fondo
sociale europeo

Elementi di Programmazione con Python e Analisi dei Dati
Lezione 8: Serie Temporali

Stefano Andreozzi, PhD

Formazione continua individuale – Id Attività: 2530775 Codice Corso: B341-1-2019-0

11 gennaio 2021

Che cos'è una serie temporale?

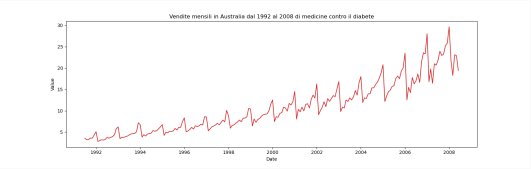
- Una serie temporale (time series) è una sequenza di osservazioni registrate a intervalli temporali regolari
- A seconda della frequenza delle osservazioni, si può avere una serie temporale con frequenza oraria, giornaliera, settimanale, ecc.
- L'analisi delle serie temporali è propedeutica allo sviluppo di una previsione (forecast)
- Le previsioni sulle serie temporali hanno un impatto significativo sia in ambito commerciale (previsione della domanda e delle vendite, numero di visitatori di un sito), sia in ambito finanziario (andamento di titoli azionari)

Caricamento e visualizzazione

```python
import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv('https://raw.githubusercontent.com/selva86/datasets/master/a10.csv',
    parse_dates=['date'], index_col='date')

def plot_df(df, x, y, title="", xlabel='Date', ylabel='Value', dpi=100):
    figura = plt.figure(figsize=(16,5), dpi=dpi)
    plt.plot(x, y, color='tab:red')
    plt.gca().set(title=title, xlabel=xlabel, ylabel=ylabel)
    plt.show()
    figura.savefig('plot_df.png')

plot_df(df, x=df.index, y=df.value, title='Vendite mensili in Australia dal 1992 al 2008
    di medicine contro il diabete')
```



- Nell'invocare `read_csv()` aggiungiamo `parse_dates=['date']` (casting esplicito delle date come data). Le date vengono utilizzate come indice.
- Pattern ripetitivo ogni anno: comparazione di ogni anno nello stesso plot.

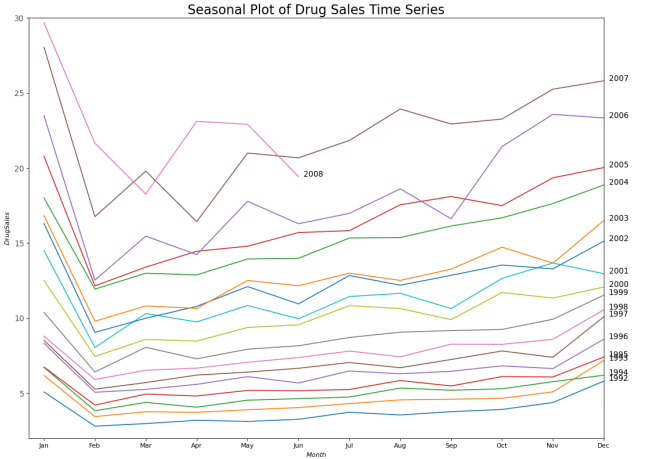
```
df.reset_index(inplace=True)

df['year'] = [d.year for d in df.date]
df['month'] = [d.strftime('%b') for d in df.date]
years = df['year'].unique()

figura2 = plt.figure(figsize=(16,12), dpi= 80)
for i, y in enumerate(years):
    if i > 0:
        plt.plot('month', 'value', data=df.loc[df.year==y, :], label=y)
        plt.text(df.loc[df.year==y, :].shape[0]-.9, df.loc[df.year==y, 'value'][-1:].
    values[0], y, fontsize=12)

plt.gca().set(xlim=(-0.3, 11), ylim=(2, 30), ylabel='$Drug Sales$', xlabel='$Month$')
plt.yticks(fontsize=12, alpha=.7)
plt.title("Seasonal Plot of Drug Sales Time Series", fontsize=20)
plt.show()
figura2.savefig('plot_df2.png')
```

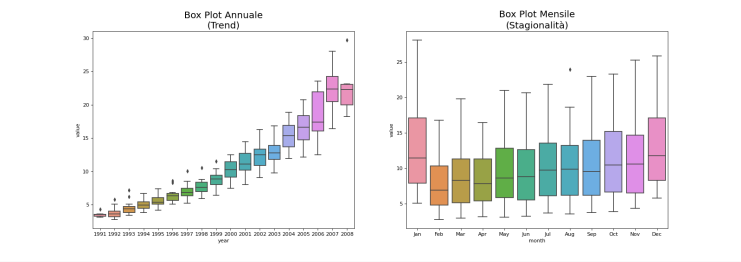
Caricamento e visualizzazione



Trend e stagionalità

```
import seaborn as sns
figura3, axes = plt.subplots(1, 2, figsize=(20,7), dpi= 80)
sns.boxplot(x='year', y='value', data=df, ax=axes[0])
sns.boxplot(x='month', y='value', data=df.loc[~df.year.isin([1991, 2008]), :])

axes[0].set_title('Box Plot Annuale\n(Trend)', fontsize=18);
axes[1].set_title('Box Plot Mensile\n(Stagionalità)', fontsize=18)
plt.show()
figura3.savefig('plot_df3.png')
```



Stazionarietà

- processo stazionario: processo stocastico la cui distribuzione di probabilità non cambia se viene traslata nel tempo
- media e varianza, se presenti, non cambiano nel tempo
- esempio: rumore bianco
- molti fenomeni studiati in economia, natura, medicina non sono stazionari, ma alcune tecniche statistiche molto utilizzate come i metodi autoregressivi a media mobile richiedono che la serie storica abbia le seguenti caratteristiche:
  - stazionarietà;
  - correlazione (i dati sono indipendentemente distribuiti?);
  - distribuzione normale.
- necessità di test statistici per verificare le caratteristiche di cui sopra