**Final Project**

**Dataset:** [Electric Production](https://www.kaggle.com/shenba/time-series-datasets?select=Electric_Production.csv)

**Report created by:** Keyur Unadkat

**Description of the dataset:**

The given dataset is about monthly production of electricity by a power plant each day in MWh (megawatt hours) from 1985 to 2018. The dataset is quite simple with only 2 columns of data to work with. This is a time series data that we would fit into one of our models to predict electricity production. Prediction of electricity production can be useful in order to buy raw materials and also forecast potential breakdowns for the power plant. Electricity demands have always been raising and therefore we would see a constant upward trend in the production as well.

**Attribute Information:**

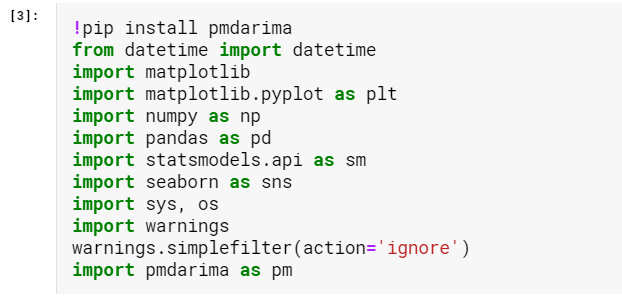
1. Date: date on which the given data is collected. It is an ‘object’ attribute in the format **D/M/YYYY.**

2. Production: Electricity produced in MWh. (float)

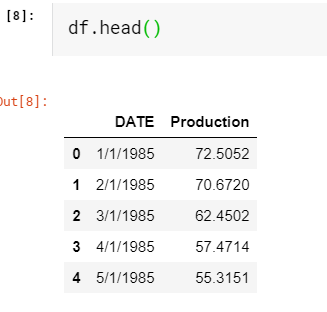
**EDA and feature Engineering**

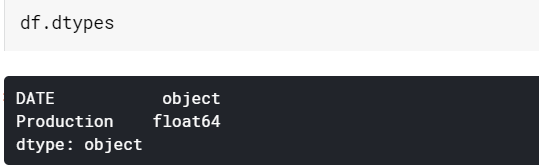
Exploratory Data Analysis (EDA) is an open-ended process where we calculate statistics and make figures to find trends, anomalies, patterns, or relationships within the data. The goal of EDA is to learn what our data can tell us. It generally starts out with a high-level overview, then narrows in to specific areas as we find intriguing areas of the data. The findings may be interesting in their own right, or they can be used to inform our modelling choices, such as by helping us decide which features to use.

🡪Importing useful libraries and data





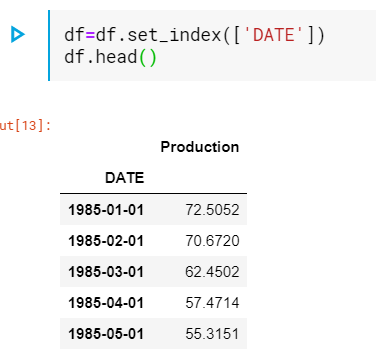




As we can see from the data, the date column is of ‘object’ type. We need to convert it into a datetime object, and also set it as an index.



🡪Setting the ‘DATE’ column as index of the dataframe.



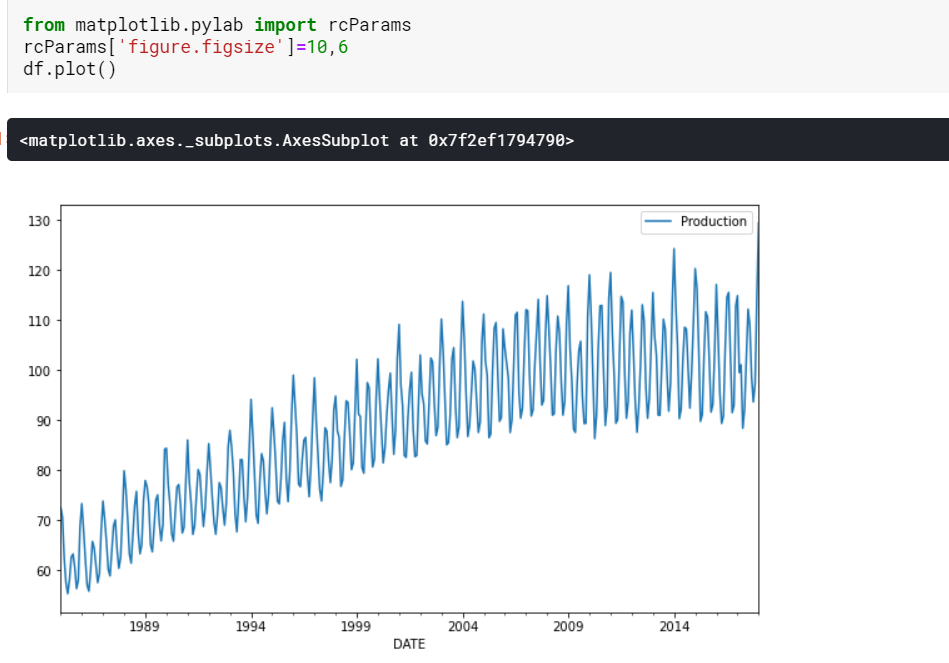
We need to check if there are any null values in our data.



Fortunately, we have a clean dataset, with no null values.

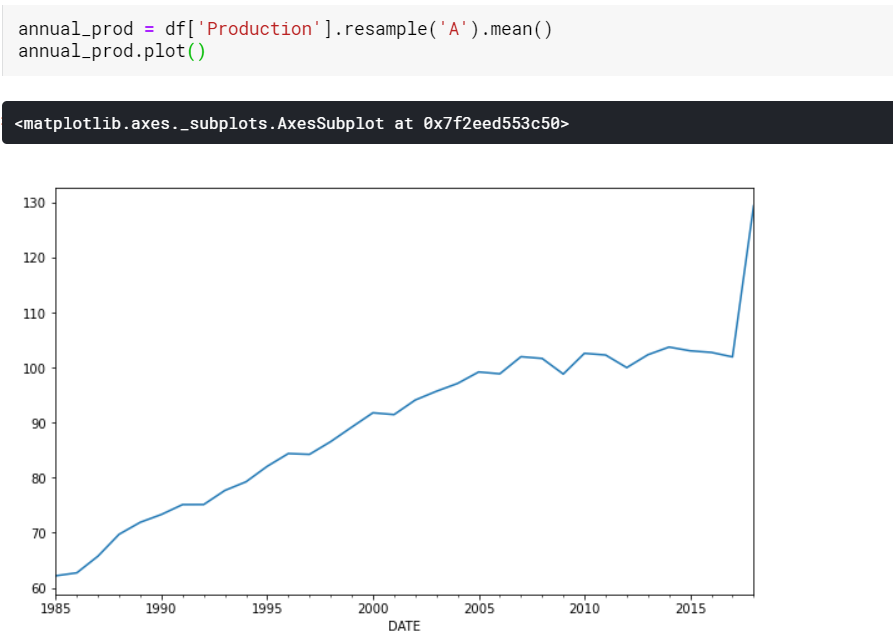
Now its time to analyse our data visually and try to get some insights out of it.

🡪Production Vs Time plot



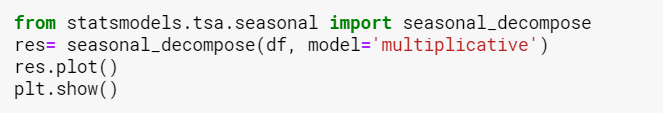
**Insights:** We can clearly see that the production has an upward trend, given the fact that demand for the electricity increases over time. Also there is a seasonality component build into it, which seems to be increasing with the trend.

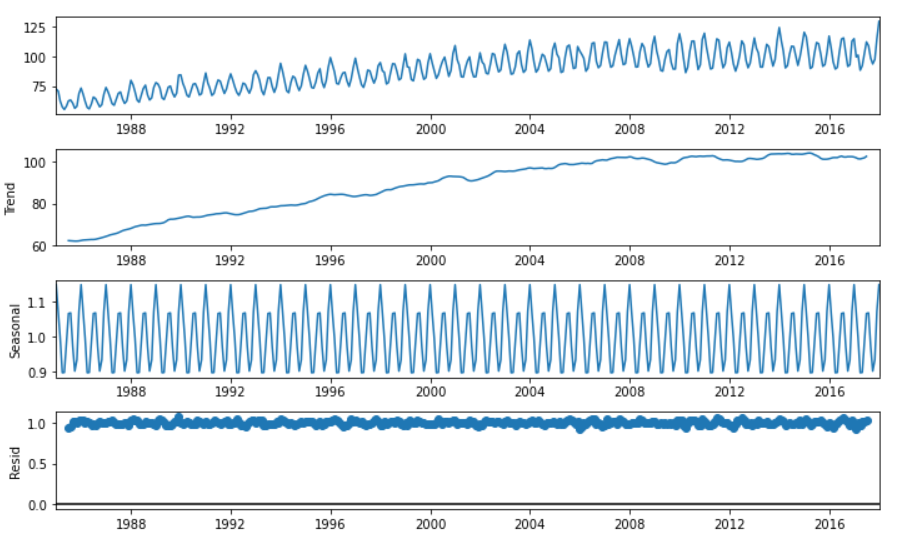
🡪Annual production plot



🡪Seasonality and trend

We would dissect the seasonality and trend from our dataset.

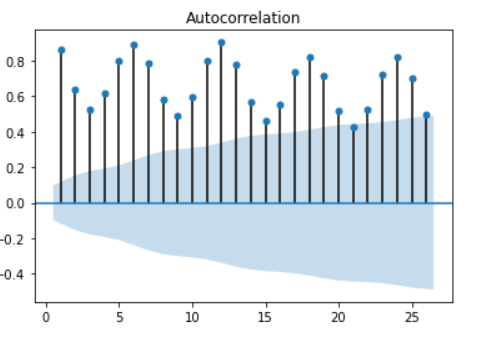


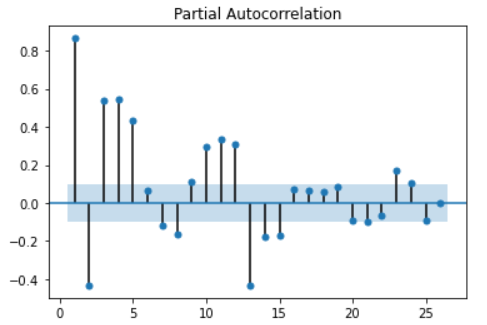


**Insights:** The data has a strong seasonality which seems to reoccur at a constant interval of 6 months. Also, the residual(noise) is quite stationary with mean=1.

🡪ACF and PACF plots

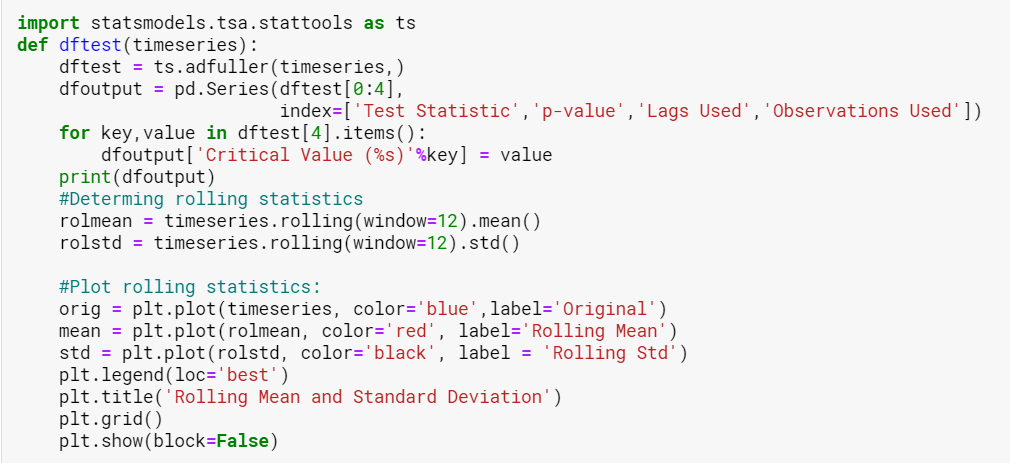


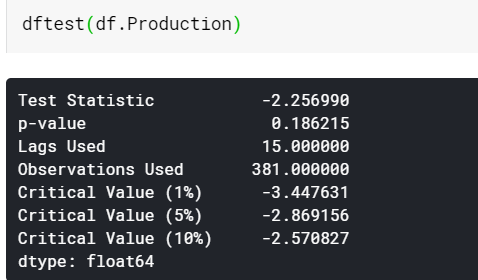




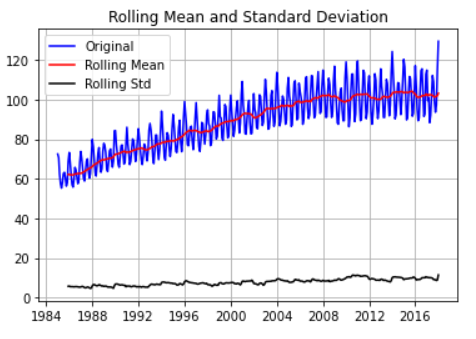
**Insight:** We can figure out from this plot that there is a strong seasonality component, and we can also infer that we can have p=3 and q=3.

🡪We would now run statistical tests (Dickey-Fuller test) in order to see if our data is stationary or not. The given function also finds rolling mean and standard deviation plots.

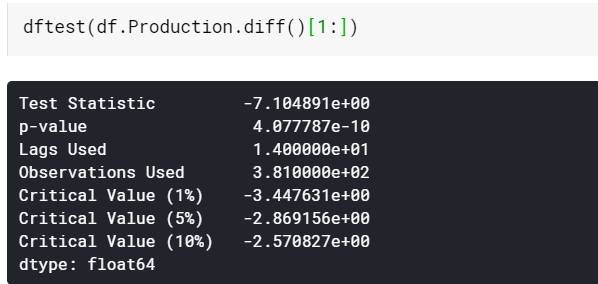




As we can see, the p value is greater than 0.05, therefore we cannot reject the null hypothesis, and thus we can say our data is not stationary.



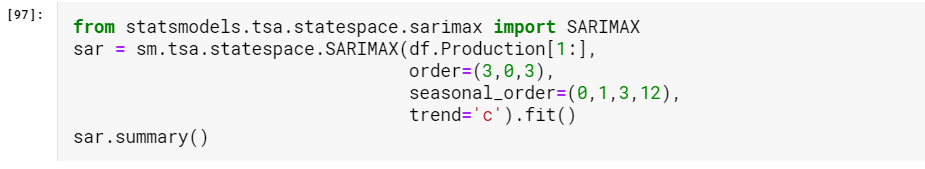
🡪To make our data stationary, we would now take difference of the consecutive values of Production column.



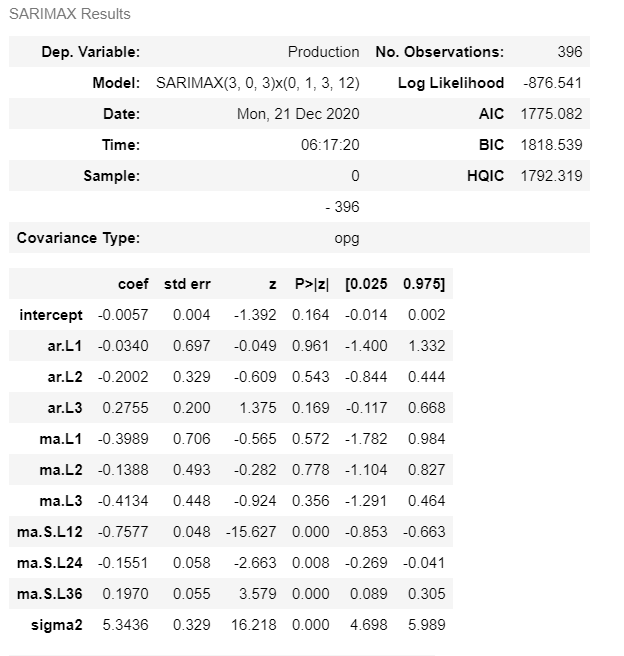
From the p value, we can make out that our data is now stationary.

Now its time to fit a model and predict values.

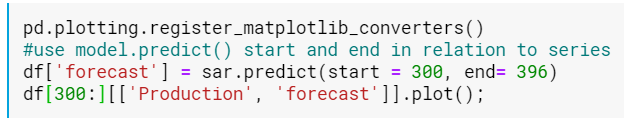
**Approach 1** Using an SARIMA model.

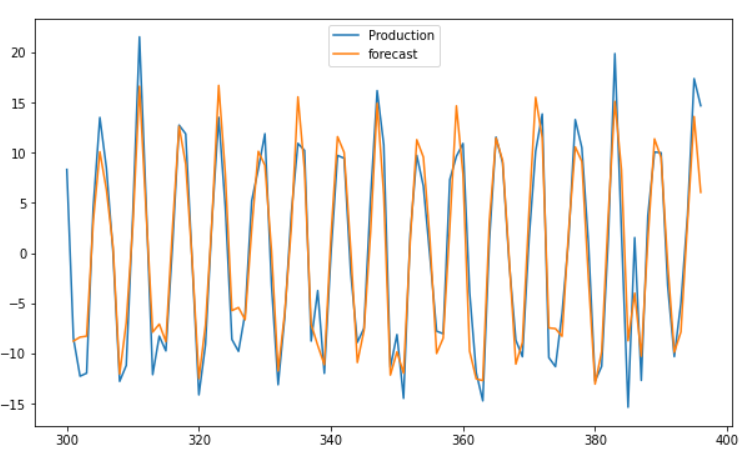


🡪Summary of the model



Lets now see visually how the model fits:





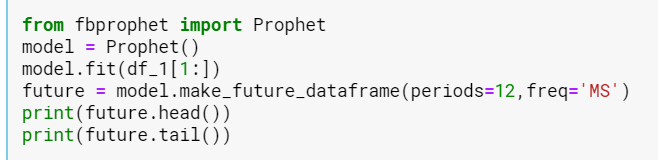
The model seems to be fitting well on the data. This is because we were able to predict p and q correctly.

**Approach 2** Using Facebook prophet.

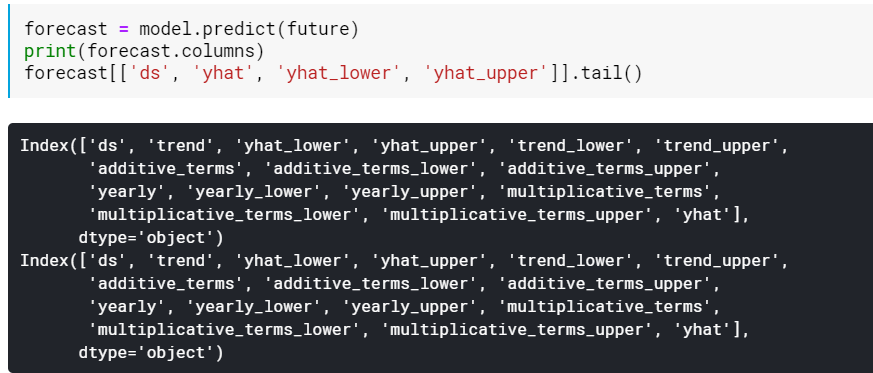
First we would need to convert the columns to ds and y respectively so that model can detect the data and work on it.

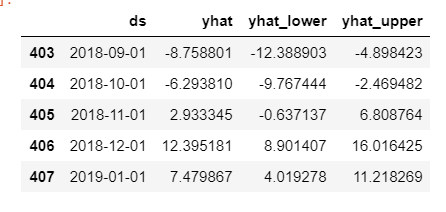


🡪Creating the model and fitting the data. Here we also create an empty dataframe for forecasting the future values.

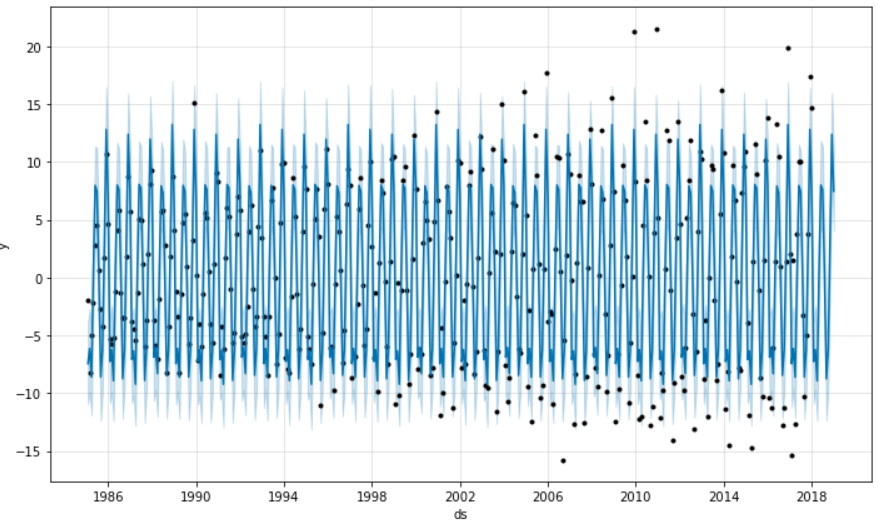


🡪Predictions:



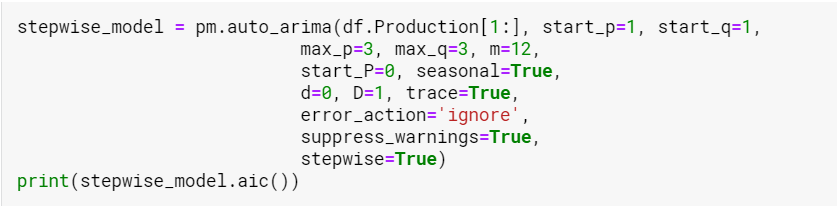


🡪Plotting the forecast:

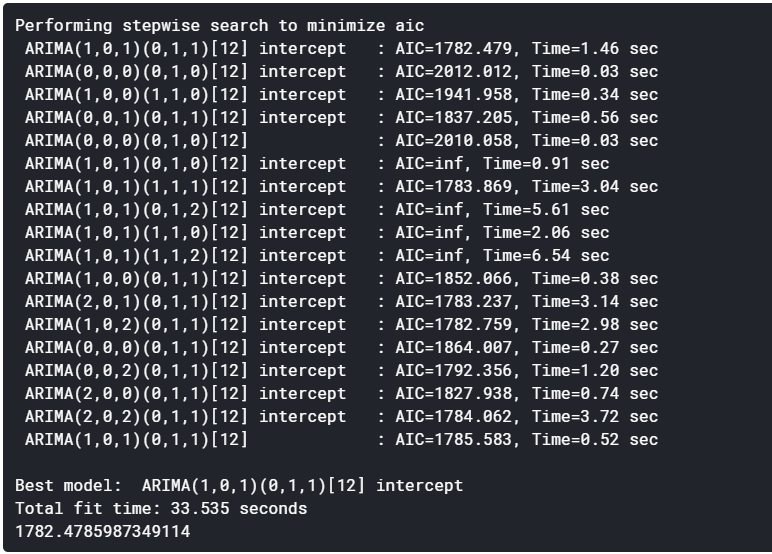


**Approach 3** Using Autofit method.

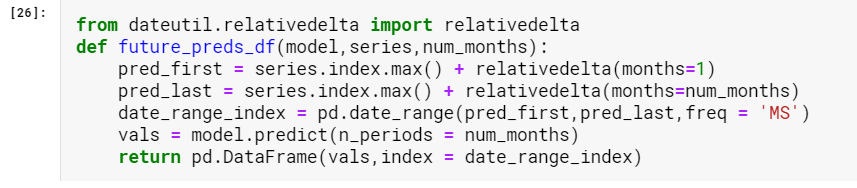
🡪We would now use pmdarima to autofit our data and find the best combination of p,q,s for the model.



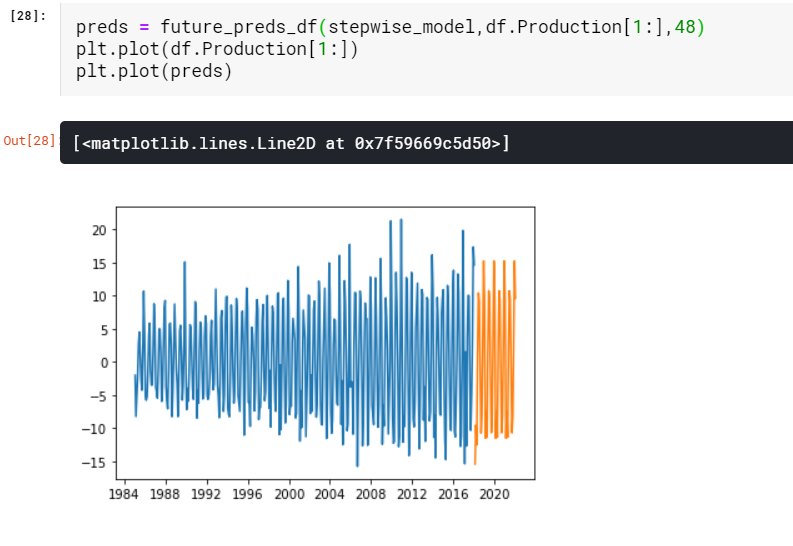
This method would find the best fit with lowest AIC value.



Now we would make a function that can predict the values, for the given model and time duration.



Result of the prediction with a plot:



The given ARIMA model predicts the data quite well, although, it does a monotonous job there. This may not be acceptable for a longer run.

**Next Steps:**

So far, we have seen that the SARIMA model fits well to the data, but we now need to explore methods of deep learning to fit the data into them. This would not only pick up the pattern well, but would also predict quite accurately than others.

We would next incorporate LSTM RNNs in order to fit the given time series data.