

# 09. Seaborn

Corso di Python per il Calcolo Scientifico

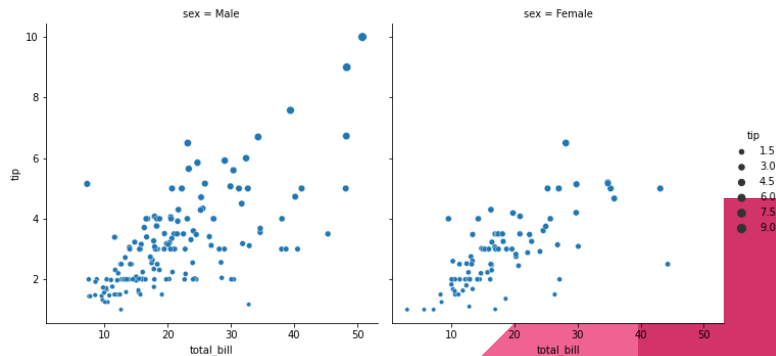
# Outline

- Visualizzazione delle relazioni tra dati
- Interpolazione e visualizzazione
- Visualizzazione delle distribuzioni
- Analisi di dati categorici
- Heatmap

# Visualizzare delle relazioni tra dati

- **Seaborn** è una libreria basata su Matplotlib per la visualizzazione avanzata dei dati
  - In tal senso, ci offre una serie di strumenti appositamente dedicati
- Possiamo visualizzare la relazione tra dati in un dataframe con la funzione **relplot()**:

```
sns.relplot(  
    data=tips,  
    x='total_bill',  
    y='tip',  
    col='sex',  
    size='tip')  
sns.relplot(  
    data=tips,  
    x='total_bill',  
    y='tip',  
    col='sex',  
    size='tip')
```

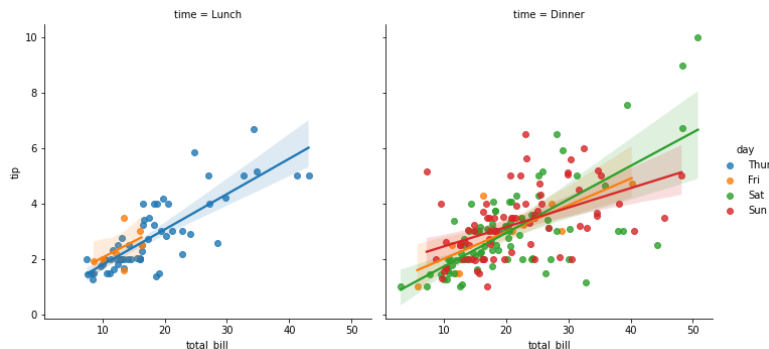


# Interpolazione e visualizzazione

- La funzione `lmplot()` estende la `relplot()` effettuando un'interpolazione dei dati:

```
sns.lmplot(  
    data=tips,  
    x='total_bill',  
    y='tip',  
    col='time',  
    hue='day')  

```

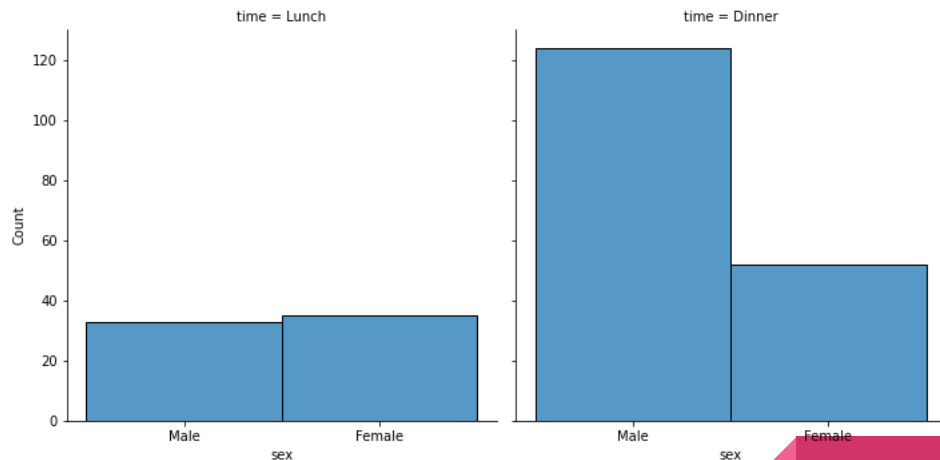


- È possibile selezionare varie tecniche di regressione passando diversi parametri alla funzione.

# Visualizzazione delle distribuzioni

- La funzione `displot()` ci permette di analizzare la distribuzione dei dati:

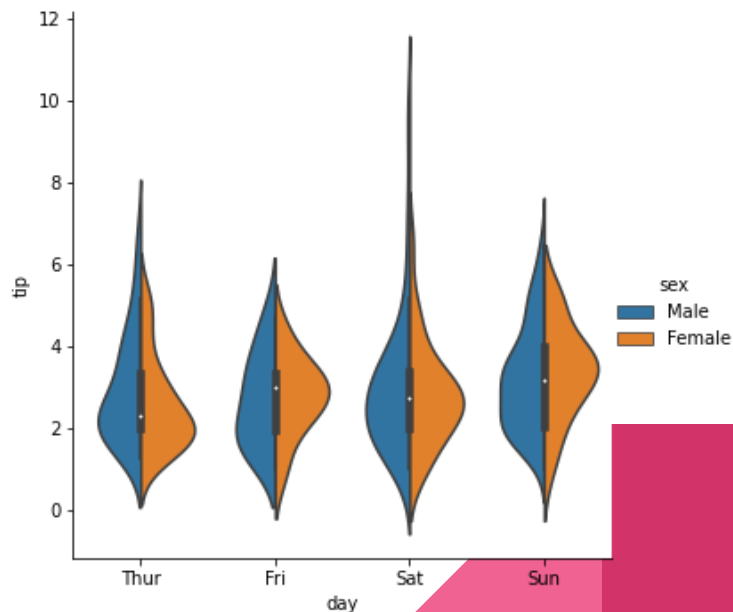
```
sns.displot(  
    data=tips,  
    x='sex',  
    col='time',  
    kde=True)
```



# Analisi di dati categorici

- Mediante la funzione `catplot()` possiamo avere ulteriori dettagli visivi per analizzare la distribuzione di dati di tipo categorico:

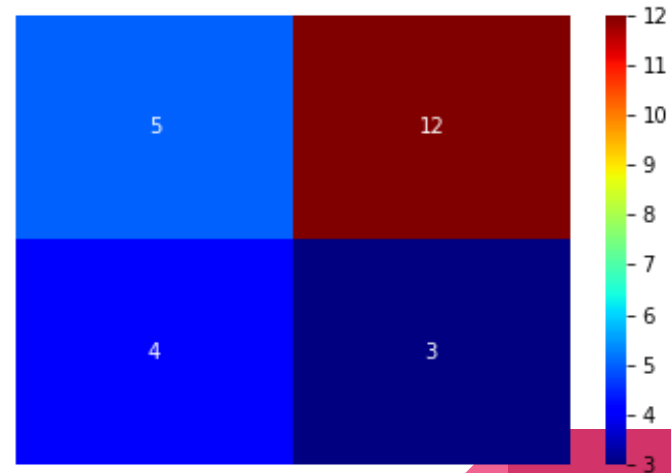
```
sns.catplot(  
    data=tips,  
    kind='violin',  
    x='day',  
    y='tip',  
    hue='sex',  
    split=True)
```



# Heatmap

- La funzione **heatmap()** ci permette di mostrare i grafici omonimi:

```
ar = np.array([[5, 12], [4, 3]])  
sns.heatmap(  
    ar,  
    cmap='jet',  
    annot=True,  
    xticklabels=False,  
    yticklabels=False)
```



# Domande?

42