We at The Data Monk hold the vision to make sure everyone in the IT industry has an equal stand to work in an open domain such as analytics. Analytics is one domain where there is no formal under-graduation degree and which is achievable to anyone and everyone in the World.

We are a team of 30+ mentors who have worked in various product-based companies in India and abroad, and we have come up with this idea to provide study materials directed to help you crack any analytics interview.

Every one of us has been interviewing for at least the last 6 to 8 years for different positions like Data Scientist, Data Analysts, Business Analysts, Product Analysts, Data Engineers, and other senior roles. We understand the gap between having good knowledge and converting an interview to a top product-based company.

Rest assured that if you follow our different mediums like our blog cum questions-answer portal www.TheDataMonk.com , our youtube channel - The Data Monk, and our e-books, then you will have a very strong candidature in whichever interview you participate in.

There are many blogs that provide free study materials or questions on different analytical tools and technologies, but we concentrate mostly on the questions which are asked in an interview. We have a set of 100+ books which are available both on Amazon and on The Data Monk e-shop page

We would recommend you to explore our website, youtube channel, and e-books to understand the type of questions covered in our articles. We went for the question-answer approach both on our website as well as our e-books just because we feel that the best way to go from beginner to advance level is by practicing a lot of questions on the topic.

We have launched a series of 50 e-books on our website on all the popular as well as niche topics. Our range of material ranges from SQL, Python, and Machine Learning algorithms to ANN, CNN, PCA, etc.

We are constantly working on our product and will keep on updating it. It is very necessary to go through all the questions present in this book.

Give a rating to the book on Amazon, do provide your feedback and if you want to help us grow then please subscribe to our Youtube channel.

TIME SERIES FORECASTING USING ARIMA, ARIMAX AND SARIMA

Q1. What is a time series?

A1. A times series is a type of data where in all the data points are in a sequence that is dependent on time. That is, the data is in the formed over some intervals of time. For example, stock market price, or weather forecast.

Q2. What is time series analysis?

A2. Time series analysis is the analysis of time series data to find out if there are any existing trends, seasonality or any other meaningful characteristics to the data.

Q3. How is time series analysis performed?

A3. Time series analysis is performed using different statistical and mathematical methods to find out any relations between one or more variables with time to perform future predictions.

Q4. What is a trend in time series?

A4. A trend in time series refers to the increase or decrease of the values present in the data, that is, there is a positive or a negative slope present in general over a long duration of time.

Q5. What is a seasonality in time series?

A5. Seasonality in time series refers to a specific time interval over which the data experiences the same changes every time. Which means that a particular pattern is observed in the time series at some regular time periods. Example: A stock price that experiences a rapid increase in its price every year in the month of June.

Q6. What is stationarity in time series?

A6. Stationarity in time series refers to a time series where the mean and variance of the data do not change over time. Note that this does not mean the data itself does not change, it doesn't even mean that the data changes itself in a uniform way. It simply means that the change in data is such that the statistical indicators remain the same.

Q7. What are the conditions for a time series to be stationary?

A7. The three conditions that are required for a time series to be labelled as stationary are:

1. The mean of the time series is constant, that is the mean of the data does not change over time.
2. The variance of the time series is constant, that is the variance of the data does not change over time.
3. There is no seasonality observed in the time series.

Q8. What is white noise?

A8. White noise in time series is data that cannot be predicted at all with data points that have no correlation between any two data points whatsoever. The conditions for a time series to be defined as data points are:

1. The mean of the time series is constant and is zero throughout the series.
2. The variance of the time series is constant, that is the variance of the data does not change over time.
3. There is no seasonality observed in the time series, that means there is no correlation between any two data points.

Q9. Is white noise a type of stationary time series?

A9. Yes, white noise is a type of stationary time series with the mean always equal to zero. All white noise time series are stationary but all stationary time series aren't white noise.

Q10. What is correlation?

A10. Correlation as the word suggests is basically the correlation between two variables, that is, it tells us about the relationship or the dependence of one variable with another. For example, if on the increase in value of one variable the other one increases as well, that will be a positive correlation. Similarly, if for the increase in value of one variable the other decreases, that will be a negative correlation.

Q11. How is correlation calculated?

A11. There are many different formulas for calculating correlation, but the most used one is the "Pearson Correlation Coefficient". It is calculated as follows:

$$r = \frac{\Sigma(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\Sigma(x_i - \bar{x})^2(y_i - \bar{y})^2}}$$

Where,

r is the correlation coefficient

x is the value of the x variable in data

x bar is the mean of the values of x in data

y is the value of the y variable in data

y bar is the mean of the values of y in data

Here,

If r = 1, that means that there is a perfect positive correlation between x and y.

If r = -1, that means that there is a perfect negative correlation between x and y.

If r = 0, that means that there is absolutely no correlation between x and y.

Q12. What does ACF stand for?

A12. ACF stands for "Autocorrelation Function".

Q13. What is Autocorrelation Function?

A13. Autocorrelation is the correlation between a time series with a lagged version of the same time series. Let's say we have a time series T and denote the value of the time series at time interval 'i' as T_i . So in autocorrelation we take the correlation of the data point T_i with the points T_{i-1}, T_{i-2}, T_{i-3} and so on for every 'i' present in the dataset.

The correlation we are talking about here is the normal Pearson correlation where in we calculate the correlation using the formula give in the previous question for every pair T_i - T_{i-1}, T_i - T_{i-2} and so on.

Q14. What does the Autocorrelation function signify?

A14. Autocorrelation shows the direct and indirect dependence of the observed variable on the lagged observation. For example, if we are considering the autocorrelation between two variables T_i and T_{i-2}, autocorrelation will consider the direct correlation of the variable T_{i-2} and T_i, but also the indirect correlation that T_{i-2} has on T_{i-1}, which will in turn have a correlation with T_i.

Q15. What does PACF stand for?

A15. PACF stands for "Partial Autocorrelation Function".

Q16. What is Partial Autocorrelation Function?

A16. Partial autocorrelation function also calculates the correlation between the data points in a time series, but in a different way. As we saw in the previous questions, Autocorrelation for the variables T_i and T_{i-2} includes the direct correlation of T_i with T_{i-2} and also the indirect relation of T_{i-2} with T_i through T_{i-1}.

Generally speaking, for any two variables T_i and T_{i-k}, autocorrelation will include all the intermediate correlations of T_{i-k} through T_i.

But in the partial autocorrelation function, we only consider the direct correlation of T_i with T_{i-k}, eliminating all the other in between intermediate indirect correlations.

Q17. How is Partial Autocorrelation calculated?

A17. To calculate Partial Autocorrelation, we need to use a different and more complex method. Let's say we need the partial correlation between the variables T_i and T_{i-2}. We will take a regression function as follows:

$$T_i = x_2 * T_{i-2} + x_2 * T_{i-1} + €$$

And the coefficient 'x_2' is what will be the PACF for T_{i-2}. Similarly, for finding out the PACF for T_i and T_{i-k}, we will take a regression function of k+1 terms and the coefficient of T_{i-k} will be the value of our PACF function.

Q18. What does the Partial Autocorrelation Function signify?

A18. The Partial Autocorrelation Function tells us about the correlation between the two variables only, eliminating all in between correlations. It signifies only the direct dependence of the variables and nothing else.

Q19. What is a persistence model in time series?

A19. The persistence model in time series is the simplest model that can be used to perform predictions on a sequence data.

Q20. How does a persistence model work?

A20. A persistence model has a very simple workflow. It simply uses the last observation to predict the next value. That is, it uses the value of the previous time step to predict the value of the next time step. So for any time step 't', a persistence model will use the value at time step 't-1' to predict the value at time step 't+1'.

Q21. Why is a persistence model used?

A21. A persistence model is usually used to set a baseline performance that can be compared to other models that we will use for the same problem statement.

Q22. What is regression?

A22. Regression is a method or a function that is used to find a relationship between two or more variables given from a dataset. Usually, you would have a dependent variable that is to be predicted using some independent variables.

For example, predicting the price of a house given its area, number of rooms, how old it is etc.

Q23. What are different types of regression?

A23. The 5 basic types of regression are:

1. Linear regression
2. Logistic regression
3. Ridge regression
4. Lasso regression
5. Polynomial regression

Q24. How does regression work?

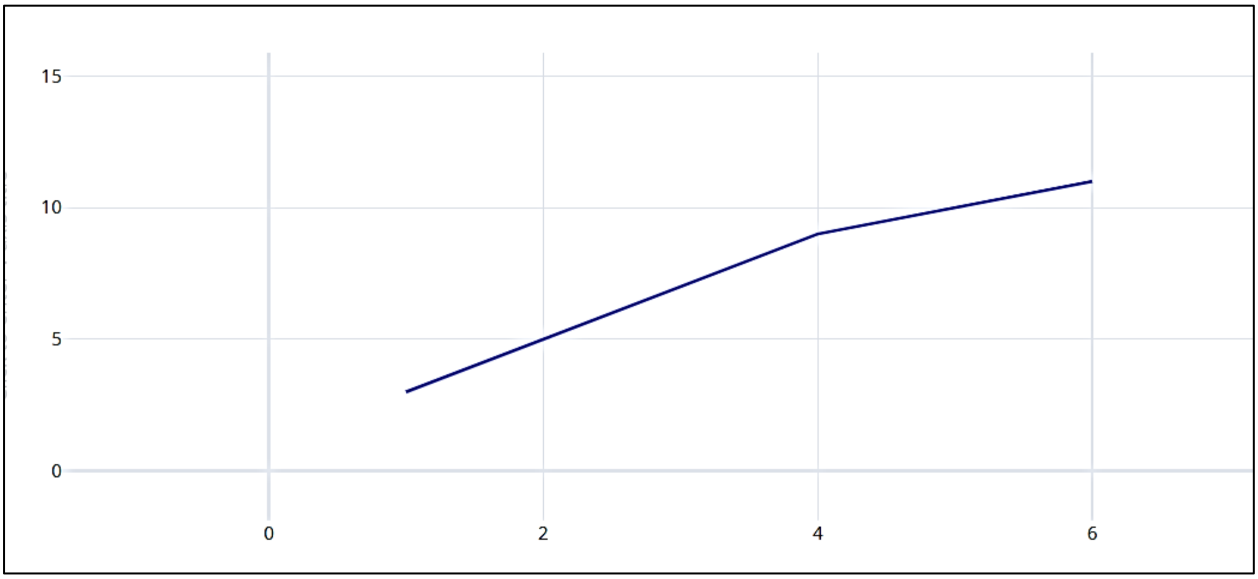
A24. To understand how regression works, let us take a simple problem statement where we have to predict the price of a house given the area it covers using linear regression.

Let’s denote the price of the house by ‘y’ and the area it covers by ‘x’.

Let’s say we have the following data:

Price of house (y)	Area it covers (x)
3	1
5	2
7	3
9	4
11	6

If we were to plot the given data on a graph, it would look something like this:



As you can see in the graph above, the data points form almost a straight line when connected together.

Now to apply linear regression, let's take a function like this:

$$y = \alpha * x + \beta$$

As you can see, the equation above is that of a straight line with some slope 'α' and intercept 'β'. Our objective is to find the correct values for the parameters α and β such that the line that it will form will be closest to the straight line represented by our data.

The way we do that is by training the model to fit to our given dataset by minimizing a loss function that will minimize the error between our line and the line represented by the dataset.

Q25. What is Auto-Regression?

A25. Auto regression is a kind of regression used in time series forecasting where you use a regression model to predict the value of a future time step using the values of previous time steps as inputs.

That means in this case, our dependent variable is the current time step T_i, and our inputs or our independent variables will be the values of the previous time steps T_{i-1}, T_{i-2}, T_{i-3} and so on. This is what the equation will look like:

$$T_i = \alpha_1 * T_{i-1} + \alpha_2 * T_{i-2} + \alpha_3 + T_{i-3} + \ldots$$

After training the above equation to fit on the dataset, we will obtain the optimal values for the parameters α_1, α_2, α_3 and so on, and we can use that model to perform time series forecasting then.

Q26. How is ACF and PACF used in Auto regression in time series?

A26. The auto correlation function or the partial auto correlation function can be used in Auto regression to choose which previous time steps should be used as inputs for the regression function. Sometimes, by using all the previous time steps as input can result in the model overfitting the data, that means that model won't generalize or make good predictions over other datasets than the training dataset.

So, we can use ACF and PACF to choose which time steps have the highest correlation with the current time step. That means we have to only consider the values that actually contribute to the value of the current time step, and eliminate the ones because the value of the current time step does not depend on those values.

Q27. What will happen if you don't use ACF or PACF to filter the parameters to be used in the auto regression model?

A27. These are the advantages of using ACF or PACF to select the input parameters for your model:

1. Only relevant time steps are used, values that don't have any contributing factor towards predictions are eliminated.
2. Model size is reduced due to decrease in number of parameters.
3. Overfitting is prevented and the model can generalize much easily on other datasets.

Q28. What is a moving average?

A28. A moving average is simply the average of a dataset observed over regular successive time intervals. Suppose for a time series with a 100 data points, you could have a moving average over 5 data points, that is first you will take the average of the first 5 points 1-5, then the average of the points 2-6, then 3-7 and so on and plot all of these averages right up to 96-10. This will be known as the moving average. You can have the intervals of any length.

Q29. What is the use of using moving average on a time series?

A29. By using moving average on a time series, you can smooth out the time series graph up to a certain extent. Using moving averages also helps to eliminate random small sized fluctuations that will help any model that you train to fit the data much better.

Q30. What is an exponential moving average (EMA)?

A30. Exponential moving average is a type of moving average where the recent datapoints are given more importance than the older datapoints. It uses the exponential function to give more weight to the recent time steps.

Q31. What is a moving average model in time series forecasting?

A31. A moving average model is a model that is kind of a like a regression model, but instead of using the values of the previous time steps as it is, a moving average model instead uses the average of the dataset and the error values from the previous time steps predictions to forecast the values at the next time step.

Let us understand this better with an example. Let's say you have to use a moving average model to perform predictions for weather forecast. Let's say that the average temperature you have observed to be is 25 degrees Celsius.

This is what a moving average model equation of degree one will look like:

$$T_i = \mu + \alpha * €_{i-1} + €_i$$

$$T'_i = \mu + \alpha * €_{i-1}$$

Where

T_i is the weather at the current time step

T'_I is the predicted value of the current time step

μ is the average value of the weather (in this case = 25)

α is the weight parameter (which is let's say = 0.5 in this case)

$€_{i-1}$ is the error value of the previous time step (= T_i – T_{i-1})

So, let's say at the first time step you predict the value 25 and the actual value is 23. So, the error at this step will be:

$$€_1 = T_1 - T_0$$

$$€_1 = -2$$

So for the prediction at second value, by using the equation

$$T'_2 = \mu + \alpha * €_1$$

We get:

$$T'_2 = 25 + 0.5 * -2$$

$$= 24$$

Now let's say at the second time step, the actual value was 28. So for the third time step:

$$€_2 = T_2 - T_1$$

$$€_1 = 4$$

$$T'_2 = 25 + 0.5 * 4$$

$$= 27$$

And this way, we can keep performing predictions on the time series. This is how a basic moving average model works.

Note: The above example was of a basic moving average model of order one. You can have higher order moving average models that consider the errors of more time steps as follows:

$$T'_i = \mu + \alpha_1 * €_{i-1} + \alpha_2 * €_{i-2} + \alpha_3 * €_{i-3} + ...$$

Q32. What does 'ARMA' stand for?

A32. ARMA stands for 'Auto Regressive Moving Average'.

Q33. What is an Auto Regressive Moving Average (ARMA) model?

A33. An Auto Regressive Moving Average or ARMA model is simply the combination of the Auto Regressive (AR) model and the Moving Average (MA) model.

Q34. What is the notation used for an ARMA model?

A34. ARMA models are usually written as follows:

ARMA(p,q)

Where 'p' is the order of the Auto Regression model and 'q' is the order of the Moving Average model.

Q35. How does an ARMA model work?

A35. An ARMA model can be formed simply by concatenating the equations of the Auto Regressive model and the Moving Average model.

So, for an ARMA (1,1) model:

Auto Regressive model (AR):

$$T_i = \alpha_1 * T_{i-1} + K$$

Moving Average model (MA):

$$T'_i = \mu + \alpha * €_{i-1}$$

Auto Regressive Moving Average model (ARMA):

$$T_i = \alpha_1 * T_{i-1} + \beta_1 * €_{i-1} + K$$

For ARMA models of higher orders, the terms in the equation will be adjusted accordingly.

For example, for an ARMA(2,3) model, this will be what the equation will look like:

$$T_i = \alpha_1 * T_{i-1} + \alpha_2 * T_{i-2} + \beta_1 * €_{i-1} + \beta_2 * €_{i-2} + \beta_3 * €_{i-3} + K$$

Q36. What does ARIMA stand for?

A36. ARIMA stands for "Auto Regressive Integrated Moving Average".

Q37. What is the notation used for ARIMA models?

A37. ARIMA models are usually written in the following way:

ARIMA(p,d,q)

Where

'p' is the order of the Auto Regression model

'q' is the order of the Moving Average model

'd' is the order of differencing

Q38. When is an ARIMA model usually used in time series forecasting?

A38. An ARIMA model is usually used in time series forecasting when the given time series is not stationary. What that means is that the given time series does not have either a constant mean or a constant standard deviation.

When a given time series isn't stationary, we can't use ARMA models as they don't perform well. In this case, we can use the ARIMA model.

Q39. How does an ARIMA model work?

A39. As the name suggests, the ARIMA model consists of both the Auto Regressive model and the Moving average model, much like the ARMA models. Let's understand what the Integrated part means.

As discussed in the previous question, we use ARIMA when the time series isn't stationary. The working of the ARIMA model is very similar to the ARMA model. In fact, the equation is exactly the same as the ARMA model.

But in ARIMA, instead of using the exact value of the variable as it is, we take the differences between the values as the variable to be predicted.

Let us take the example of weather forecast. Let's say you have the following data:

Day (x)	Temperature (t)
1	25
2	28
3	24
4	27
5	26
6	31
7	28

Now for implementing an ARMA model where d=1, it means the difference is to be taken once.

So, we will generate a new series where we take the differences between two consecutive values. Let's call this series D where

$$d_i = t_{i+1} - t_i$$

x	D
1	3
2	4
3	3
4	1
5	5
6	-3

Now we will implement an ARMA model where instead of the value 't' we will be using the variable 'd':

$$d_i = \alpha_1 * d_{i-1} + \beta_1 * €_{i-1} + K$$

And for predicting the value of 't', we will simply use the equation mentioned before to transform the value back:

$$d_i = t_{i+1} - t_i$$

Therefore,

$$t_i = d_{i-1} + t_{i-1}$$

Note, you can substitute t_{i-1} with $d_{i-2} + t_{i-2}$

Therefore,

$$t_i = d_{i-1} + d_{i-2} + t_{i-2}$$

If you keep on substituting, you will end up with the following equation:

$$t_i = \sum_{i=1}^{K} d_{K-i} + t_K$$

Note: In case of change in values of ‘p’ and ‘q’, the equation will change accordingly as it did in ARMA.

In the case of change in value in ‘d’, we will simply take the difference that many times.

For example, if d = 2, we will create a new series that has the differences of the series ‘D’, let’s say the new series is called ‘E’.

If d = 3, we will create another series ‘F’ that will have the differences of successive values of the series ‘E’.

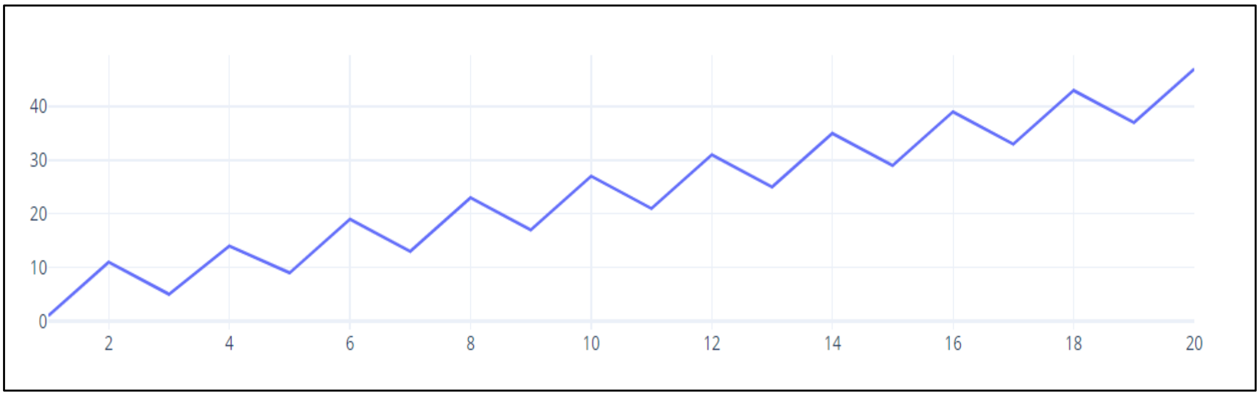
And accordingly, we will keep applying reverse transforms on every level to get the value of the original variable ‘t’.

Usually, one or at most two differences are enough to get a decent performance on the time series.

Q39. What is the use of using the differences in ARIMA model?

A39. The use of using differences in ARIMA is that it helps to make the time series stationary somewhat. Let us understand this better with an example.

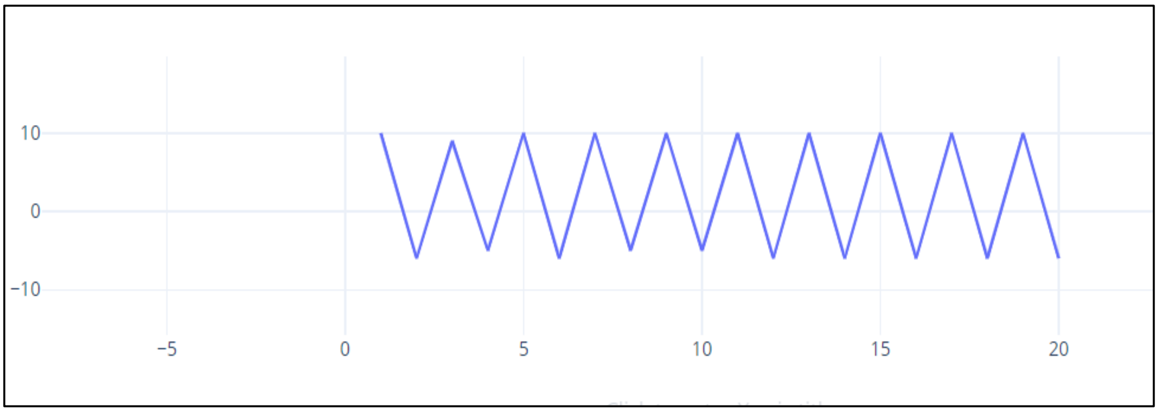
Let’s say you have a time series which when plotted looks like the graph shown below:



As you can see, the graph above has a pretty constant standard deviation. It does not show any seasonality too. But it definitely does not have a constant mean. The mean seems to be gradually increasing with time.

Clearly, the graph isn't stationary. But one thing that is visible is that the graph seems to be linear in nature, that is the mean of the graph seems to be increasing linearly at a constant rate slowly. What that means basically that the mean seems to have a constant gradient.

So, when we take the differences of every two consecutive values and plot them, this is what it might look like:



As you can see, this series has a constant mean as well as a constant variance. This is a stationary series. We can now implement ARMA models for this series which will perform very well.

This is why we use differences in ARIMA model so that the model works on non-stationary series.

Q40. What are some variations of the ARIMA model?

A40. Some variations of the ARIMA model are the “SARIMA” model and the “ARIMAX” model.

Q41. What does ‘SARIMA’ stand for?

A41. SARIMA stands for “Seasonal Auto Regressive Integrated Moving Average”.

Q42. When is the “SARIMA” model used in time series forecasting?

A42. The SARIMA model is used in time series forecasting when you would usually use an ARIMA model but the time series given has some seasonality to it. In this case you would use the SARIMA model instead as it works better on seasonal time series data.

Q43. What is the notation used for SARIMA?

A43. This is how we usually denote SARIMA models:

$$\text{SARIMA } (p,d,q)\ (P,D,Q)_m$$

Where

‘p’ is the order of the Auto Regression model

‘q’ is the order of the Moving Average model

‘d’ is the order of differencing

‘P’ is the seasonal order of the Auto Regression model

‘Q’ is the seasonal order of the Moving Average model

‘D’ is the seasonal order of differencing

‘m’ is the seasonality factor

Q45. What does the seasonality factor ‘m’ in the SARIMA model signify?

A45. The seasonality factor ‘m’ in the SARIMA model is basically the number of time steps required for the seasonality to repeat. It simply means how many more time steps at a current time step for the time series to show the same cycle again.

Q46. How does a SARIMA model work?

A46. The SARIMA model works exactly like the ARIMA model, just along with the normal orders, the model also considers the lagged observations to factor in seasonality.

For example, if you have a time series where the seasonal effect occurs every year, we will set m = 12 for 12 months so that the SARIMA model at every time step will use the observations from 12 months before to perform forecasting.

Q47. What does ARIMAX stand for?

A47. ARIMAX stands for "Auto Regressive Integrated Moving Average with Explanatory Variable".

Q48. What are two types of time series classified according to the number of variables?

A48. The two types of time series classified according to the number of variables are:

1. Univariate time series: In a univariate time series, there is only one variable that is observed that changes with time.
2. Multivariate time series: In a multivariate time series, there are more than one variables observed that are changing with time.

Q49. When is an ARIMAX model used in time series forecasting?

A49. The ARIMAX model is used in time series forecasting when the time series given is not univariate but multivariate. Which means that it has more than one variables that might factor in the on deciding the value of the dependent variable.

Q50. How does an ARIMAX model work?

A50. An ARIMAX model works the same way as an ARIMA model, just with the addition of another variable in the equation called the Exogenous variable:

$$d_i = \alpha_1 * d_{i-1} + \beta_1 * €_{i-1} + K + \gamma * X$$

Where 'X' is the new Exogenous variable.

X could be anything like a technical indicator or a statistic value that could in any way influence the value to be predicted.

CODE

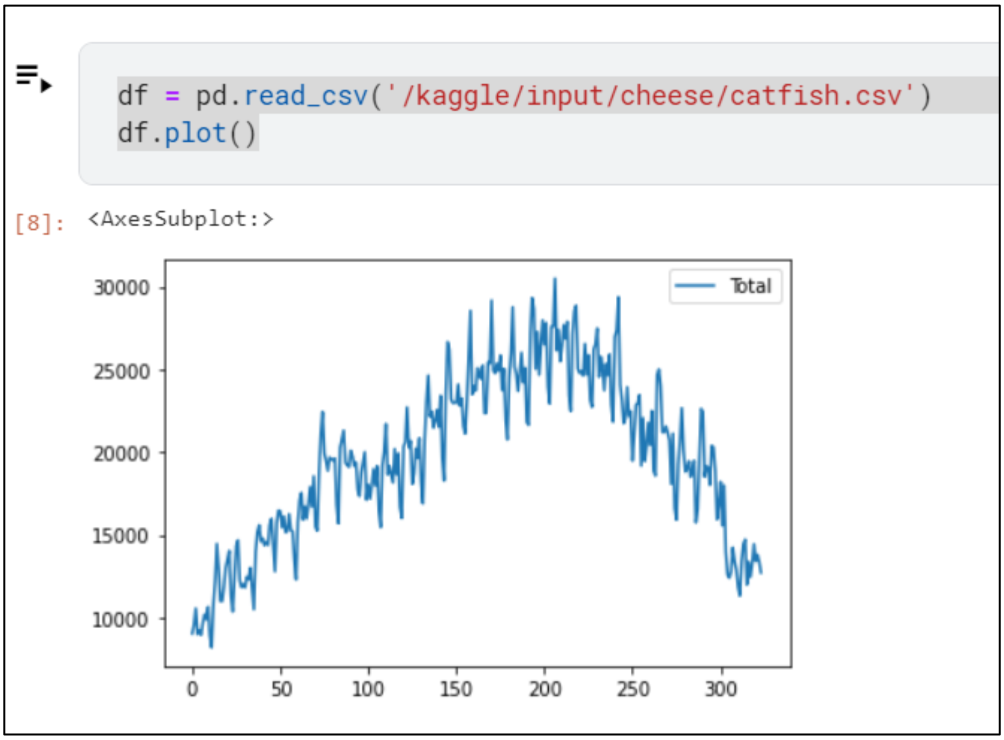
First, let us import the required libraries:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.stattools import acf, pacf
from statsmodels.tsa.arima_model import ARIMA
```

For this example, I am using a dataset on Kaggle, let us import it:

```
df = pd.read_csv('/kaggle/input/cheese/catfish.csv')
df.plot()
```

This is what the output graph after the code above is executed looks like:



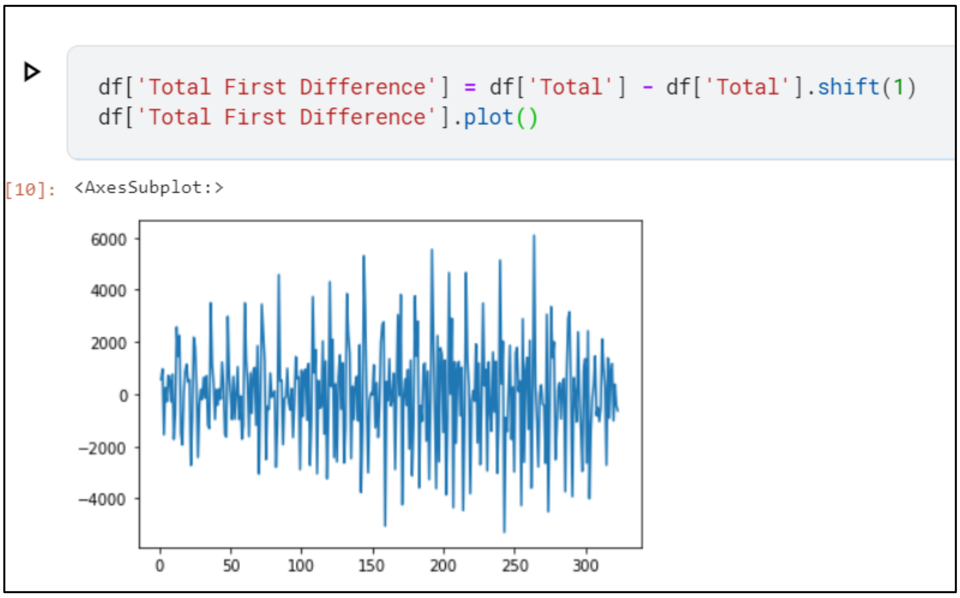
As you can see, this time series is definitely not stationary as the mean can be seen first increasing somewhat linearly and then after a point decreasing again.

Since the graph is not stationary, we cannot straight up apply ARMA. Let us see if ARIMA should work well by taking the differences once.

Let us look at the first difference graph and what it looks like:

```
df['Total First Difference'] = df['Total'] - df['Total'].shift(1)

df['Total First Difference'].plot()
```

This is what the output graph after the code above is executed looks like:



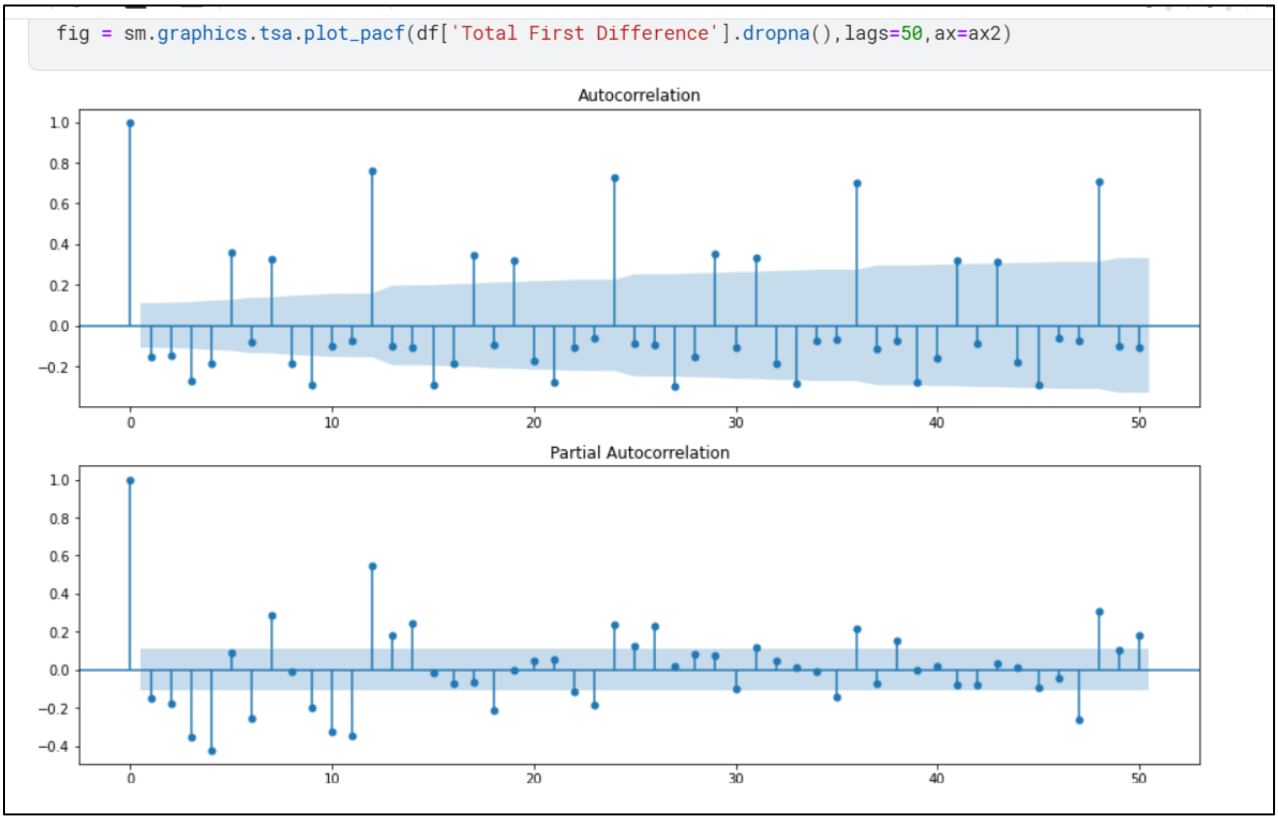
As you can see now, after applying the differences once, the series seems to have a pretty constant mean throughout the series. Now the series looks stationary, and we can apply ARMA to it.

Let us now look at the autocorrelation function plot and the partial autocorrelation function plots now:

```
from statsmodels.graphics.tsaplots import plot_acf,plot_pacf

import statsmodels.api as sm

fig = plt.figure(figsize=(15,10))

ax1 = fig.add_subplot(211)
```

```
fig = sm.graphics.tsa.plot_acf(df['Total First Difference'].dropna(),lags=50,ax=ax1)

ax2 = fig.add_subplot(212)

fig = sm.graphics.tsa.plot_pacf(df['Total First Difference'].dropna(),lags=50,ax=ax2)
```

This is what the output graphs after the code above is executed looks like:



Let us now finally use the ARIMA model!

Let's start with using a simple ARIMA (1,1,1) model:

```
from statsmodels.tsa.arima_model import ARIMA

model=ARIMA(df['Total'],order=(1,1,1))

model_fit=model.fit()

model_fit.summary()
```

This is what the model summary looks like:

ARIMA Model Results			
Dep. Variable:	D.Total	**No. Observations:**	323
Model:	ARIMA(1, 1, 1)	**Log Likelihood**	-2873.468
Method:	css-mle	**S.D. of innovations**	1765.871
Date:	Mon, 28 Jun 2021	**AIC**	5754.937
Time:	14:59:52	**BIC**	5770.047
Sample:	1	**HQIC**	5760.969

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
const	11.3998	27.435	0.416	0.678	-42.372	65.172
ar.L1.D.Total	0.4187	0.064	6.517	0.000	0.293	0.545
ma.L1.D.Total	-0.8400	0.031	-26.748	0.000	-0.902	-0.778

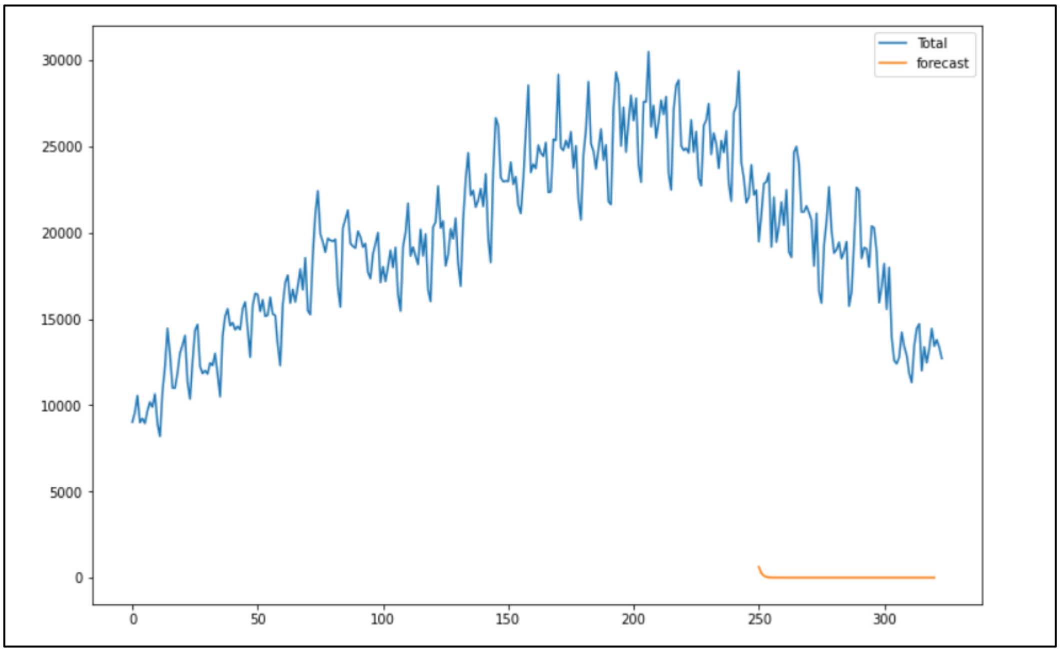
Roots

	Real	**Imaginary**	**Modulus**	**Frequency**
AR.1	2.3886	+0.0000j	2.3886	0.0000
MA.1	1.1905	+0.0000j	1.1905	0.0000

Now let us look at the performance of our model!

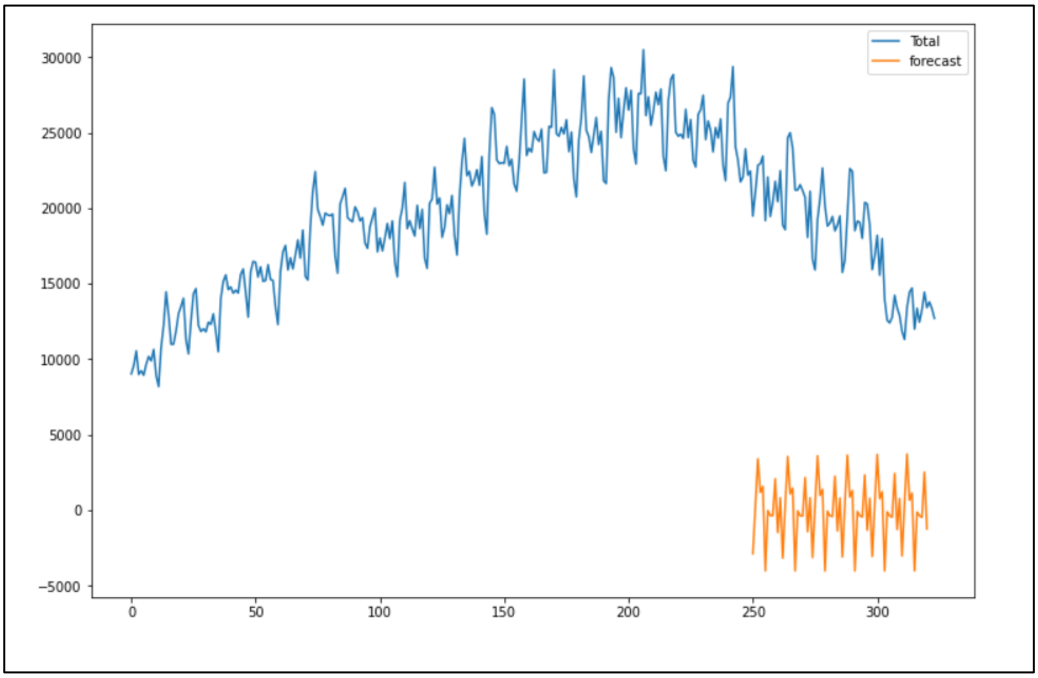
```
df['forecast']=model_fit.predict(start=250,end=320,dynamic=True)
df[['Total','forecast']].plot(figsize=(12,8))
```

This is what the output graphs after the code above is executed looks like:



That looks like a really poor performance! Let us try increasing the model parameters to ARIMA(10,1,10)

This is what the performance looks like for an ARIMA(10,1,10):



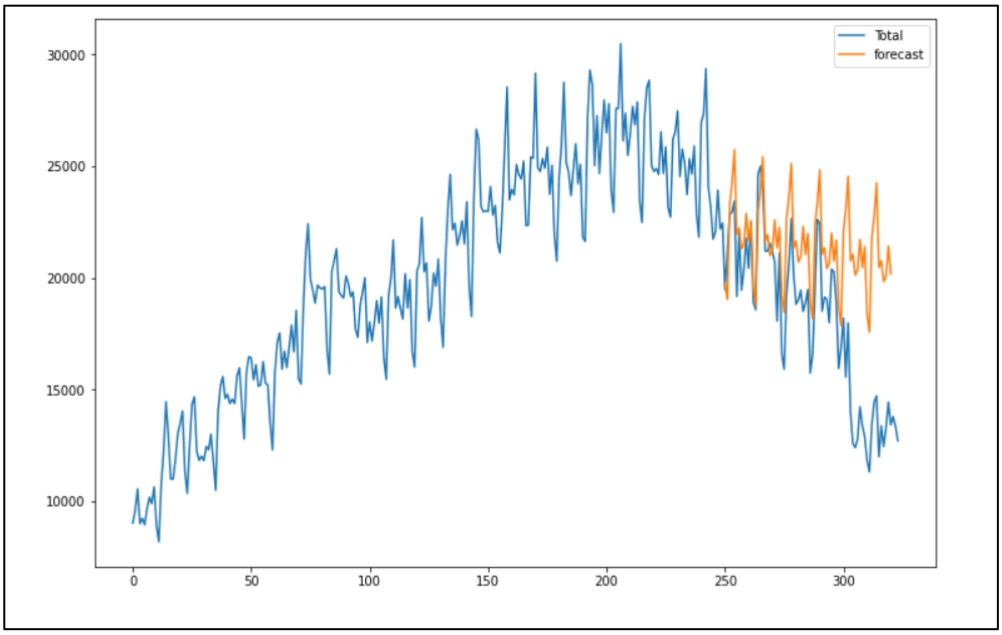
Still not a good performance, but somewhat of an improvement.

The problem here if you can see it is that the time series given shows some seasonality to it. This means that a SARIMAX model should work much better.

Let's find out!

```
import statsmodels.api as sm

model=sm.tsa.statespace.SARIMAX(df['Total'],order=(1, 1,
1),seasonal_order=(1,1,1,12))

results=model.fit()

df['forecast']=results.predict(start=250,end=320,dynamic=True)

df[['Total','forecast']].plot(figsize=(12,8))
```

This is what the output graphs after the code above is executed looks like:



Not a perfect accuracy, but a much better performance than an ARIMA model. With some more tweaking and hyper parameter optimization, we can get this model to give a much better performance.