

`.as_type(str) / .describe()` / `independent()`

`.value_counts()` / `.value_counts(normalize=True)`
`.value_counts(bins=5)` / `.value_counts(bins=bins)`
`.kurt()` / `.corr()`

`sns.countplot(x, data)` → bar.

`sns.countplot(data, x, hue)`
`df["column"].value_counts().plot(kind='pie', autopct='%1.2f%%')`
`df.sns.load_dataset("titanic")`

`sns.histplot(df["column"])` # bins # frequency distribution

`# df[df["price"] < 5000]` [price]

`sns.distplot()` # smooth version of histogram
probability distribution function

`df.value_counts(bins=20)` # shape of distribution

`sns.boxplot()` # identifying outliers

`sns.violinplot(df["col"])` # combo Distribution & boxplot

`sns.boxplot(x, y, data)` # central tendency

`num. category (y)` # central tendency

`sns.boxplot(x, y, hue, data)`

`estimator = "mean" (default)`
`= "sum"`
`= "median"`

sex age

`sns.boxplot(x, y, data)` # outliers

the line in each boxplot represents the median of
indicates range of age within 10%

`sns.boxplot(x="sex", y, hue, data)`

`sns.violinplot(x, y, data)`

distribution plot

`distplot(df[df["survived"] = 0][df["age"] > 35, list=False])`
`dist` `survived = 0` `age > 35` `list=False`

numerical variable across category

`boxplot(x, y, data, hue)` # avg

`stripplot(x, y, hue, data)` # pivot table

`heatmap()` `x = pd.crosstab(df["class"], df["survived"])`

`print(x)` `3rd` `2nd`

`void, sns.heatmap(x, annot=True, fmt="0.1f")`


```
sns.load_dataset("iris")
df.head(10).plot(kind="bar")
df.head(10).plot(kind="bar", stacked=True)
```

```
sns.pairplot(df2)
```

each scattered plot represents the relⁿ b/w two numerical columns
 # diagonal plots "shows the distribution" of individual
 → histogram

```
sns.scatterplot(x, y, data) # 2 Numerical Variable
```

each point individual data

X: Age
 Y: fair

```
sns.load_dataset("tips")
```

```
sns.scatterplot(x, y, hue, style, data)
```

total bill → x
 tip → y
 smoker → hue

```
sns.jointplot(x, y, data)
```

total bill → x
 tip → y

scatterplot in center
 Histogram showing distribution of two variable

```
# sns.jointplot(x, y, hue, data)
```

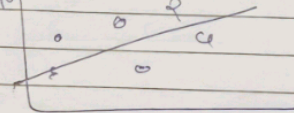
combine scatter & histogram

total bill tip smoker

regression line

```
sns.jointplot(x, y, data, kind="reg")
```

Add best fit line to show correlation



* +ive correlation

* Higher bill → higher tip

Lineplot for time series

```
sns.load_dataset("downjones")
```

```
sns.lineplot(x, y, data)
```

date → x
 price → y

visual trend over time

for time-series data, natural ordering

Zoom in timeperiod

```
df["year"] = df["Date"].dt.year
df["month"] = df["Date"].dt.month
```

sns.lmplot(y='price', x='month',
data = df[df['year']==1996])

can we

or not

but not