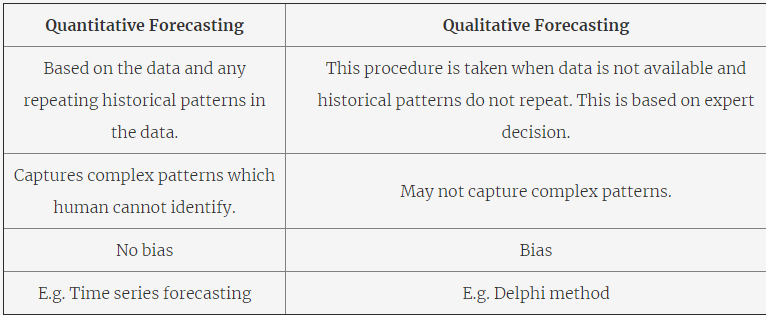
**Timeseries forecasting Notes**:



The one thing to keep in mind before moving forward is that there are some caveats associated with a time series forecasting. These caveats revolve around the steps you learnt about while defining the problem.

1. **The Granularity Rule:**The more aggregate your forecasts, the more accurate you are in your predictions simply because aggregated data has lesser variance and hence, lesser noise. As a thought experiment, suppose you work at Netflix and you want to predict the number of views for a few newly launched TV shows in Mumbai for the next one year. Now, would you be more accurate in your predictions if you predicted at the city-level or if you go at an area-level? Obviously, accurately predicting the views from each area might be difficult but when you sum up the number of views for each area and present your final predictions at a city-level, your predictions might be surprisingly accurate. This is because, for some areas, you might have predicted lower views than the actual whereas, for some, the number of predicted views might be higher. And when you sum all of these up, the uncalled noise and variance cancel each other out, leaving you with a good prediction. Hence, don't try to make predictions at very granular levels.
2. **The Frequency Rule:**This rule tells you to keep updating your forecasts regularly to capture any new information that comes in. Let's take the Netflix example taken above where the problem is to predict the number of views in Mumbai for the next one year. Now, if you keep the frequency too low, you might not be able to capture the new information coming in. For example, say, your frequency for updating the forecasts is 3 months. Now, due to the COVID-19 pandemic, the residents may be locked in their homes for around 2-3 months during which the number of views will significantly increase. Now, if the frequency of your forecast is 3 months, you will not be able to capture this increase in views which may incur significant losses and lead to mismanagement.
3. **The Horizon Rule:**When you have the horizon planned for a large number of months, you are more likely to be accurate in the earlier months as compared to the later ones. Let's again go back the to Netflix example. Suppose that Netflix made a prediction for the number of views for the next 6 months in December 2019. Now, it may have been quite accurate for the first two months, but due to the unforeseen COVID-19 situation, the actual number of view in the next couple of months would have been significantly higher than predicted because of everyone staying at home. The more far ahead we go into the future, the more uncertain we are about it.

Now that you have understood the steps in defining the problem, let’s apply them to the air passenger traffic problem.

1. **Quantity:** Number of passengers
2. **Granularity:** Flights from city A to city B; i.e., flights for a particular route
3. **Frequency:** Monthly
4. **Horizon:** 1 year (12 months)

There are three important characteristics that every time series data must exhibit in order for us to make a good forecast.

1. **Relevant:** The time-series data should be relevant for the set objective that we want to achieve.
2. **Accurate:**The data should be accurate in terms of capturing the timestamps and capturing the observation correctly.
3. **Long enough:** The data should be long enough to forecast. This is because it is important to identify all the patterns in the past and forecast which patterns repeat in the future.

You also saw the various types of data sources to get a time-series data. These were:

1. **Private enterprise data:** E.g. financial information about the quarterly results of any private organisation.
2. **Public data:** E.g. government publishes the economic indicators
3. **System/Sensor data:** E.g. Logs generated by the servers during their 24/7 working hours.

It seems like there are quite a few components associated with time series. Let's look at them one by one once again.

1. **Level:**This is the baseline of a time series. This gives the baseline to which we add the different other components.
2. **Trend:** Over a longterm, this gives whether the time series moves lower or higher. For example, in the following Sensex graph. You can clearly observe, that with passing time, the overall value is increasing, i.e., this particular time series data has an increasing trend.
3. **Seasonality:** It is a pattern in a time-series data that repeats itself after a given period of time. For example, in the following graph 'Monthly sales data of company X', you can clearly observe that a fixed pattern is repeating for every one year. The simplest example to explain this could be, say, the sales of winter wear in India. Now, during months, say, November-January, you would expect these sales to be very high whereas, for the other months, the sales might be low. This shows a seasonality pattern and proves out to be very useful while making forecasts.
4. **Cyclicity:**It is also a repeating pattern in data that repeats themselves aperiodically. We don’t get into the more details of this component as it is out of the scope of this module.
5. **Noise:** This is completely random fluctuations and we cannot use this component to forecast into the future. These are the components of the time series that no one can explain and is completely random.

**Components of a time series**

Which of the following is/are compulsory in a Time-series?

**Level**

Every time series has a level and noise, while trend, seasonal and cyclic patterns are optional. **Noise**

Every time series has a level and noise, while trend, seasonal and cyclic patterns are optional.

**Missing values**

How do you deal with missing values in time series data with trends?

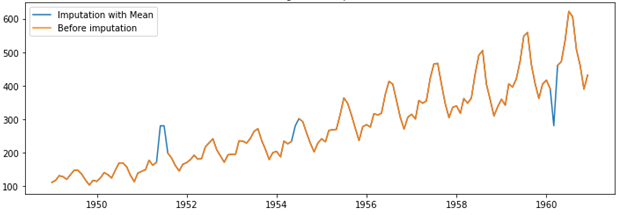
**Linear Interpolation.**

**Feedback :**

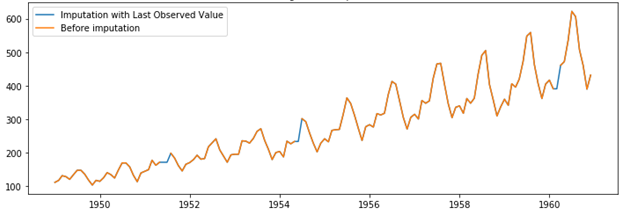
Imputing the missing value with mean, median and mode can reduce the variance. Imputing the missing value with the next observed value and last observed value can introduce bias in analysis and perform poorly when data has a visible trend. Linear interpolation imputes the missing value with the average of previous and next values.

Let's revisit the methods of handling missing values.

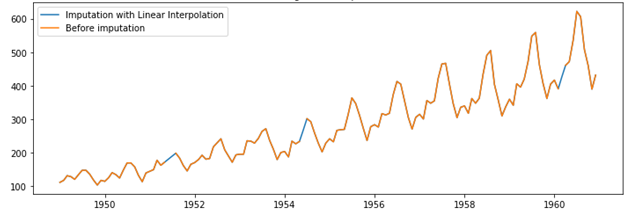
1. **Mean Imputation:** Imputing the missing values with the overall mean of the data.



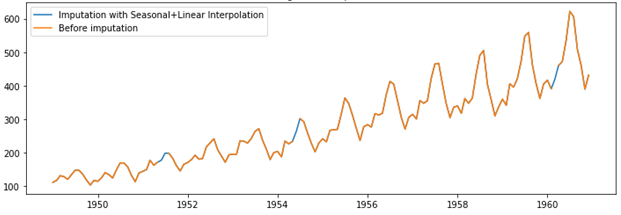
1. **Last observation carried forward:** We impute the missing values with its previous value in the data.



1. **Linear interpolation:** You draw a straight line joining the next and previous points of the missing values in the data.



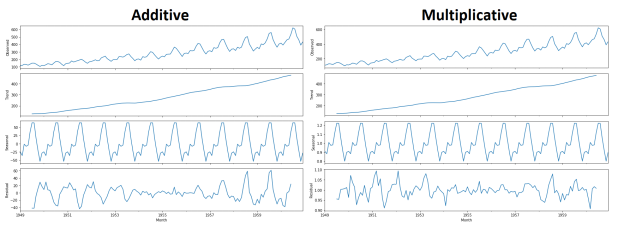
1. **Seasonal + Linear interpolation:** This method is best applicable for the data with trend and seasonality. Here, the missing value is imputed with the average of the corresponding data point in the previous seasonal period and the next seasonal period of the missing value.



* **Additive Seasonal Decomposition**- the individual components can be added to get the time-series data
* **Multiplicative Seasonal Decomposition** - the individual components can be multiplied to get the time-series data

You learnt that time series can be decomposed in two ways.

1. **Additive Seasonal Decomposition**  
   When the magnitude of the seasonal pattern in the data does not directly correlate with the value of the series, the additive seasonal decomposition may be a better choice to split the time series so that the residual does not have any pattern.
2. **Multiplicative Seasonal Decomposition**  
   When the magnitude of the seasonal pattern in the data increases with an increase in data values and decreases with a decrease in the data values, the multiplicative seasonal decomposition may be a better choice.

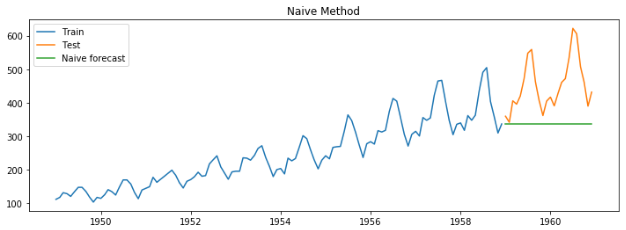


So, the two most important differences between time series and regression are:

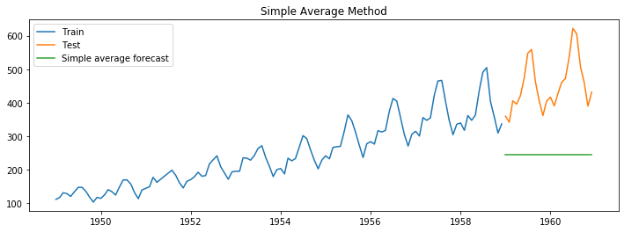
1. Time series have a strong temporal (time-based) dependence — each of these data sets essentially consists of a series of time-stamped observations, i.e., each observation is tied to a specific time instance. Thus, unlike regression, the order of the data is important in a time series.
2. In a time series, you are not concerned with the causal relationship between the response and explanatory variables. The cause behind the changes in the response variable is very much a black box.

Let's quickly recap the forecasting methods you just learnt.

* Naive method
  + Forecast = Last month’s sales



* Simple average method
  + Forecast = Average of all past months’ sales



Let's quickly summarise the three popular error measures that you learnt in the video above.

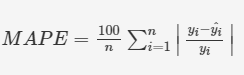
* **Mean Forecast Error (MFE):**In this naive method, you simply subtract the actual values of the dependent variable, i.e., 'y' with the forecasted values of 'y'. This can be represented using the equation below.



* **Mean Absolute Error (MAE):**Since MFE might cancel out a lot of overestimated and underestimated forecasts, measuring the mean absolute error or MAE. In this method, you take the absolute values of the difference between the actual and forecasted values.



* **Mean Absolute Percentage Error (MAPE):**The problem with MAE is that even if you get an error value, you have nothing to compare it against. For example, if the MAE that you get is 1.5, you cannot tell just on the basis of this number whether you have made a good forecast or not. If the actual values are in single digits, this error of 1.5 is obviously high but if the actual values are, say in the order of thousands, an error of 1.5 indicates a good forecast. So in order to capture how the forecast is doing based upon the actual values, you evaluate mean absolute error where you take the mean absolute error (MAE) as the percentage of the actual values of 'y'.



* **Mean Squared Error (MSE):**The idea behind mean squared error is the same as mean absolute error, i.e., you want to capture the absolute deviations so that the negative and positive deviations do not cancel each other out. In order to achieve this, you simply square the error values, sum them up, and take their average. This is known as mean squared error or MSE which can be represented using the equation below.



* **Root Mean Squared Error (RMSE):**Since the error term you go from SME is not in the same dimension as the target variable, i.e., 'y' (since you have squared it), you deploy a metric known as RMSE wherein you just take the square root of the MSE value obtained.



Accuracy = 1 - MAPE