ReDI School of Digital Integration

Time Series

Lecture 6

David Nagy, Mohan Sukumar

We use tech to connect human potential and opportunity with dignity & humility

Learning Objectives

- Working with Dates and Times in Python
- What is time series?
- Time series terminology
- Time Series Analysis
- Decomposition of Time Series
- Additive and Multiplicative Time Series
- Stationary and Non-Stationary Time Series
- Time Series analysis techniques

Working with Dates and Times in Python

Definitions of Date and Time

Date: Handles dates without time

POSIXct: Handles date & time in calendar time (ct)

POSIXlt: Handles date & time in local time (lt)

Hms: Parses periods with hour(h), minute(m), and second(s)

Timestamp: Represents a single pandas date & time

Interval: Defines an open or closed range between dates and time

Time delta: Computes time difference between different datetimes

ISO8601 datetime format

Standardization of working with time:

- YYYY-MM-DD HH:MM:SS TZ
  - Y -year, M-month, D-day,
  - H - hour, M - minute, S - second,
  - TZ - timezone

iso
1969-07-20 20:17:40
1969-11-19 06:54:35
1971-02-05 09:18:11

us
07/20/1969 20:17:40
11/19/1969 06:54:35
02/05/1971 09:18:11

non_us
20/07/1969 20:17:40
19/11/1969 06:54:35
05/02/1971 09:18:11

Usage of Date and Time

Useful libraries: datetime, time, pytz, pandas & many … many more

Cheatsheet: [LINK]

Arithmetic with Date and Time

```python
# Create two datetimes
now = dt.datetime.now()
print(now)
then = pd.Timestamp('2021-09-15 10:03:30')
print(then)
# Get time elapsed as timedelta object
print(now - then)
# Get time elapsed in seconds
print((now - then).total_seconds())
# Adding a day to a datetime
print(dt.datetime(2022,8,5,11,13,50) + dt.timedelta(days=1))
```

✓ 0.0s

```
2023-04-23 12:36:22.022385
2021-09-15 10:03:30
585 days 02:32:52.022385
50553172.022385
2022-08-06 11:13:50
```

Parsing dates, datetimes, and times

```python
# Parse dates in ISO format
iso = pd.to_datetime('2021-09-15 10:03:30')
print(iso)
# Parse dates in US format
us = pd.to_datetime('09/15/2021 10:03:30', dayfirst=False)
print(us)
# Parse dates in Danish format
dk = pd.to_datetime('15-09-2021 10:03:30',dayfirst=True)
print(dk)
```

✓ 0.0s

```
2021-09-15 10:03:30
2021-09-15 10:03:30
2021-09-15 10:03:30
```

Extracting datetime components

```python
# Get year from datetime pandas series
year = iso.year
print(year)
# Get day of the year from datetime pandas series
day_of_year = iso.day_of_year
print(day_of_year)
# Get month name from datetime pandas series
month = iso.month_name()
print(month)
# Get day name from datetime pandas series
day_name = iso.day_name()
print(day_name)
# Get datetime.datetime format from datetime pandas series
dt_format = iso.to_pydatetime()
print(dt_format)
```

✓ 0.0s

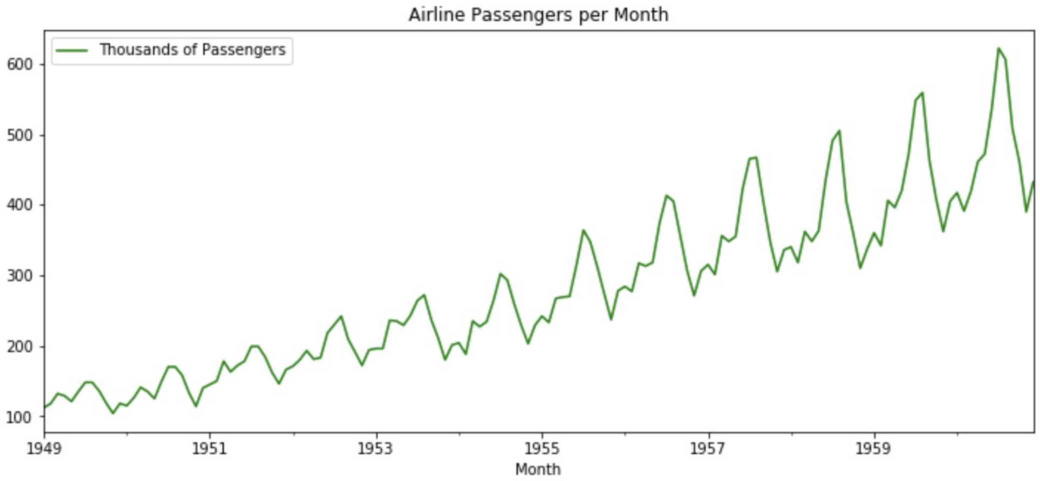
```
2021
258
September
Wednesday
2021-09-15 10:03:30
```

Break

What is Time Series

Time Series

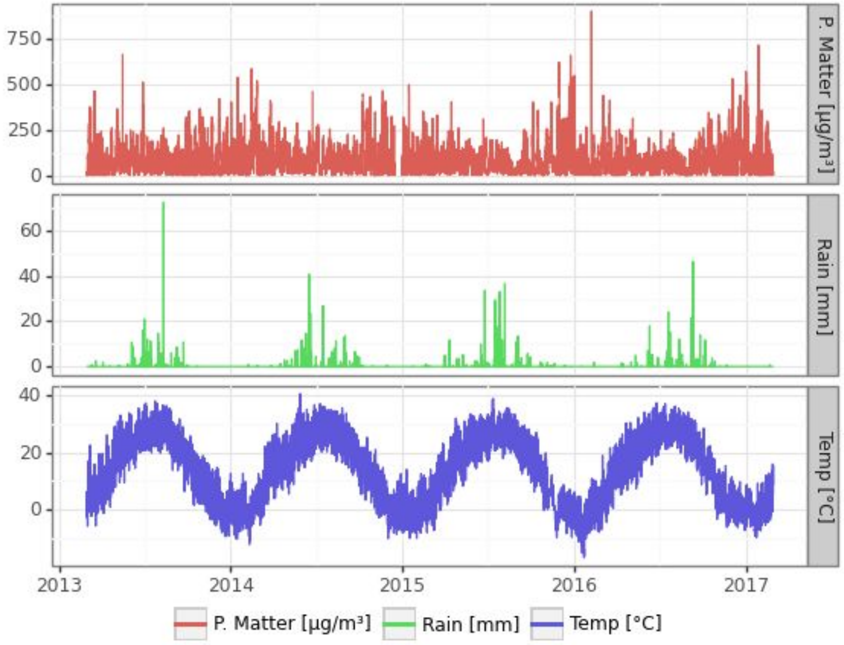
- Sequence of data points ordered in time
- Typically measured at regular intervals.
- Visualized using line charts or time plots,
- Data points plotted against time on the x-axis.



Univariate - Multivariate

- One-dimensional or multidimensional (several variables measured over time).

- Patterns or trends in the data (seasonal fluctuations or long-term trends)



Time Series Terminology

Terminology

Time interval: The frequency at which the data points are collected, such as hourly, daily, weekly, or monthly.

Time stamp: The specific time and date when a data point was collected.

Trend: The long-term increase or decrease in the data over time.

Seasonality: Regular and predictable fluctuations in the data that occur at fixed intervals, such as daily, weekly, or monthly.

Cyclicity: Longer-term periodic patterns in the data that do not occur at fixed intervals.

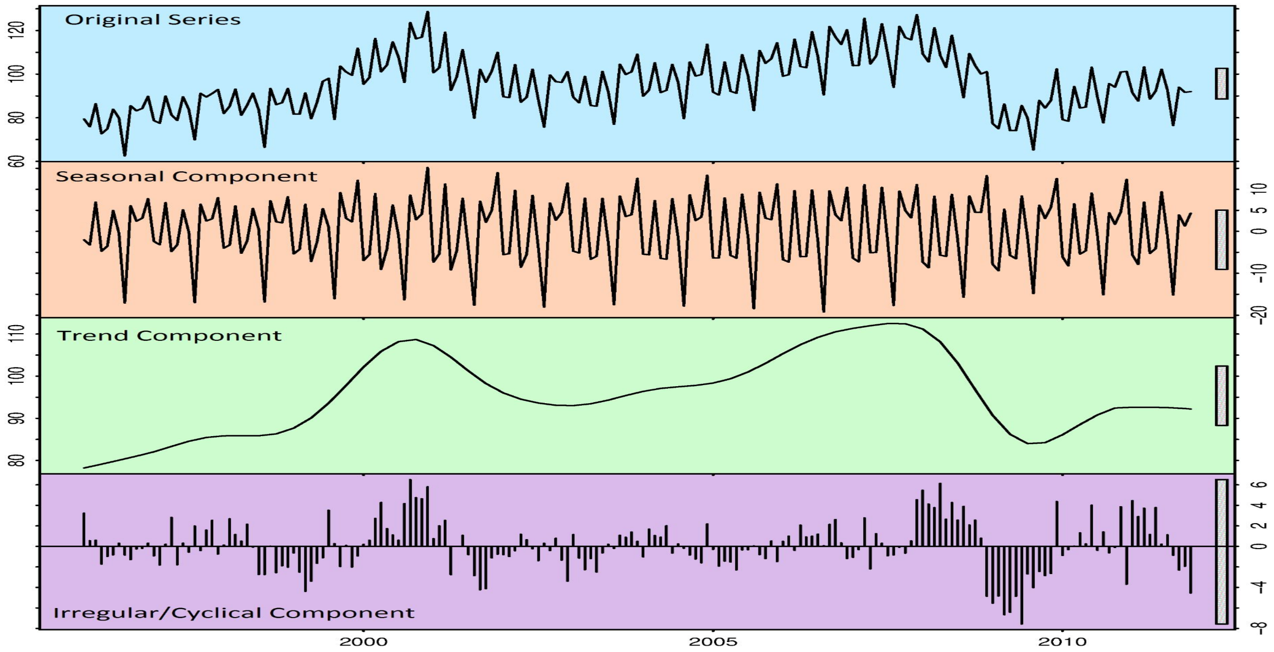
Stationarity: A time series is stationary if its statistical properties, such as mean and variance, remain constant over time.

Autocorrelation: The correlation between a time series and a lagged version of itself.

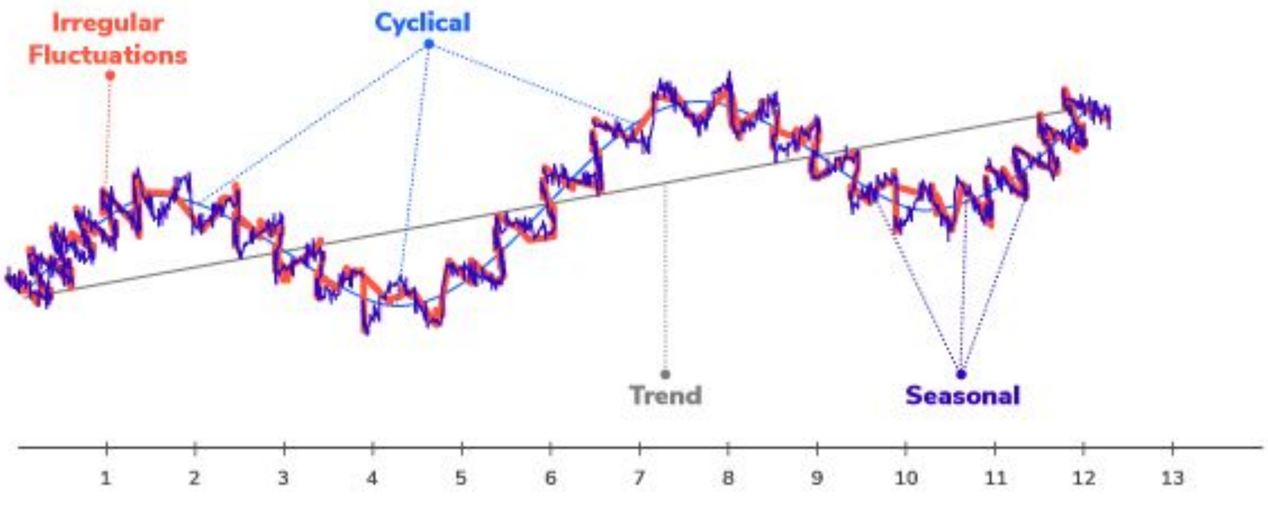
White noise: A time series where each data point is a random and uncorrelated value with a constant mean and variance.

Moving average: A smoothing technique that averages out fluctuations in the data to highlight underlying trends.

Terminology



Terminology



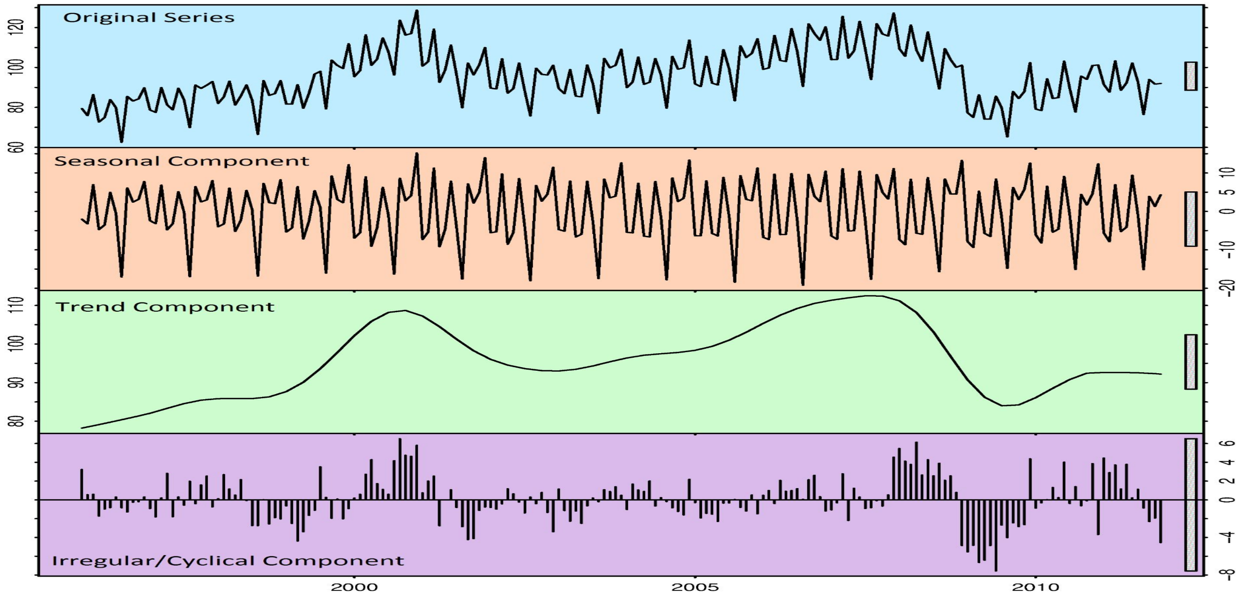
Time Series Analysis

Break

Decomposition of Time Series

Decomposition of Time Series

The process of breaking down the observed data into its constituent parts, such as trend, seasonal, and random components.

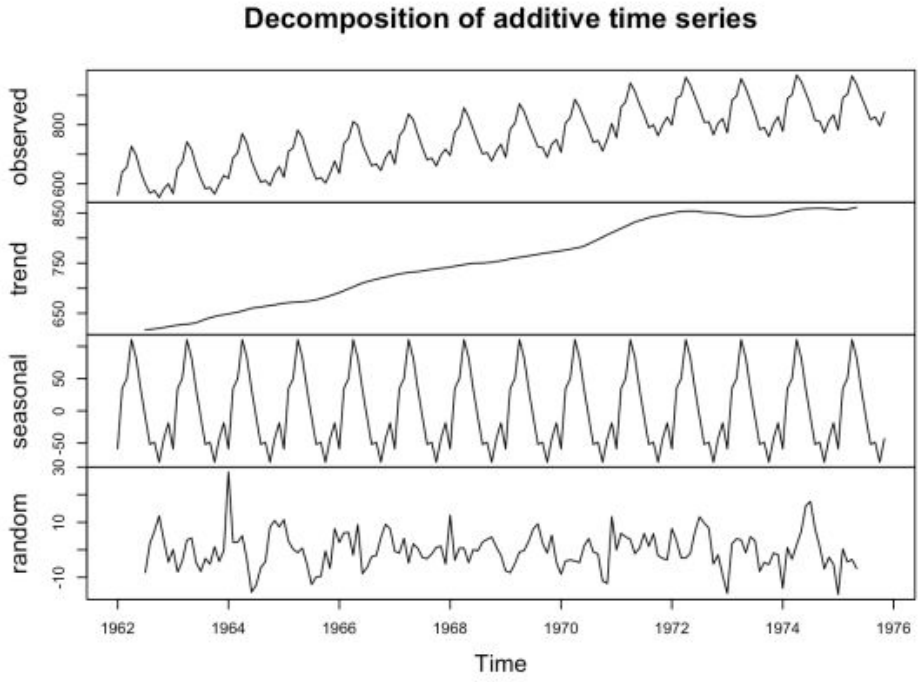


Additive and Multiplicative Time Series

Additive Time Series

Model assumes that the trend, seasonality, and random noise components of the time series are additive.

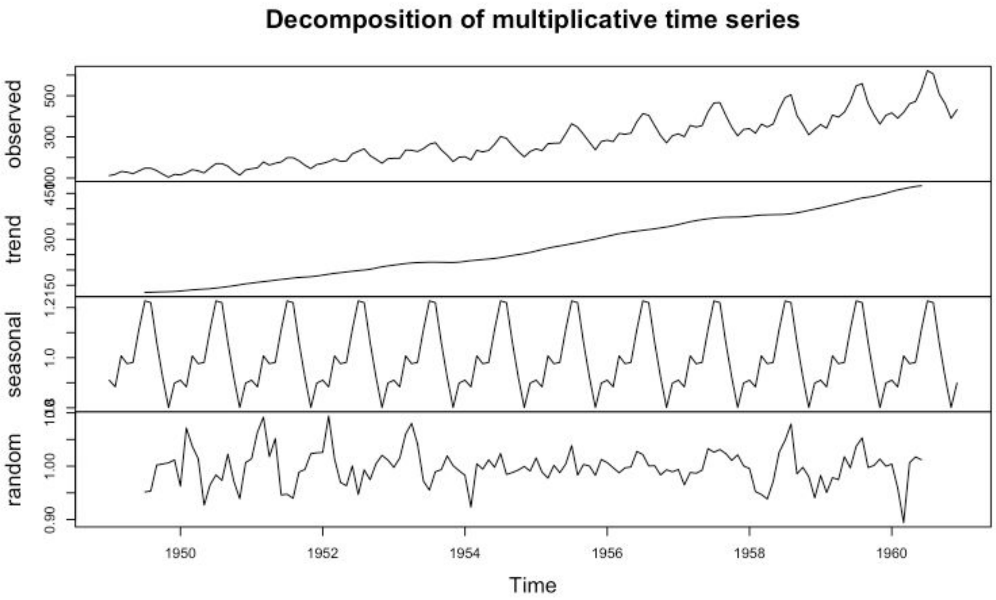
$$Y(t) = T(t) + S(t) + e(t)$$



Multiplicative Time Series

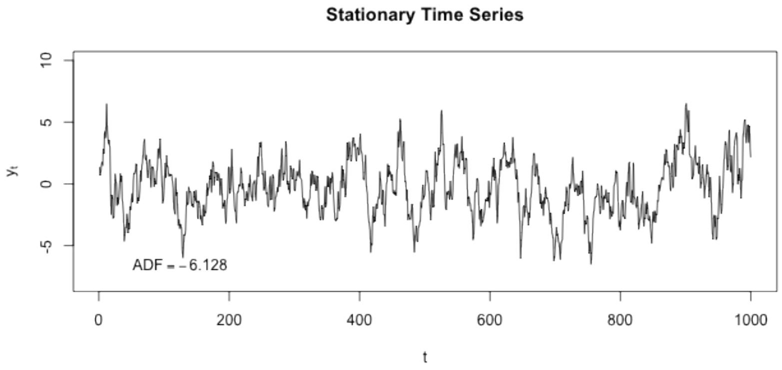
Model assumes that the trend, seasonality, and random noise components of the time series are multiplicative.

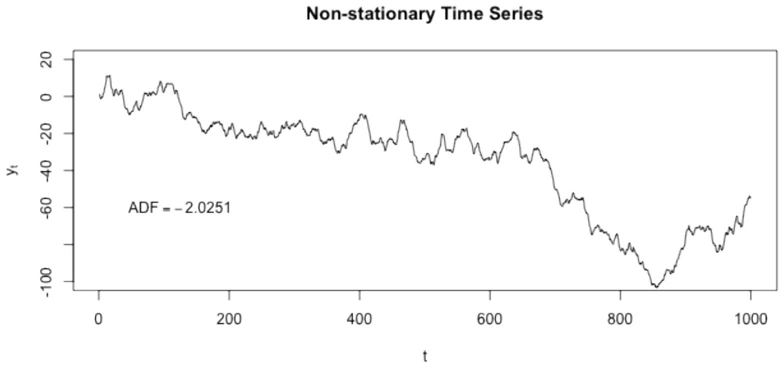
Y(t) = T(t) * S(t) * e(t)



Stationary vs Non Stationary Time Series

Stationary vs Non Stationary





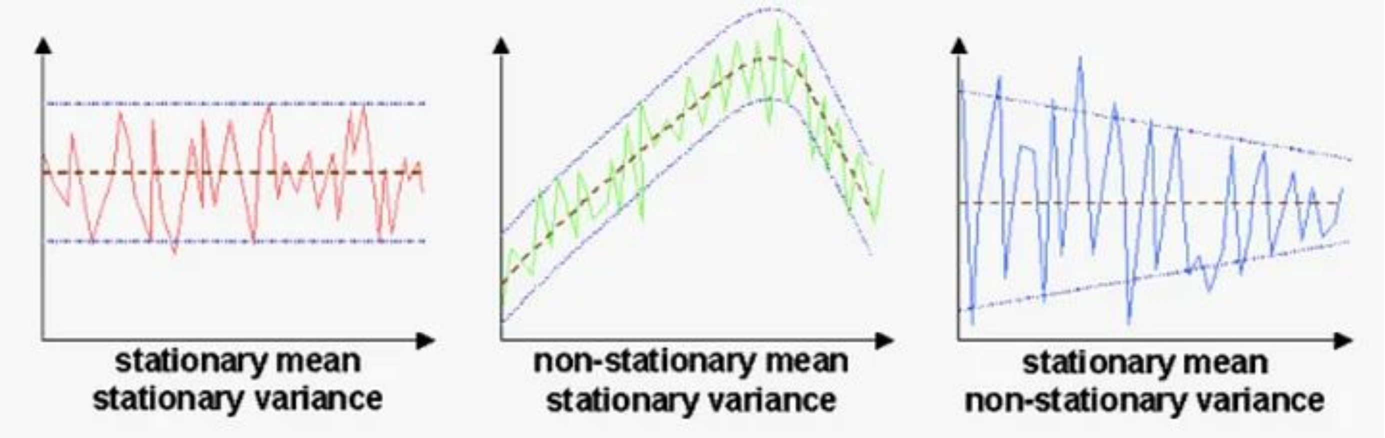
Stationary Time Series

Stationary Time Series

Statistical properties, such as mean and variance, remain constant over time.

Why is this important?

- Easier to analyze
- Most statistical model and technique assumes the data generated by stationary processes.



Non Stationary Time Series

Non Stationary Time Series

Statistical properties, such as mean and variance, changes over time.

Why is this important?

- Underlying in relationship between datapoints can bias the modelling.

Trending time series: upward or downward trend over time.
Seasonal time series: regular seasonal patterns,
Cyclical time series: irregular, non-seasonal cycles over time.
Random walk time series: each observation is a random deviation from the previous observation.

How to test for stationarity?

How to test for stationarity?

- Looking at the plot of the series.
- Split the series into 2 or more continuous parts and computing the summary statistics and the autocorrelation.
- There are several quantitative methods we can use to determine if a given series is stationary or not.
  - Augmented Dickey Fuller test (ADF Test)
  - Kwiatkowski-Phillips-Schmidt-Shin – KPSS test (trend stationary)
  - Philips Perron test (PP Test)

Break

Time Series Analysis Techniques

Differencing

Differencing

A technique used in time series analysis to remove the dependence of the observations on time.

- Help stabilize the mean of the time series by removing the trend.
- Help remove the seasonal component.
- After differencing, the resulting time series is said to be stationary.

Differencing

The first-order difference - is the difference between the current observation and the previous observation.

The second-order difference - is the difference between the first-order difference and the previous first-order difference, and so on..

diff() method in pandas to perform differencing.
(the default value of periods=1 is used to compute the difference between consecutive values)

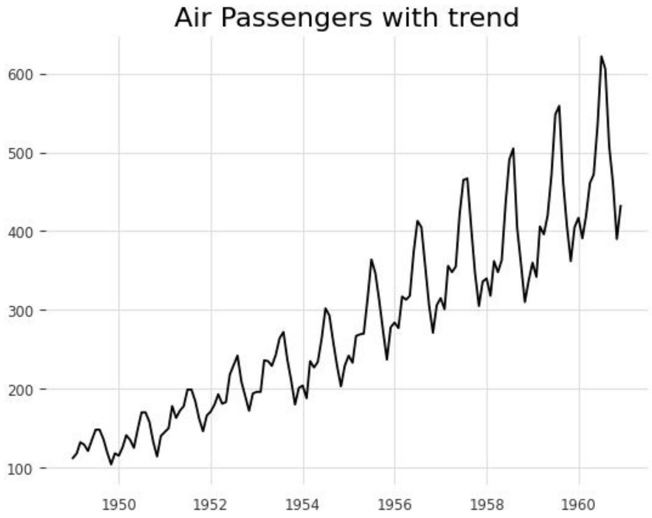
```
import pandas as pd
import random

df = pd.DataFrame({"ts": random.sample(range(10, 30), 8)})

df["diff"] = df.diff()

df
```

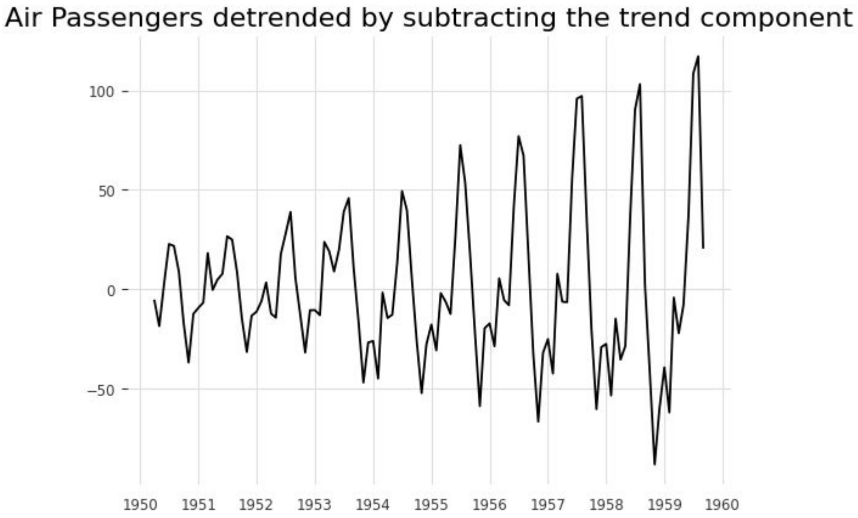
	ts	diff
0	24	NaN
1	13	-11.0
2	27	14.0
3	17	-10.0
4	10	-7.0
5	23	13.0
6	20	-3.0
7	26	6.0

Detrend and Deseasonalize

Detrending

It means to remove the trend component from the time series





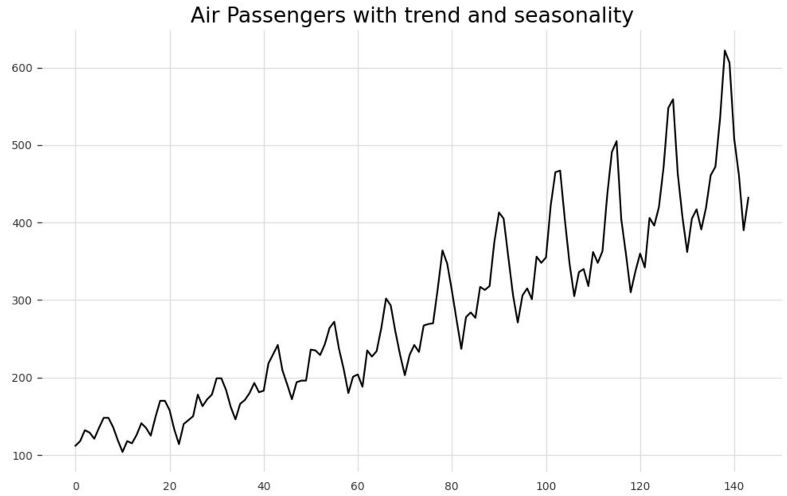
Detrending

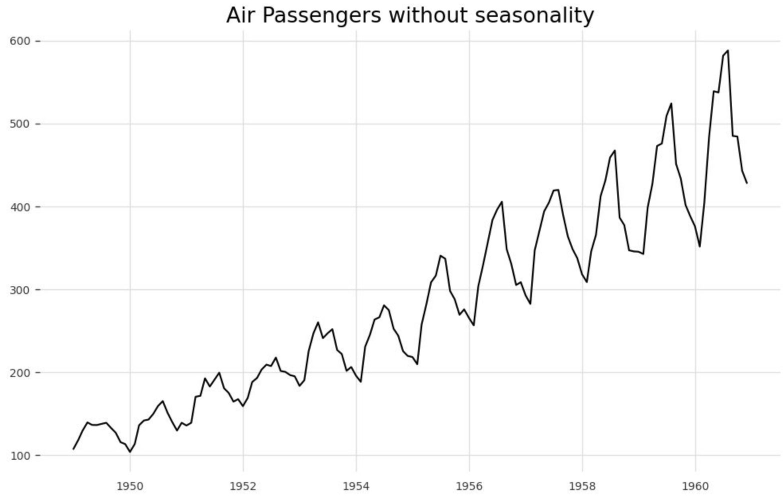
It means to remove the trend component from the time series

```python
# Using statmodels: Subtracting the Trend Component
from statsmodels.tsa.seasonal import seasonal_decompose
result_mul = seasonal_decompose(df['#Passengers'], model='multiplicative', period=30)
detrended = df['#Passengers'].values - result_mul.trend
plt.plot(detrended)
plt.title('Air Passengers detrended by subtracting the trend component', fontsize=16)
```

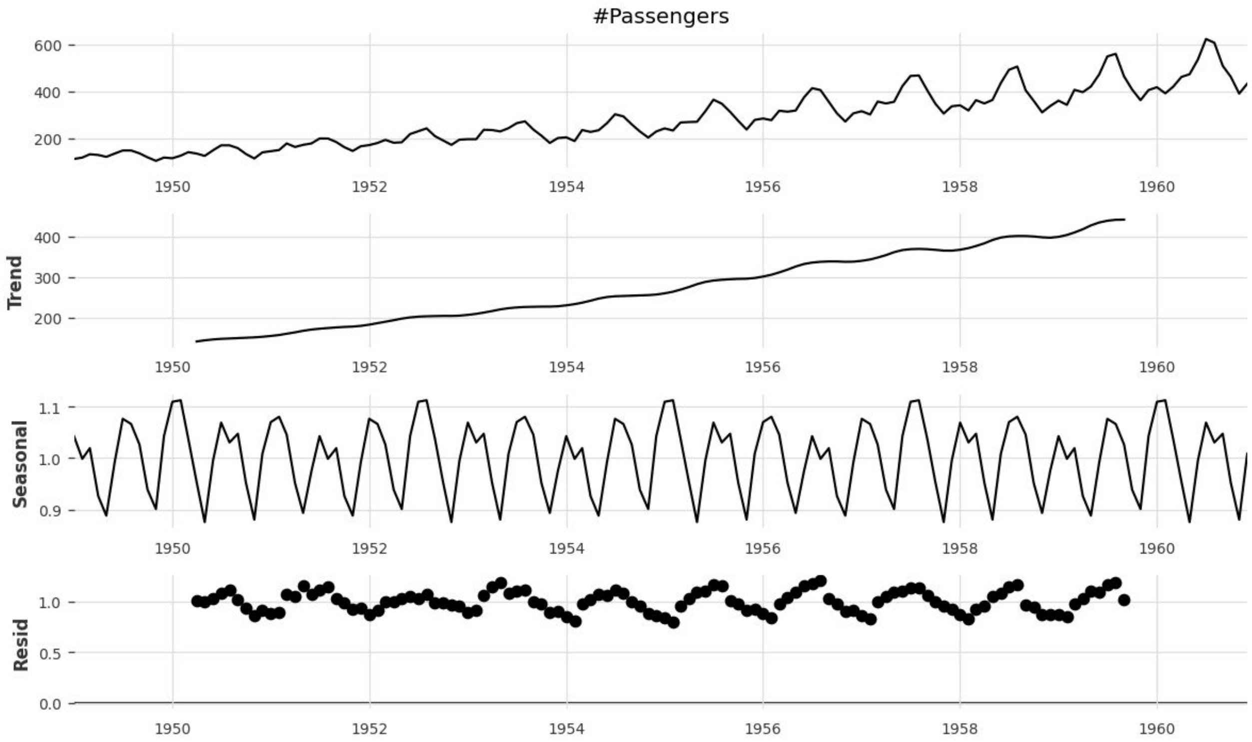
Detrending

It means to remove the seasonal component from the time series



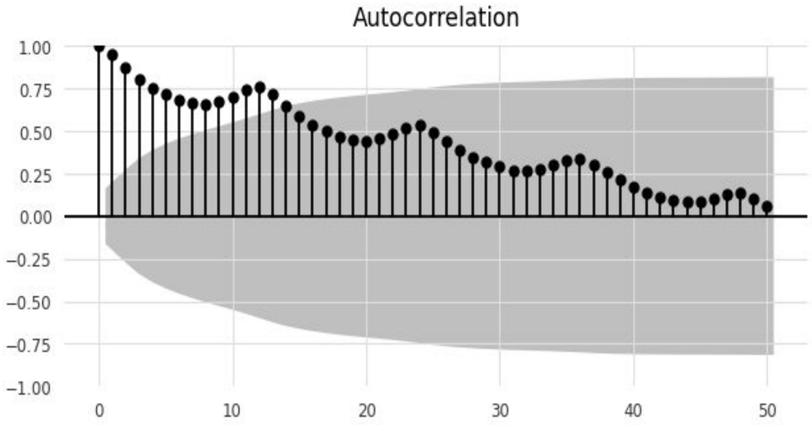


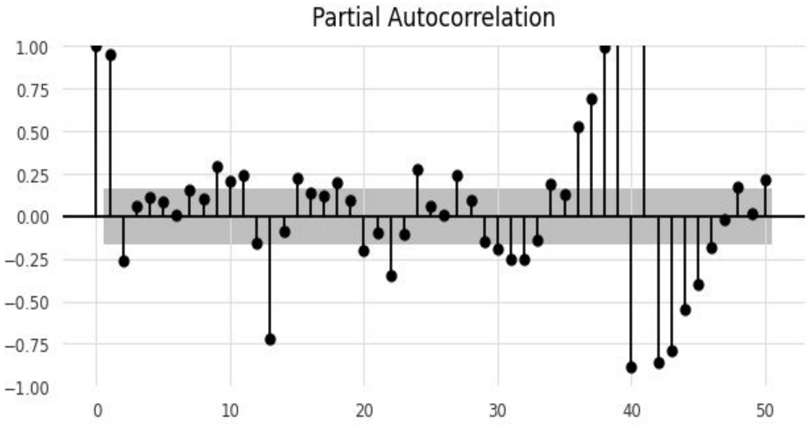
Complete decomposition



Autocorrelation

Autocorrelation and Partial Autocorrelation





WE DID IT!