# TSA Final Project

Shubhangi Gupta, Aditi Jackson, David Robinson

2024-03-29

## Introduction

### Motivation, Relevance, Objectives

As one of the world's fastest growing emerging economies, India's demand for electricity is increasing every year - fueled by newly electrified homes, growing industry, urbanization, rising income levels, a higher demand for cooling and in a low carbon future - the electrification of end use sectors like industry and transport. To address this demand, the country plans to almost double its power capacity to 900 GW by 2030, up from 427 GW today. Source. At the same time, as a signatory of the Paris Agreement, India has committed to reducing its national emissions to net zero by 2070, with an intermediate target of transitioning 50% of its electric generation capacity to clean sources by 2030 as part of its NDC to the UNFCCC Source. In line with this growing demand for electricity and simultaneous need to decarbonize, India has announced a complementary target of achieving 500 GW of renewable energy capacity by 2030. Source. However, to ensure that this new clean energy translates into emissions reductions, integrating such high levels of variable renewable energy (VRE) into the electric system requires a concurrent expansion and modernisation of the grid, so that issues around connecting new RE capacity to load centres, power flow management and congestion, and managing higher load volumes do not impede the clean energy transition.

With this context in mind, in this study, we aim to: Explore how India's transmission capacity has changed over the last several years and thus forecast it based on historical trends. This would allow us to identify what capacity it will reach in 2030 in a "business as usual" scenario. Compare our finding to what is needed to integrate the additional 500 GW of RE into the grid as assessed in the literature.

### Dataset information and methods

For this study, we used data on the line length (in ckm) of transmission lines installed in India, taken from the "India Climate and Energy Dashboard (ICED)" developed by the Government of India's inhouse think tank called the NITI Aayog. The data is monthly and extends from April 2015 to January 2024 and represents the length of new transmission lines (in '00 kms) added across the country in each month during this timeframe. While line length does not reflect voltage levels and different types of transmission and distribution, the portal from which we acquired the data clearly states that this length refers to transmission only. We also explored the additions to transmission capacity by voltage and found no significant change in trend - rendering line length a simple yet effective metric to explore how transmission has expanded in India over the last decade. Source.

The original dataset included two columns of interest - additional line length (ckm) in each month, and month/year of completion. In order to create a time series dataset of total line length in each month, we wrangled it using the following process:

Stage 1: Wrangling and methods Importing the dataset, subsetting it to only retain these two columns of interest, and renaming them to simpler names. Checking if there were any NAs (there weren't). Splitting the "month/ year of completion column" that was a string into the month and year separately, converting the year from two digits to four digits (ex: "15" to "2015"), pasting that back with the month column separated by a "-" and then using lubridate to convert it into a date object. Using the group_by(Date) function to add

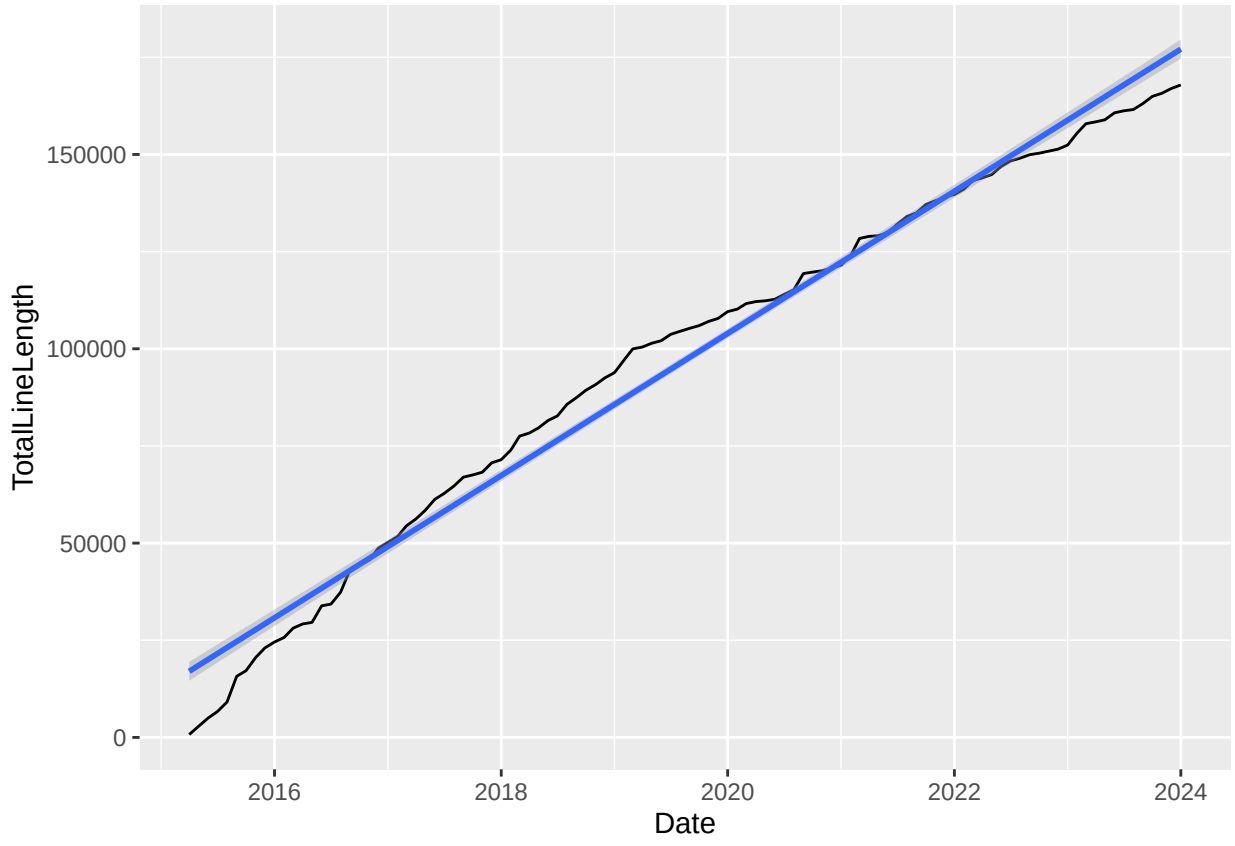
up all the capacity additions in the same month - that in the original dataset were broken up across multiple rows based on the regions of India that they occurred in (we only consider total nation-wide capacity and do not look at this data's breakup across states/ regions). Using the cumsum() function to sum the line length of the previous month's total capacity (calculated) to the current month's capacity addition (original data). This gave us a dataset with two columns: Date and total transmission line length (ckm) until that month. Plotting these two columns along with an lm line to check the trend. This concluded the first stage of wrangling the data to acquire our final dataset to be used for the analysis.

Stage 2: Initial Exploration Converting the data into a time series object. Plotting the ACF and PACF. Decomposing the time series object (multiplicative) Running an SMK and ADF test.

Stage 3: Fitting the model and forecasting to training data Breaking up the dataset into training and testing: Training: April 2015 to March 2023 Testing: April 2023 - January 2024 (these follow India's financial year cycle of April-March) Fitting the models: For ARIMA, identifying the best ARIMA model using auto.arima. Besides that, we fit the SARIMA, TBATS and Neural Network. Fitting the models mentioned in the previous step to the training data, and using the summary() and checkresiduals() functions to check the result. Forecasting the fitted model to the next one year (testing data) using the forecast function. Plotting the result along with the original data using autoplot and autolayer, as well as using the accuracy() function to explore the goodness of fit and forecast of the model to the data.

# Data Wrangling

```
## [1] TRUE
```




# Data Structure

Table 1: Data Structure Summary - NITI Aayog

| Detail | Description |
|---|---|
| Data Source | India Climate and Energy Dashboard |
| Retrieved from | https://iced.niti.gov.in/energy/electricity/transmission/transmission-lines |
| Variables Used | Line Length (cKM), Month of Completion |
| Data Range | 2015 - 2023 |

# Analysis

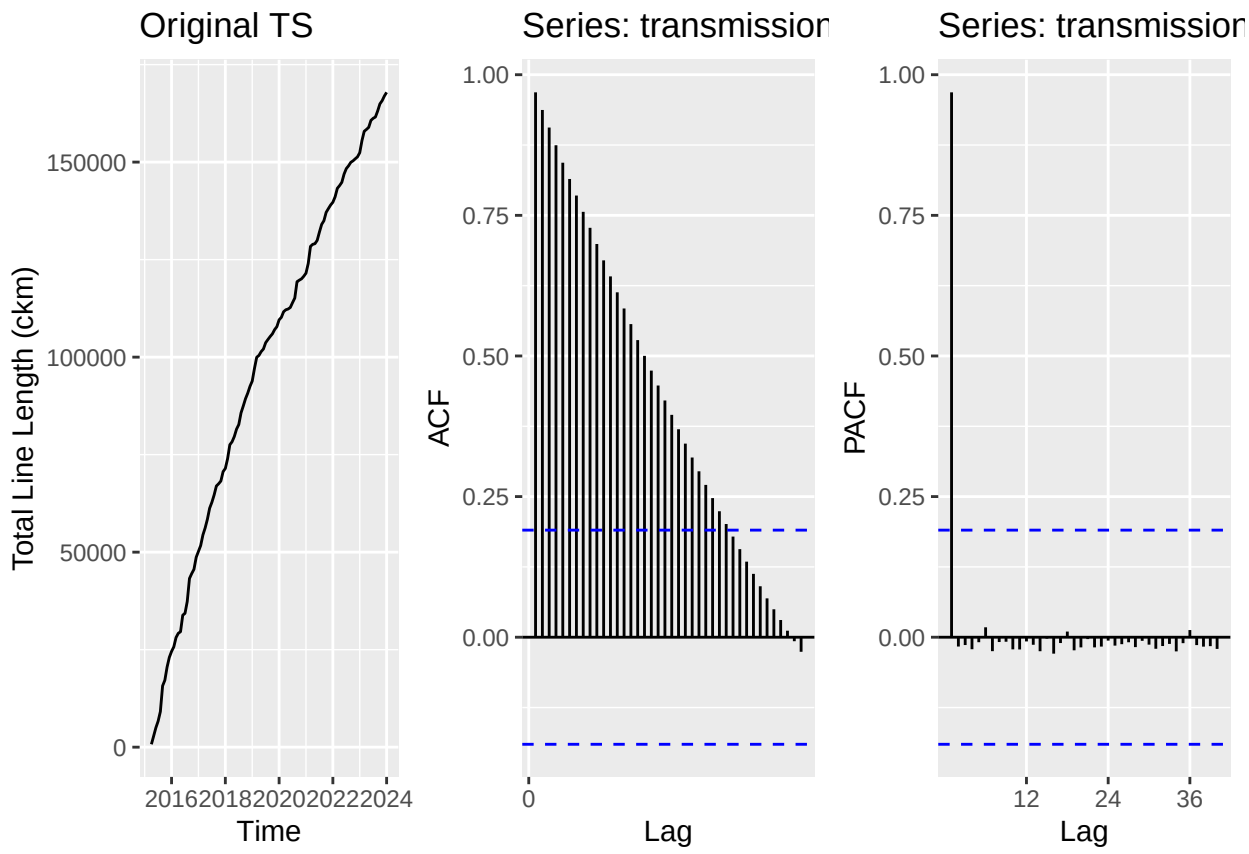
## Background on Analysis

Originally, the data included columns for date and annual line additions in ckm per month. However, plotting annual line additions over time revealed a strong declining trend. This led us to explore whether the data represented only transmission expansion or also included distribution expansion. While the original data source (Government of India) states that the data is transmission-level only, questions remain about the granularity of the data as voltage levels for the transmission lines were not included in the data or otherwise made easily accessible. We further explored the voltage levels of the data to determine if the declining trend could be attributed to the addition of higher efficiency voltage lines but found, based on research from Indian think-tank Prayas, that this is not the case.

Given this process, we came to the conclusion that forecasting annual additions might be misleading given our research question, which was to forecast transmission growth in India. Upon reflection, we decided to add a column to our dataset to account for cumulative line additions. This allowed us to represent how total transmission capacity has grown over time based on line length (ckm). Plotting cumulative line capacity over time showed a clear increasing trend, which started to plateau after 2019.

## Plots

Initial analysis included creating a time series of Total Line Length (ckm), plotting the series, and generating its ACF and PACF. The time series plot shows a strong, positive trend. The ACF plot decays exponentially, and the PACF plot is only significant at the first lag. Taken together, the ACF and PACF plots suggest that the series follows an autoregressive process.

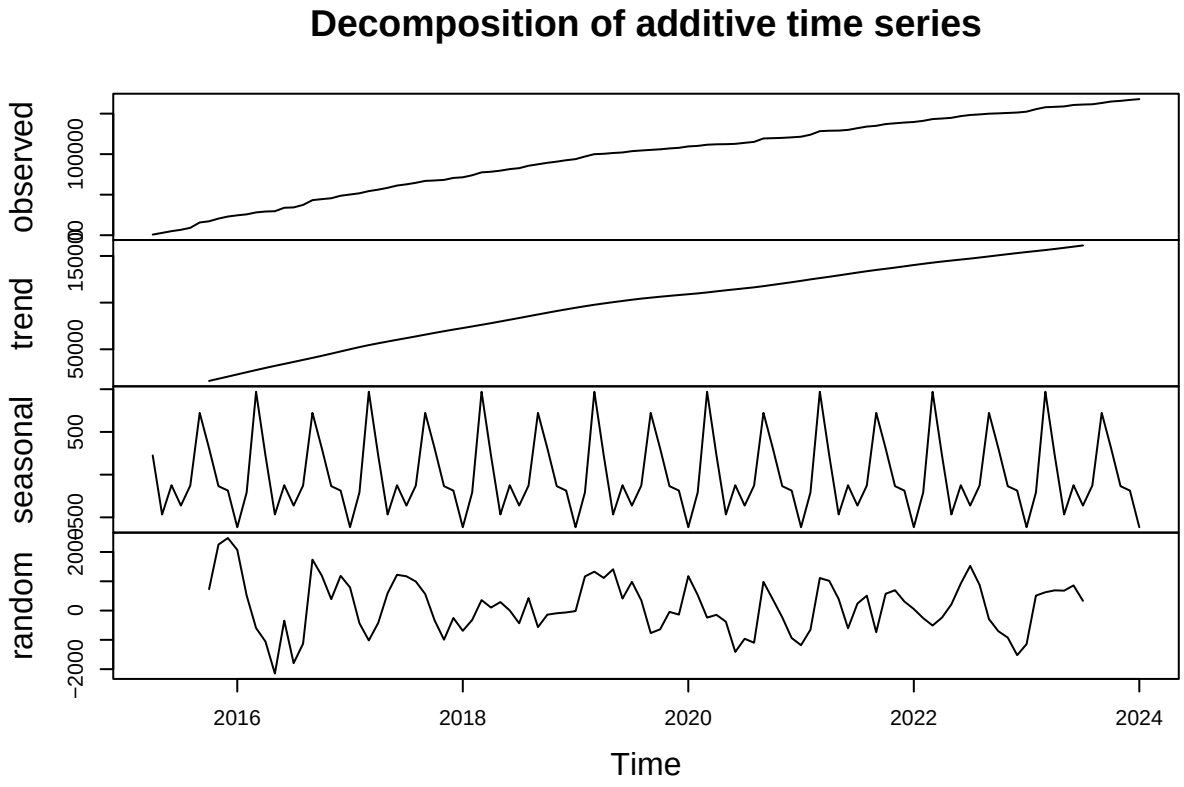



## Decomposition

The series was then decomposed into observed, trend, seasonality, and residuals. The observed and trend components both increase similarly over time. However, the seasonal component is not representative of the dataset. Upon further inspection, we observed that the seasonality was too uniform in magnitude and may not be part of the dataset. While there may be some seasonality due to construction start and end dates for transmission projects (e.g. before and after Monsoon Season), it is more likely that the decompose function in R is forcing a seasonal component that does not exist within the data. As such, we did not de-season the data and instead chose to fit models that could handle seasonality (see below). The residuals looked fairly random, so we did not see the need for further manipulation before fitting models.

## Statistical Tests

To determine whether or not the series was stationary (i.e. if its statistical properties like mean and variance do not change over time), we employed the Seasonal Mann-Kendall (SMK) and Augmented Dickey-Fuller (ADF) tests. The SMK test produced a positive test statistic (Tau = 1) and a significant p-value (2-sided p-value =< 2.22e-16) at the 95% confidence level. This indicates non-stationarity and a positive deterministic trend. The ADF test produced a test a negative test statistic (Dickey-Fuller = -3.2942) and an insignificant p-value (p = 0.07586) at a 95% confidence level. Thus we do not have enough evidence to reject the null hypothesis that the series has a unit root (i.e. non-stationarity). Instead, we would lean towards accepting the alternative hypothesis that the time series in question is stationary (i.e. does not possess a unit root).




```
## NULL
```

```
## Score =  416 , Var(Score) = 1050.667
## denominator =  416
## tau = 1, 2-sided pvalue =< 2.22e-16
## NULL
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  transmission_ts
## Dickey-Fuller = -3.2942, Lag order = 4, p-value = 0.07586
## alternative hypothesis: stationary
```

### Training and Testing Data

In order to check model performance later on, we split the cleaned transmission data into training and testing data. Training data spans April 2015 to March 2023 while the testing data spans April 2023 to January 2024. We then created time series objects for both.

### Mode Fitting & Forecasting

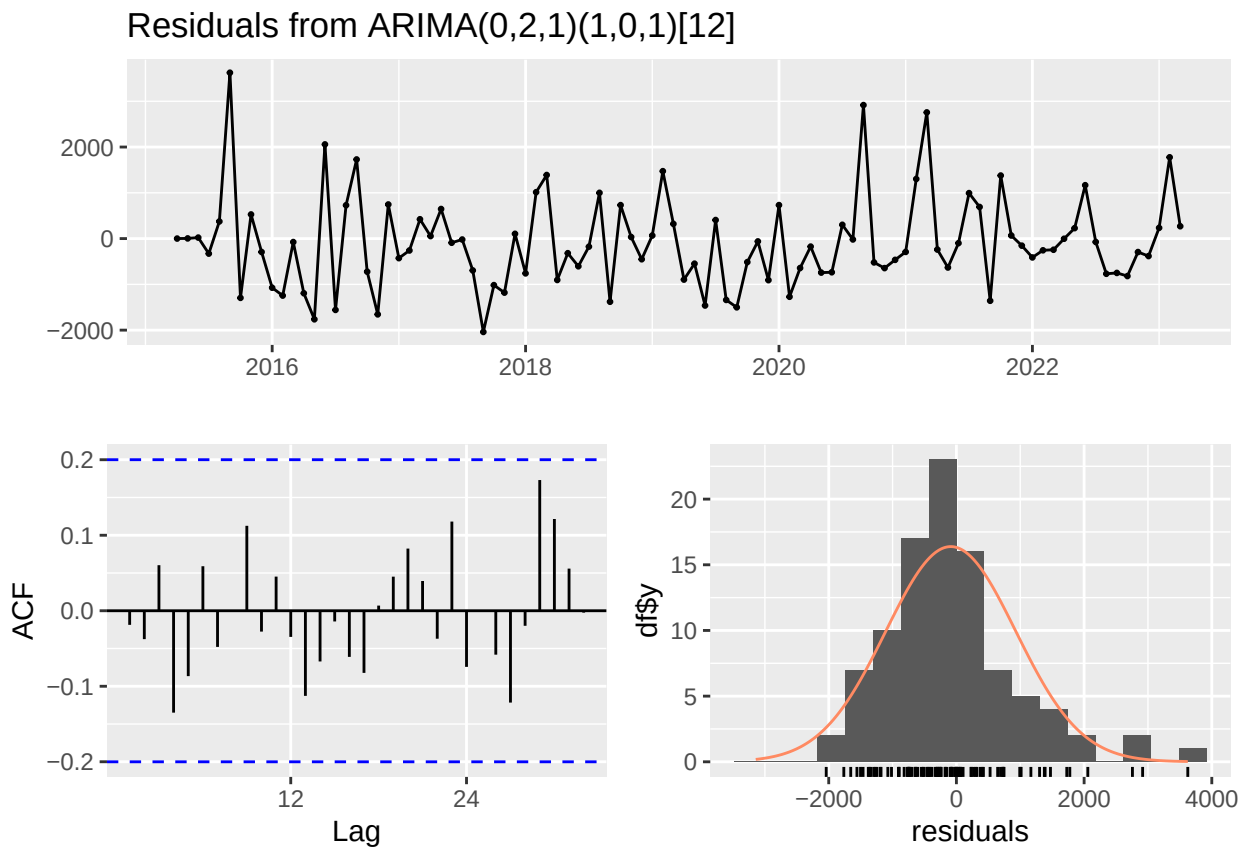
We fit three models to our transmission time series data: Seasonal ARIMA (SARIMA), TBATS, and Neural Network.

**SARIMA - Model and Forecast**

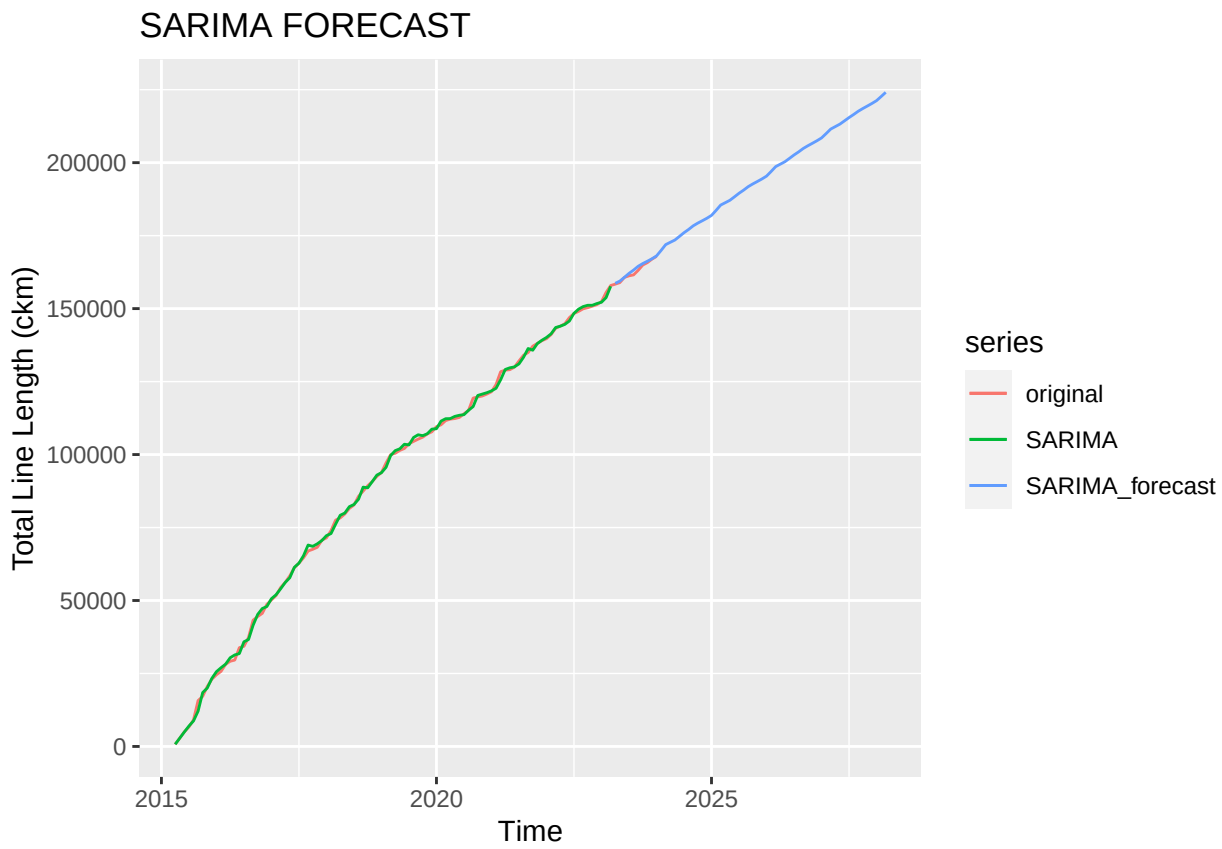
We chose to start with the SARIMA model given its relative simplicity and ability to handle seasonality. To fit the SARIMA model, we used the auto.arima() function to generate the order of parameters. The function produced the following: p = 0, d = 2, q= 1, P = 1, D = 0, Q = 1. The non-seasonal parameters suggest no autoregressive component (p=0), 2 degrees of differencing (d=2), and a moving average component (q=1). The seasonal part of the model suggests that there is some seasonal autoregression (P=1), no differencing, and some seasonal moving average (Q=1). After fitting the SARIMA model, we used the forecast function to create a 6-month forecast and compared the values with those in our testing data. The model appears to fit well based on visual inspection. An accuracy table is provided below as well as a discussion of the results. We then created a 6-year forecast to ascertain a value for transmission build-out by 2030.

```
## Series: transmission_ts_training
## ARIMA(0,2,1)(1,0,1)[12]
##
## Coefficients:
##           ma1    sar1     sma1
##       -0.9451  0.8338  -0.5295
## s.e.   0.0320  0.1292   0.2064
##
## sigma^2 = 1090869:  log likelihood = -788.27
## AIC=1584.53   AICc=1584.98   BIC=1594.7
```

```
## Series: transmission_ts_training
## ARIMA(0,2,1)(1,0,1)[12]
##
## Coefficients:
##           ma1    sar1     sma1
##       -0.9451  0.8338  -0.5295
## s.e.   0.0320  0.1292   0.2064
##
## sigma^2 = 1090869:  log likelihood = -788.27
## AIC=1584.53   AICc=1584.98   BIC=1594.7
##
## Training set error measures:
##                      ME     RMSE      MAE         MPE     MAPE       MASE
## Training set -88.36536 1016.884 760.9757 -0.09538527 1.445726 0.03940281
##                      ACF1
## Training set -0.01866278
```



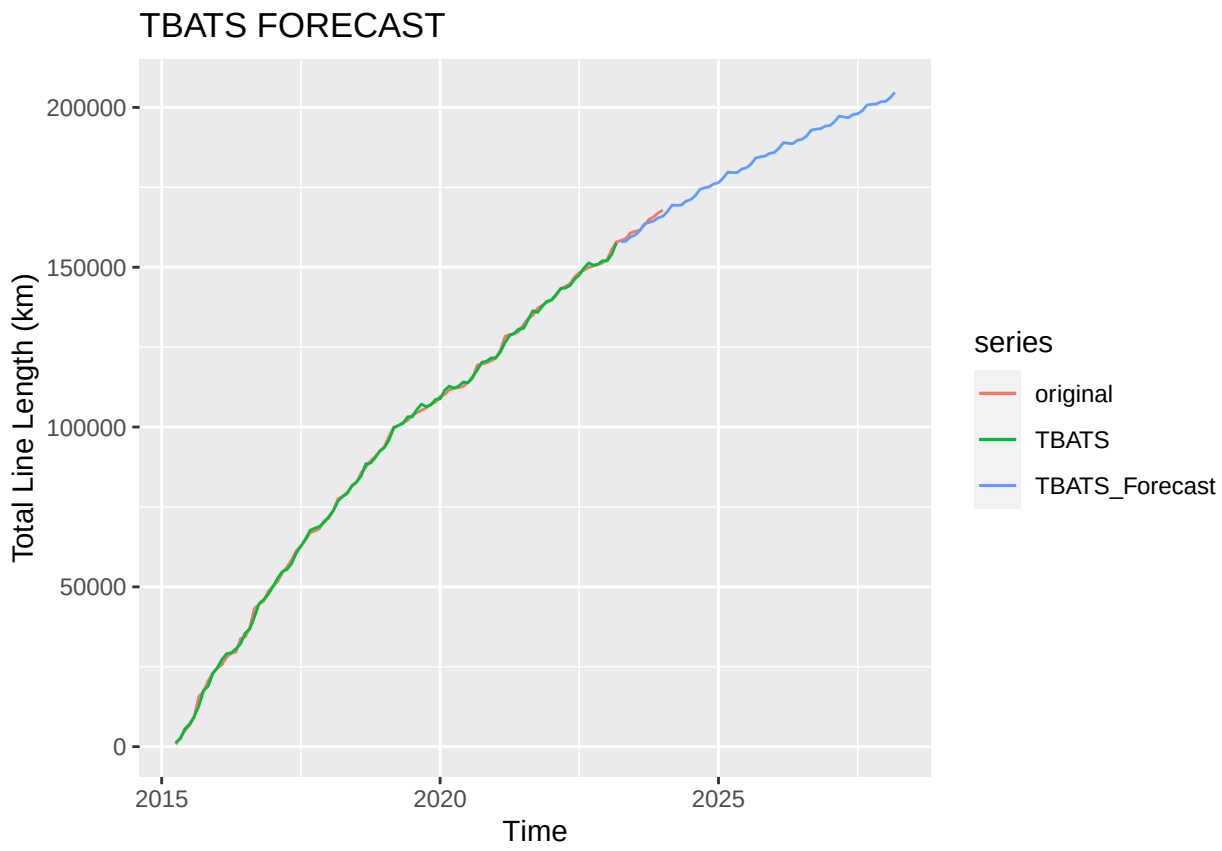

```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(0,2,1)(1,0,1)[12]
## Q* = 9.0798, df = 16, p-value = 0.9101
##
## Model df: 3.   Total lags used: 19
```




```
##                       ME     RMSE      MAE         MPE     MAPE       MASE
## Training set -88.36536 1016.884 760.9757 -0.09538527 1.445726 0.03940281
##                       ACF1
## Training set -0.01866278
```

**TBATS - Model and Forecast**

TBATS was the next model we fit to our time series. Given the uncertainty around seasonality in our data, we opted for TBATS since the model can handle seasonal variation. We fit the model using the tbats() function from the forecast package. We then used the TBATS model to generate a forecast and compared the forecast to our original data. Again, we forecasted 6 years of values in order to obtain a value for transmission build-out by 2030. Upon inspection, the forecast compares favorably with the test values from our original data.



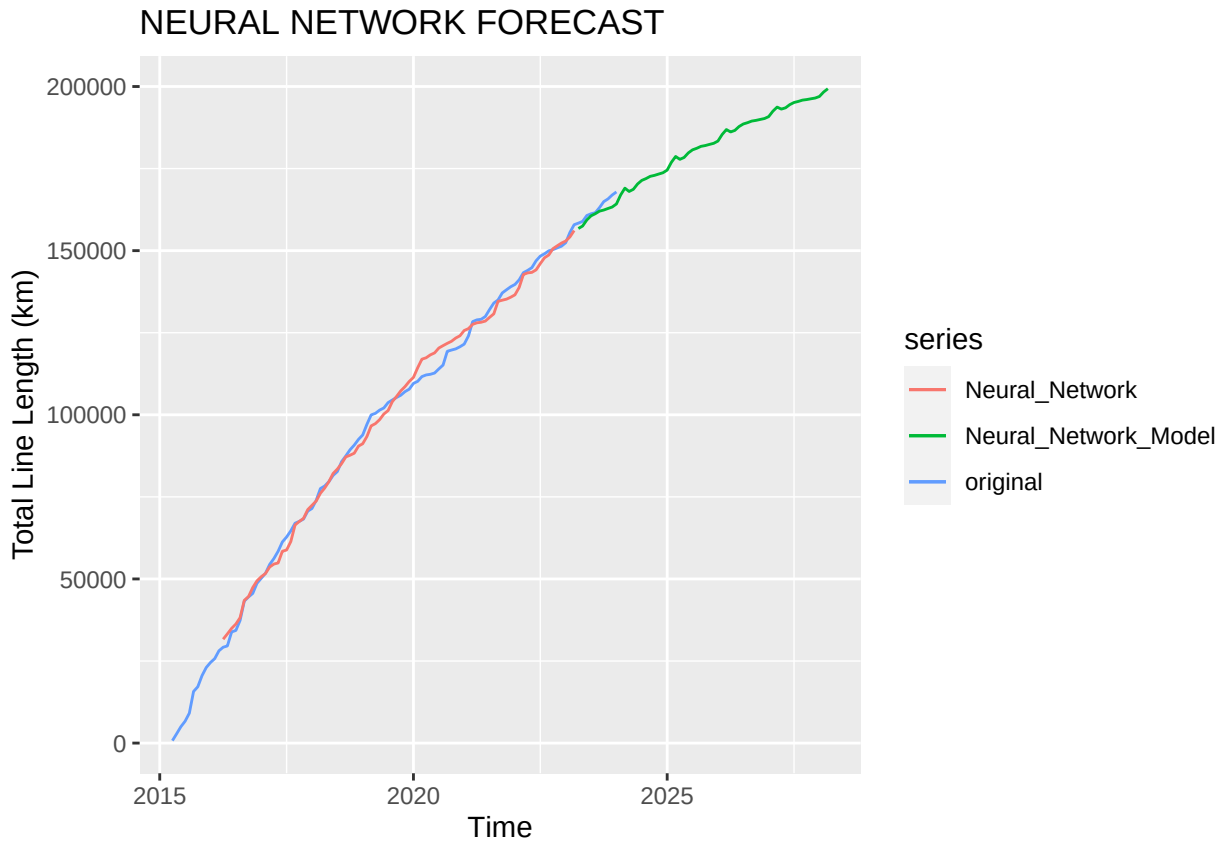

```
##                     ME     RMSE     MAE        MPE     MAPE       MASE
## Training set 13.7543 887.3142 685.352 -0.6353815 2.303935 0.03548707
##                      ACF1
## Training set -0.00187256
```

**Neural Network - Model and Forecast**

The last model we fit was a neural network since they are able to capture complex patterns in time series data. We used the function nnetar() from package 'forecast' with p = 1 and P = 1 (taken from the parameters of our SARIMA model). Similar to our workflow with SARIMA and TBATS, we created a forecast with the Neural Network model and compared the values to our original data. Then, we created a 6-year forecast to obtain a value for transmission build-out by 2030. Upon inspection, the forecast compares favorably with the test values from our original data.

Table 2: Forecast Accuracy Table

| | ME | RMSE | MAE | MPE | MAPE | MASE | ACF1 |
|---|---|---|---|---|---|---|---|
| SARIMA | -88.36536 | 1016.8839 | 760.9757 | -0.09539 | 1.44573 | 0.03940 | -0.01866 |
| TBATS | 13.75430 | 887.3142 | 685.3520 | -0.63538 | 2.30393 | 0.03549 | -0.00187 |
| NN | 8.32924 | 2520.7498 | 1986.4633 | -0.18504 | 2.21727 | 0.10286 | 0.88787 |




```
##                     ME    RMSE      MAE        MPE    MAPE      MASE      ACF1
## Training set 8.329245 2520.75 1986.463 -0.1850448 2.21727 0.1028577 0.8878727
```

### Checking Accuracy

Across the three models and forecasts that were run, we evaluated the accuracy of each model to understand the best fit. By Root Mean Squared Error or RMSE equal to 887, the best model is TBATS. By Mean Absolute Percentage Error or MAPE equal to 1.44, the best model is SARIMA. Additionally, we added a table to show various accuracy metrics in detail across SARIMA, TBATS, and NN models for detailed, quantitative comparison across a variety of accuracy measures.

```
## The best model by RMSE is: TBATS
```

```
## The best model by MAPE is: SARIMA
```

## Summary and Conclusions

Regarding forecasting, our primary research question was "How does this compare to the transmission capacity needed in a net zero-aligned renewable-heavy future?"

In seeking a reference point from the literature by which to measure the success of our model, we found that in order to achieve India's goal of integrating 500 GW of renewable energy by 2030, 50,890 Ckm of transmission line length would need to be built. Source.

India currently has 487,367 Ckm of transmission line length. Source.

Adding these estimates of projected transmission line build-out and current transmission line length, the literature suggests that cumulative transmission line capacity would need to reach 529,257 Ckm by 2030 to achieve its decarbonization goals.

From our SARIMA forecast, the projected value for January of 2030 is 246,096 Ckm. Thus, our output underestimates the estimate from the literature source we identified by approximately one-half.