# Exploratory Data Analysis (EDA)

# **Outline**

# Features and Feature Types

# Steps for Doing Machine Learning

1. Acquire and load the data
2. Explore and analyze the data with Pandas and visualization
3. Data Preprocessing (DP): Clean and transform the data as necessary
   - E.g., Scikit-Learn requires numeric data
4. <u>Split the data</u> for training and testing
5. Create the machine learning model
6. <u>Train and test</u> the model
7. Tune the model and evaluate its accuracy
8. Use the model to <u>make predictions</u> on live data that the model hasn't seen before

# Feature

- A feature or a variable is any characteristic, number, or quantity that can be measured or counted

- E.g.,
    - Age (21, 35, 62, …)
    - Gender (male, female)
    - Income ($25000, $35000, $50000, …)
    - House price ($450000, $980000, …)
    - Country of birth (Qatar, Australia, Saudi, …)
    - Eye colour (blue, brown, green, …)
    - Vehicle make (Toyota, Kia, …)

# Feature Types

| Type | Subtype | Examples |
|------|---------|----------|
| **Categorical** (Qualitative) | **Nominal** | Product type, name |
| | **Ordinal** | Size measured as small<medium<large |
| | **Binary** | Spam email (yes/no, true/false, 0/1) |
| | **Date / Time** | Job start date |
| **Numerical** (Quantitative) | **Discrete** | Number of students in a class |
| | **Continuous** | Height, weight |

**Understanding the type of variables is crucial for selecting appropriate statistical methods, visualization techniques, and ML algorithms**

# Categorical Features

- **Categorical data are strings that represent qualitative data**

  o  **Often selected from a group of categories, also called labels**

- **Nominal**, e.g., country of birth, gender, eye color, etc.
  - No inherent order or ranking
  - Operators applicable: **=, ≠**
  - 1:1 transformation permissible, e.g. ID:  974 ⇒ Qatar

- **Ordinal**, e.g. grade (A, B, C, D, F), degree (bachelor, master, PhD), height (tall, medium, short), etc.
  - Represent categories that can be meaningfully ordered
  - Operator applicable: **=, ≠, <, >, ≥, ≤**
  - Order-preserving transformation permitted,
  - e.g. height (tall, medium, short) to (1, 2, 3)

# Numerical Features

- **Discrete**
  - Whole numbers (counts) typically integers
  - E.g., The number of cars in a parking lot, the number of students in a class, or the count of items in a basket.
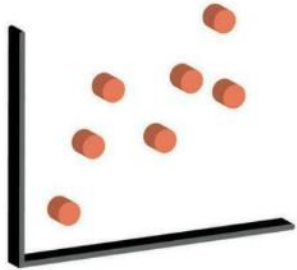
- **Continuous**
  - Measurable numeric variable that may contain any value within a range
  - Typically represented decimal numbers and fractions
  - E.g., Height, weight, temperature, or distance

# Features and Data Objects

- **Data object:** (also known as record, sample, or entity) individual object/event
    - Characterized by its recorded values on a fixed set of features

- **Features:** (also known as attribute, variable, field, or characteristic) a specific property or characteristic of the data object
    - **Raw Features:**
        - **Collected or measured** value of an attribute according to an appropriate measurement scale

    - **Derived Features**
        - Constructed from data in one or more raw features

**Features**

| Tid | Refund | Marital Status | Taxable Income | Cheat |
|-----|--------|----------------|----------------|-------|
| 1 | Yes | Single | 125K | No |
| 2 | No | Married | 100K | No |
| 3 | No | Single | 70K | No |
| 4 | Yes | Married | 120K | No |
| 5 | No | Divorced | 95K | Yes |
| 6 | No | Married | 60K | No |
| 7 | Yes | Divorced | 220K | No |
| 8 | No | Single | 85K | Yes |
| 9 | No | Married | 75K | No |
| 10 | No | Single | 90K | Yes |

**Objects**

# Derived Features

- **Aggregates:** defined over a group or period, e.g., count, sum, average, minimum, or maximum of the values

- **Flags:** indicate presence or absence of some characteristic within a dataset, e.g., a flag indicating whether or not a bank account has ever been overdrawn

- **Ratios:** capture relationship between two or more raw data values, e.g., a ratio between a loan applicant's salary and the amount for which they are requesting

- **Mappings:** convert continuous features into categorical features, e.g., map the salary values to low, medium, and high

- **Others:** no restrictions to the ways in which we can combine data to make derived features, e.g., use satellite photos to count the number of cars in the parking lots and use this as a proxy measure of activity within a competitor's stores!

# Goals for Derived Features

- To **improve** the accuracy and performance of machine learning models by transforming the raw data into a more meaningful representation that can better capture the underlying relationships in the data

- To help to **reduce** the dimensionality of a dataset and make it easier to visualize and understand the relationships between variables

# Dataset Types, Properties and Sources

# Dataset Types

| Age Group | Own Car | Income Band | Class |
|---|---|---|---|
| young | yes | low | risky |
| young | no | low | risky |
| middle aged | yes | middle | risky |
| middle aged | no | high | safe |
| middle aged | yes | low | risky |
| young | yes | high | risky |
| middle aged | no | low | safe |
| retired | yes | middle | safe |
| retired | no | middle | safe |
| retired | yes | high | safe |

Relational Table

| No. | studentID Numeric | Homework1 Numeric | Homework2 Numeric | Homework3 Numeric | Final Exam Numeric |
|---|---|---|---|---|---|
| 1 | 1.0 | | 94.0 | 34.0 | 42.0 |
| 2 | 2.0 | 35.0 | 94.0 | 85.0 | 45.0 |
| 3 | 3.0 | 31.0 | 46.0 | 22.0 | 48.0 |
| 4 | 4.0 | 46.0 | 90.0 | 60.0 | 50.0 |
| 5 | 5.0 | 52.0 | 94.0 | 49.0 | 50.0 |
| 6 | 6.0 | 58.0 | 94.0 | 30.0 | 51.0 |
| 7 | 7.0 | 47.0 | 90.0 | | 52.0 |
| 8 | 8.0 | 37.0 | 94.0 | 25.0 | 52.0 |
| 9 | 9.0 | 35.0 | 94.0 | 45.0 | 54.0 |
| 10 | 10.0 | 57.0 | 94.0 | 100.0 | 54.0 |
| 11 | 11.0 | 51.0 | 94.0 | 5.0 | 54.0 |
| 12 | 12.0 | 45.0 | 94.0 | 33.0 | 55.0 |
| 13 | 13.0 | 44.0 | 0.0 | 35.0 | 55.0 |
| 14 | 14.0 | 52.0 | 95.0 | 56.0 | 56.0 |
| 15 | 15.0 | 35.0 | 94.0 | | 57.0 |
| 16 | 16.0 | 57.0 | 97.0 | 57.0 | 57.0 |
| 17 | 17.0 | 45.0 | 90.0 | 71.0 | 57.0 |
| 18 | 18.0 | 39.0 | 94.0 | 54.0 | 57.0 |
| 19 | 19.0 | 31.0 | 94.0 | 63.0 | 57.0 |
| 20 | 20.0 | 45.0 | 94.0 | | 59.0 |
| 21 | 21.0 | 35.0 | 90.0 | 84.0 | 59.0 |
| 22 | 22.0 | 37.0 | 90.0 | 40.0 | 61.0 |
| 23 | 23.0 | 83.0 | 97.0 | 26.0 | 61.0 |
| 24 | 24.0 | 68.0 | 97.0 | 55.0 | 62.0 |
| 25 | 25.0 | 50.0 | 95.0 | 56.0 | 62.0 |
| 26 | 26.0 | 77.0 | 93.0 | | 63.0 |
| 27 | 27.0 | 84.0 | 48.0 | 18.0 | 63.0 |

Data Matrix

| TID | Items |
|---|---|
| 100 | apple, milk, newspaper |
| 200 | apple, beef, milk, newspaper, potato |
| 300 | beef, potato |
| 400 | beef, noodles |
| 500 | beef, potato |

Transaction Data

| | team | coach | play | ball | score | game | win | lost | timeout | season |
|---|---|---|---|---|---|---|---|---|---|---|
| Document 1 | 3 | 0 | 5 | 0 | 2 | 6 | 0 | 2 | 0 | 2 |
| Document 2 | 0 | 7 | 0 | 2 | 1 | 0 | 0 | 3 | 0 | 0 |
| Document 3 | 0 | 1 | 0 | 0 | 1 | 2 | 2 | 0 | 3 | 0 |

Document-term Matrix

# Types of data sets (cont.)

Page1
link1
link2

Page2
link3

Page3
xxxx
yyyy

Page4
www
zzzz
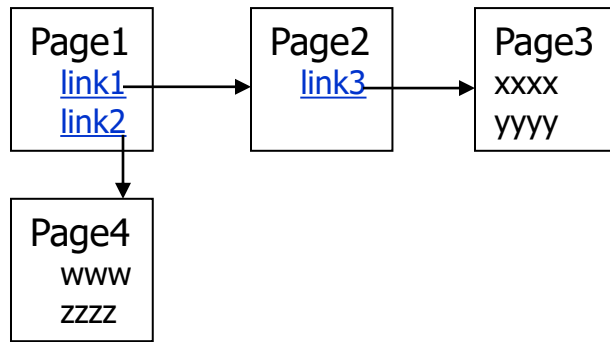
Web Structure
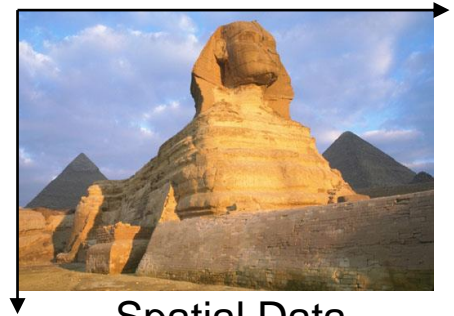

Spatial Data

GGTTCCGCCTTCAGCC
CCGCGCCCGCAGGG...

Data Sequence
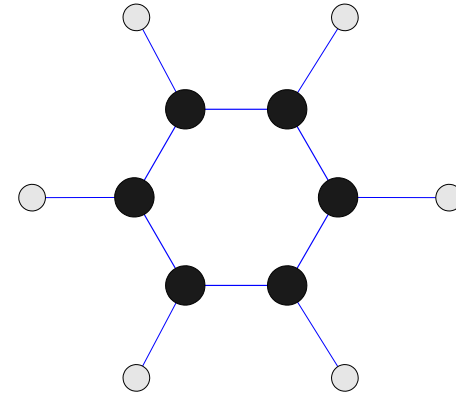
# Types of data sets (cont.)
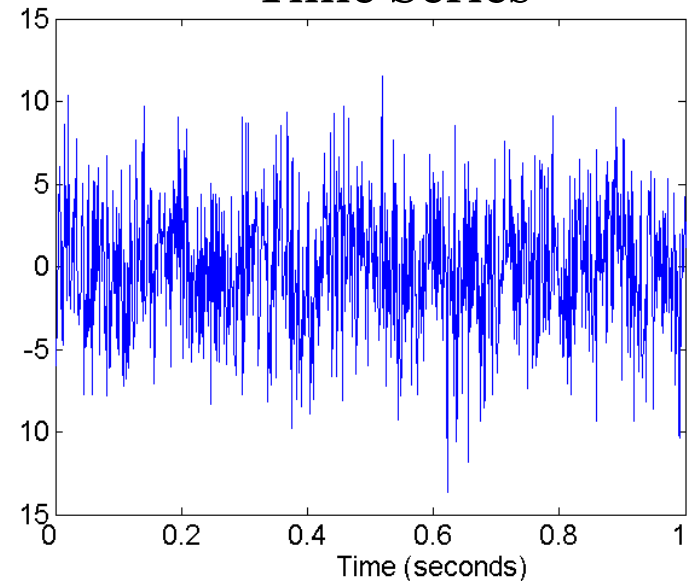
## Chemical Data



Benzene Molecule: $C_6H_6$

## Spatio-Temporal Data

Jan
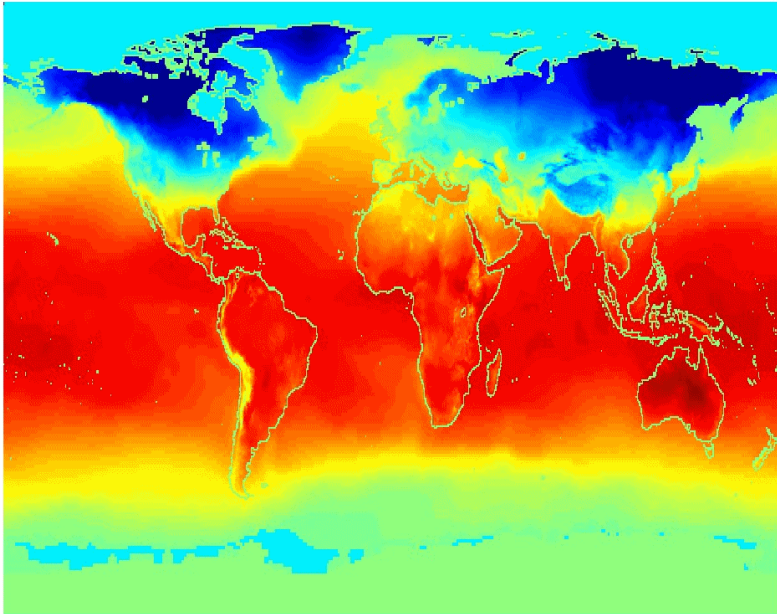


Average Monthly Temperature of land and ocean

## Time Series

# Data Matrix

- Data can often be represented or abstracted as an $n \times d$ *data matrix*, with $n$ rows and $d$ columns, given as

$$\mathbf{D} = \begin{pmatrix} & X_1 & X_2 & \cdots & X_d \\ \mathbf{x}_1 & x_{11} & x_{12} & \cdots & x_{1d} \\ \mathbf{x}_2 & x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_n & x_{n1} & x_{n2} & \cdots & x_{nd} \end{pmatrix}$$

- **Rows:** Also called *instances*, *examples*, *records*, *transactions*, *objects*, *points*, *feature-vectors*, etc. Given as a $d$-tuple

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{id})$$

- **Columns:** Also called *attributes*, *properties*, *features*, *dimensions*, *variables*, *fields*, etc. Given as an $n$-tuple

$$X_j = (x_{1j}, x_{2j}, \ldots, x_{nj})$$

# Iris Dataset Extract

Data to quantify the [morphologic](#) variation of *[Iris](#)* flowers
[Wikipedia](#)



iris setosa     iris versicolor     iris virginica

|  | Sepal length $X_1$ | Sepal width $X_2$ | Petal length $X_3$ | Petal width $X_4$ | Class $X_5$ |
|---|---|---|---|---|---|
| $x_1$ | 5.9 | 3.0 | 4.2 | 1.5 | Iris-versicolor |
| $x_2$ | 6.9 | 3.1 | 4.9 | 1.5 | Iris-versicolor |
| $x_3$ | 6.6 | 2.9 | 4.6 | 1.3 | Iris-versicolor |
| $x_4$ | 4.6 | 3.2 | 1.4 | 0.2 | Iris-setosa |
| $x_5$ | 6.0 | 2.2 | 4.0 | 1.0 | Iris-versicolor |
| $x_6$ | 4.7 | 3.2 | 1.3 | 0.2 | Iris-setosa |
| $x_7$ | 6.5 | 3.0 | 5.8 | 2.2 | Iris-virginica |
| $x_8$ | 5.8 | 2.7 | 5.1 | 1.9 | Iris-virginica |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $x_{149}$ | 7.7 | 3.8 | 6.7 | 2.2 | Iris-virginica |
| $x_{150}$ | 5.1 | 3.4 | 1.5 | 0.2 | Iris-setosa |
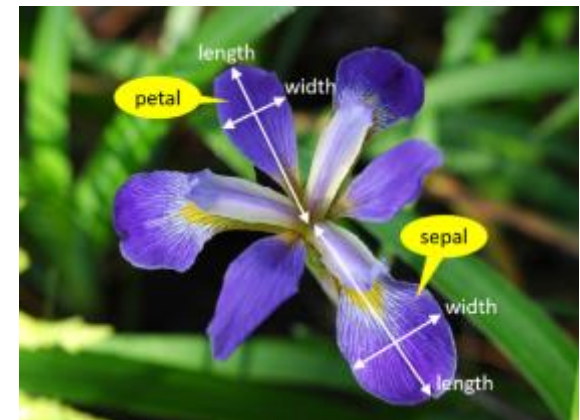
# Dataset Properties

**Size:** Measured in terms of the total number of records or total number of bytes, e.g. Small (MB), medium (GB) and large (TB)

**Dimensionality:** Number of attributes

**Sparsity:**
- Values are skewed to some extreme or sub-ranges
- Asymmetric values (some are more important than others)

**Resolution:**
- Right level of data details
- Related to the intended purpose

# Data Sources

- **Public data**

  – Data hubs https://www.kaggle.com/datasets https://www.openml.org , GitHub
  – Open data such as https://www.data.gov.qa/ & https://data.gov/
  – Data conferences
  – Many others…

- **Enterprise/Organisational data warehouse**

  - An organisational database for decision making
  - A central data repository separate from operational systems
  - Equipped with data analysis and reporting tools

- **Your own generated/collected data**

# Data Exploration - Univariate

# Exploratory Data Analysis (EDA)

- Exploratory Data Analysis (EDA): exploring data through summary statistics and visual charts, and graphs.

  **Purpose:**
  - Better understanding of the characteristics of data
  - Spot anomalies (e.g., missing data, outliers)
  - Better decision regarding data pre-processing tasks

- The three main types of EDA:

  - **Univariate EDA** explore a single feature at a time to understand the data distribution and identify any outliers.

  - **Bivariate EDA** looking at two features at a time to understand the relationship and identify any patterns that might exist

  - **Multivariate EDA** looking at three or more features at a time to understand the relationships and identify any patterns

# Summary Statistics - Central Tendency

- Mean and Median for continuous attributes:

  - **Mean**

  $$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$$
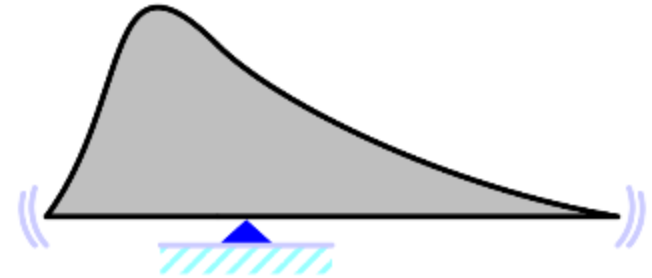
  - **Median** (Middle value if odd number of values, or average of the middle two values otherwise)
  
  ➢ Median is a better indication of "average" when data distribution is skewed, or outliers are present

  50%  50%
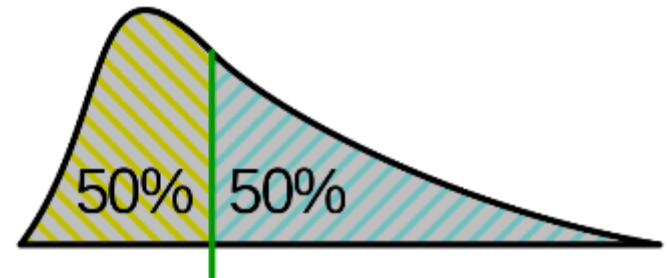
  - Trimmed Mean and Median (after trimming top and bottom p%)

# Summary Statistics - Central Tendency

- **Mode** for categorical attributes:
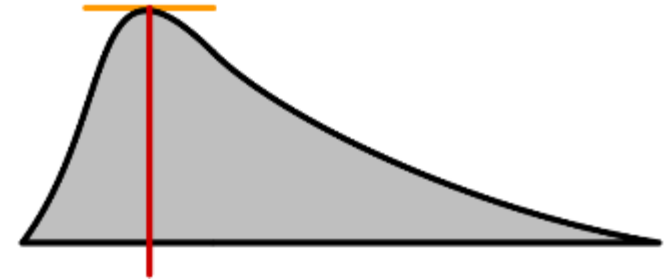


  o Frequency counts of values that a feature takes

  o Proportion: Frequency count for a value divided by the total sample size

  o **Mode**: the most frequently occurred value

# Summary Statistics - Measures of Spread

- Measure how "spread out" the values are

- Range $range(x) = \max(x) - \min(x)$

- Variance ($\sigma^2$) $\sigma^2 = \dfrac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2$

- Standard Deviation ($\sigma$) $\sigma = \sqrt{\dfrac{1}{m-1}\sum_{i=1}^{m}(x_i - \bar{x})^2}$

- Percentiles of continuous attributes:

  o Given an attribute $x$ and an integer $p$ ($0 \leq p \leq 100$), the percentile $x_p$ is a value of x such that $p\%$ observed values of x are less than $x_p$

  o $Q_1$ (25th percentile), $Q_3$ (75th percentile). $Q_3$ means 75% of the data values are less than $Q_3$

  o Inter-quartile range: IQR $= Q_3 - Q_1$

# Measures of Spread, *cont'd*

- Measure how the values are stretched or squeezed

  o aka **Measures of dispersion**

Two datasets with the <u>same mean</u> but **different dispersion**. The blue dataset is much more **dispersed** than the red dataset.
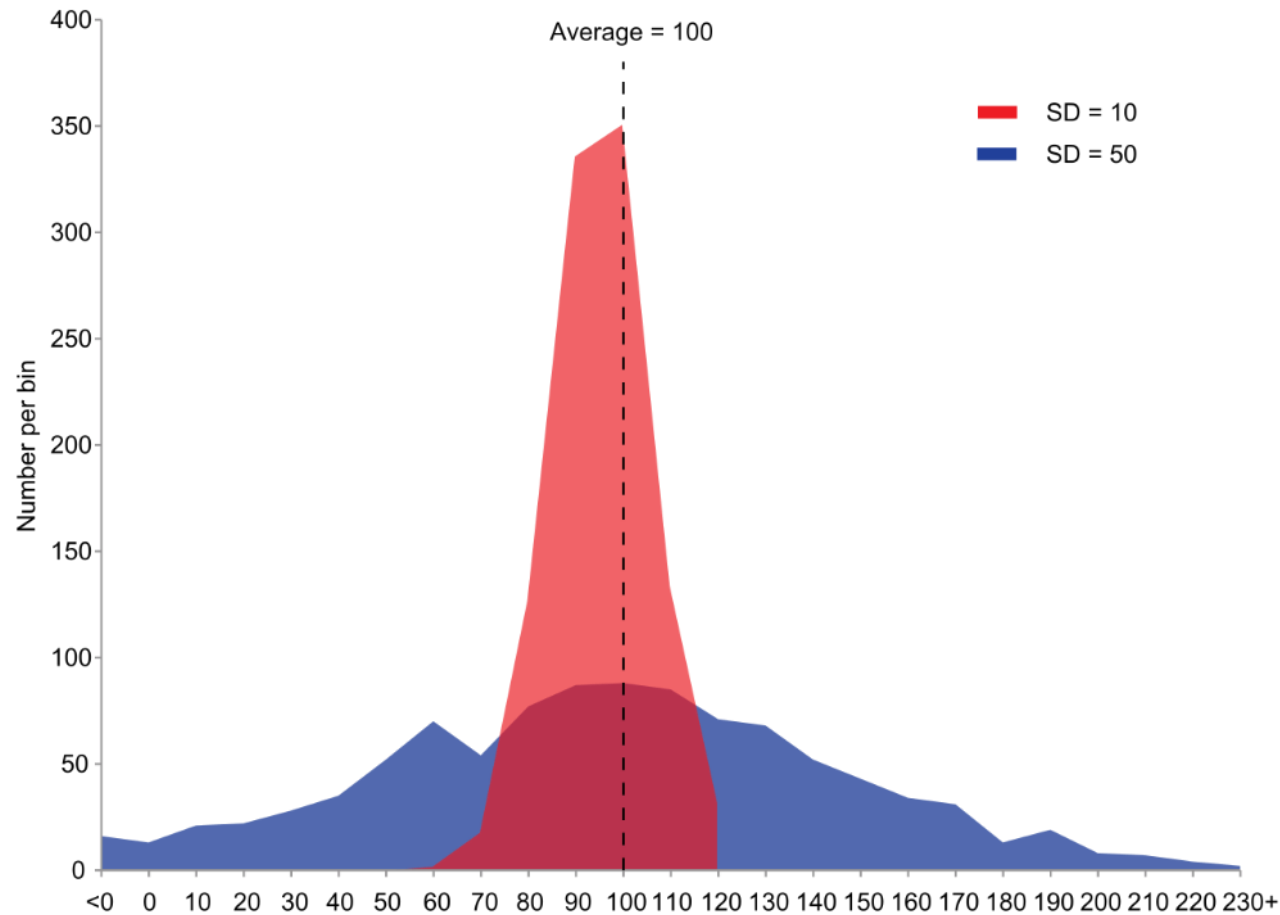
# Summary Statistics using Pandas

```
df.describe()

df[['DepTime', 'DepDelay', 'ArrTime',
'ArrDelay']].agg(['mean', 'min', 'max'])


price_mean = df['price'].mean()

price_median = df['price'].median()

price_std = df['price'].std()

price_var = df['price'].var()

price_quantiles =

    df['price'].quantile([0.25,0.5,0.75])
```

# Motto: Visualize Before Analyzing!

- Data visualization gives us a more holistic sense

- Allows understanding patterns, distributions, and relationships among different features

- *Anscombe's quartet* datasets having the same mean, standard deviation, and regression line, but which are qualitatively different.

  - It illustrates the importance of looking at a set of data graphically and not only relying on basic statistic properties.

`03.eda\`**`4.why-visualization-anscombe.`**`ipynb`

# Data visualization for categorical data

- **Bar Chart**: categories on one axis and the corresponding frequencies or proportions on the other axis



- **Pie Chart**: shows relative size of each category within the whole



`03.eda\`**`2-categorical-variables.`**`ipynb`
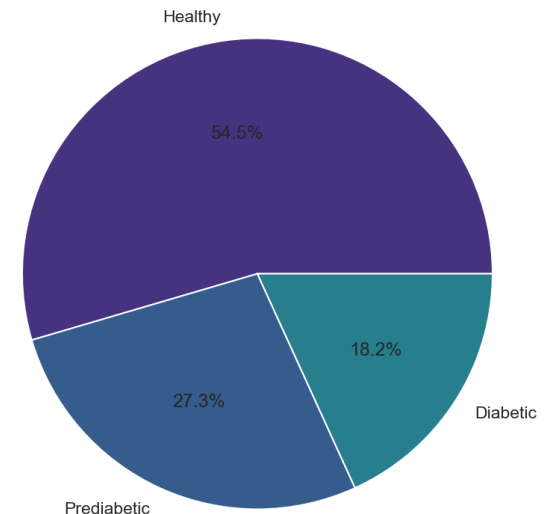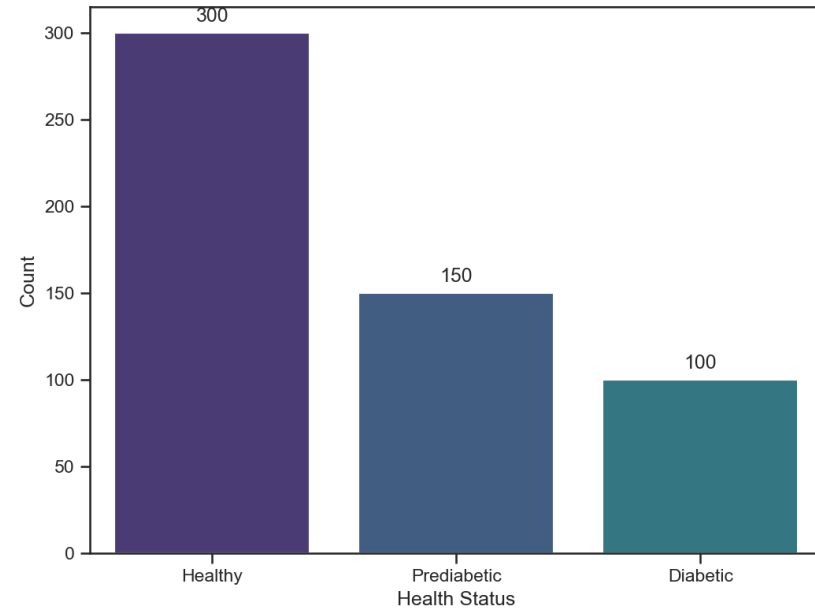
# Data visualization for numerical data

- **Histogram**: represent the distribution of a continuous numerical variable

- **Boxplot**: Depicts the spread and central tendency of the data, including median, quartiles, and potential outliers

- **Line plot**: visualize trends in data over time
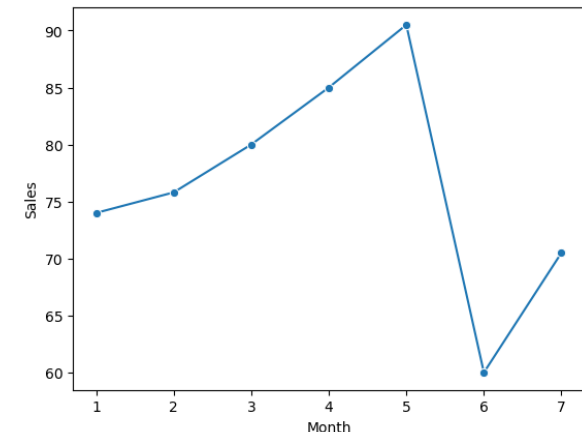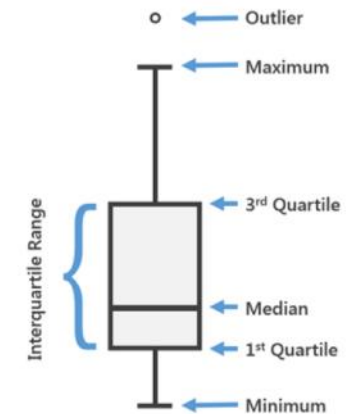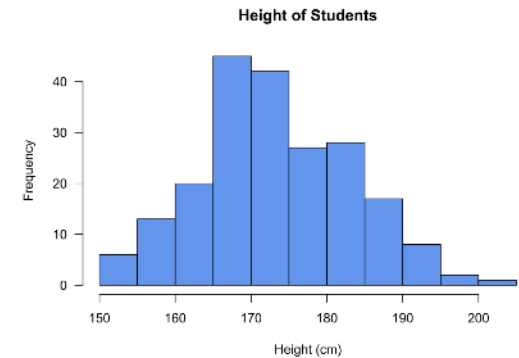
`03.eda\`**`1-numerical-variables.`**`ipynb`

# Histogram to probability distribution

- Divide the count for each interval by the total number of observations in the dataset multiplied by the width of the interval

$$\text{Probability distribution} = \frac{\text{Count for each interval}}{\text{Count of observations in the dataset} \times \text{Width of the interval}}$$
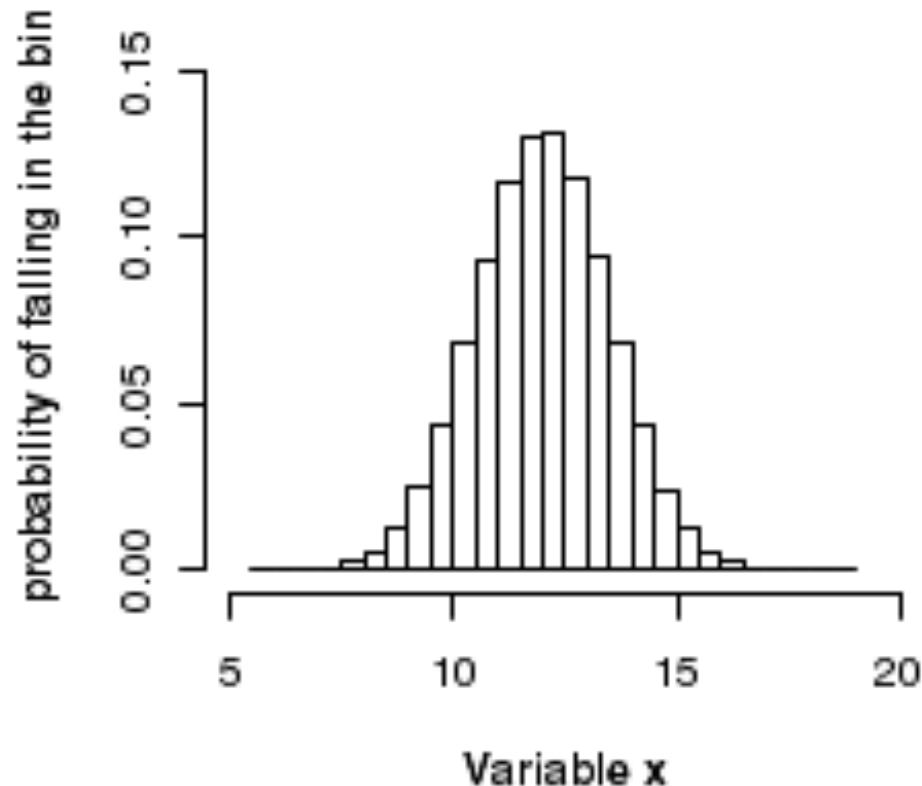
# Normal Distribution

- Many phenomena follow a normal distribution

  o Height, blood pressure, exam scores, etc.
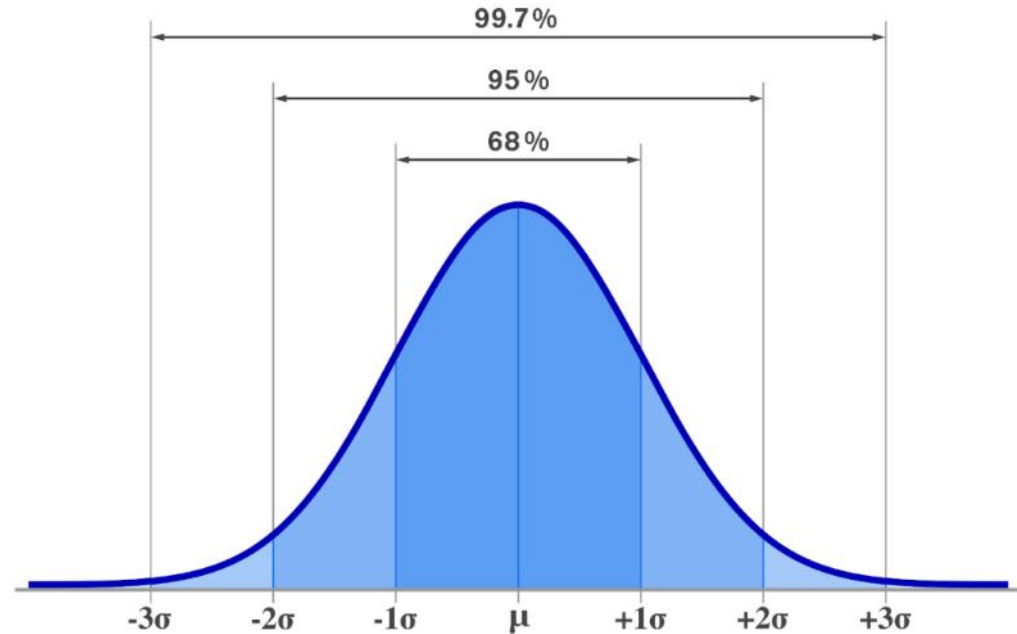


- Symmetric:

  o Most of the observations occur around the central peak

  o Probabilities for values further away from the center decrease equally in both directions

  o Extreme values in both tails of the distribution are similarly unlikely

# Spread Properties of Normal Distribution Curve

- The 68−95−99.7 rule:
  - From μ−σ to μ+σ: contains about 68% of the values
  - From μ−2σ to μ+2σ: contains about 95% of it
  - From μ−3σ to μ+3σ: contains about 99.7% of it
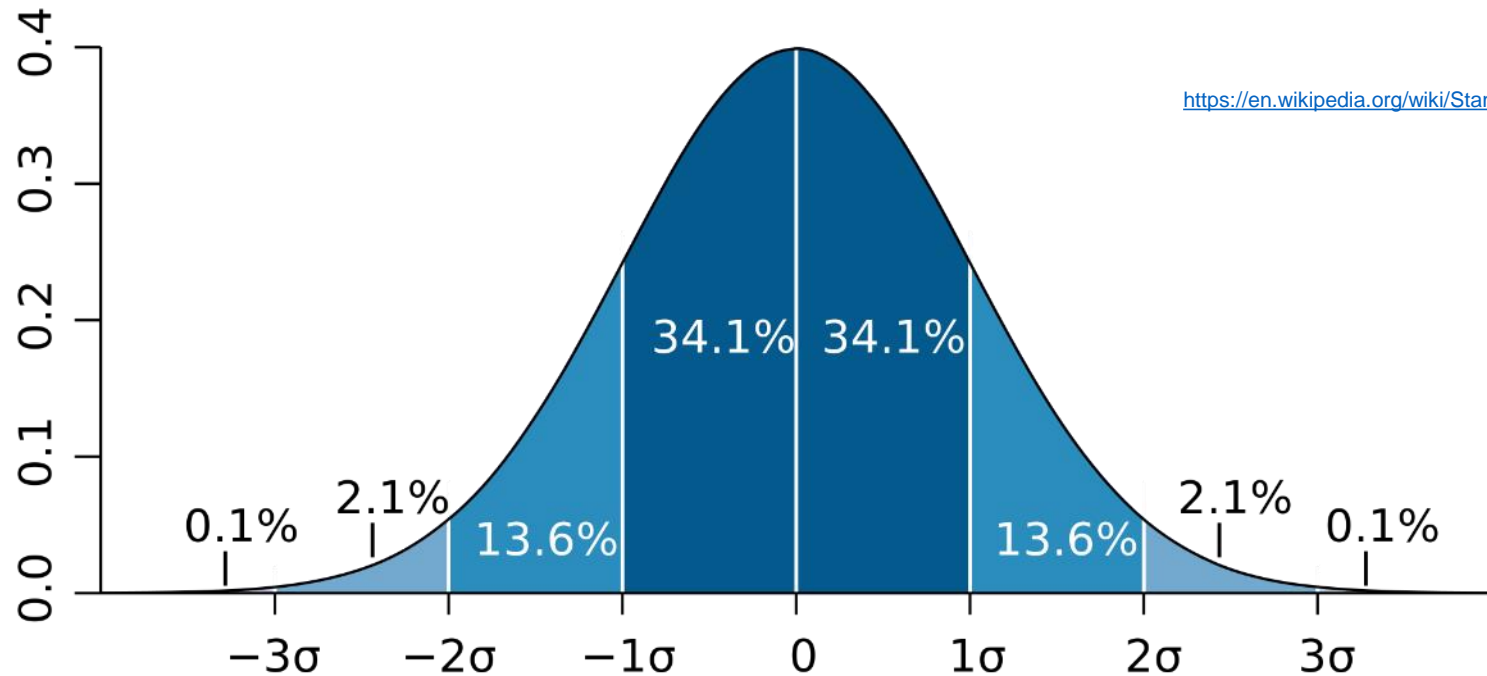
Dark blue is one standard deviation on either side of the mean, or 68% of the values

Mean Median Mode
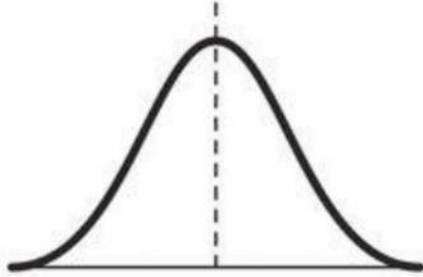
- In normal **symmetric distribution**, the mean, median and mode are the same



Median

Mean — Mode
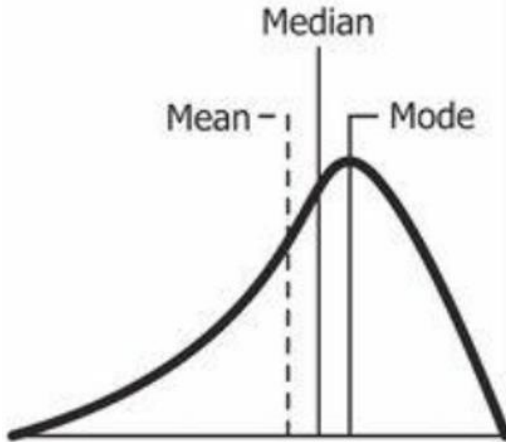
- A distribution is skewed if one of its tails is longer than the other

- A **left-skewed** (negative-skewed) distribution has a long left tail



Median

Mode — Mean
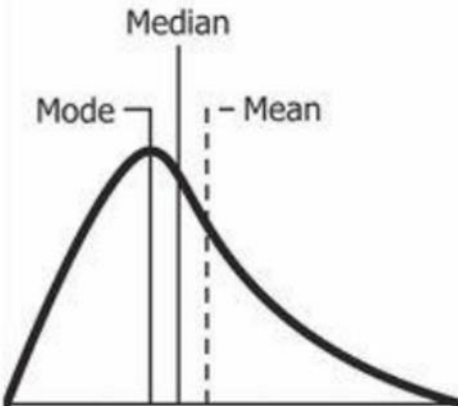
- A **right-skewed** (positive-skewed) distribution has a long right tail

# Coefficient of Skewness

- The coefficient of skewness quantifies the asymmetry of a probability distribution around its mean, indicating how much the data distribution diverges from symmetry
  - Skewness > 0 denotes a right-skewed distribution, where the tail extends towards higher values
  - Skewness < 0 signifies a left-skewed distribution, where the tail extends towards lower values
  - Skewness = 0 signals a perfectly symmetrical distribution
- The formula for skewness is often given as:

$$\text{Skewness} = \frac{3(\text{Mean} - \text{Median})}{\text{Standard Deviation}}$$

pandas

```
skewness = 
df.price.skew()
```

# Detecting Outliers

- An outlier is a data point which is **significantly different** from the remaining data
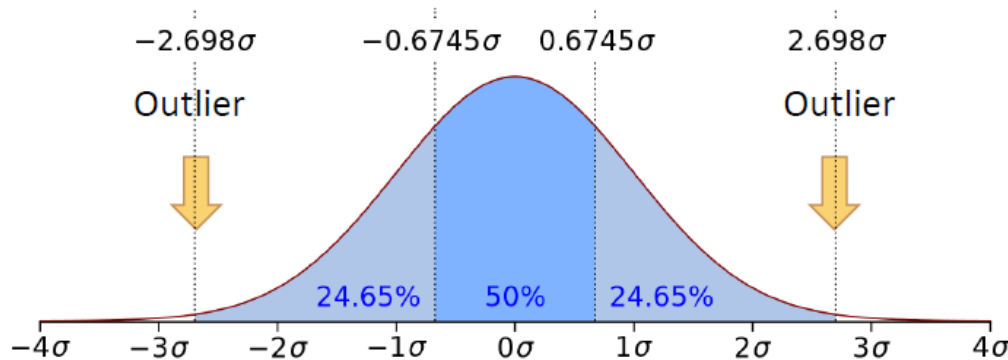


`03.eda\`**`5-univariate-`**
**`eda`**`.ipynb`

- ≈99% of the observations of a normally distributed variable lie within the mean ± 3 x standard deviations

- Values outside mean **± 3 x standard deviations** are considered outliers

# Outliers for skewed distributions
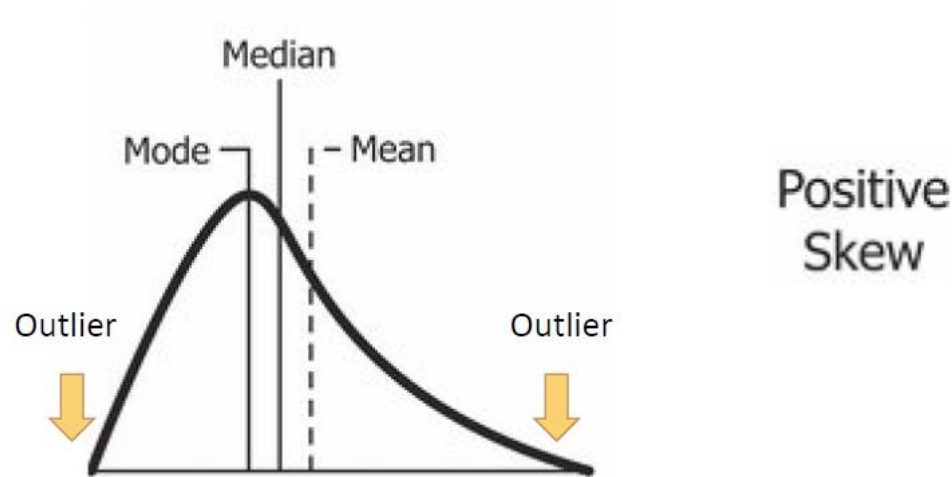


Calculate the quantiles, and then the inter-quantile range (IQR), as follows:

- IQR = 75$^{th}$ Quantile - 25$^{th}$ Quantile
- Upper limit = 75$^{th}$ Quantile + IQR × 1.5
- Lower limit = 25$^{th}$ Quantile - IQR × 1.5
- Values outside the limits are considered **outliers** that can be dropped or replaced by the mean or median

# Visualizing outliers using Boxplots



Images taken from pro.arcgis.com and wiki.commons
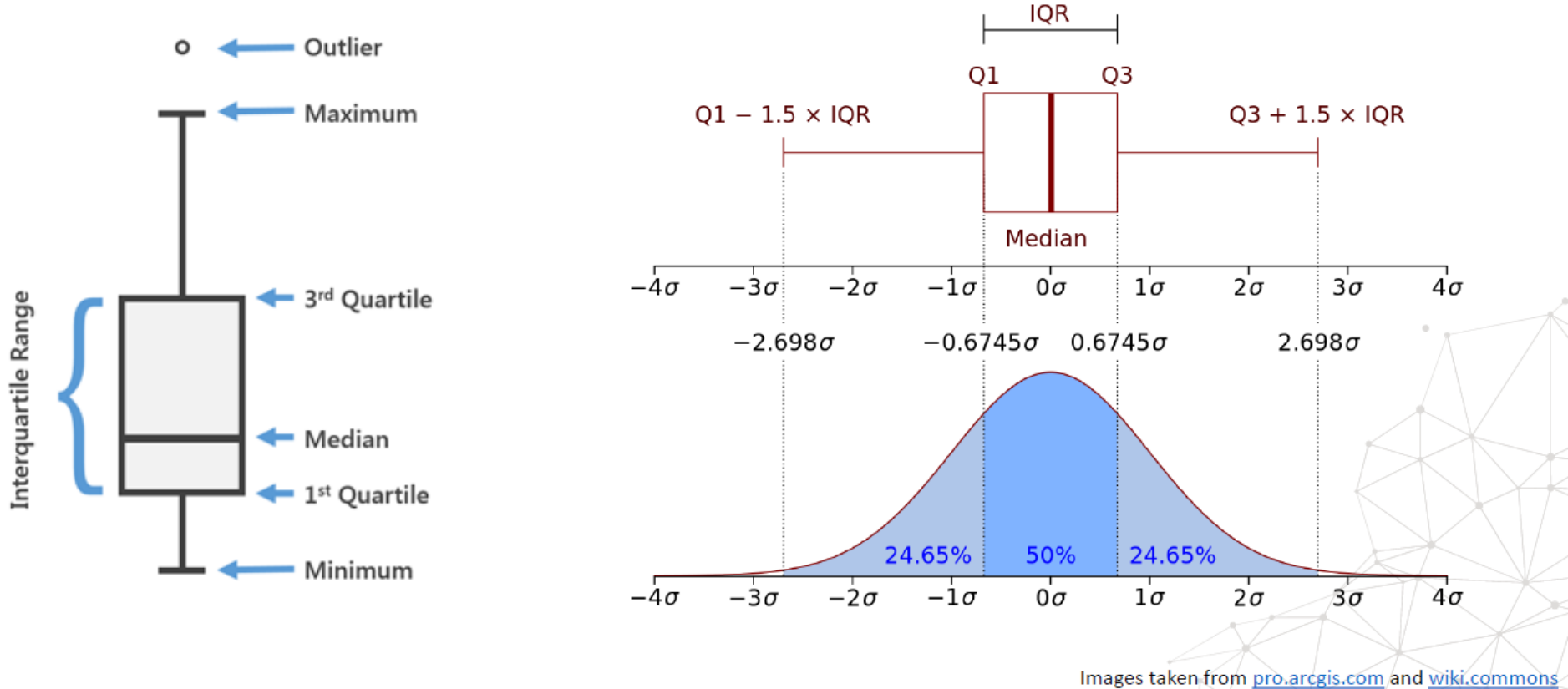
Boxplot: data is represented with a box

- The ends of the box are at the 1st and 3rd quartiles, i.e., the height of the box is IQR
- The median is marked by a line within the box
- Whiskers: two lines outside the box extended to $\pm$ `IQR × 1.5`

# Line Plot - analyzing a Single Variable over Time

- Line Plot display data points connected by straight lines
- It is particularly effective for visualizing trends in data over time
  - E.g., Tracking stock prices over months to identify trends or patterns
- They help identify seasonal patterns such as increasing, decreasing, or cyclical trends
  - E.g., Peak air travel around June-August

e.g., LinePlot visualizes the monthly sales.
We can observe any fluctuations or any seasonal patterns in sales over the course of the year
e.g. upward trend with growth in sales then a sharp drop in the month of June
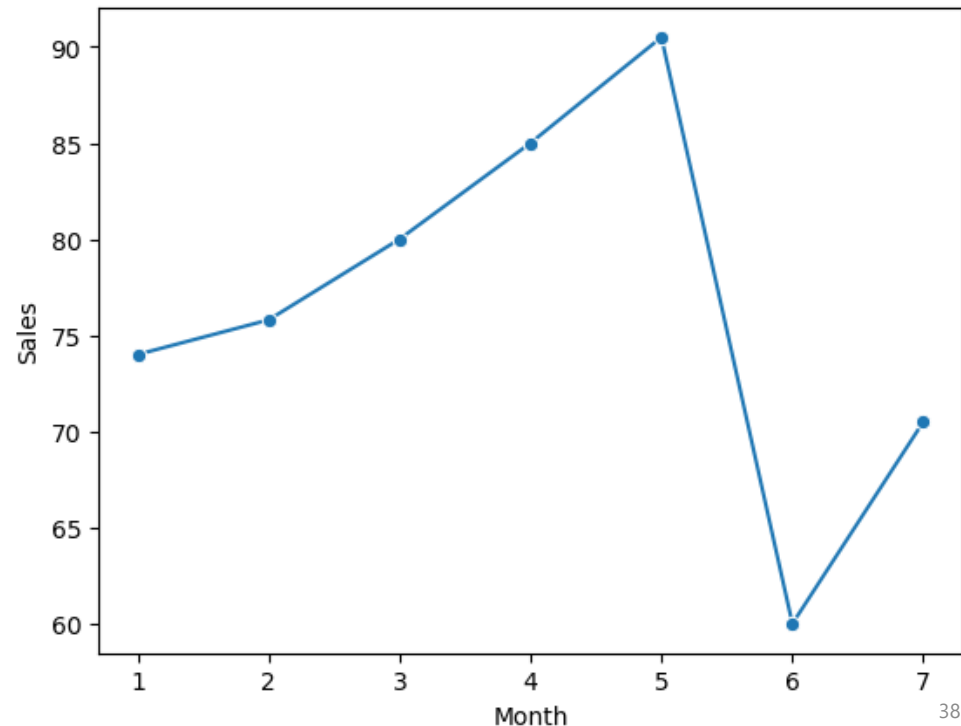
# Data Exploration - Bivariate
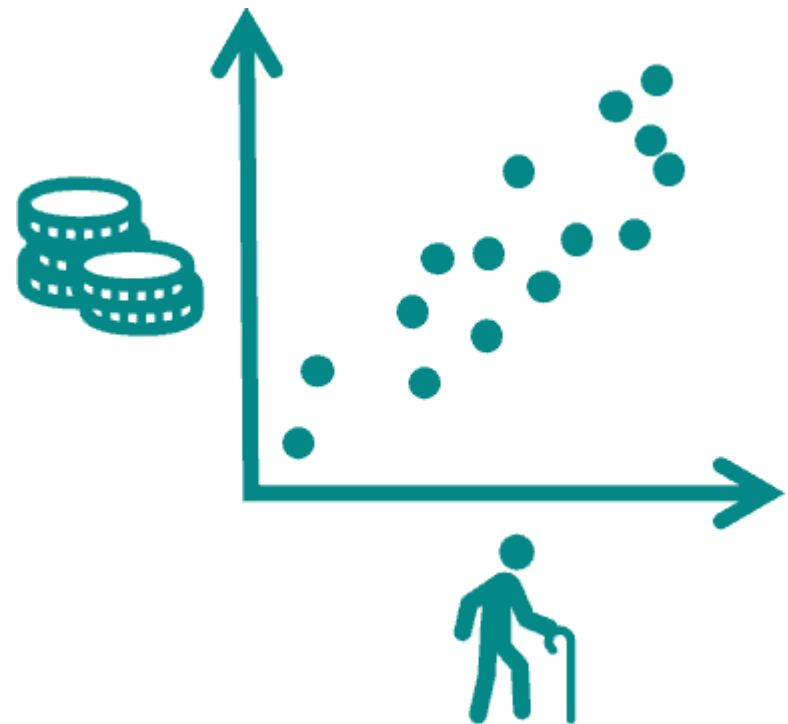
# Types of Bivariate Analysis

- 📈 **Numerical vs. Numerical**: examines the relationship between two numerical variables.
  - E.g.,: In a real estate dataset, we analyze the correlation between the house size in $m^2$ of a house and its price. Are larger houses more expensive?
- 📊 **Categorical vs. Numerical**: explore how a categorical variable affects a numerical one.
  - E.g.,: Studying how the type of car (SUV, sedan, etc.) impacts fuel efficiency (miles per gallon) in an automotive dataset.
  - E.g., Connection between the level of education and income in demographic studies
- 📚 **Categorical vs. Categorical**: focuses on the association between two categorical variables.
  - E.g.,: In an e-commerce dataset, we assess if there's a connection between a customer's gender and their preferred payment method

# Scatter Plot

- **Scatter Plot** visualize the relationship between two continuous numerical variables by plotting one variable along the x-axis and the other variable along the y-axis

- Useful to determine whether there is a correlation between the variables (positive, negative, or none), the strength of the correlation, and the presence of any outliers
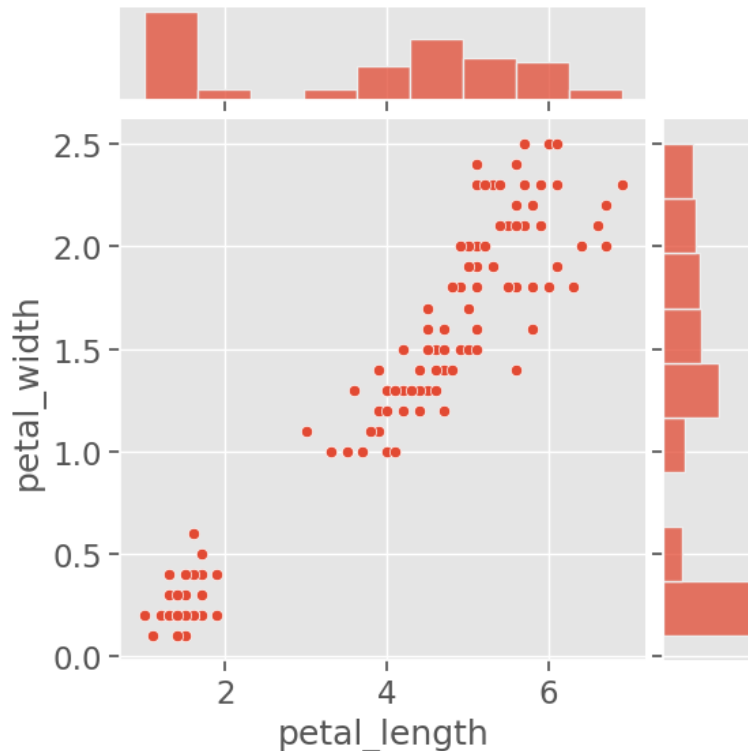
Positive correlation

Negative correlation
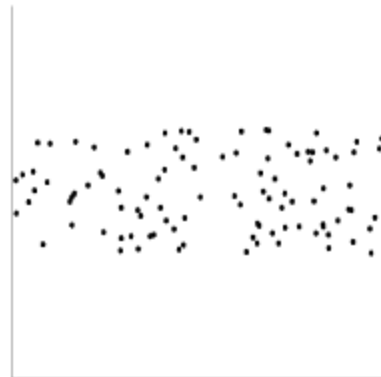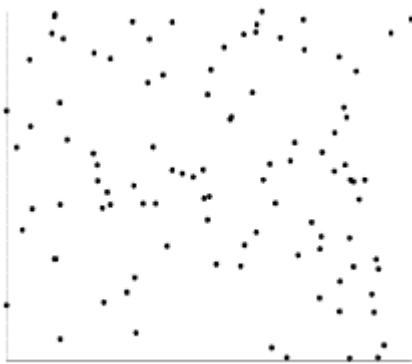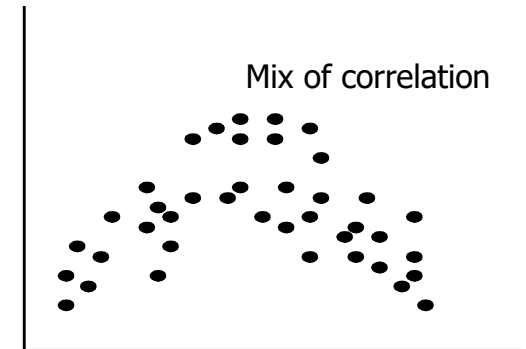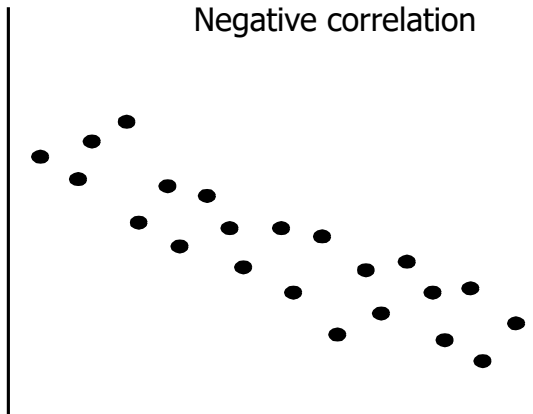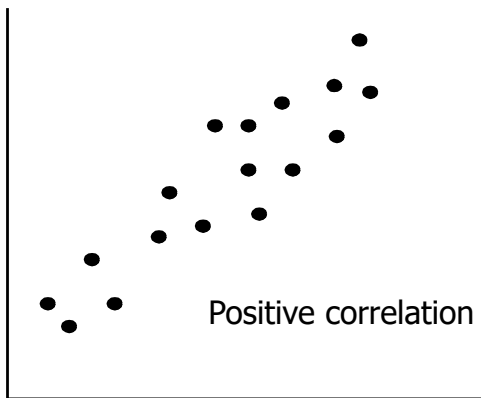
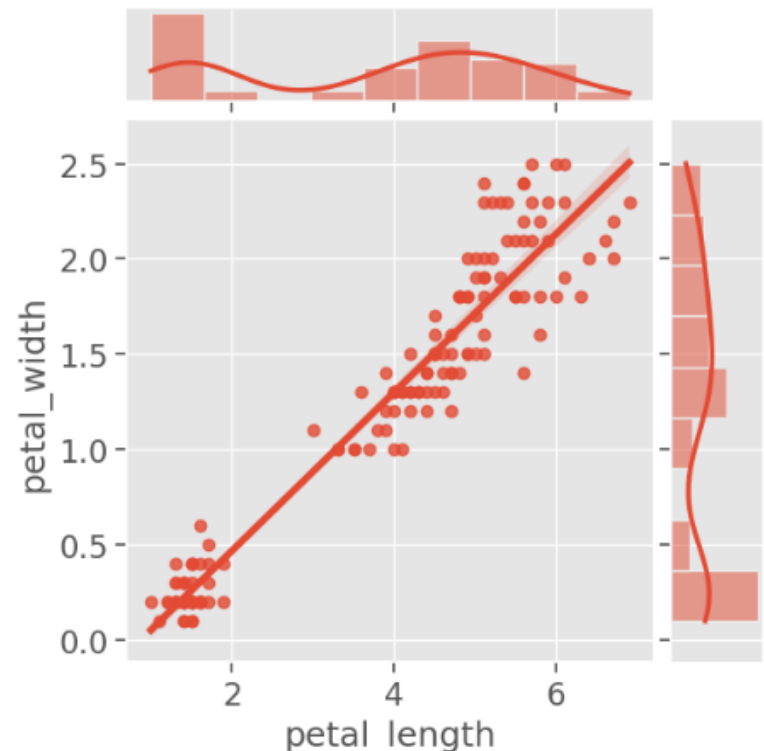Mix of correlation

Not Correlated Data

# Joint Plot -  Visualizing Pairs of Continuous Features

- **Joint Plot** combines multiple plots, such as scatter plot and histogram, to visualize the correlation between the variables, including their individual distributions



**Iris Characteristics: Strong linear relationship between petal length and width**

# Pearson's correlation

- Pearson correlation is a statistical measure that **quantifies the linear relationship between two continuous variables**. It provides insights into how closely related two variables are and the direction of their relationship (positive or negative)
  - The Pearson correlation coefficient, denoted by r, ranges from -1 to 1
  - r=1 indicates a perfect positive linear relationship
  - r=−1 indicates a perfect negative linear relationship
  - r=0 indicates no linear relationship
  - A positive r value suggests that as one variable increases, the other tends to increase as well
  - A negative r value indicates that as one variable increases, the other tends to decrease
- In Python, you can calculate the Pearson correlation coefficient using the corr() function from pandas or pearsonr() function from the scipy.stats module

# Pearson correlation coefficient

$$r = \frac{\sum(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum(X_i - \bar{X})^2 \sum(Y_i - \bar{Y})^2}}$$

**Where:**

- $X_i$ and $Y_i$ are individual data points.
- $\bar{X}$ and $\bar{Y}$ are the means of $X$ and $Y$ respectively.

- The numerator calculates the covariance between X and Y, which measures how they vary together from their means
- The denominator normalizes the covariance by the standard deviations of X and Y, ensuring that the correlation coefficient is scaled appropriately

# Pearson's correlation assumptions

- Pearson's correlation coefficient is a parametric measure of the linear relationship between two continuous variables. It makes certain assumptions about the data, including:

  - **Linearity**: there is a linear relationship between the two variables. If the relationship between the variables is not linear, Pearson's correlation may not accurately reflect the relationship.

  - **Normality**: the data is normally distributed. This means that the distribution of the residuals (the difference between the values) should follow a normal distribution.

  - **Independence**: the observations are independent of one another. This means that the value of one observation does not influence the value of another observation.

- If these assumptions are not met, Pearson's correlation may not accurately reflect the relationship between the variables. In these cases, non-parametric methods, such as Spearman's rank correlation, may be more appropriate

# Spearman's Rank Correlation

- Non-parametric: does not assume a specific distribution of the data

- Measure of the monotonic relations: the variables tend to move in the same direction, but not necessarily at a constant

- Measure the degree of correlation between two variables

- Calculated based on the ranks of the data points instead of the actual values.
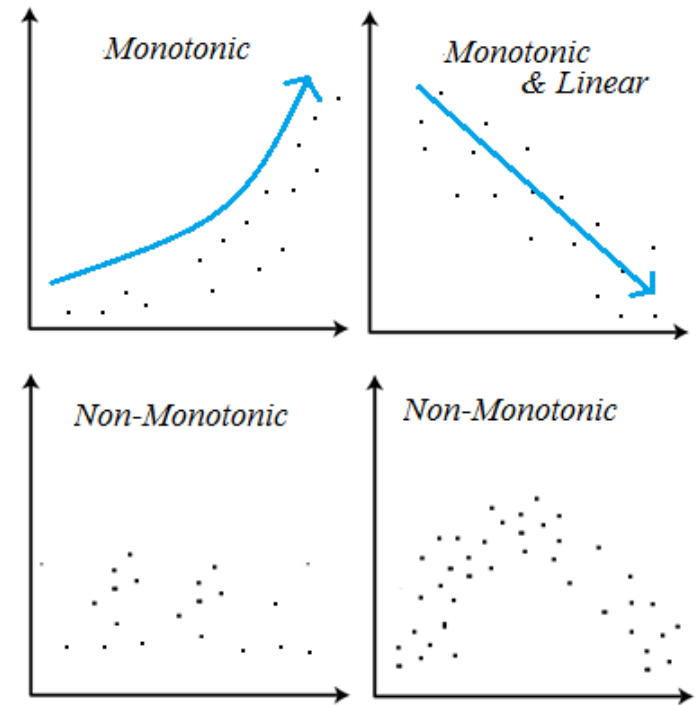


| Students | Maths | Science |
|----------|-------|---------|
| A | 35 | 24 |
| B | 20 | 35 |
| C | 49 | 39 |
| D | 44 | 48 |
| E | 30 | 45 |

| Students | Maths | Rank | Science | Rank | d | d square |
|----------|-------|------|---------|------|---|----------|
| A | 35 | 3 | 24 | 5 | 2 | 4 |
| B | 20 | 5 | 35 | 4 | 1 | 1 |
| C | 49 | 1 | 39 | 3 | 2 | 4 |
| D | 44 | 2 | 48 | 1 | 1 | 1 |
| E | 30 | 4 | 45 | 2 | 2 | 4 |
| | | | | | | 14 |

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

$\rho$ = Spearman's rank correlation coefficient

$d_i$ = difference between the two ranks of each observation

$n$ = number of observations

1 - (6 * 14) / 5(25 - 1) = 0.3

The Spearman's Rank Correlation for the given data is 0.3. The value is near 0, which means that there is a weak correlation between the two ranks.

# Grouped Bar Plots - Numerical by Categories

- **Grouped Bar Plots** visualize and compare how a numerical feature varies across different categories of a single categorical feature. This allows identifying patterns and trends

  - E.g., visualize how the sales of each product category vary across different regions and quarters (using searborn `sns.catplot`)
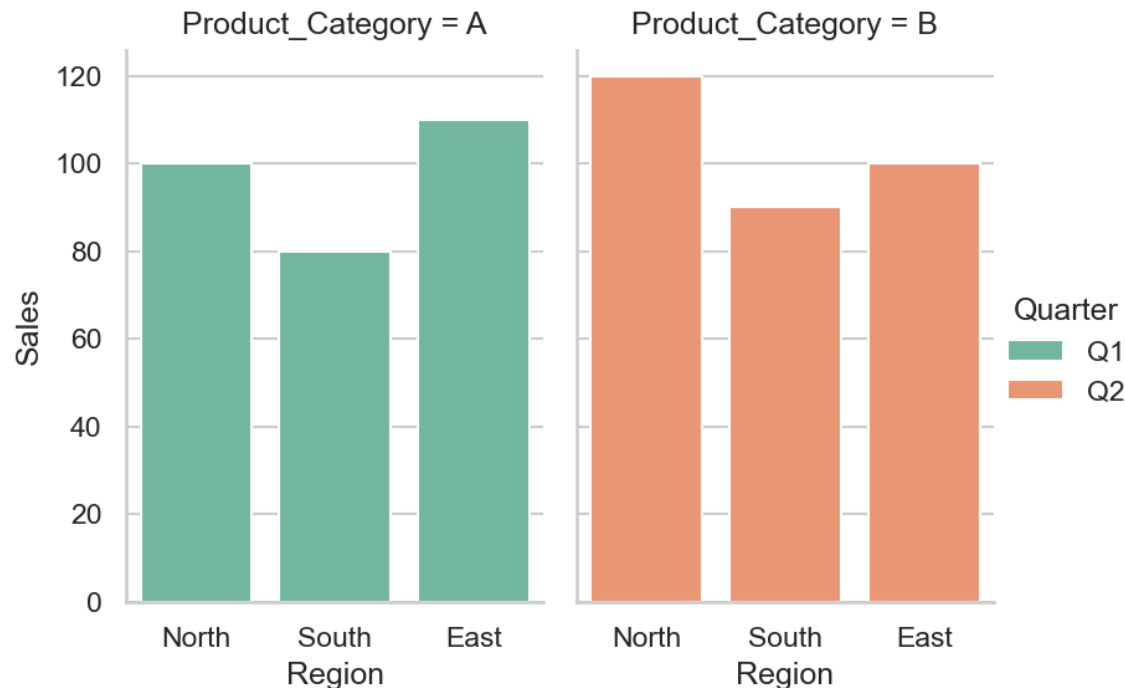
# Stacked Bar Plot - Numerical by Categories

- **Stacked Bar Plot** visualize how a numerical feature varies across a categorical feature while showcasing the contribution of each subcategory to the total (e.g., `df.set_index('Month').plot(kind='bar', stacked=True)`)
  - Each bar in the plot represents the **total value**, and the segments within the bar correspond to contribution of each subcategory to the total

This chart allows for a quick comparison of the sales **composition** across different months and categories



Stacked Bar Plot of Sales by Category
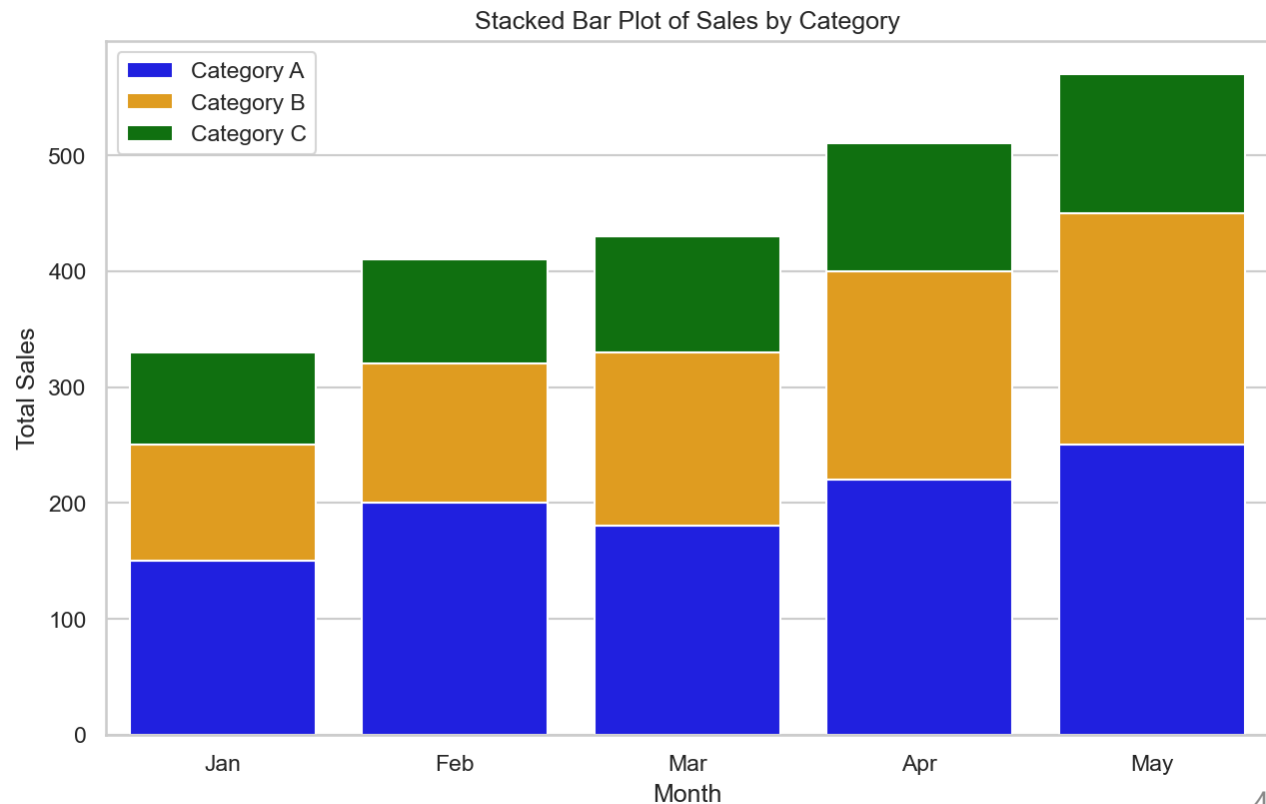
# Grouped Box Plots

- **Grouped Box Plots**, allows comparing the distributions of a continuous variable across **multiple categories**
  - This visualization is particularly useful for identifying patterns, differences, and relationships between the categorical and continuous variables



Distribution of Total Bills by Day
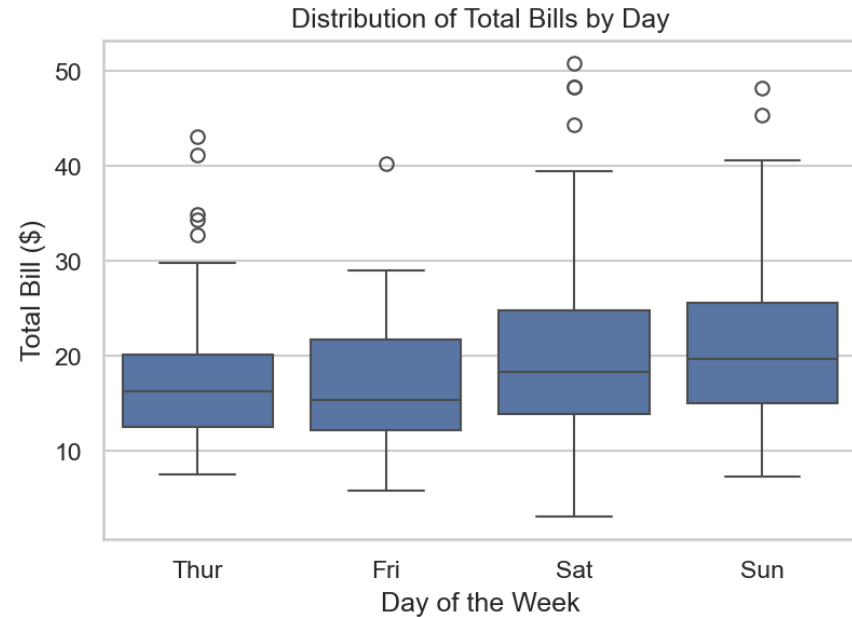
# Data Exploration - Multivariate
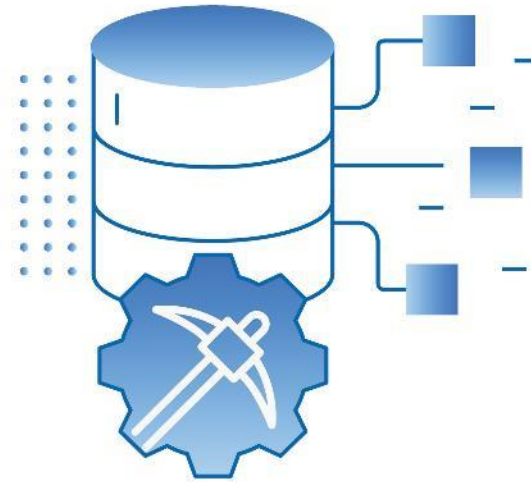
# Scatter Plot matrix

- **A Scatter Plot matrix** (used for Numerical vs. Numerical), also known as a **Pairs Plot**, is used to examine the relationships between multiple variables simultaneously.

  - It consists of a grid of scatter plots where each variable is plotted against every other variable in the dataset
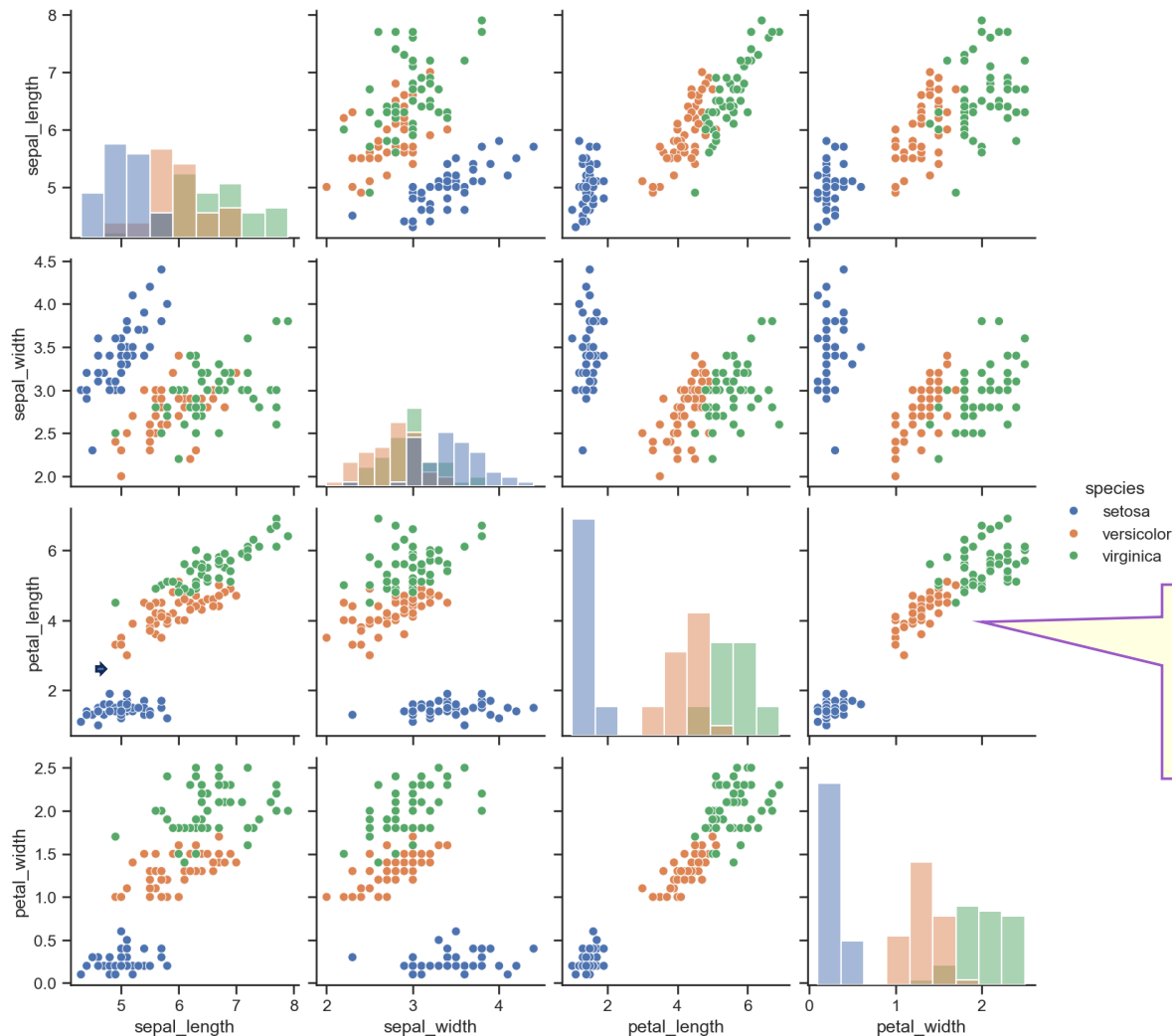


Petal dimensions discriminate species more strongly than sepal dimensions

**Strong linear relationship** between petal length and width

# Correlation Matrix & Heatmap

- A correlation matrix (used for Numerical vs. Numerical) is a table that shows the correlation coefficients between many variables

- A heatmap plots data as a color-encoded matrix

```
cormat = iris.corr(method="pearson")
round(cormat,2)
```

]:

|  | sepal_length | sepal_width | petal_length | petal_width |
|---|---|---|---|---|
| **sepal_length** | 1.00 | -0.12 | 0.87 | 0.82 |
| **sepal_width** | -0.12 | 1.00 | -0.43 | -0.37 |
| **petal_length** | 0.87 | -0.43 | 1.00 | 0.96 |
| **petal_width** | 0.82 | -0.37 | 0.96 | 1.00 |

```
In [65]:  cormat = iris.corr(method="spearman")
          round(cormat,2)
```

Out[65]:

|  | sepal_length | sepal_width | petal_length | petal_width |
|---|---|---|---|---|
| **sepal_length** | 1.00 | -0.17 | 0.88 | 0.83 |
| **sepal_width** | -0.17 | 1.00 | -0.31 | -0.29 |
| **petal_length** | 0.88 | -0.31 | 1.00 | 0.94 |
| **petal_width** | 0.83 | -0.29 | 0.94 | 1.00 |

```
sns.heatmap(cormat)
```



Correlation Heatmap
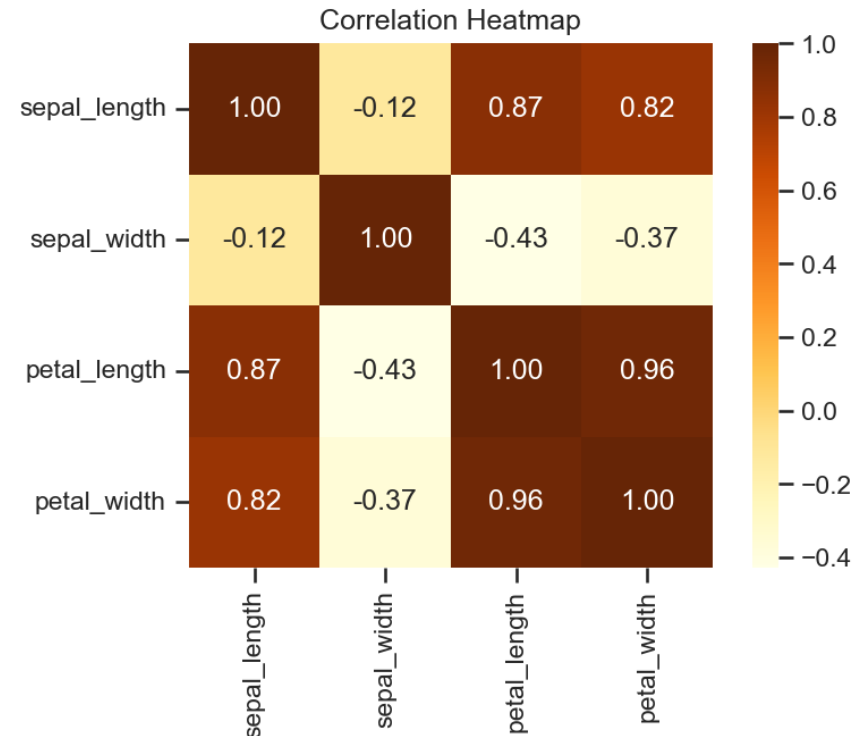
We can observe a strong positive correlation between petal length and width
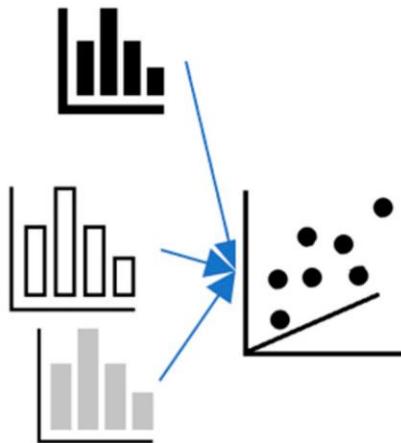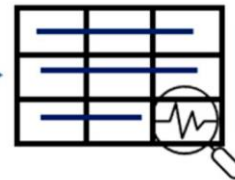
# Summary



**Train ML Model**

**Load Data**    **Univariate EDA**    **Bi/Multivariate EDA**    **Spot Anomalies (e.g., missing data, outliers)**    **Data Preprocessing (DP)**