

Significance of ACF and PACF Plots In Time Series Analysis



Jayesh Salvi · Follow

Published in Towards Data Science

6 min read · Mar 27, 2019



Listen



Share

... More



This article is for folks who want to know the intuition behind determining the order of auto-regressive (AR) and moving average (MA) series using ACF and PACF plots. Most of us know how to use ACF and PACF plots to obtain the values of p and q to feed into the AR-I-MA model, but we lack the intuition behind why we use PACF and ACF to obtain p and q respectively and not the other way around.

Let's understand what do we mean by ACF and PACF first,

ACF is an (complete) auto-correlation function which gives us values of auto-correlation of any series with its lagged values. We plot these values along with the confidence band and tada! We have an ACF plot. In simple terms, it describes how well the present value of the series is related with its past values. A time series can have components like trend, seasonality, cyclic and residual. ACF considers all these components while finding correlations hence it's a 'complete auto-correlation plot'.

PACF is a partial auto-correlation function. Basically instead of finding correlations of present with lags like ACF, it finds correlation of the residuals (which remains after removing the effects which are already explained by the earlier lag(s)) with the next lag value hence 'partial' and not 'complete' as we remove already found variations before we find the next correlation. So if there is any hidden information

Open in app ↗



Search



can create multicollinearity issues. Hence we need to retain only the relevant features.

Now let's see what is an AR and MA time series process,

Auto regressive (AR) process, a time series is said to be AR when present value of the time series can be obtained using previous values of the same time series i.e the present value is weighted average of its past values. Stock prices and global temperature rise can be thought of as an AR processes.

The AR process of an order p can be written as,

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \dots + \phi_p y_{t-p} + \varepsilon_t$$

Where ε_t is a white noise and y_{t-1} and y_{t-2} are the lags. Order p is the lag value after which PACF plot crosses the upper confidence interval for the first time. These p lags will act as our features while forecasting the AR time series. We cannot use the ACF plot here because it will show good correlations even for the lags which are far in the past. If we consider those many features, we will have multicollinearity issues. This is not a problem with PACF plot as it removes components already explained by earlier lags, so we only get the lags which have the correlation with the residual i.e the component not explained by earlier lags.

In the below code, I have defined a simple AR process and found its order using the PACF plot. We should expect our AR process to show gradual decrease in the ACF plot since being an AR process its present has good correlation with the past lags. We expect PACF to have sharp fall after near lags as these lags near to present can capture the variation so well that we don't need past lags to predict present.

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

import matplotlib
matplotlib.rc('xtick', labels=40)
matplotlib.rc('ytick', labels=40)

import seaborn as sns
sns.set(style="whitegrid", color_codes=True)
from statsmodels.tsa.stattools import acf, pacf

In [2]: t = np.linspace(0, 10, 500)
#normal distributed values
ys = np.random.normal(0,5,500)

In [3]: #exponential series to get the trend
ye = np.exp(t**0.5)
#adding normally distributed series in exponential series
y = ys+ye
#plot
plt.figure(figsize=(16,7))
plt.plot(t,y)
```

AR.ipynb hosted with ❤ by GitHub

[view raw](#)

Now let's discuss the second type of process,

Moving average (MA) process, a process where the present value of series is defined as a linear combination of past errors. We assume the errors to be independently distributed with the normal distribution. The MA process of order q is defined as ,

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \dots + \theta_q \varepsilon_{t-q},$$

Here ε_t is a white noise. To get intuition of MA process let's consider order 1 MA process which will look like,

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1}$$

let's consider y_t as the crude oil price and ε_t is the change in the oil price due to hurricane. Assume that $c=10$ (mean value of crude oil price when there is no hurricane) and $\theta_1=0.5$. Suppose, there is a hurricane today and it was not present yesterday, so y_t will be 15 assuming the change in the oil price due to hurricane as $\varepsilon_t=5$. Tomorrow there is no hurricane so y_t will be 12.5 as $\varepsilon_t=0$ and $\varepsilon_{t-1}=5$. Suppose there is no hurricane day after tomorrow. In that case the oil price would be 10 which means it got stabilized back to mean after getting varied by hurricane. So the effect of hurricane only stays for one lagged value in our case. Hurricane in this case is an independent phenomenon.

Order q of the MA process is obtained from the ACF plot, this is the lag after which ACF crosses the upper confidence interval for the first time. As we know PACF captures correlations of residuals and the time series lags, we might get good correlations for nearest lags as well as for past lags. Why would that be? Since our series is linear combination of the residuals and none of time series own lag can directly explain its present (since its not an AR), which is the essence of PACF plot as it subtracts variations already explained by earlier lags, its kind of PACF losing its power here! On the other hand being a MA process, it doesn't have the seasonal or trend components so the ACF plot will capture the correlations due residual components only. You can also think of it as ACF which is a complete plot (capturing trend, seasonality, cyclic and residual correlations) acting as a partial plot since we don't have trends, seasons, etc.

In the below code, I have defined a simple MA process and found its order using the ACF plot. We can expect the ACF plot to show good correlation with nearest lags and then sharp fall as its not an AR process to have good correlation with past lags. Also we would expect the PACF plot to have gradual decrease as being an MA process, nearest lag values of time series cannot really predict its present value unlike AR process. So, we will get good correlations of residuals with further lags as well, hence the gradual decrease.

```
In [13]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

import matplotlib
matplotlib.rc('xtick', labels=40)
matplotlib.rc('ytick', labels=40)

import seaborn as sns
sns.set(style="whitegrid", color_codes=True)

#ACF and PACF plots:
from statsmodels.tsa.stattools import acf, pacf

In [14]: #xma = np.random.randint(1,50000,100)
xma = np.random.normal(0,25,1000)

In [18]: #Lets create a MA series having mean 2 and of order 2
y5 = 2 + xma + 0.8*np.roll(xma,-1) + 0.6*np.roll(xma,-2) #+ 0.6 *np.ro

In [19]:
```

MA.ipynb hosted with ❤ by GitHub

[view raw](#)

To summarize,

While building a machine learning model we should avoid multicollinear features. The same applies to time series models as well.

We find optimum features or order of the AR process using the PACF plot, as it removes variations explained by earlier lags so we get only the relevant features.

We find optimum features or order of the MA process using the ACF plot, as being an MA process it doesn't have seasonal and trend components so we get only the residual relationship with the lags of time series in the ACF plot. ACF acting as a partial plot.

I hope you liked the article, I have tried to keep it as simple as possible. As this being my first article on Medium I would really appreciate feedback and suggestions to improve myself in the future posts.

Thanks for reading and happy learning! See ya!

Data Science

Timeseries Forecasting

Machine Learning



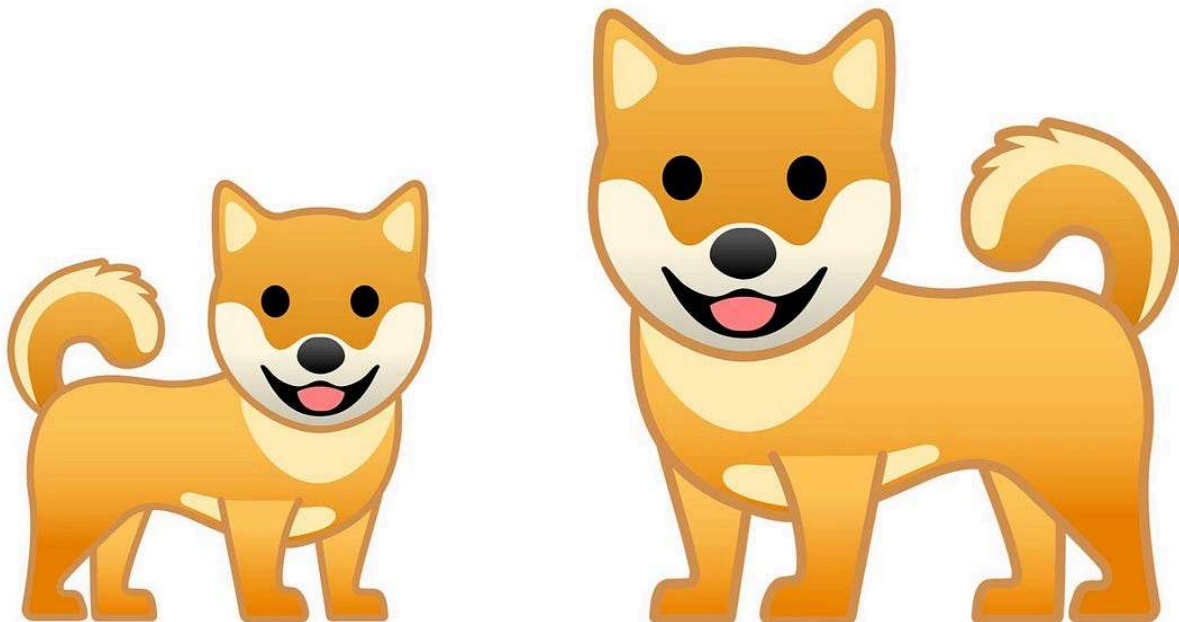
Follow

Written by Jayesh Salvi

164 Followers · Writer for Towards Data Science

I am a data enthusiast with love for travelling and Indian food.

More from Jayesh Salvi and Towards Data Science



Jayesh Salvi in Towards Data Science

Familiarity With Coefficients Of Similarity

When you were doing a project on the recommendation system or the semantic segmentation of the images, you must have come across...

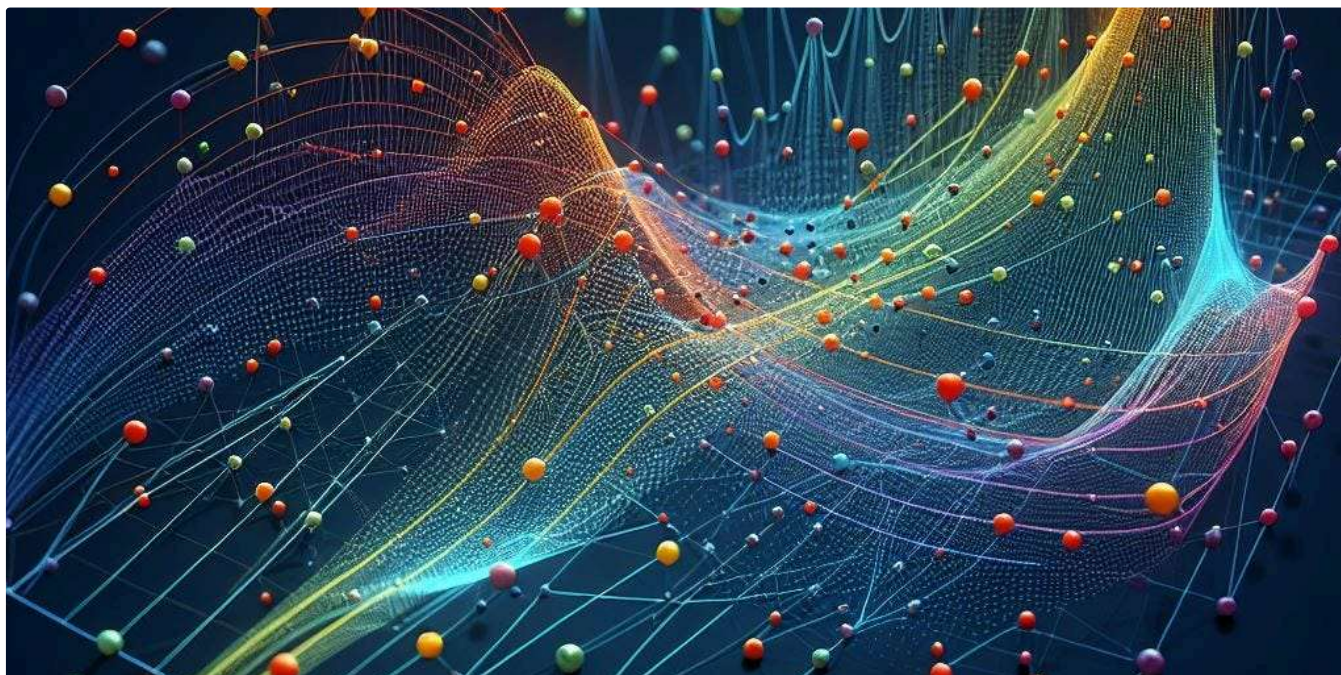
8 min read · Apr 5, 2019



96



1



Tim Sumner in Towards Data Science

A New Coefficient of Correlation

What if you were told there exists a new way to measure the relationship between two variables just like correlation except possibly...

10 min read · Mar 31, 2024



3K



36





 Rami Krispin in Towards Data Science

Setting A Dockerized Python Environment—The Elegant Way

This post provides a step-by-step guide for setting up a Python dockerized development environment with VScode and the Dev Containers...

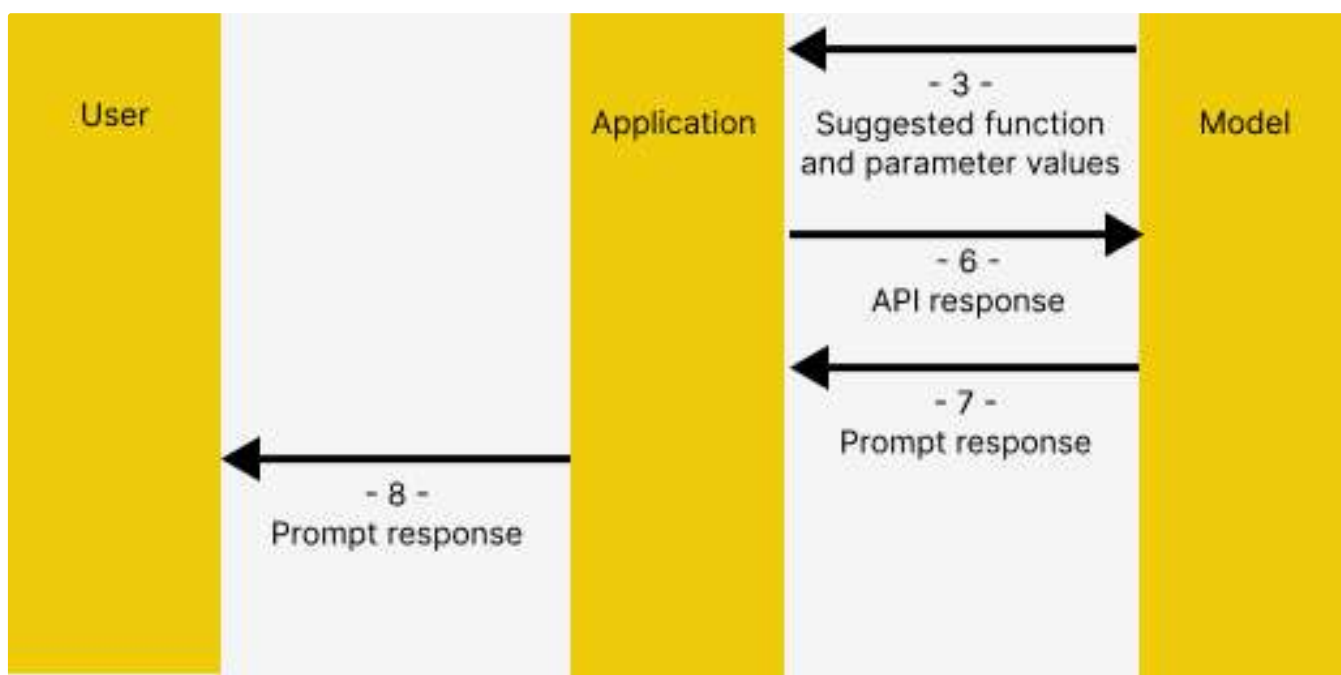
9 min read · Apr 3, 2024



792



1



 Julian Yip in Towards Data Science

Build Autonomous AI Agents with Function Calling

Transform your chatbot into an agent that can interact with external APIs

11 min read · Apr 2, 2024

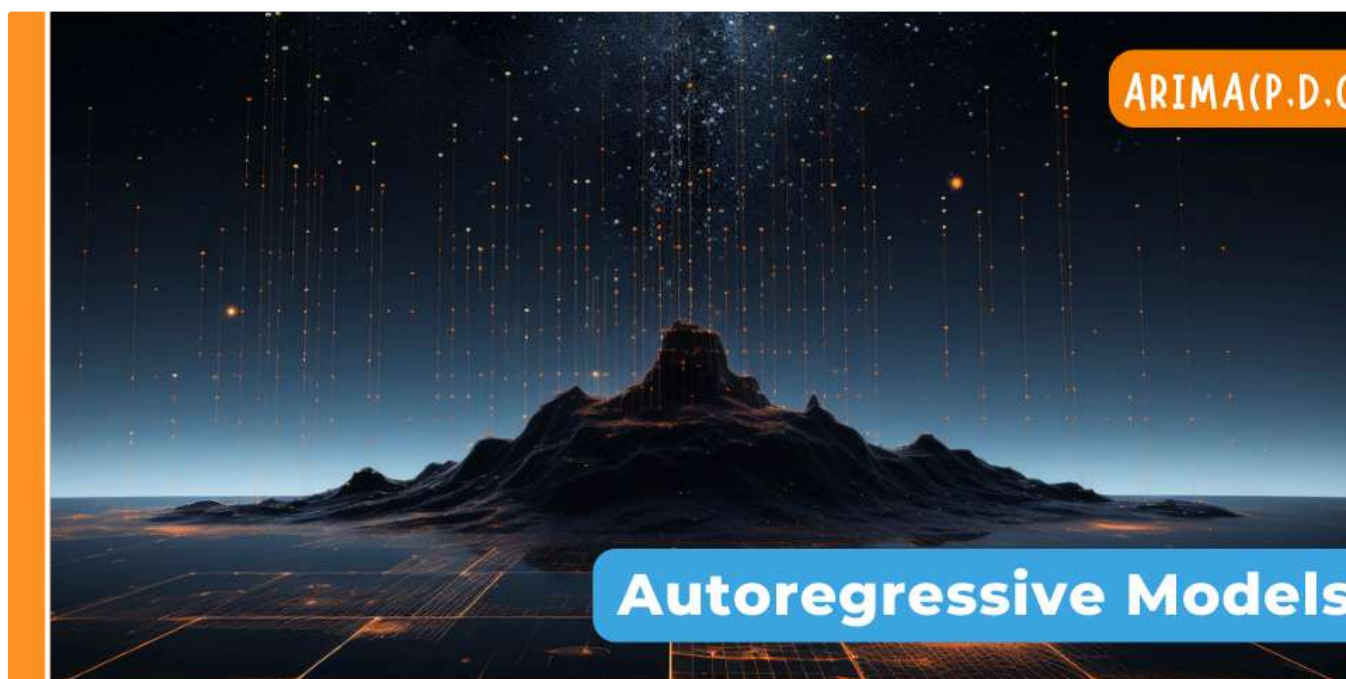
 1K  8



See all from Jayesh Salvi

See all from Towards Data Science

Recommended from Medium



Prof. Frenzel

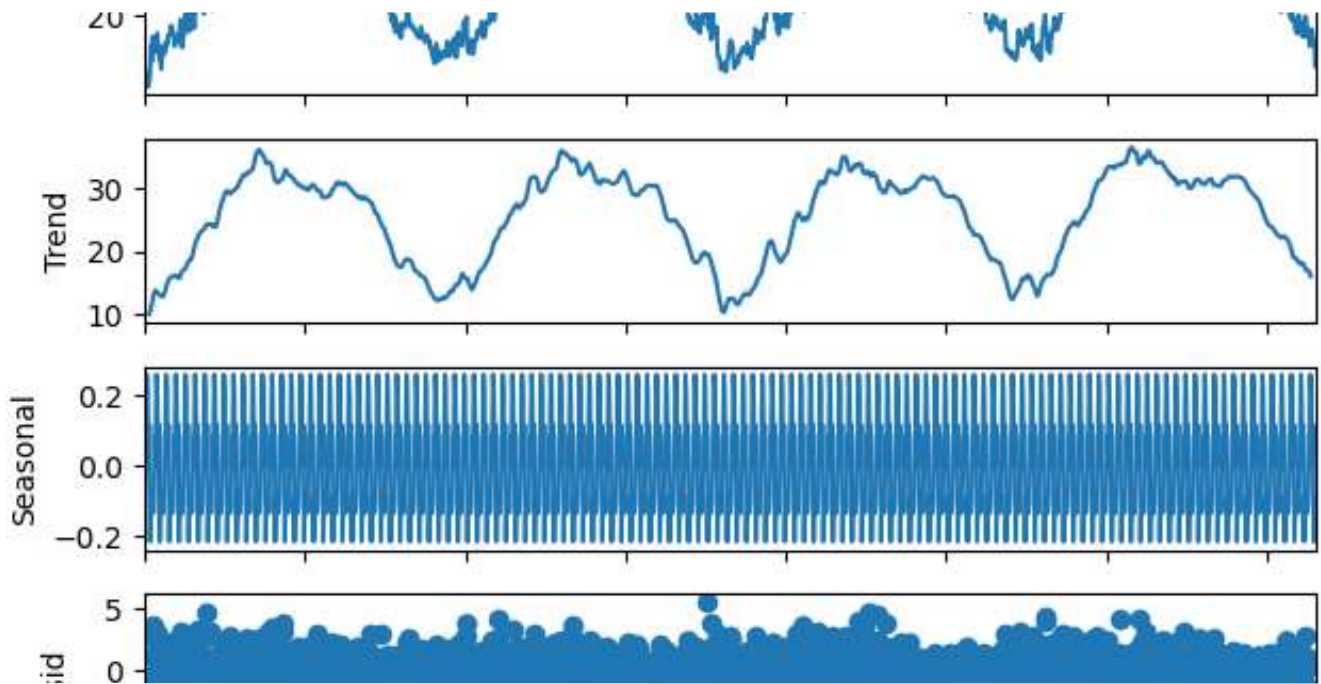
#KB Time-Series Analysis—Part 4—Autoregressive Models

Gear up for an in-depth exploration of Autoregressive Models, the benchmark for advanced time series forecasting.

17 min read · Nov 27, 2023

 11






 Tirtha Mutha

Time Series forecasting using SARIMA in Python

A time series is a series of data points ordered in time. In a time series, time is often the independent variable, and the goal is usually...

5 min read · Nov 10, 2023

 33 

Lists



Predictive Modeling w/ Python

20 stories · 1119 saves



Practical Guides to Machine Learning

10 stories · 1343 saves



Natural Language Processing

1396 stories · 894 saves



data science and AI

40 stories · 136 saves



 İpek Şahbazoglu

Time Series Decomposition and Moving Average Smoothing

In the most general sense time series are a list of numbers, along with information on when those numbers were recorded.

5 min read · Feb 20, 2024



 S. Do. in LatinXinAI

Time series forecasting—ARIMA and SARIMA

Time series forecasting is one of the most useful (and complex) fields of Machine Learning. In this article, second part of the...

7 min read · Dec 11, 2023



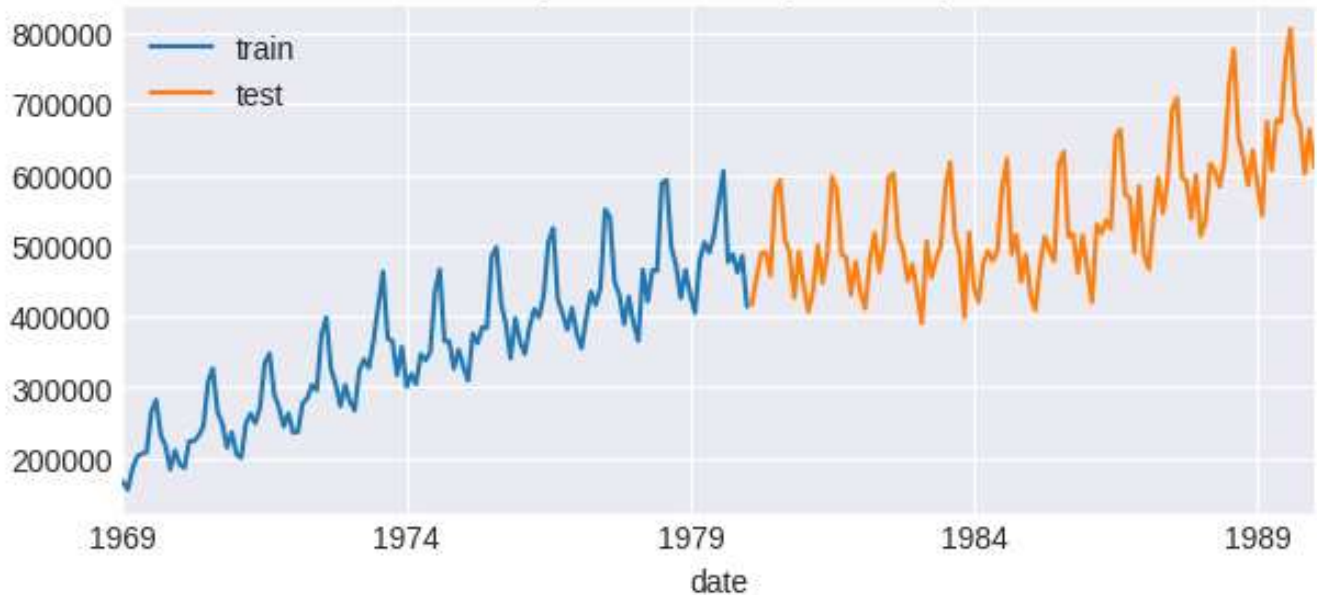
18



1



Monthly Fuel Consumption in Spain



azhar in azhar labs

The Python Forecasting Toolkit: ARIMA and SARIMAX for Time Series Mastery

The world of time series forecasting using ARIMA (AutoRegressive Integrated Moving Average) and SARIMAX (Seasonal AutoRegressive Integrated...

🌟 · 19 min read · Jan 6, 2024

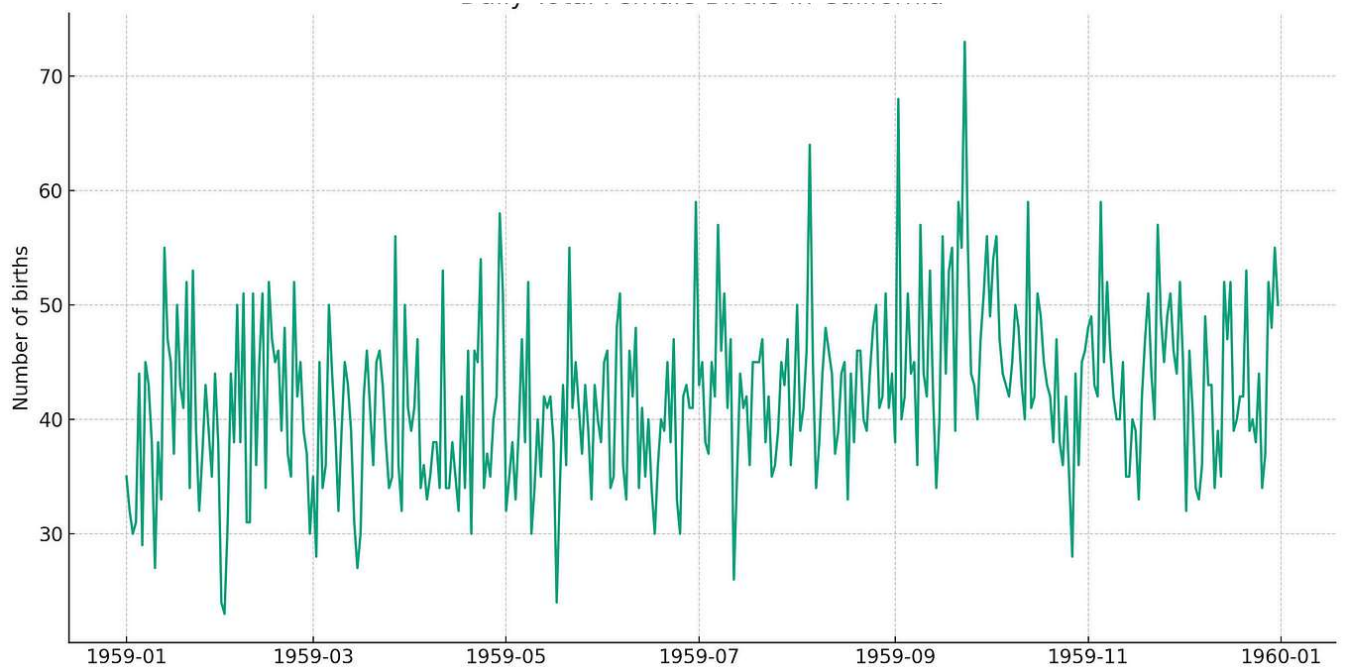


6



2





Divyesh Bhatt in The ML Classroom

Time Series Analysis and Forecasting with ARIMA in Python

Time series forecasting is a crucial area of machine learning that predicts future points in a series based on past data. It is especially...

4 min read · Nov 3, 2023



10



See more recommendations