April 22, 2022

Report on
Prediction of Future Energy Production

Spring 2022

AGNI SIDDHANTA
ABHISHEK BHAGAT
KENNEDY CRANDELL
KUNAL JHA

Objective

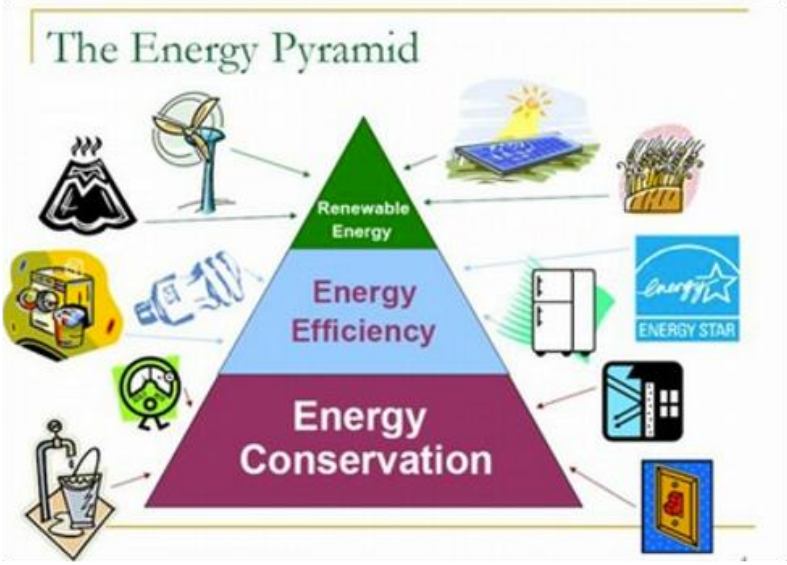
Predicting future energy production to ensure the provision of adequate, secure and cost-effective energy supply.

Introduction

With modernization in technology, the need for electric and gas utilities has been on the rise resulting in an increase in their production. Therefore, energy production of a country is regarded as an important indicator of economic development.

Problem Statement

Accurately forecasting energy demand among consumers is a challenge of its own. The match in demand and supply of energy consumption is crucial because overestimating leads to exploitation of resources and incurs additional cost and underestimating will lead to power outage. Clearly, there are tangible benefits in closely monitoring the energy production and consumption.



How has the production from the gas and utilities increased with increasing demands over the years?

What is the forecasted production for the electric and gas industry in the near future?

Using the IP index data of electric and gas utility production facilities in the United States, we will answer the above questions.

Data Description

This data covers the Industrial production index for gas and electric utilities in the United States.

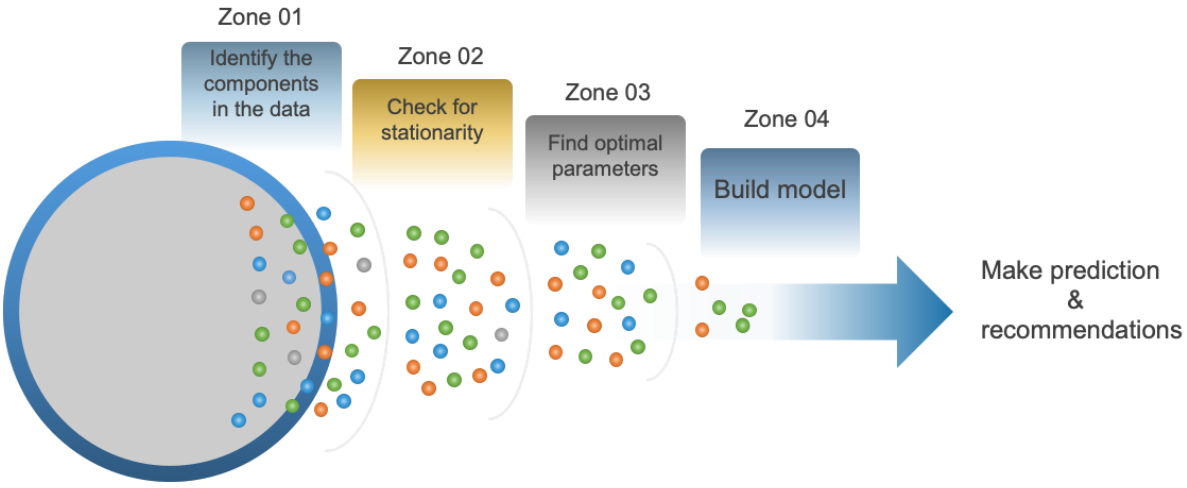
Our data consists of 2 features which include Date ranging from 1985 to 2017, and IPG2211A2N, which is the measure of energy production for gas and utilities.

	A	B
1	DATE	IPG2211A2N
2	1/1/1985	72.5052
3	2/1/1985	70.672
4	3/1/1985	62.4502
5	4/1/1985	57.4714
6	5/1/1985	55.3151
7	6/1/1985	58.0904
8	7/1/1985	62.6202
9	8/1/1985	63.2485
10	9/1/1985	60.5846
11	10/1/1985	56.3154

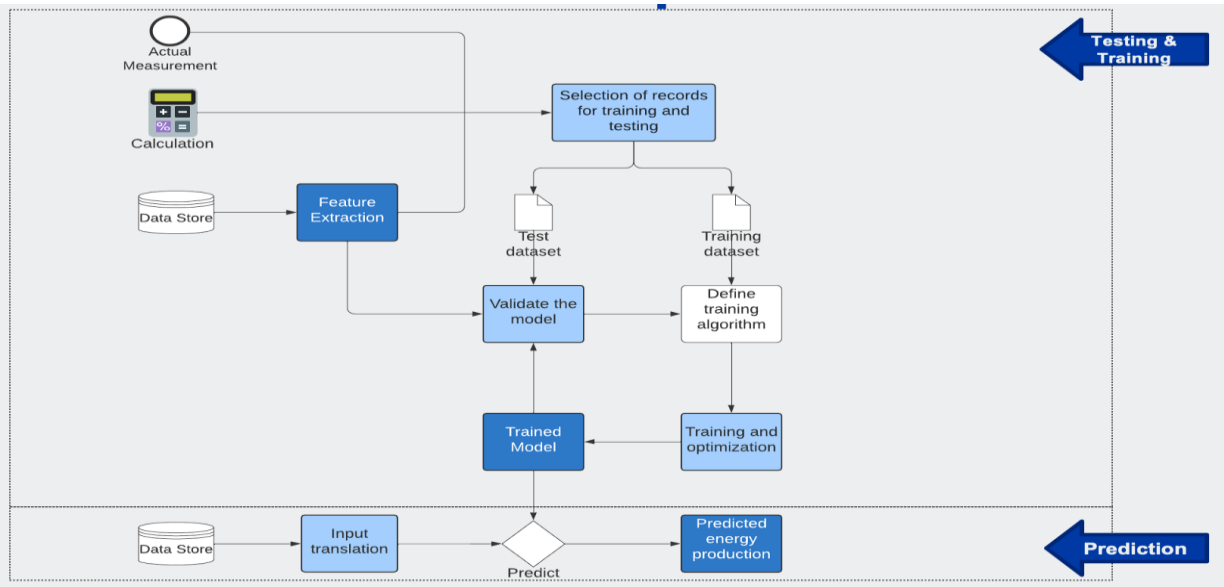
Feature	Definition
DATE	Our date is in monthly format (i.e. 01-01-99)
IPG2211A2N	Industrial production index measuring production output for gas and electric utilities

Process Flow

The image below illustrates the overall process that we essentially take in building any time series model. After acquiring the ample amount of data, we identify the components and stationarize the data. Now finding the optimal parameters to test and train the model and lastly predicting the target variable.



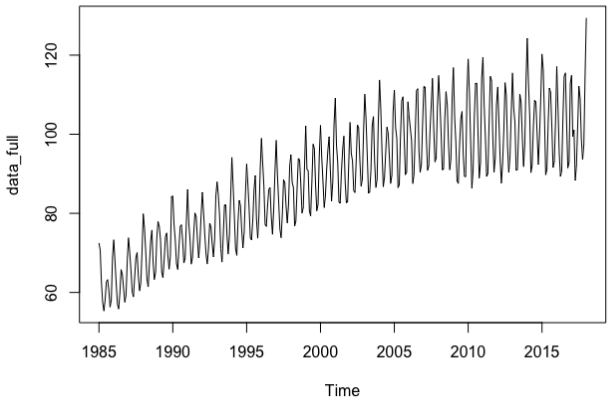
The architectural diagram represents the testing, training, and prediction components of our model.



Architectural Diagram

Data Exploration

The first step in any time series problem is to make the time series stationary i.e., its statistical properties (mean and variance) should not change with time. So, we plotted the time series to check its stationarity. Visual inspection of the plot reveals a prominent increasing trend as well as non-stationarity of the time series. To cross-check, we performed the augmented Dickey-Fuller test which resulted in a p-value greater than 0.05 meaning the null hypothesis is true i.e. The time series is non-stationary.



After the initial plotting of our data, we observe an increasing trend, along with high variances among the data points. We use differentiation techniques to make stationary and use log function to reduce variance. After pre-processing, the plots of the data show that the data is stationary and variation is reduced. The decomposition also shows that there is seasonality present and the lags are significant. After looking at the ACF and PACF plots, we looked at the residuals of different combinations for our p and q based on the lags. We found that SARIMA(2,1,3)(2,1,1), SARIMA(2,1,3)(1,1,1), SARIMA(3,1,3)(2,1,1), and SARIMA(3,1,1)(4,1,0) were the best fit because all p-values were on or above the blue

line which indicates that we fail to reject the null hypothesis that the residuals are independent.

Data Modeling

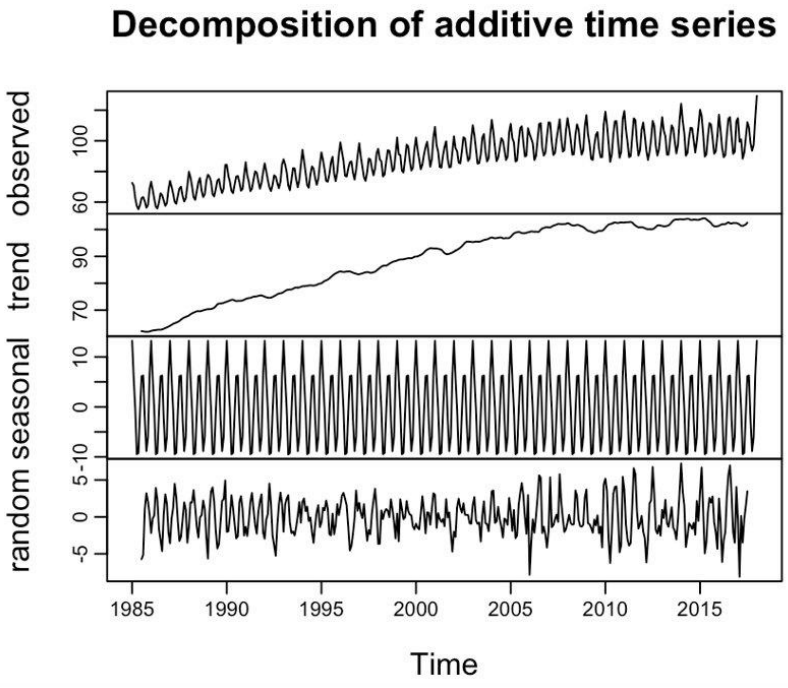
SARIMA

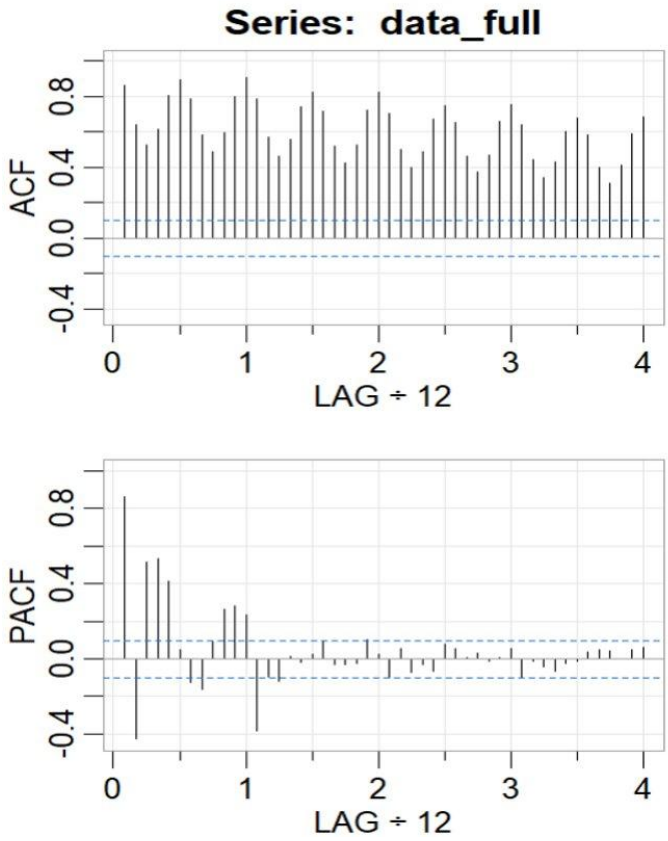
In the ARIMA model, AR is auto regressive, we predict the time series values based on some periods in the past integrating an upward or downward trend and to get rid of it, we use differencing. MA is moving average, which is informing the errors from the previous period to the next period. The new thing we see here is S – seasonality.

In SARIMA (P, D, Q) m: m is the seasonal factor. It's the number of time steps for a single seasonal period.

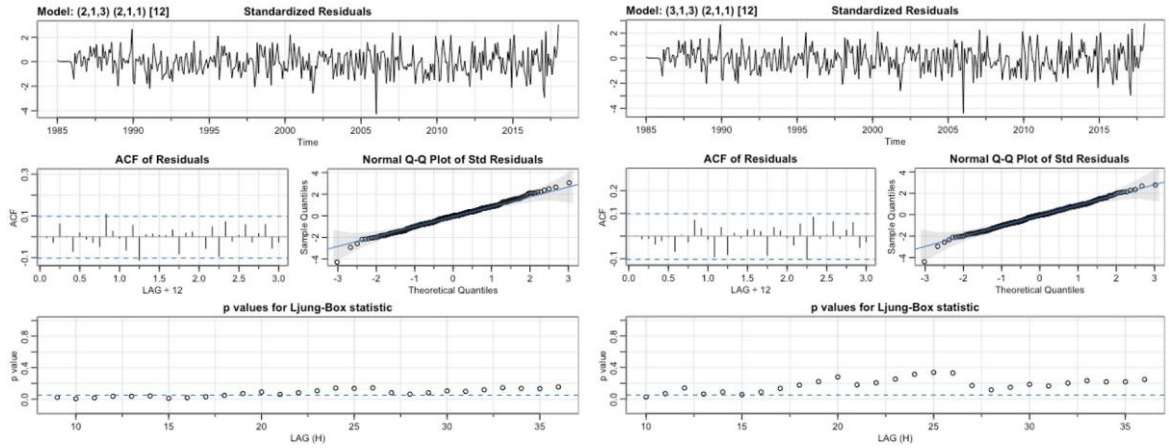
Procedure For Checking Model Diagnostic for SARIMA Models

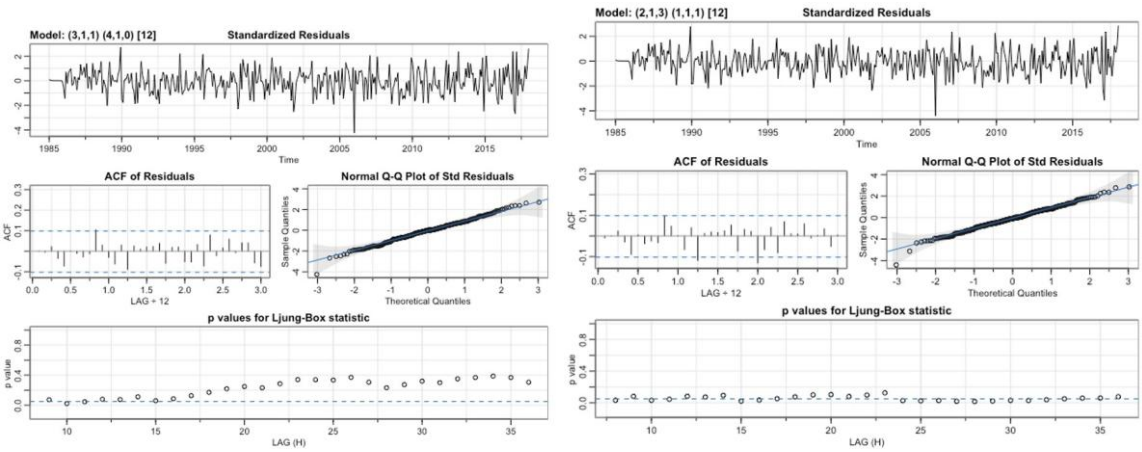
We split the dataset into train and test with 80:20 ratio. We manually try each possible combination of P, D, Q from ACF and PACF plots to find with the best model based on the output of sarima i.e. P-value plot, residual plot and Q-Q plot.





The initial plot of our data displays an upward trend and an increasing variance, therefore we decided to take the difference and log of our dataset.



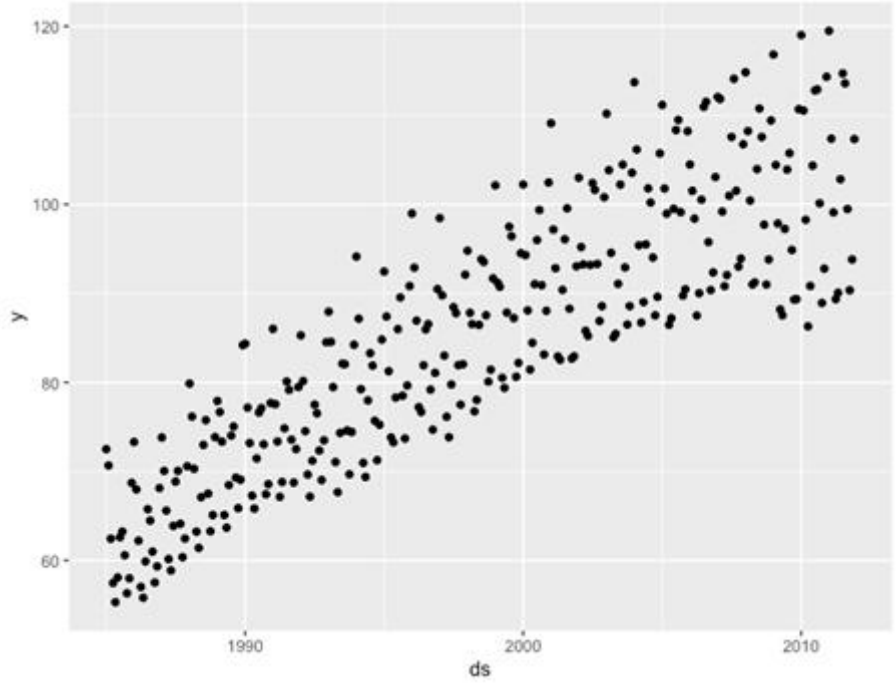


None of the lags are significant for the ACF residuals. There are few outliers in the Q-Q plot, which indicates minor departure from normality. Most of the P-values are above the blue dotted line for the Ljung-Box statistic.

Prophet

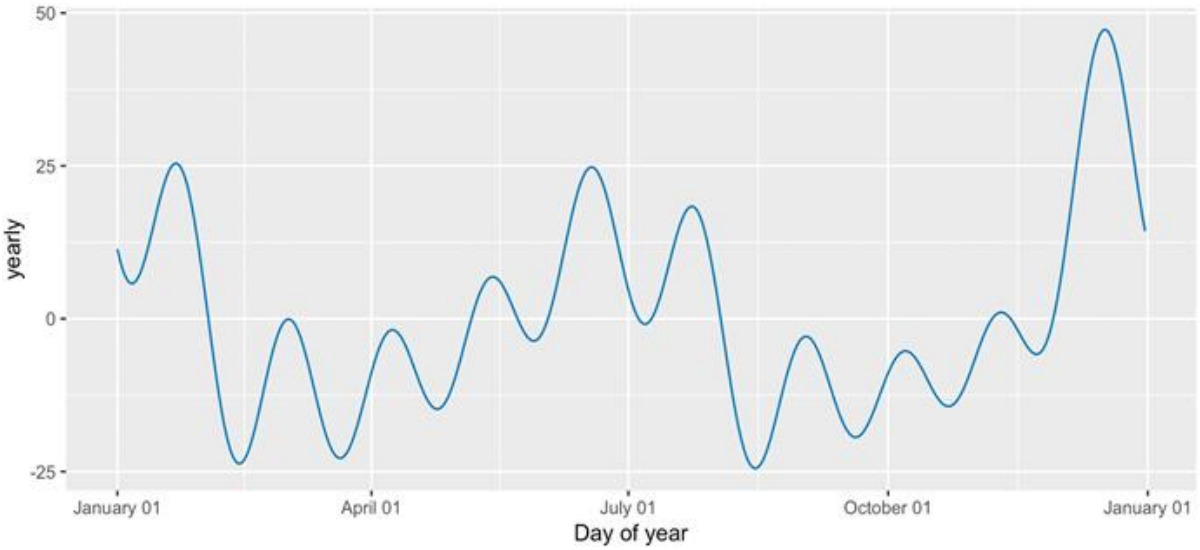
Prophet is a procedure for forecasting time series data based on an additive model, where non-linear trends are fit with monthly, seasonality and holiday effects. It detects the following trend and seasonality from the data first, then combines them together to get the forecasted values.

Following are the visualizations from the Prophet model:



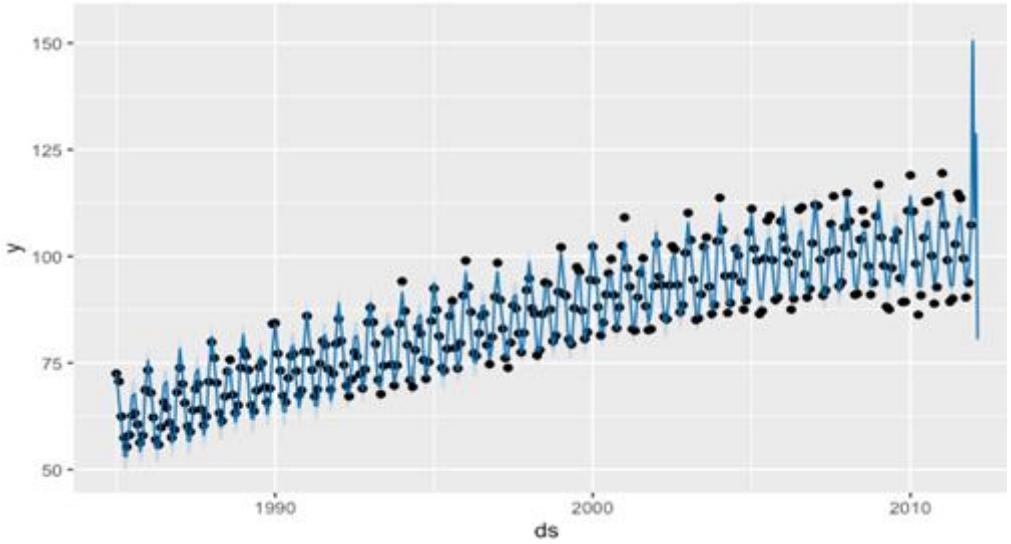
Plot of the raw data

The above graph represents a scatter plot of the target variable with respect to the time which clearly indicates an increasing trend with time.



Yearly trend

This plot depicts the yearly trend of the target variable and as expected, the energy production is maximum during the winter months due to abundant usage of gas for heating purposes followed by during the summer months when the energy demand is also high to cater to the air conditioning needs. There is also a weekly trend detected in the above plot which suggests that there might be more usage of energy on the weekdays than weekends since most work activities slow down on weekends. However, since we do not have weekly data, any further investigation was not possible.



Plot of forecast

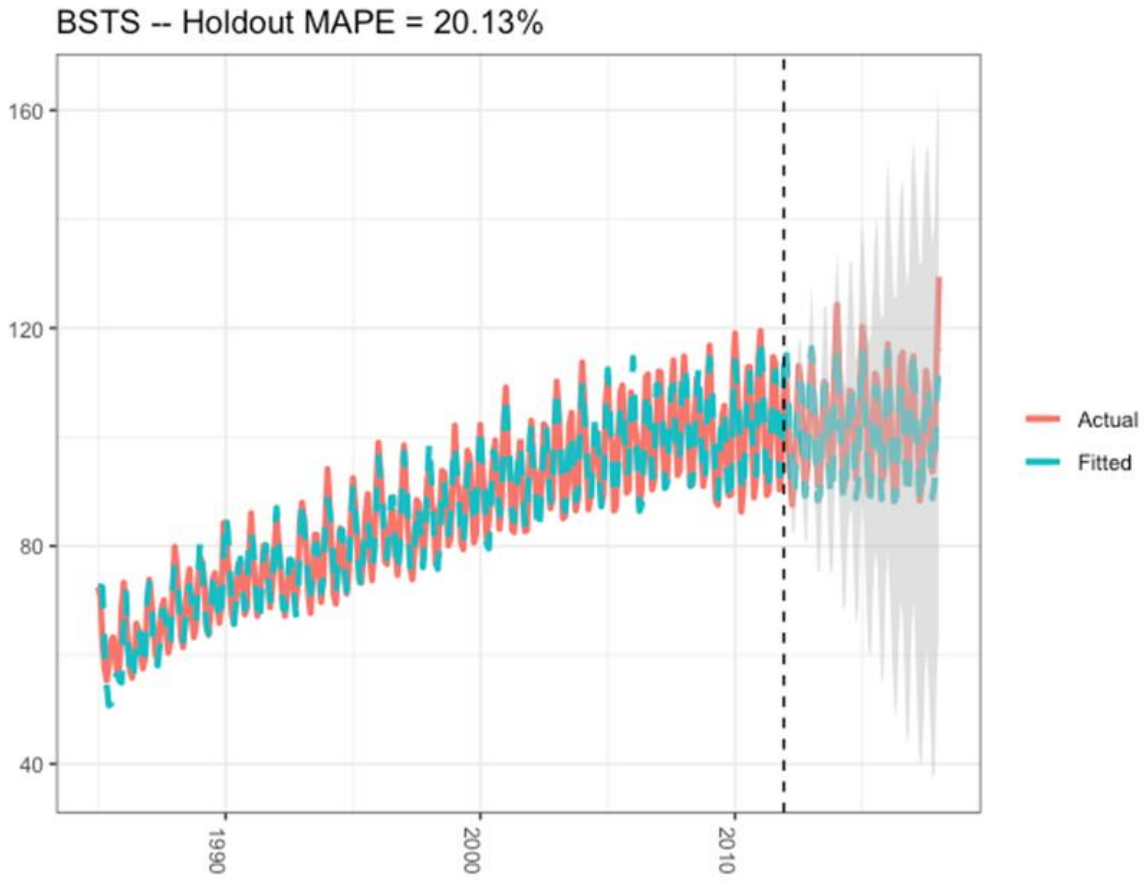
This plot indicates how closely the Prophet model (blue lines) predicted data points

are following the actual data points and as evident, the model is doing a decent job. The MAPE on the test data was 24.10%

Bayesian Structural Time Series (BSTS)

Bayesian Structural Time Series is a specific approach to solving "structural time series" models. A structural time series is a member of the very broad class of state space models which model time series as observations of a hidden state that evolves over time.

Here, the BSTS model was trained using 500 MCMC draws and then the forecast was created by averaging across the MCMC draws. Also, the initial MCMC interactions were discarded. The Mean Absolute Percentage Error calculated on the test data was 20.13%



Long Short-Term Memory (LSTM)

It is an advanced version of recurrent neural network (RNN) architecture that was designed to model chronological sequences and their long-range dependencies more precisely than conventional. When performing normal text modeling, most of the preprocessing task and modeling task focuses on creating data sequentially. Examples of such tasks can be POS tagging, stop words elimination, sequencing of the text.

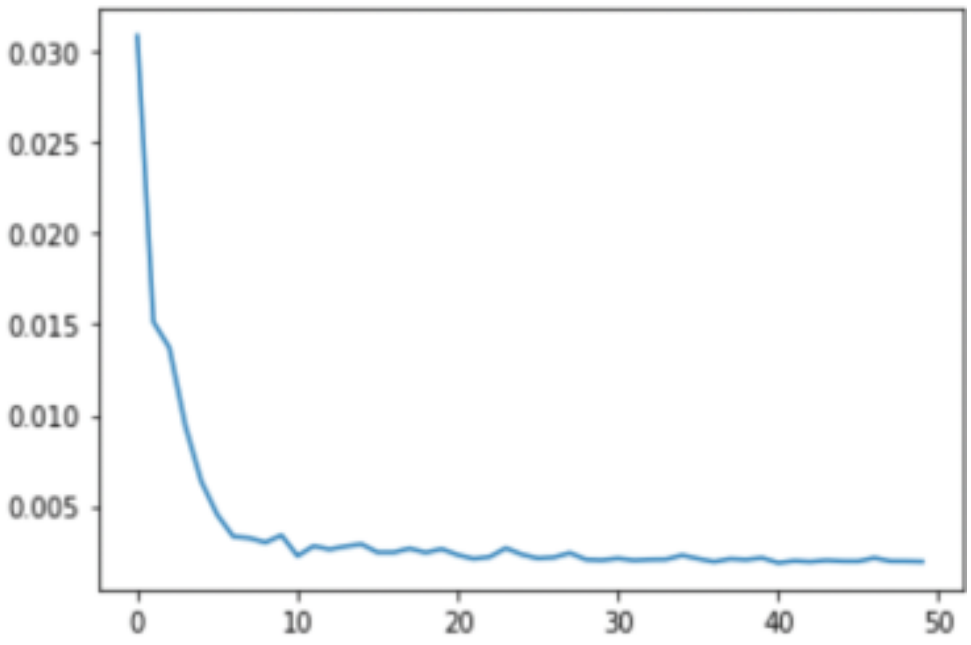
LSTM has a feature through which it can memorize the sequence of the data. It has one more feature that it works on the elimination of unused information and as we know the text data always consists of a lot of unused information which can be eliminated by the LSTM so that the calculation timing and cost can be reduced. The LSTM can take inputs with different lengths which is useful to build forecasting models for specific industries. The different gates inside LSTM boost its capability for capturing non-linear relationships for forecasting. Causal factors generally have a non-linear impact on demand. When these factors are used as part of the input variable, the LSTM could learn the nonlinear relationship for forecasting.

Model Summary:

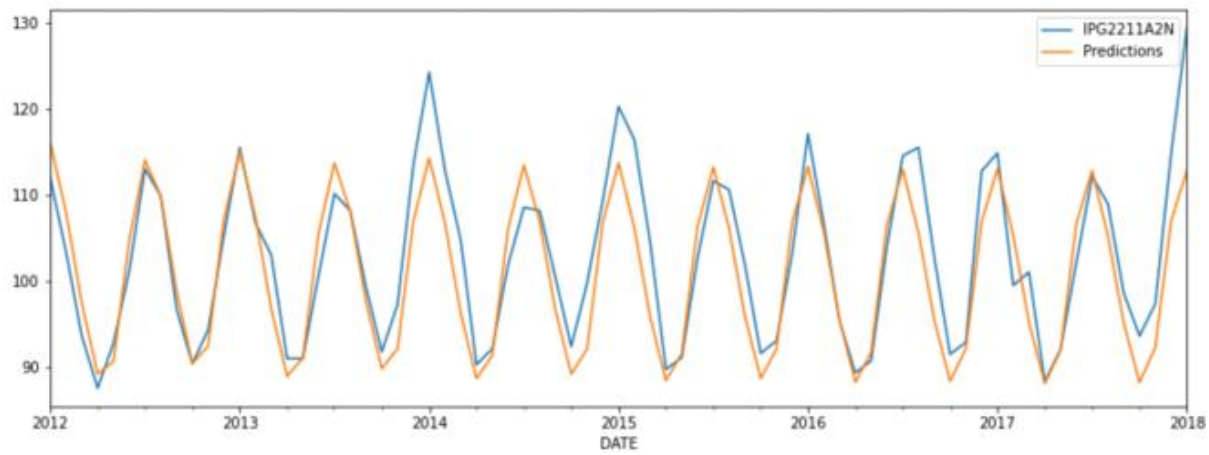
```
model.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                 Output Shape              Param #
=================================================================
 lstm_1 (LSTM)                (None, 100)               40800

 dense_1 (Dense)              (None, 1)                 101

=================================================================
Total params: 40,901
Trainable params: 40,901
Non-trainable params: 0
_________________________________________________________________
```

Loss By Number of Epochs:



Actual vs Predicted Values:



Model Evaluation and Results

We trained and fit each model with the best combination of P,D,Q for the non-seasonal and seasonal parts. We then applied the model on the test dataset to calculate the MAPE for each combination and plotted on the Bar graph.

Model Summaries:

```
ARIMA(3,0,0)(0,1,1)[12] with drift

Coefficients:
         ar1      ar2     ar3     sma1   drift
      0.6128  -0.0986  0.1543  -0.6676  0.1307
s.e.  0.0577   0.0664  0.0569   0.0440  0.0113
```

AUTO ARIMA COEFFICIENTS

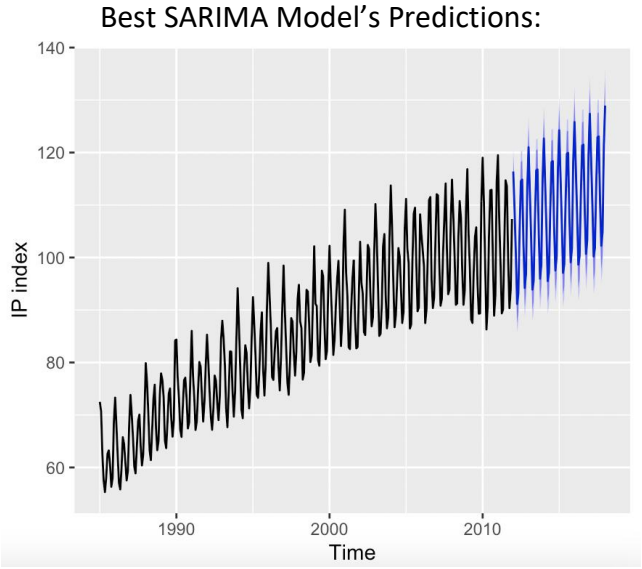
```
ARIMA(2,1,3)(2,1,1)[12]
Box Cox transformation: lambda= 0

Coefficients:
         ar1      ar2      ma1     ma2     ma3     sar1     sar2     sma1
      1.4104  -0.4317  -1.7739  0.6798  0.0985  -0.0218  -0.1833  -0.7341
s.e.  0.1600   0.1519   0.1700  0.2892  0.1279   0.0790   0.0688   0.0666
```

BEST SARIMA MODEL (2, 1, 3) (2, 1, 1) COEFFICIENTS

```
ARIMA(3,1,3)(2,1,1)[12]
Box Cox transformation: lambda= 0

Coefficients:
          ar1     ar2     ar3      ma1     ma2      ma3    sar1     sar2     sma1
      -0.0954  -0.548  0.5100  -0.2839  0.2649  -0.7944  0.0111  -0.1682  -0.7308
s.e.   0.2631   0.080  0.0686   0.2813  0.1531   0.1313  0.0823   0.0782   0.0690
```

SARIMA MODEL (3, 1, 3) (2, 1, 1) COEFFICIENTS

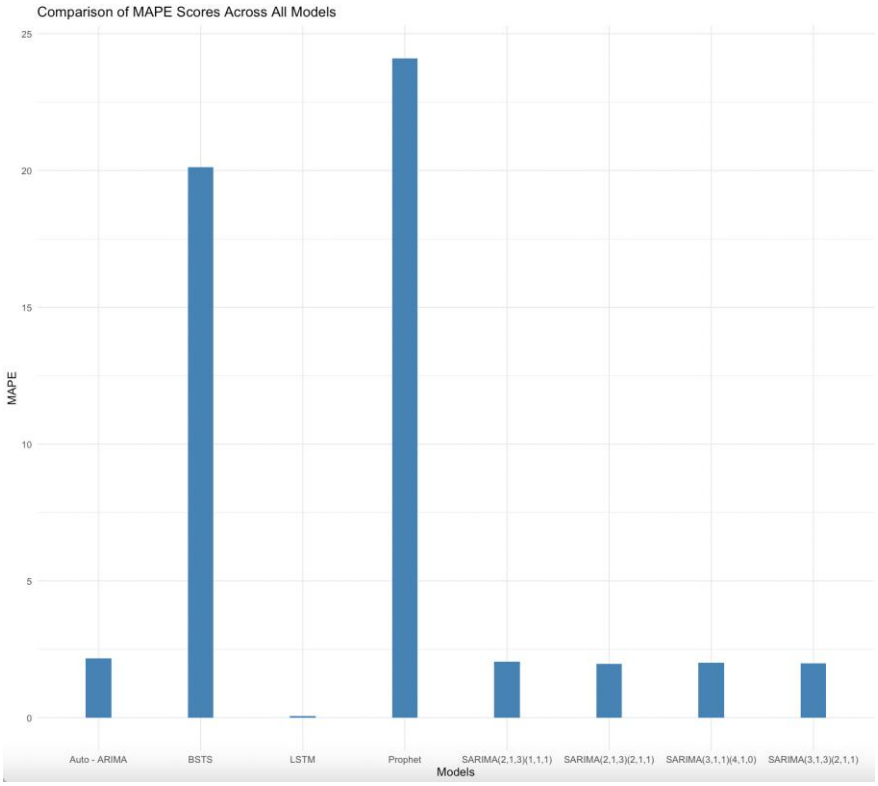
The standard error for our best model is lower than the auto-ARIMA and other models' standard errors.



These are the predictions of our best performing SARIMA model (2, 1, 3) (2, 1, 1).

Conclusion and Challenges

Model	MAPE Score
Auto - ARIMA	2.17%
SARIMA(2,1,3)(2,1,1)	1.97%
SARIMA(2,1,3)(1,1,1)	2.05%
SARIMA(3,1,3)(2,1,1)	1.98%
SARIMA(3,1,1)(4,1,0)	2.00%
Prophet	24.10%
LSTM	0.05%
BSTS	20.13%



Plot of the time series indicated long term trend and differencing removed the trend which was confirmed by ADF test. Compared to Auto-ARIMA, Manual SARIMA gave better results. Increasing epochs led to better accuracy of LSTM model. Though some of the p-values were below the blue dotted line for SARIMA (2,1,3) (2,1,1) and (2,1,3) (1,1,1), we received good model accuracies.

Prophet and BSTS did not give the results we expected as the MAPE score for both models were large. We could not increase the number of epochs to more than 50 due to computational challenges.

LSTM is our best performing model because it can answer our problem statement with highest precision.

Dataset

Electric_Production.csv

Code:

Readme .docx

predictive_main_project_LSTM_final.ipy

Final_project_Predictive.R

Presentation:

Predictive final project presentatior

References:

https://alfred.stlouisfed.org/series?seid=IPG2211A2N&utm_source=series_page&utm_medium=related_content&utm_term=related_resources&utm_campaign=alfred
https://developers.refinitiv.com/en/article-catalog/article/forecasting-inflation-romanian-case-study-using-sarima-models
https://towardsdatascience.com/exploring-the-lstm-neural-network-model-for-time-series-8b7685aa8cf
https://cran.microsoft.com/snapshot/2021-10-23/web/packages/bsts/bsts.pdf
https://www.analyticsvidhya.com/blog/2018/10/predicting-stock-price-machinelearningnd-deep-learning-techniques-python/
https://machinelearningmastery.com/time-series-forecasting-with-prophet-in-python/