

# Time Series

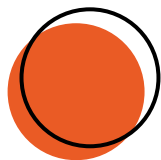
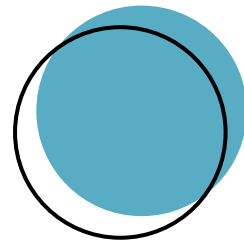
Lecture 6

David Nagy, Mohan Sukumar

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

# Ціль навчання

- Робота з Датами і Часом у програмі Пайтон
- Що таке Часовий ряд (Ряд динаміки)?
- Термінологія
- Аналіз Часових рядів
- Декомпозиція (розклад) Часових рядів
- Адитивна(додавання) та мультиплікативна(множення) моделі Часових рядів
- Стаціонарні та нестаціонарні Часові ряди
- Методи аналізу Часових рядів



# Робота з Датими і Часом у програмі Пайтон

# Визначення Дат і Часу

**Дата:** Виключно дати, без часу

**POSIXct:** Дати і час у календарному форматі (ct)

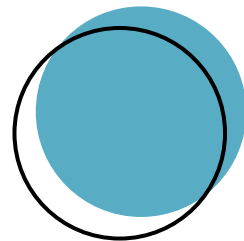
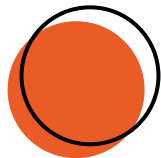
**POSIXlt:** Дати і час у місцевому форматі (lt)

**Hms:** Ідентифікує періоди за годинами(h), хвилинами(m), і секундами(s)

**Timestamp (Часовий штамп):** Відокремлює Дати і час в бібліотеці «Panda»

**Interval (Інтервал):** Визначає відкритий або закритий діапазом між Датою і Часом

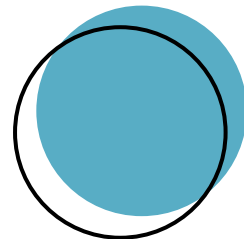
**Time delta (Дельта часу):** Обчислює різницю в часі між різними Датами



# ISO8601 формат Дати і Часу

## Стандартні виміри часу:

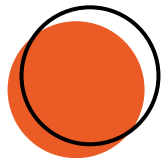
- YYYY-MM-DD HH:MM:SS TZ
  - Y – year (рік), M – month (місяць), D – day (день),
  - H – hour (години), M – minute (хвилини), S – second (секунди),
  - TZ – timezone (часовий пояс)



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

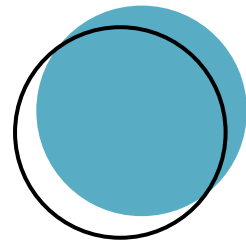
non_us
20/07/1969 20:17:40
19/11/1969 06:54:35
05/02/1971 09:18:11



# Використання Дати і Часу

**Корисні бібліотеки:** datetime, time, pytz, pandas & many ... many more

Cheatsheet: [LINK](#)



# Арифметичні розрахунки з Датами і Часом

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

✓ 0.0s

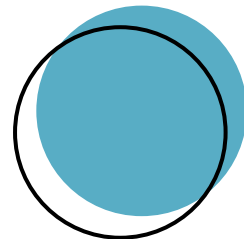
2023-04-23 12:36:22.022385

2021-09-15 10:03:30

585 days 02:32:52.022385

50553172.022385

2022-08-06 11:13:50

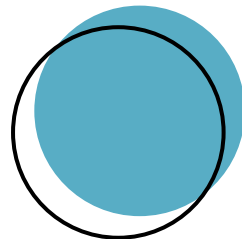


# Розбір на dates, datetimes, and times

```
1 # Parse dates in ISO format
2 iso = pd.to_datetime('2021-09-15 10:03:30')
3 print(iso)
4 # Parse dates in US format
5 us = pd.to_datetime('09/15/2021 10:03:30', dayfirst=False)
6 print(us)
7 # Parse dates in Danish format
8 dk = pd.to_datetime('15-09-2021 10:03:30', dayfirst=True)
9 print(dk)
10
```

✓ 0.0s

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



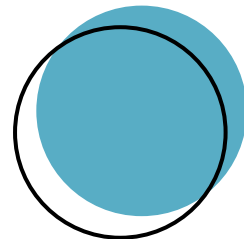


# Вилучення datetime компонентів

```
1 # Get year from datetime pandas series
2 year = iso.year
3 print(year)
4 # Get day of the year from datetime pandas series
5 day_of_year = iso.day_of_year
6 print(day_of_year)
7 # Get month name from datetime pandas series
8 month = iso.month_name()
9 print(month)
10 # Get day name from datetime pandas series
11 day_name = iso.day_name()
12 print(day_name)
13 # Get datetime.datetime format from datetime pandas series
14 dt_format = iso.to_pydatetime()
15 print(dt_format)
```

✓ 0.0s

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



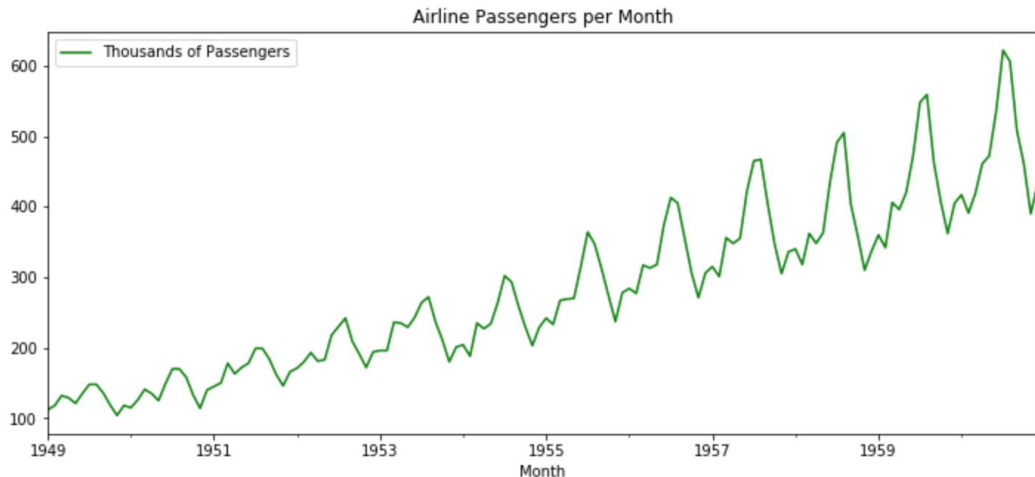
# Перерва



**Що таке Часовий ряд (Ряд динаміки)?**

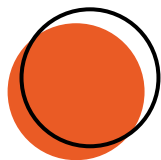
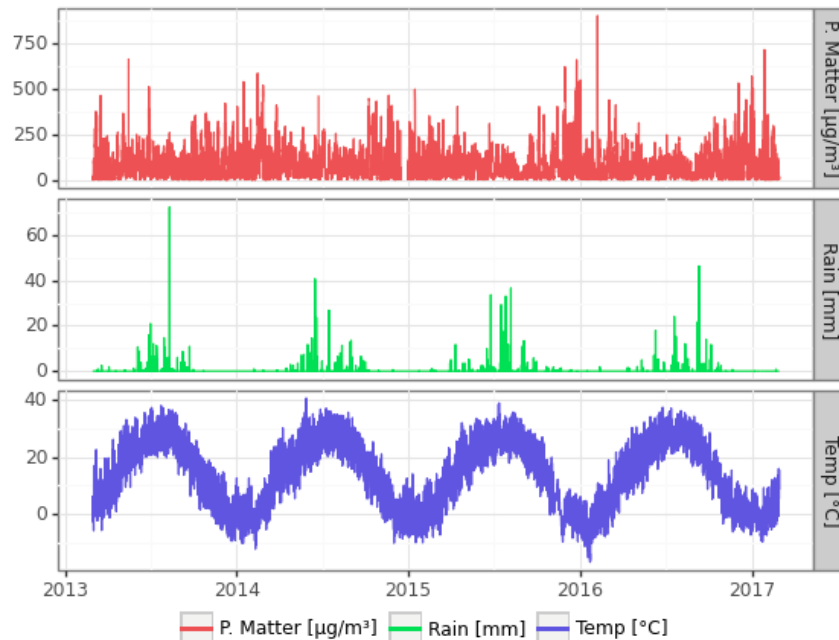
# Часовий Ряд

- Послідовні точки даних, упорядкованих у часі
- Зазвичай вимірюються регулярними проміжками часу
- Зобаржуються у вигляді лінійних або часових діаграм
- Точки даних наносяться на графіку відносно часу на осі X



# Univariate (Одно-варіантний) – Multivariate (Багато-варіантний)

- Одно- або Багатовимірний (декілька змінних виміряних у часі).
- Шаблон або тенденція в даних (сезонні коливання або довгострокові тенденції)



# Термінологія

# Термінологія

**Time interval (Часовий інтервал):** Частота з якою збираються дані, наприклад щогодини, щотижня тощо

**Time stamp (Часовий штамп):** Конкретний час і дата коли було зібрано дані

**Trend (Тенденція):** Довгострокове збільшення або зменшення даних протягом часу

**Seasonality (Сезонність):** Регулярні або передбачуванні коливання, які відбуваються у фіксований період часу. Наприклад щодня, щотижня чи щомісяця

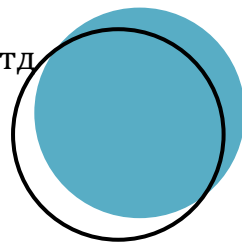
**Cyclicity (Циклічність):** Довготривалі періодичні коливання, які виникають в НЕ фіксований проміжок часу

**Stationarity (Стаціонарність):** Часовий ряд є стаціонарним, якщо його статичні властивості, такі як середнє значення або дисперсія, залишається незмінним протягом часу

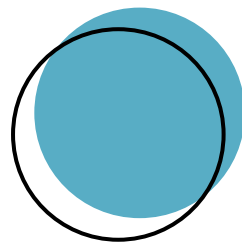
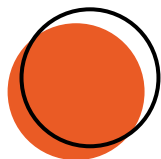
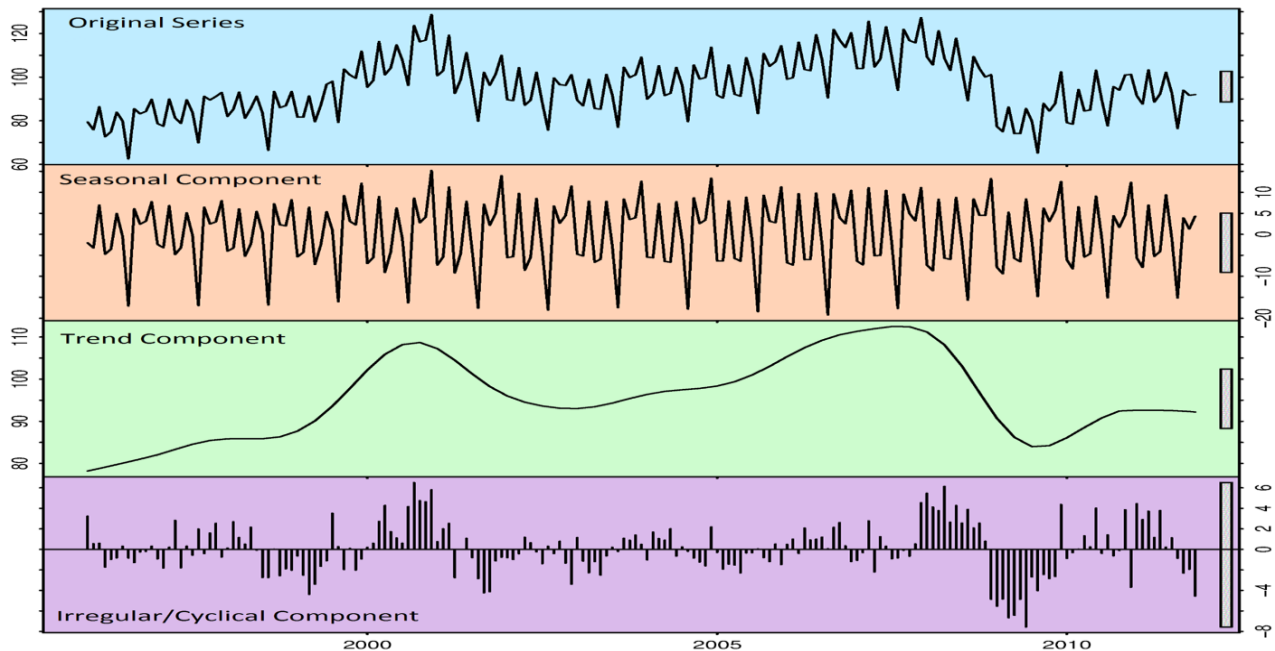
**Autocorrelation (Автокореляція):** Кореляція між часовим рядом і його запізнілою (тобто переміщеною в часі) версією

**White noise (Білий шум):** Часовий ряд де кожна точка даних є випадковим некорельованим значенням з постійним середнім значенням і дисперсією

**Moving average (Рухоме середнє):** Техніка згладжування, яка усереднює коливання даних, щоб виділити основні тенденції

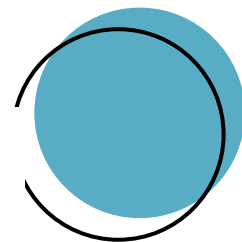
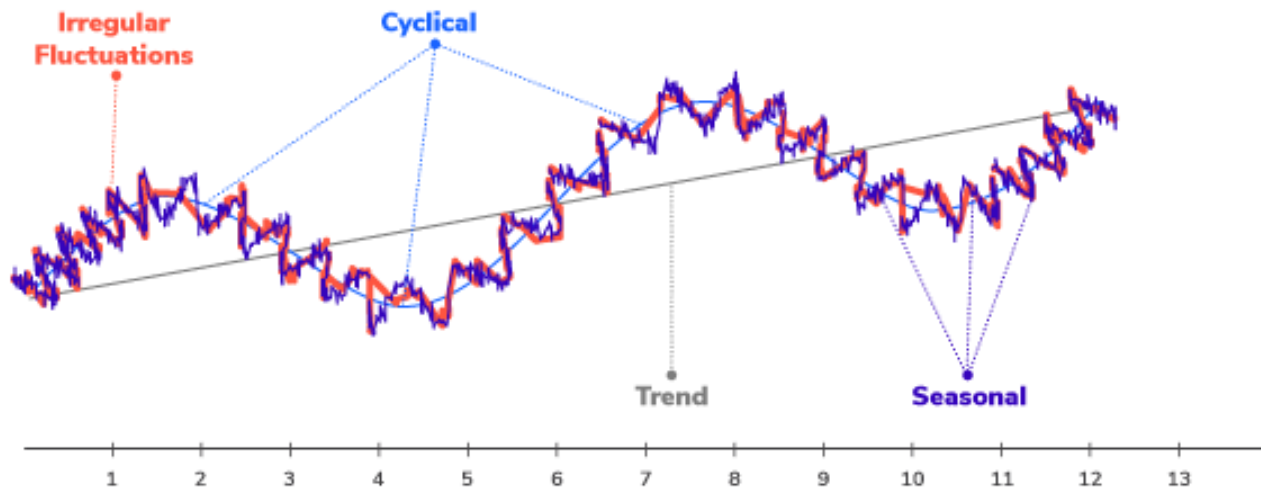


# Термінологія





# Термінологія



# Аналіз Часових рядів

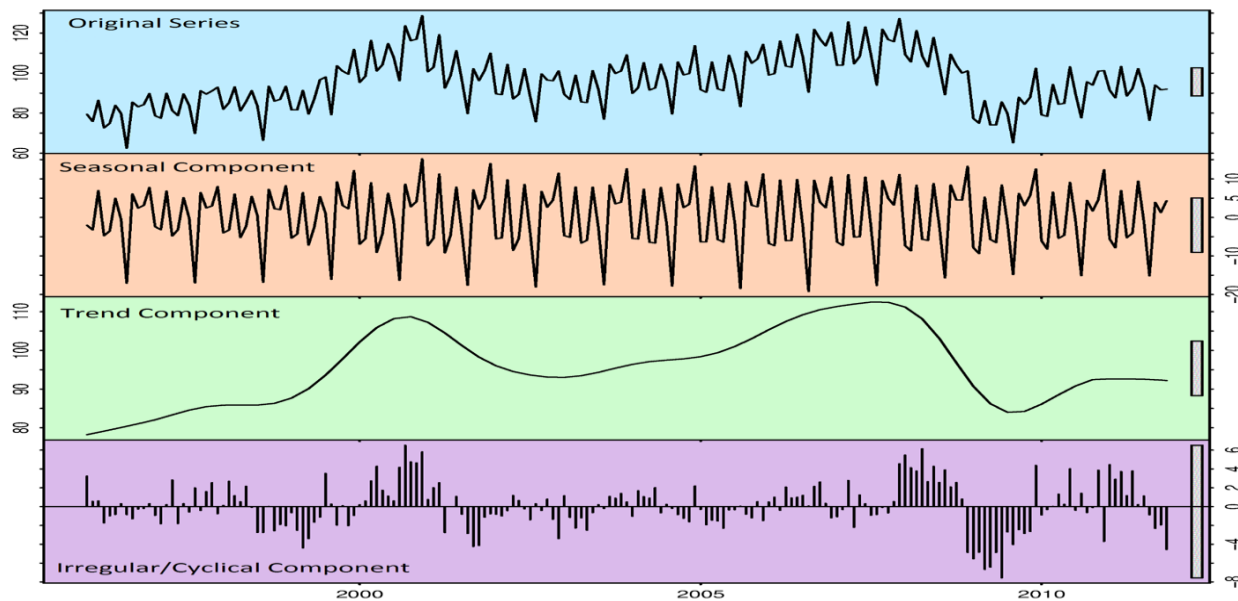
# Перерва



# Декомпозиція (розклад) Часових рядів

# Декомпозиція Часових рядів

Процес розбиття спрострежувальних даних на складові частини, такі як тренди, сезонність та випадкові компоненти

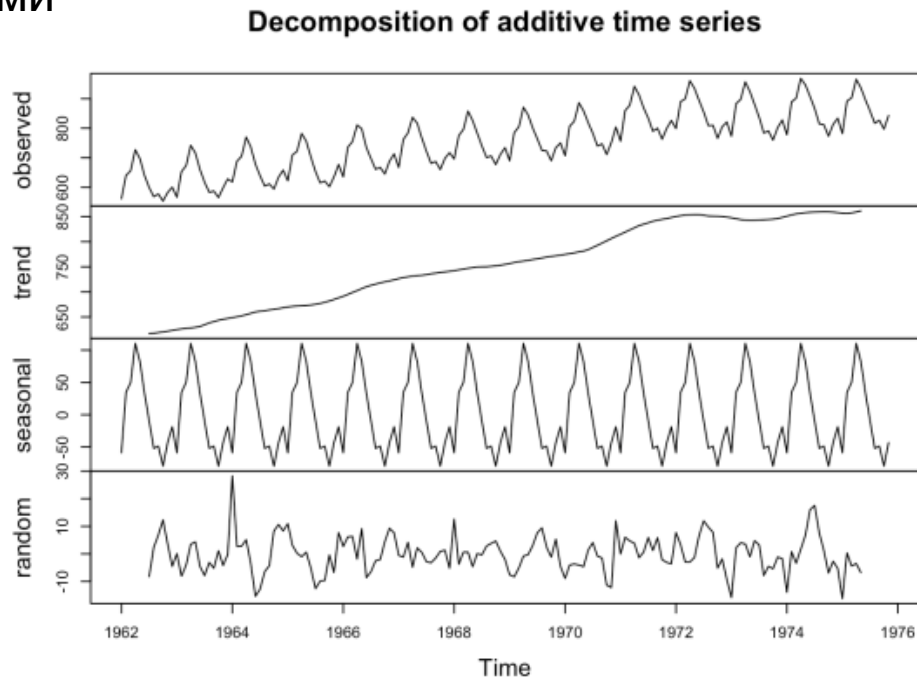


# Адитивна (додавання) та мультиплікативна (множення) моделі Часових рядів

# Адитивний Часовий ряд

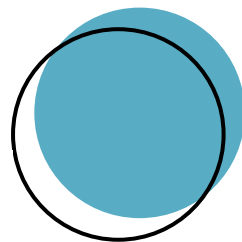
Модель припускає що тренд, сезонність і випадкові компоненти шуму часового ряду є адитивними

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

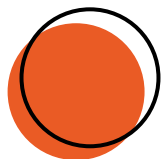
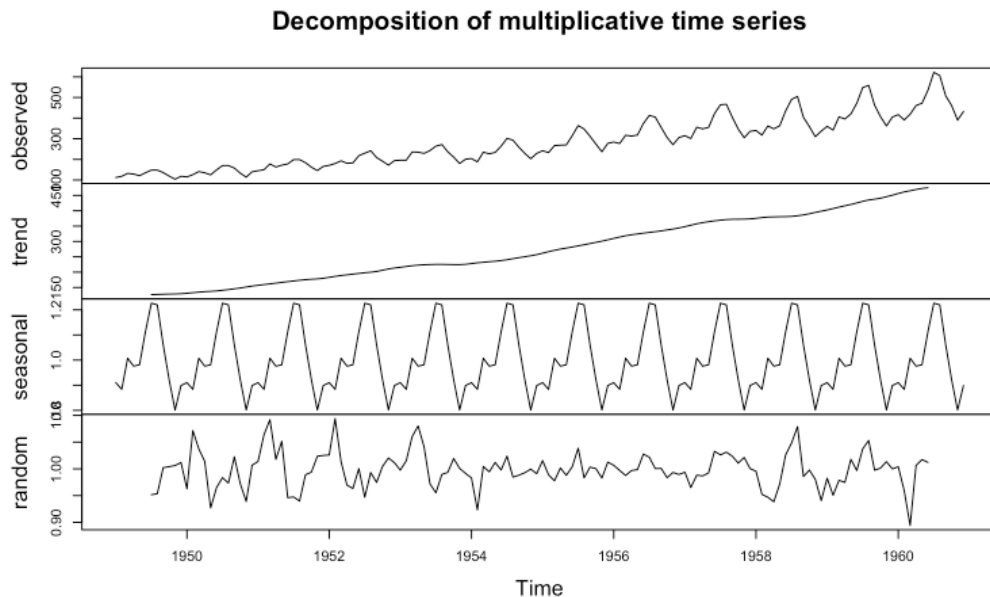


# Мультиплікативний Часовий ряд

Модель припускає що тред, сезонність і випадкові компоненти шуму часового ряду є мультиплікативними



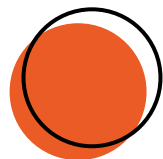
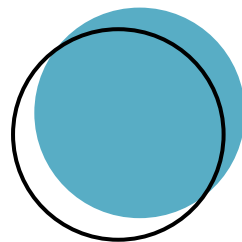
$$Y(t) = T(t) * S(t) * e(t)$$



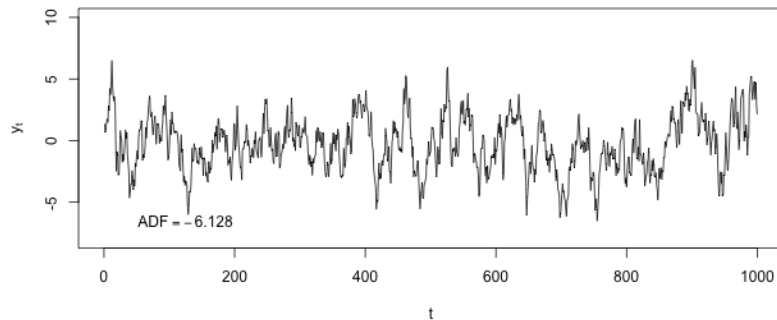


# Стационарні та нестационарні Часові ряди

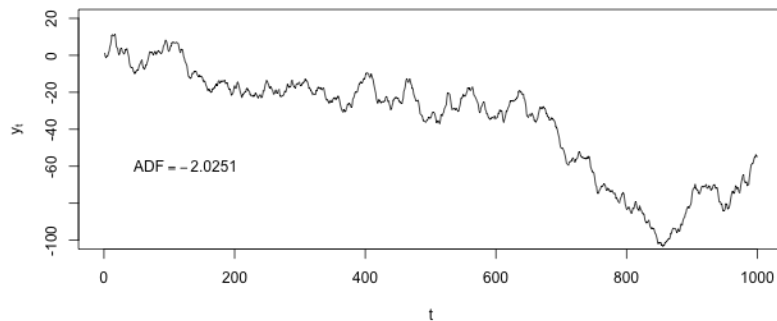
# Різниця між стаціонарними чи нестаціонарними Часовими ряди



Stationary Time Series



Non-stationary Time Series



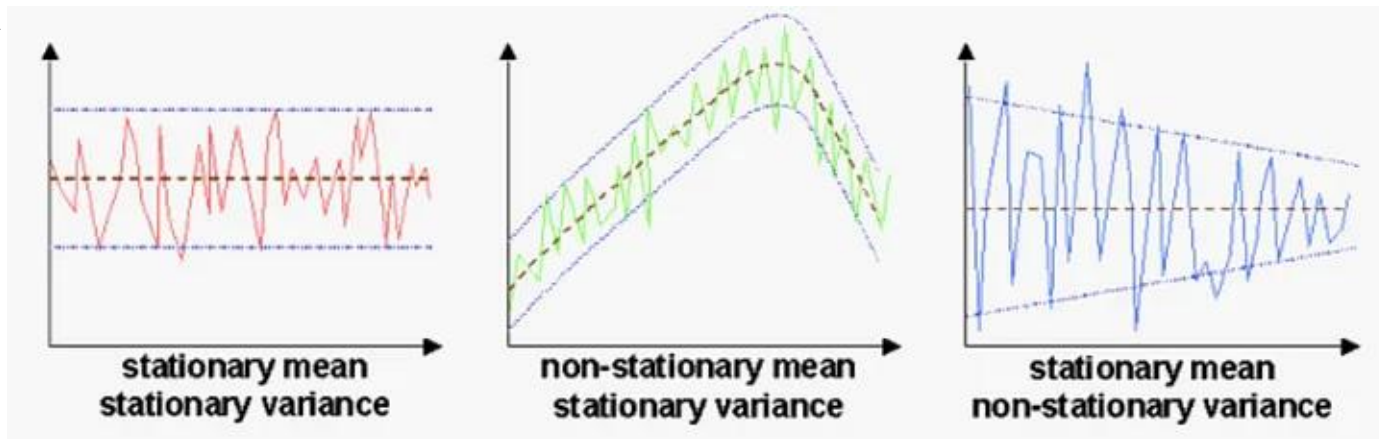
# Стационарний Часовий ряд

# Стаціонарний Часовий ряд

Статичні властивості, такі як середнє значення та дисперсія, залишаються постійними з часом

## Чому це важливо?

- Легше аналізувати
- Більшість статистичних моделей і методів припускають що отримані дані є стаціонарні



# Нестаціонарний Часовий ряд

# Нестационарний Часовий ряд

Статичні властивості, такі як середнє значення або дисперсія, змінюються з часом

## Чому це важливо?

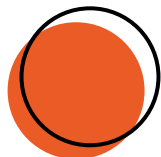
- Осановий зв'язок між точками даними може фальсифікувати результати моделювання

**Trending time series:** зростаючий або спадаючий тренд

**Seasonal time series:** регулярні сезонні моделі

**Cyclical time series:** нерегулярні, несезонні цикли

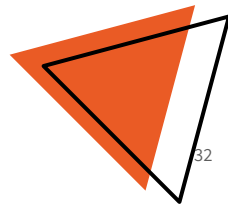
**Random walk time series (Випадкове блукання):** кожне спостереження є випадковим відхиленням від попереднього спостереження



**Як перевірити стаціонарність?**

# Як перевірити стаціонарність?

- Дивдячись на сюжет ряду
- Розібрати ряд на 2 або більше безперервних частин та обчислити статичтні підсумки і автокореляцію
- Існує декілька методів які можна використати щоб визначити чи ряд стаціонарний чи ні:
  - Augmented Dickey Fuller test ([ADF Test](#))
  - Kwiatkowski-Phillips-Schmidt-Shin – [KPSS test](#) (trend stationary)
  - Philips Perron test ([PP Test](#))





# Перерва

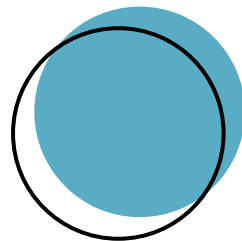


# Методи аналізу Часових рядів

# Differencing (Розрізнення)

# Розрізнення

Техніка, яка використовується в аналізі часових рядів з усуненням залежності від часу спостереження.



- Стабілізує середнє значення часового ряду за допомогою видалення тренду
- Допомогає прибрати сезонні компоненти
- Після застосування розрізнення, часовий ряд називають стаціонарним



# Розрізнення

Різниця першого порядку – це різниця між поточним та попереднім спостереженням

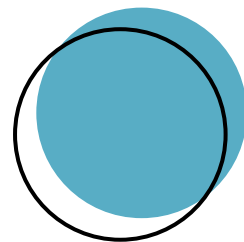
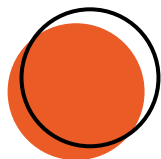
Різниця другого порядку – це різниця між різницею першого порядку та попередньою різницею першого порядку тощо...

**diff()** method in pandas to perform differencing.

( the default value of periods=1 is used to compute the difference between consecutive values)

```
1 import pandas as pd
2 import random
3
4 df = pd.DataFrame({"ts": random.sample(range(10, 30), 8)})
5
6 df["diff"] = df.diff()
7
8 df
```

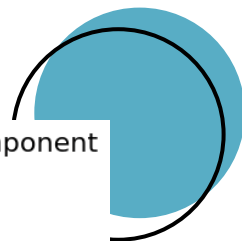
	ts	diff
0	24	NaN
1	13	-11.0
2	27	14.0
3	17	-10.0
4	10	-7.0
5	23	13.0
6	20	-3.0
7	26	6.0



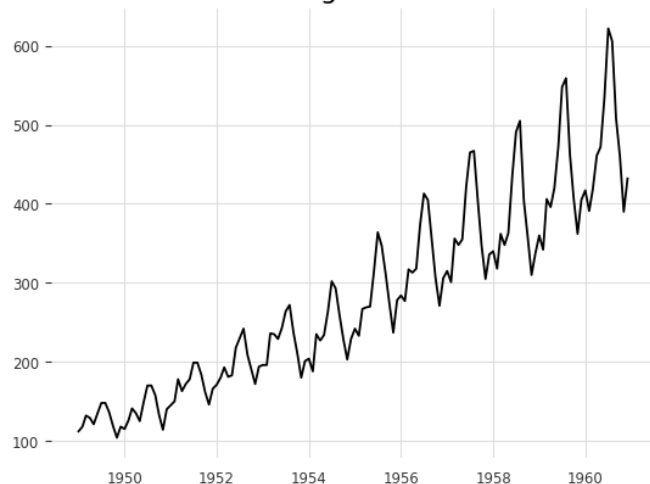
# Усунення тренду чи сезонності

# Усунення тренду

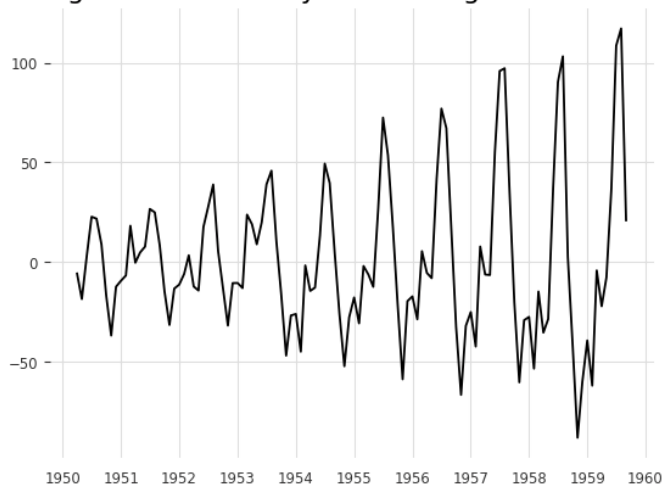
Має на увазі видадання компоненту тренду з часового ряду



Air Passengers with trend



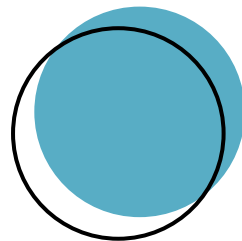
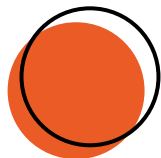
Air Passengers detrended by subtracting the trend component



# Усунення тренду

Має на увазі видалення компонента тренду з часового ряду

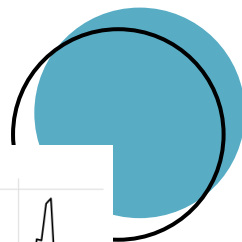
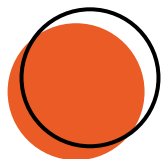
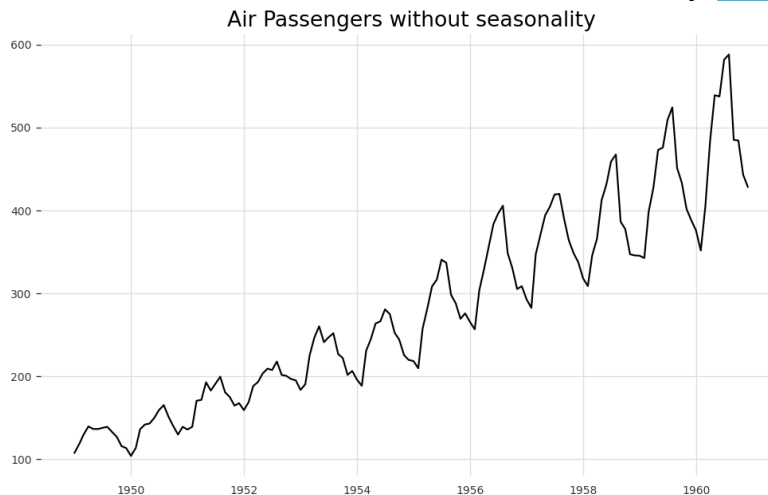
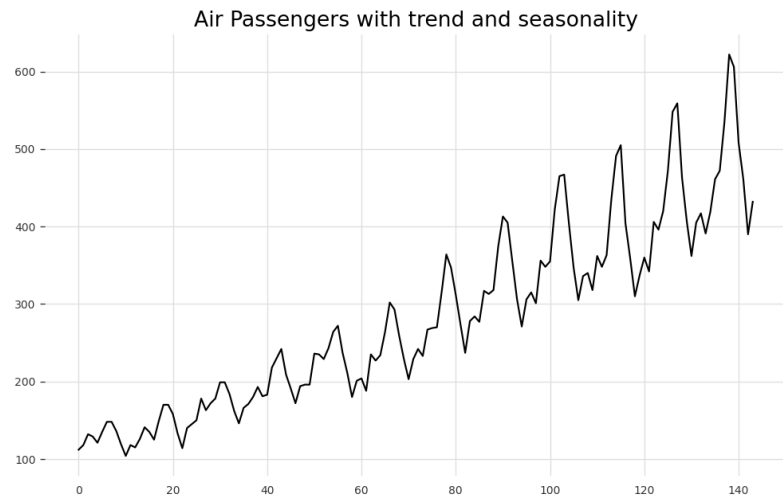
```
1 # Using statmodels: Subtracting the Trend Component
2 from statsmodels.tsa.seasonal import seasonal_decompose
3 result_mul = seasonal_decompose(df['#Passengers'], model='multiplicative', period=30)
4 detrended = df['#Passengers'].values - result_mul.trend
5 plt.plot(detrended)
6 plt.title('Air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



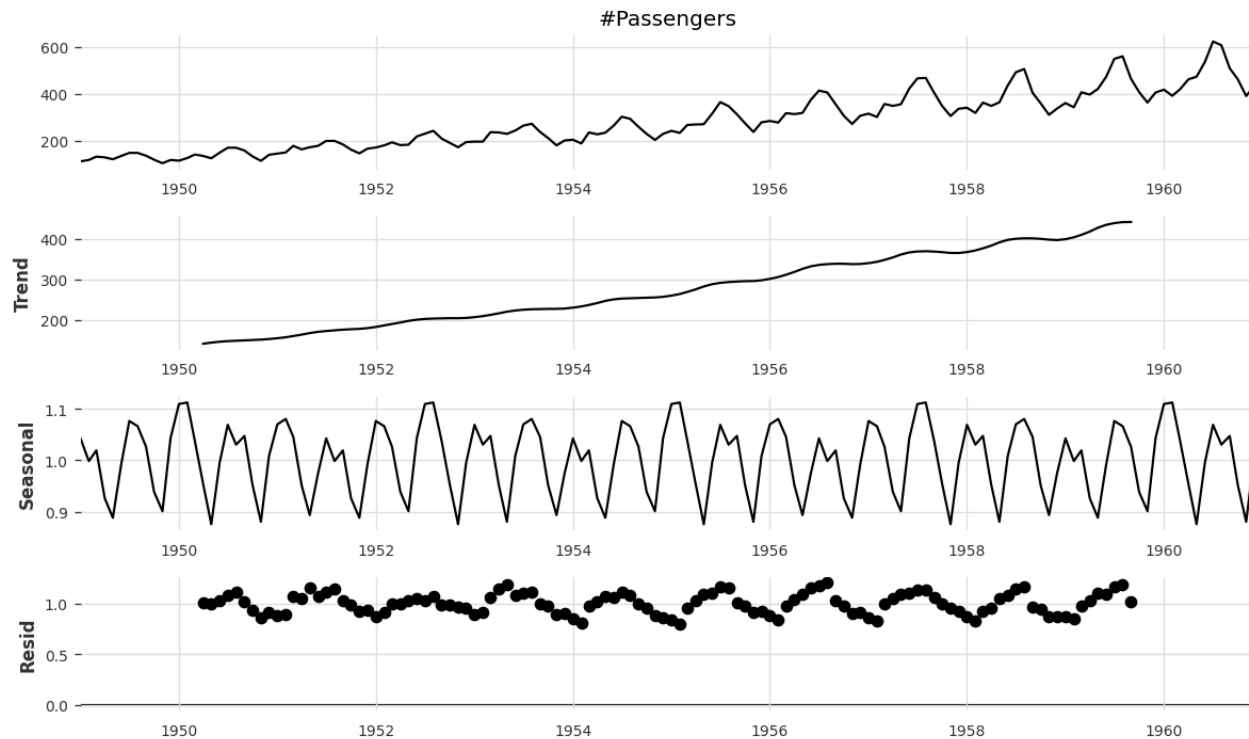


# Усунення сезонності

Має на увазі видадання сезонної складової з часового ряду

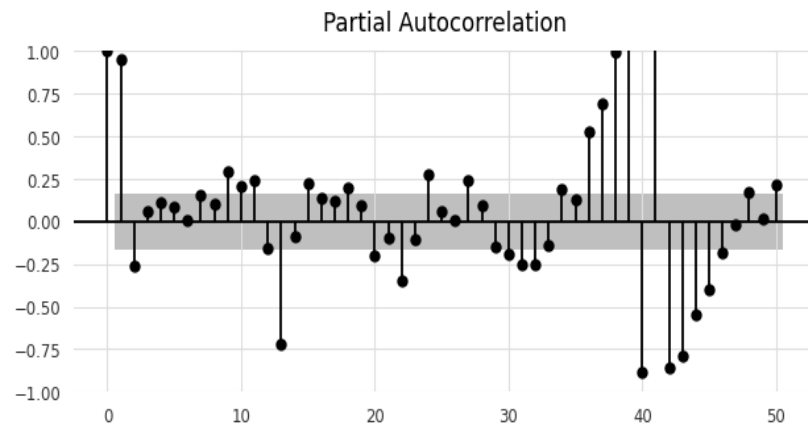
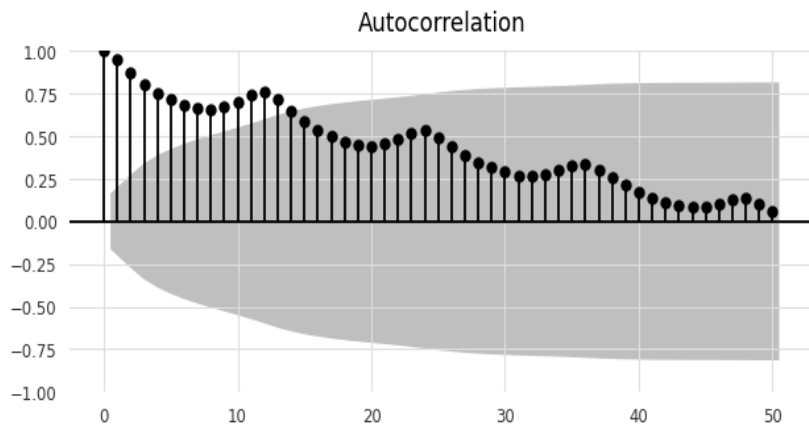


# Повне розкладання



# Автокореляція

# Автокореляція та Часткова автокореляція



Ми це зробили!

