SALES FORECASTING OF FURNITURE USING TIME SERIES MODELLING

SHUBHAM SADANAND KUTE

PROJECT=03

Step 1: Prototype Selection

Abstract:

"*Time series forecasting* is a key ingredient in the automation and optimization of business processes: in retail, deciding which products to order and where to store them depends on the forecasts of future demand in different regions; in cloud computing, the estimated future usage of services and infrastructure components guides capacity planning; and workforce scheduling in warehouses, call centers, factories requires forecasts of the future workload. Recent years have witnessed a paradigm shift in forecasting techniques and applications, from computer-assisted model- and assumption-based to data-driven and fully-automated. This shift can be attributed to the availability of large, rich, and diverse timeseries data sources, posing unprecedented challenges to traditional time series forecasting methods. As such, how can we build statistical models to efficiently and effectively learn to forecast from large and diverse data sources? How can we leverage the statistical power of "similar" time series to improve forecasts in the case of limited observations? What are the implications for building forecasting systems that can handle large data volumes?"

Time Series DataBase

TSDB is optimized for storing and serving time series through associated pairs of time(s) and value(s). Although it is possible to store time-series data with many different database formats, the design of these systems with time as a key index is distinctly different from relational databases. Early time series databases are associated with industrial applications, which could efficiently store measured values from sensory equipment, but are now used in support of a much wider range of applications. Data compression is used to reduce the volume of data. Data storage and access in time-series databases is discussed in (Dunning and Friedman, 2014) and alphabetically and by revenue here.

Step 2:Prototype Development

Problem Statement

Time series analysis comprises methods for analyzing time series data in order to extract meaningful statistics and other characteristics of the data.

This dataset consists of daily sales data of various products at a superstore.

We will need to apply Time Series (ARIMA) to build model to predict and forecast the sales of furniture for the next one year i.e. predict future values based on previously observed values. We have a 4-year furniture sales data.

Data Preparation

We remove unwanted columns that is not needed and check missing values. Aggregate sales data by date and finally index it with the time series data.

Feature Engineering

We check the stationarity of the data and decide the next step to be taken. Also decompose the data for further clarification and apply the time series model on the data.

Model Comparison

We perform parameter selection to find optimal set of parameters that yields the best performance for the model.

Importing The Required Libraries:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

Loading The Dataset:

```
data=pd.read_excel(r'C:\Users\HP\SHUBHAM PYTHON\Sample - Superstore.xls',header=0)
```

```
data.head()
```

	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	Country	City	...	Postal Code	Region	Product ID	Category	Sub-Category	Pro N
0	1	CA-2016-152156	2016-11-08	2016-11-11	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	...	42420	South	FUR-BO-10001798	Furniture	Bookcases	I Som Colle Book
1	2	CA-2016-152156	2016-11-08	2016-11-11	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	...	42420	South	FUR-CH-10000454	Furniture	Chairs	Hon De F Upholst Sta Cha
2	3	CA-2016-138688	2016-06-12	2016-06-16	Second Class	DV-13045	Darrin Van Huff	Corporate	United States	Los Angeles	...	90036	West	OFF-LA-10000240	Office Supplies	Labels	Adhe Adc Label Typewi
3	4	US-2015-108966	2015-10-11	2015-10-18	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	33311	South	FUR-TA-10000577	Furniture	Tables	Bre CR Series Rectan
4	5	US-2015-108966	2015-10-11	2015-10-18	Standard Class	SO-20335	Sean O'Donnell	Consumer	United States	Fort Lauderdale	...	33311	South	OFF-ST-10000760	Office Supplies	Storage	Eldon 'N Roll Sy

5 rows × 21 columns

Categorywise Counting:

```
data["Category"].value_counts()
```

```
Office Supplies    6026
Furniture          2121
Technology         1847
Name: Category, dtype: int64
```

Here we can see that the office Supplies variable has 6026 counts.Technology has 1847 counts and our required variable has 2121 counts.

Finding Out Min Max Order Date

```
furniture['Order Date'].min(),furniture['Order Date'].max()
```

```
(Timestamp('2014-01-06 00:00:00'), Timestamp('2017-12-30 00:00:00'))
```

Here we can see that the Minimum Order Date is 2014-01-06 and the Maximum Order Date is 2017-12-30.

Total Furniture Data

```
furniture.head(10)
```

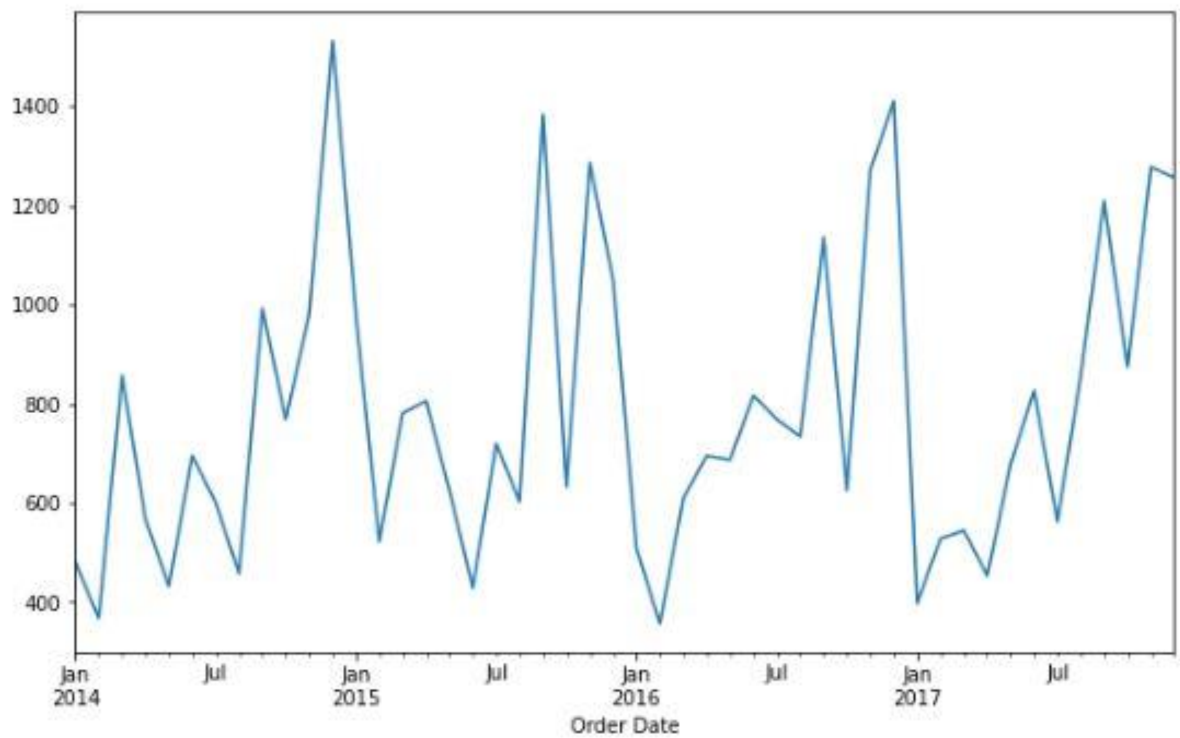
	Order Date	Sales
7474	2014-01-06	2573.820
7660	2014-01-07	76.728
866	2014-01-10	51.940
716	2014-01-11	9.940
2978	2014-01-13	545.940
4938	2014-01-13	333.999
6474	2014-01-14	61.960
970	2014-01-16	127.104
5465	2014-01-19	181.470
6327	2014-01-20	272.940

Value Counts of orders in Each date

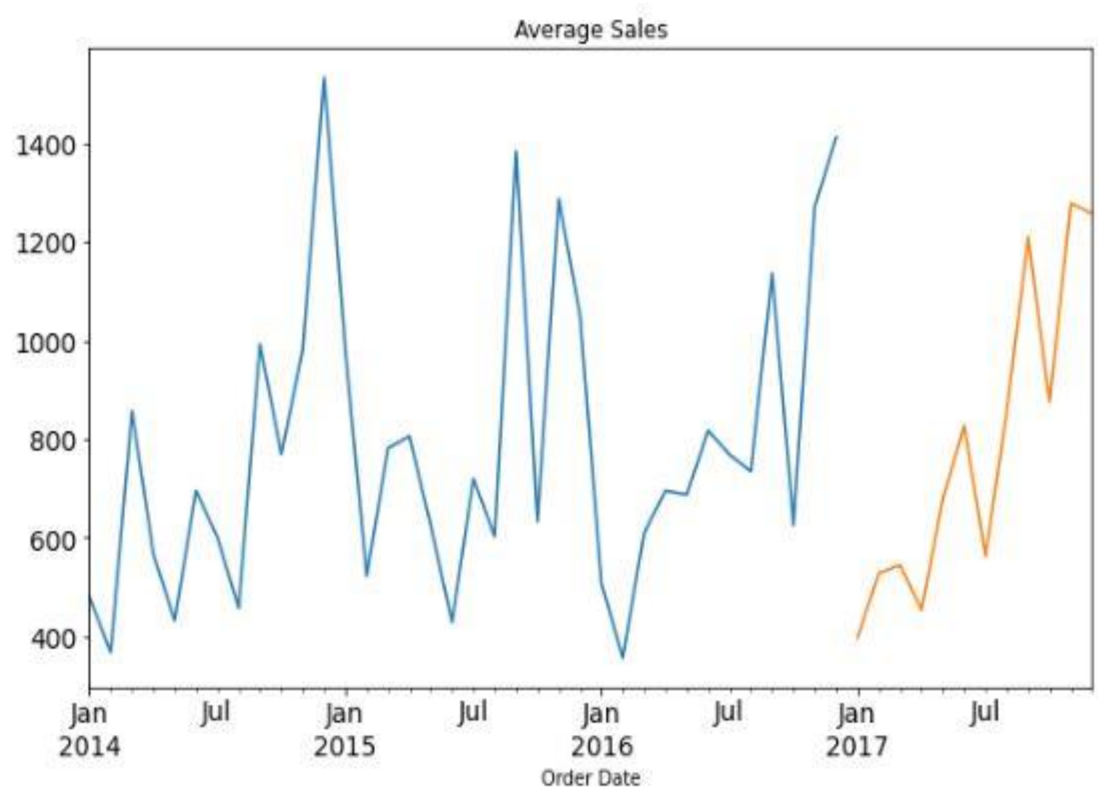
```
furniture["Order Date"].value_counts()
```

```
2016-09-05    10
2016-12-25     9
2016-12-01     9
2017-11-19     9
2017-10-30     9
              ..
2015-11-26     1
2015-11-24     1
2015-11-22     1
2015-11-19     1
2017-12-30     1
Name: Order Date, Length: 889, dtype: int64
```

Visually checking the Time Series for Trend And Other Components



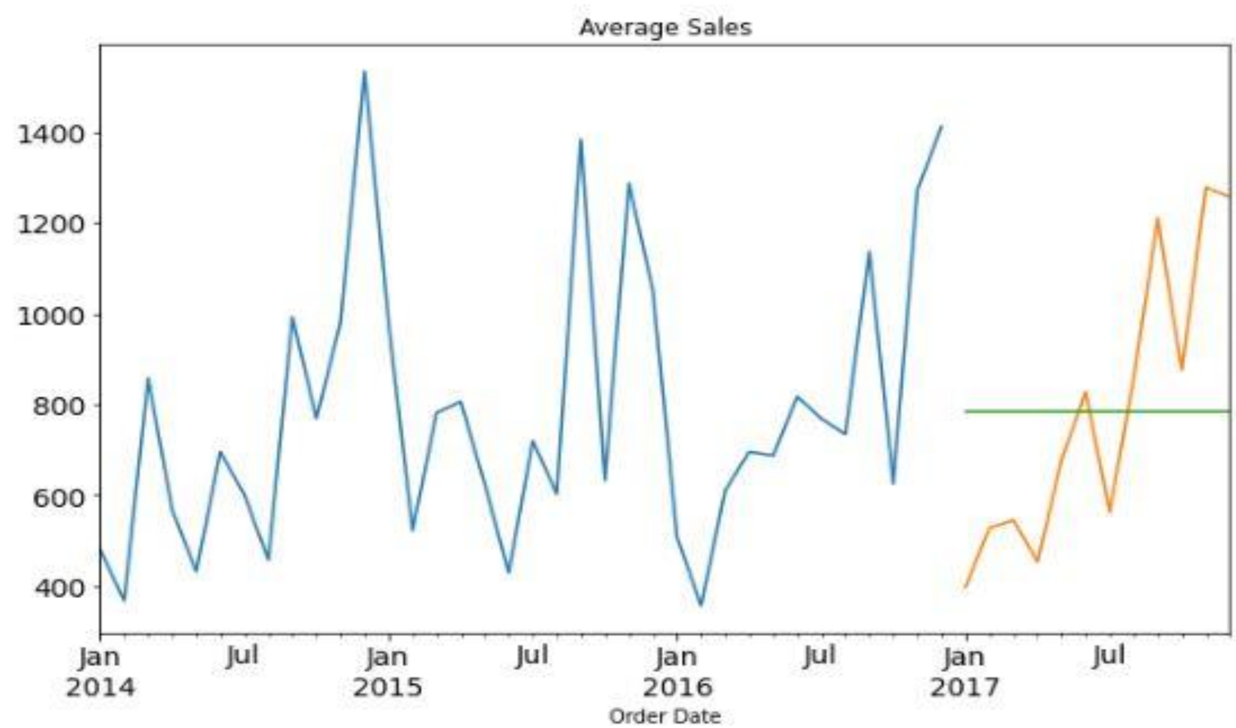
Visualizing Average Sales Graphically



Model Building

```
from statsmodels.tsa.api import SimpleExpSmoothing
Exp_Smooth = test.copy()
#smoothing level=alpha
#alpha value should be in the range of 0 to 1.
#values close to 0 indicate that older values are very less imp compared to the recent values
#values close to 1 indicate that older values are equally imp
fit1 = SimpleExpSmoothing(train).fit(smoothing_level=0.01)
# fit1 is a model object
Exp_Smooth['SES'] = fit1.forecast(steps=len(test))
# whatever is the forcasted value will be stored in Exp_Smooth['SES']

train.plot(figsize=(10,6), title= 'Average Sales', fontsize=14)
test.plot(figsize=(10,6), title= 'Average Sales', fontsize=14)
Exp_Smooth['SES'].plot(figsize=(10,6), title= 'Average Sales', fontsize=14)
plt.show()
```



Here we can see that the actual values are very much far away from the predicted values so the errors are goin to be high.

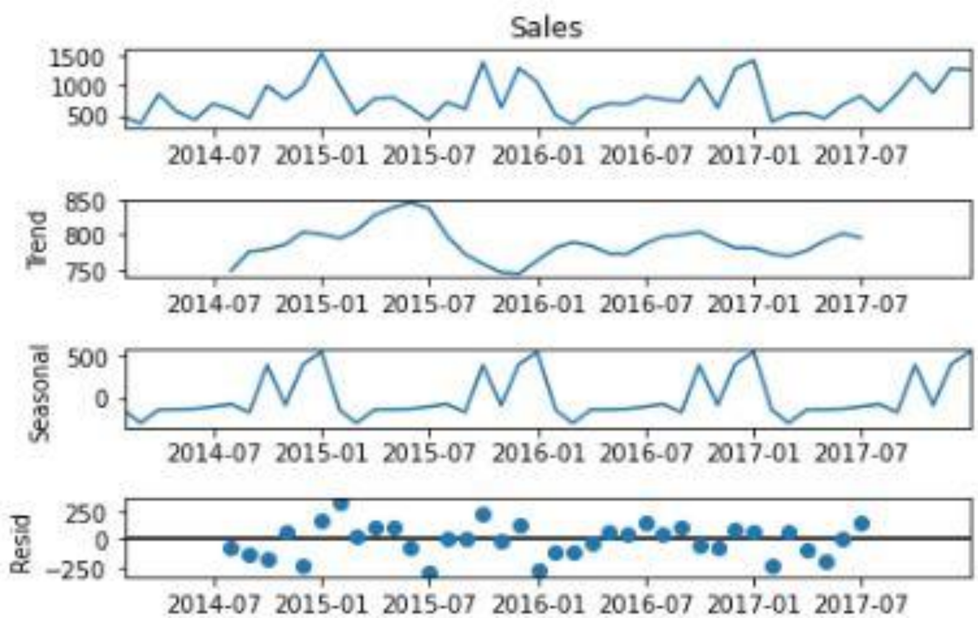
To check the error between actual values and the predicted values,choose the model with the lowest RMSE values.

Seasonal Decomposition:

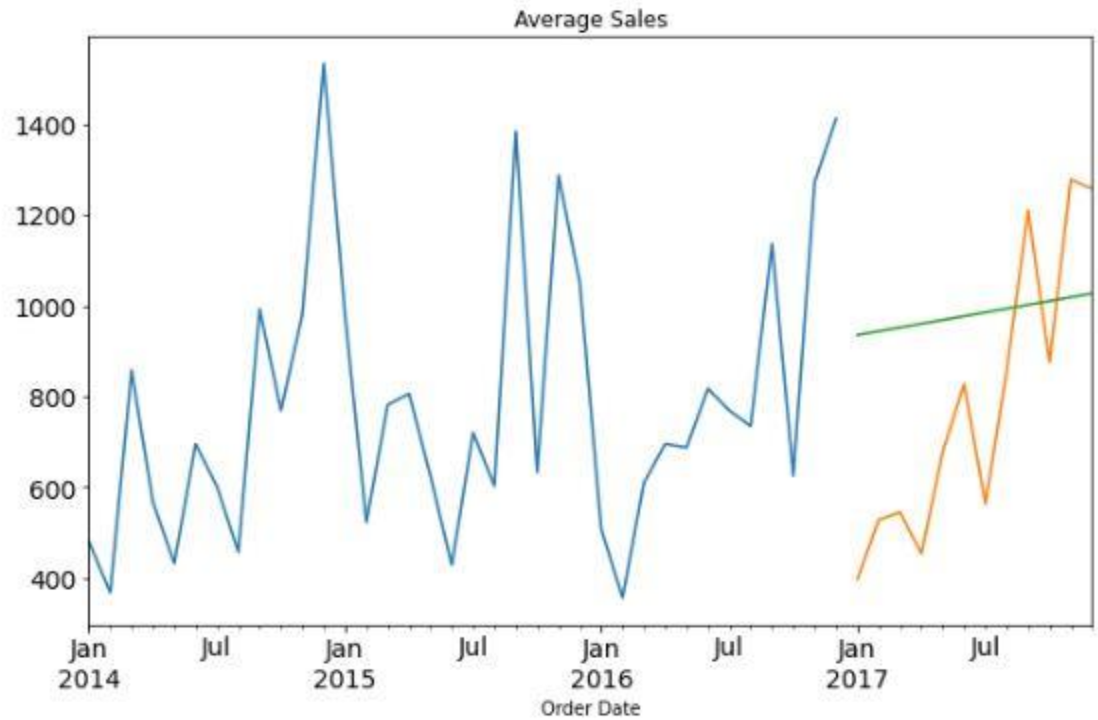
Decomposition procedures are used in Time Series to describe the trends and seasonal factors in Time Series.

```
import statsmodels.api as sm
decomposition=sm.tsa.seasonal_decompose(y)
fig=decomposition.plot()
plt.show()
```

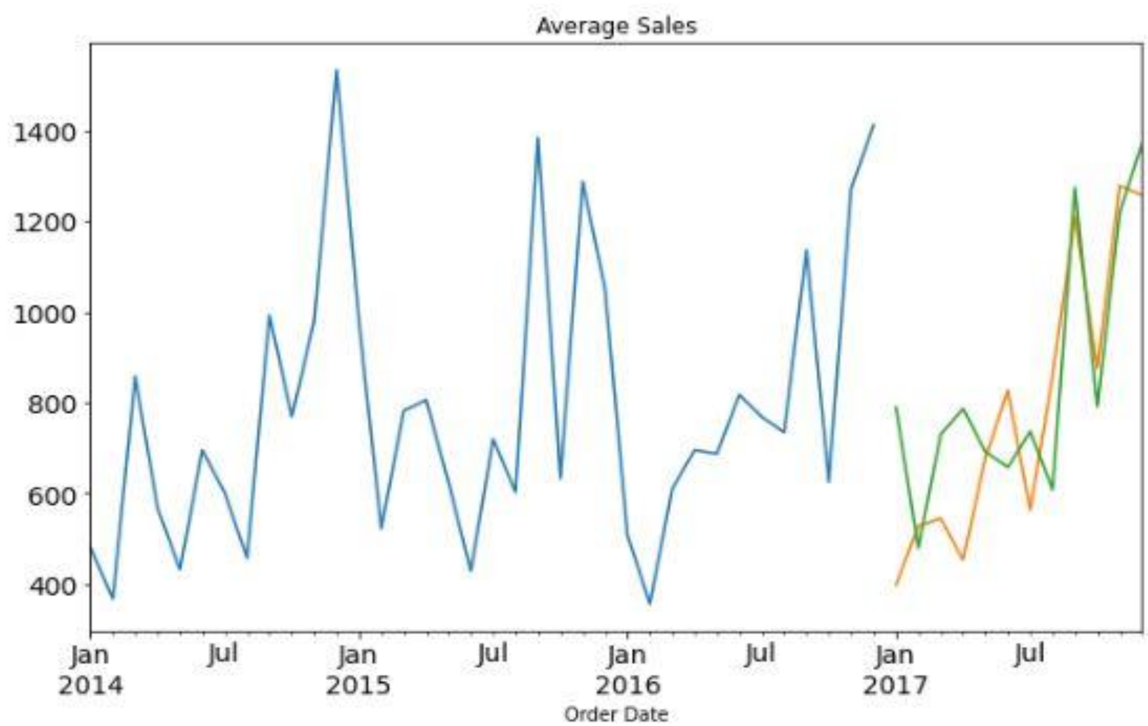
Decomposing the Time Series into three different components " Trends ", " Seasonal ", " Residual ".



Holt Linear Plot



Holt Winter Plot



The graph looks quite good and the model performs well with Holt Winter because we can see the model has handeled spikes and trends very well and it has done quite accurate predictions.

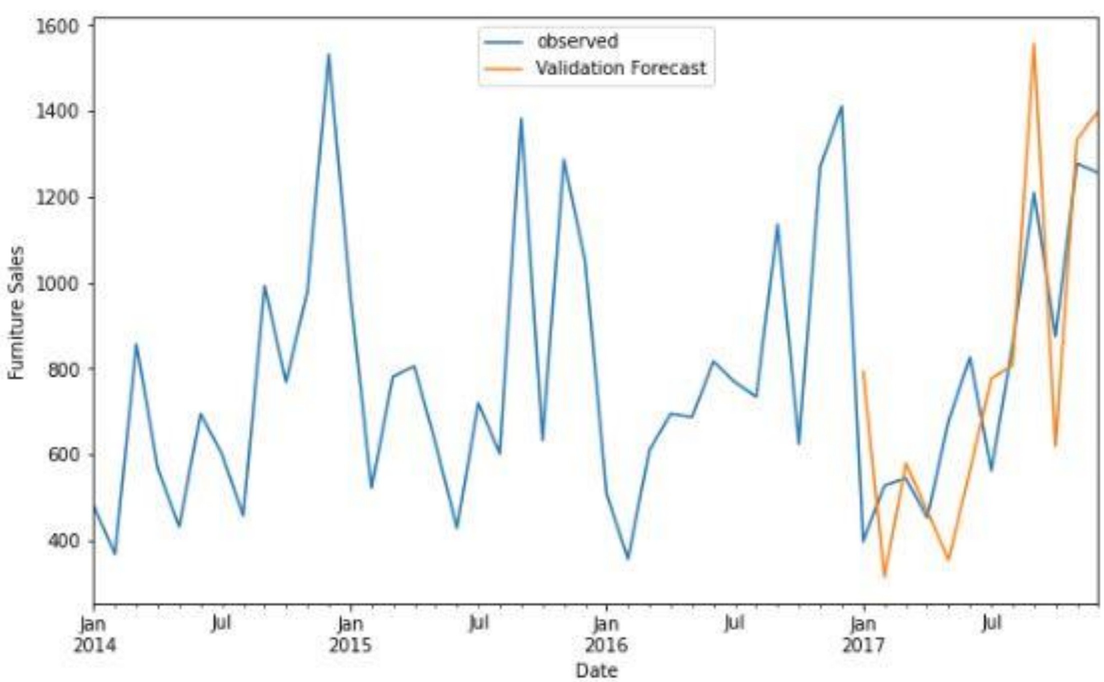
ARIMA :

AutoRegressive Integrated Moving Average is a statistical analysis model that leverages Time Series Data to forecast future trends.

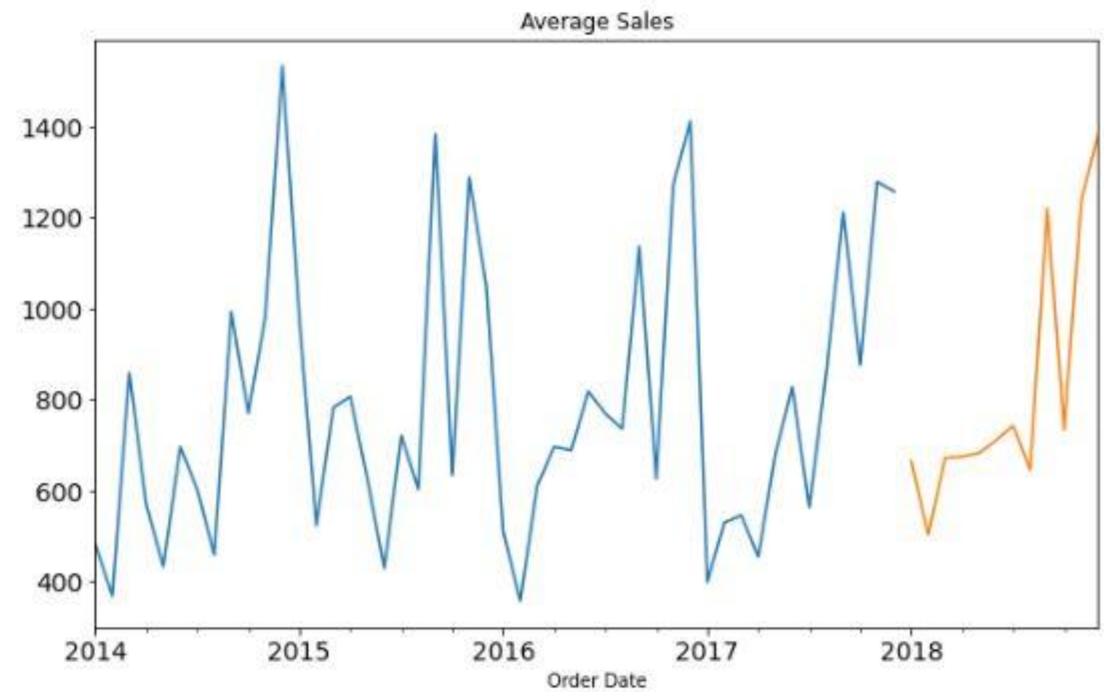
SARIMAX:

Seasonal AutoRegressive Integrated Moving Average extends the ARIMA framework by seamlessly integrating seasonal patterns and exogenous variables.

Visualization of Observed value and the Validation Forecast by using SARIMAX



Final visualizations



The orange line defines the forecated values for the year 2018.

Step 3: Business Modelling

Time series analysis is critical for businesses to predict future outcomes, assess past performances, or identify underlying patterns and trends in various metrics. Time series analysis can offer valuable insights into stock prices, sales figures, customer behavior, and other time-dependent variables.