Sales Forecasting Using ARIMA

System Diagnostics project

Author:

Nandan Hegde

179042

Objective:

The objective of this project is to get a proper dataset of daily based sales may be from a supermarket or any production industry and use that dataset to generate a Time Series plot of the sales. Next step is to use ARIMA model and predict the future sales and plotting the forecast of the same. In this project I am considering 10 days for sales forecasting.

Introduction:

A time series in general is a sequence where a collection of data or a matrix is recorded over regular time intervals.

Depending on the frequency of the interval a time series can be yearly, quarterly, monthly, weekly, daily, hourly, minutes and even can be seconds wise. For example, the data of airline ticket sales per day is a time series.

Forecasting is a step where we want to predict the future value collection the series in going continue. Forecasting a time series (Like future sales) is often of tremendous commercial value.

Every Time Series will consist of several key features such as trend, seasonality etc. We are going to analyse those features of a time series data set for any sales, and then we use mathematical models to forecast the sales into the future.

Forecasting using ARIMA:

Forecasting is a process of predicting the future data based on past and present data. One of the most commonly used method for forecasting a time series is ARIMA model, which stands for Auto Regressive Integrated Moving Average.

In an ARIMA model, 3 important parameters are used to aid model the key features of a Time series, that is seasonality, trend and noise. Those parameters are labelled as **p, d** and **q.**

p is related to auto-regressive aspect of the model; **d** is associated with the integrated part of the model and **q** is the parameter associated with the moving average part of the model.

Our model has seasonal component. So, we will be using seasonal ARIMA model.

Dataset:

Found a dataset for daily sales of a supermarket for a year stored in an Excel file. Total number of sales recorded in the dataset is 1048575.

Since I am plotting time series and forecast on daily basis, I have done group by dates.

Using Date and daily sale amount for training the ARIMA model. Neglecting the other fields in the dataset.

Data is read from Excel file as a Pandas DataFrames.

Shape of the dataset:

```
data.shape
```

```
(1048575, 5)
```

Top 5 Rows:

Date	date_block_num	shop_id	item_id	Price	item_cnt_day
2013-02-01	0	59	22154	999.00	1
2013-03-01	0	25	2552	899.00	1
2013-05-01	0	25	2552	899.00	-1
2013-06-01	0	25	2554	1709.05	1
2013-01-15	0	25	2555	1099.00	1

Shape of dataset after GroupBy date: (334 days). Date column is used as Index column.

```
data.shape
```

```
(334, 1)
```

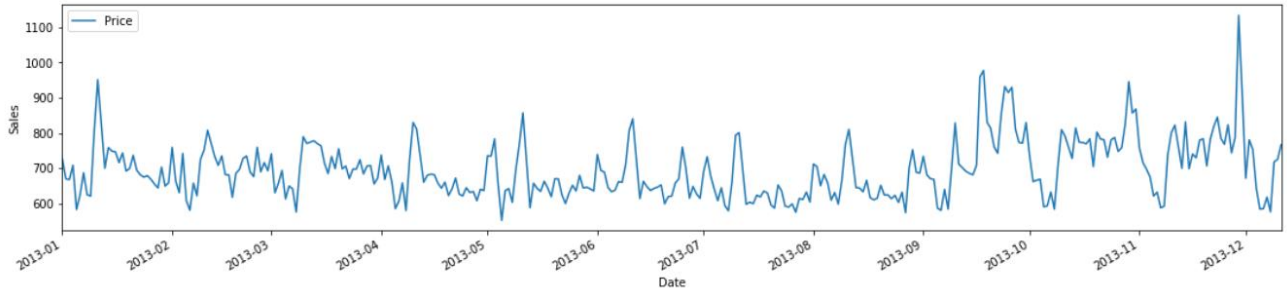
Top 5 Rows:

Date	Price
2013-01-01	726.05
2013-01-02	669.48
2013-01-03	667.68
2013-01-04	708.27
2013-01-05	582.60

Actual Time Series from sales data:

X-Axis: Date

Y-Axis: Sale amount per day



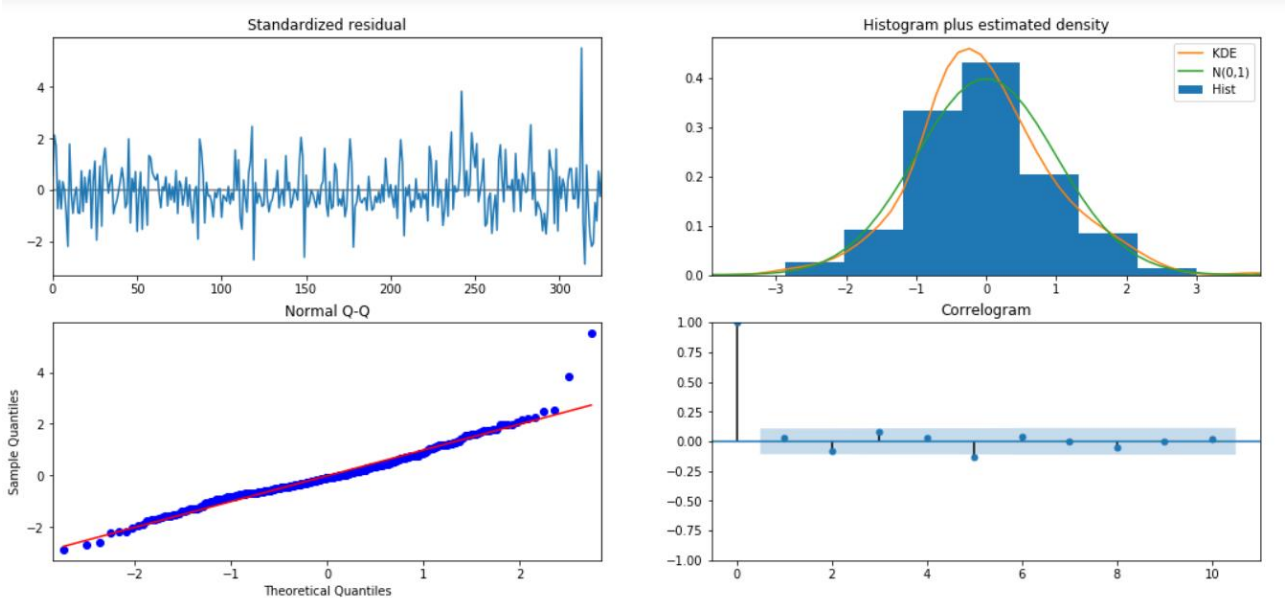
Actual sales per day includes multiple product sold in a day and taken sum of overall sale on each day.

ARIMA model:

- Next step is to create Seasonal ARIMA model with frequency assigned to 1 (D=1). Time series in for daily plot so parameter m is set to 1 (**m=1**).
- We need to enable seasonal parameter and stepwise algorithm.
- Let's plot the summary of the ARIMA model and diagnostics.

```
Fit ARIMA: order=(1, 1, 1) seasonal_order=(0, 1, 1, 7); AIC=3657.816, BIC=3676.751, Fit time=3.305 seconds
Fit ARIMA: order=(0, 1, 0) seasonal_order=(0, 1, 0, 7); AIC=3879.597, BIC=3887.170, Fit time=0.042 seconds
Fit ARIMA: order=(1, 1, 0) seasonal_order=(1, 1, 0, 7); AIC=3802.100, BIC=3817.247, Fit time=0.543 seconds
Fit ARIMA: order=(0, 1, 1) seasonal_order=(0, 1, 1, 7); AIC=3702.117, BIC=3717.265, Fit time=0.627 seconds
Fit ARIMA: order=(1, 1, 1) seasonal_order=(1, 1, 1, 7); AIC=3656.082, BIC=3678.803, Fit time=1.630 seconds
Fit ARIMA: order=(1, 1, 1) seasonal_order=(1, 1, 0, 7); AIC=3748.267, BIC=3767.202, Fit time=0.808 seconds
Fit ARIMA: order=(1, 1, 1) seasonal_order=(1, 1, 2, 7); AIC=3661.700, BIC=3688.209, Fit time=4.826 seconds
Fit ARIMA: order=(1, 1, 1) seasonal_order=(0, 1, 0, 7); AIC=3814.028, BIC=3829.175, Fit time=0.513 seconds
Fit ARIMA: order=(1, 1, 1) seasonal_order=(2, 1, 2, 7); AIC=3659.983, BIC=3690.278, Fit time=4.854 seconds
Fit ARIMA: order=(0, 1, 1) seasonal_order=(1, 1, 1, 7); AIC=3700.988, BIC=3719.923, Fit time=0.917 seconds
Fit ARIMA: order=(2, 1, 1) seasonal_order=(1, 1, 1, 7); AIC=3657.492, BIC=3684.000, Fit time=2.403 seconds
Fit ARIMA: order=(1, 1, 0) seasonal_order=(1, 1, 1, 7); AIC=3709.609, BIC=3728.543, Fit time=0.996 seconds
Fit ARIMA: order=(1, 1, 2) seasonal_order=(1, 1, 1, 7); AIC=3657.209, BIC=3683.717, Fit time=2.714 seconds
Fit ARIMA: order=(0, 1, 0) seasonal_order=(1, 1, 1, 7); AIC=3716.795, BIC=3731.943, Fit time=0.697 seconds
Fit ARIMA: order=(2, 1, 2) seasonal_order=(1, 1, 1, 7); AIC=3658.873, BIC=3689.168, Fit time=3.418 seconds
Fit ARIMA: order=(1, 1, 1) seasonal_order=(2, 1, 1, 7); AIC=3657.212, BIC=3683.720, Fit time=5.359 seconds
Total fit time: 33.673 seconds
================================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
--------------------------------------------------------------------------------
intercept      0.0154      0.007      2.148      0.032       0.001       0.029
ar.L1          0.5964      0.039     15.165      0.000       0.519       0.674
ma.L1         -0.9993      0.352     -2.841      0.005      -1.689      -0.310
ar.S.L7        0.1098      0.060      1.835      0.066      -0.007       0.227
ma.S.L7       -0.9988      1.479     -0.675      0.499      -3.897       1.899
sigma2      3772.3774   5357.970      0.704      0.481   -6729.050    1.43e+04
================================================================================
```

Diagnostics of the model fitting will include Standardized residuals (Normalized), Sample vs Theoretical quantities, Histogram with estimated density and Correlation diagram.



Forecasting:

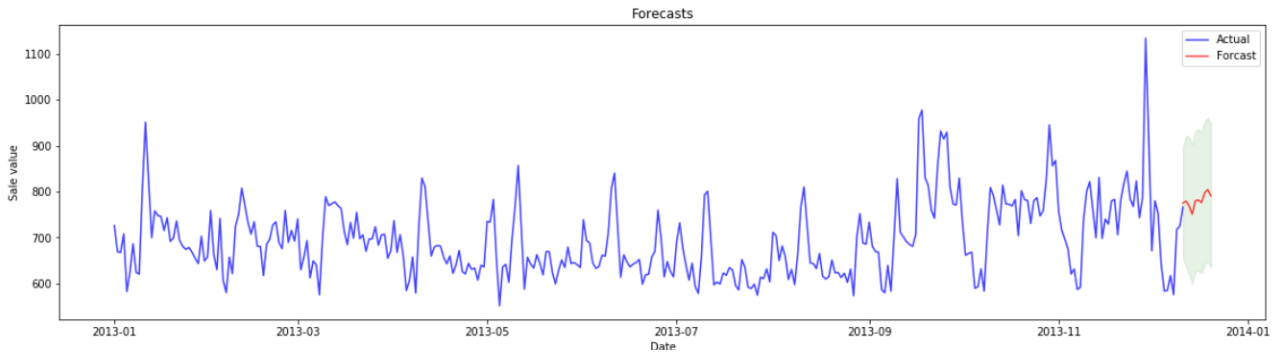
Number of days considered for Forecasting: 10 days

We need to generate Range of dates (for 10 days) using **pandas.date_range()** function with frequency set to **D** (day)

We will predict the forecast values using **model.predict()** with forecasting period (10). This function will return 2 NumPy array.

One is predicted forecasting values; another is an array of minimum and maximum range of values where Forecast can vary.

Next step is to plot the predicted value by concatenating with the actual sales Time Series plotted before and plotting the range of values that forecast can vary and shade that area with Grey colour.



Range of values where Forecast can vary:

Actual sale:

Predicted sale (Forecast):

Predicted Sale amounts:

```
predicted
```

Date	Amount
2013-12-11	775.34
2013-12-12	779.49
2013-12-13	768.98
2013-12-14	751.21
2013-12-15	780.18
2013-12-16	781.70
2013-12-17	776.46
2013-12-18	797.41
2013-12-19	804.59
2013-12-20	791.00

Conclusion:

I have calculated the future sale and successfully plotted the sales forecast using Seasonal ARIMA model.

With this project I have learnt to use Pandas DataFrames, plotting a Time Series data. I have also learnt to create and train seasonal ARIMA model for different frequencies such as yearly, monthly, daily, hourly etc.
Also, I have learnt to plot the graphs by concatenating the 2 separate NumPy array.

Reference:

Brownlee, Jason. "How to Create an ARIMA Model for Time Series Forecasting in Python." *Machine Learning Mastery*, 17 Sept. 2019, machinelearningmastery.com/arima-for-time-series-forecasting-with-python/.

"Pmdarima.arima.auto_arima¶." *Pmdarima.arima.auto_arima - Pmdarima 1.0.0 Documentation*, alkaline-ml.com/pmdarima/1.0.0/modules/generated/pmdarima.arima.auto_arima.html.

Portilla, Jose Marcial. "Using Python and Auto ARIMA to Forecast Seasonal Time Series." *Medium*, Medium, 10 June 2018, medium.com/@josemarcialportilla/using-python-and-auto-arima-to-forecast-seasonal-time-series-90877adff03c.

Prabhakaran, Selva. "ARIMA Model - Complete Guide to Time Series Forecasting in Python: ML+." *Machine Learning Plus*, 13 Oct. 2019, www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/.