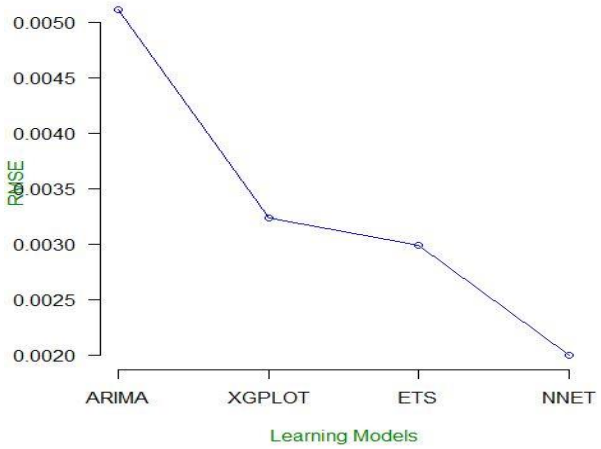


Figure 13: RMSE values for indicator, 4.1

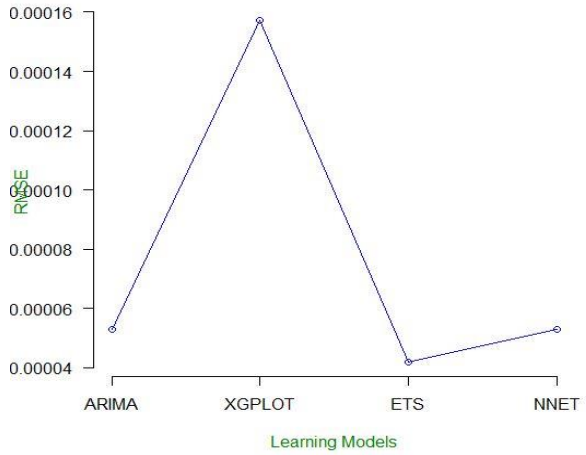


Figure 14: RMSE values for indicator, 5.1

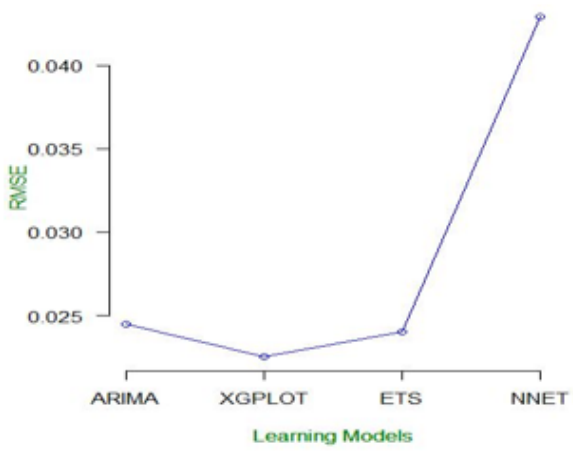


Figure 15: RMSE values for indicator, 3.2

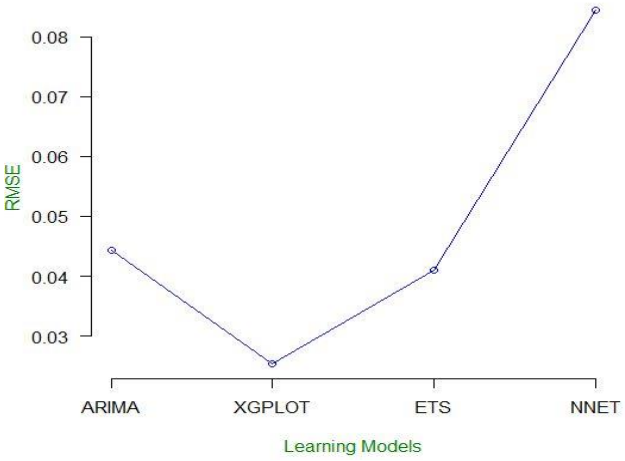


Figure 16: RMSE values for indicator, 8.16

We could predict the indicator values for years 2008 through 2012 with minimal RMSE. We compared the performance of each regression model. We observed that the model performance varied for each indicator based on the data quality.