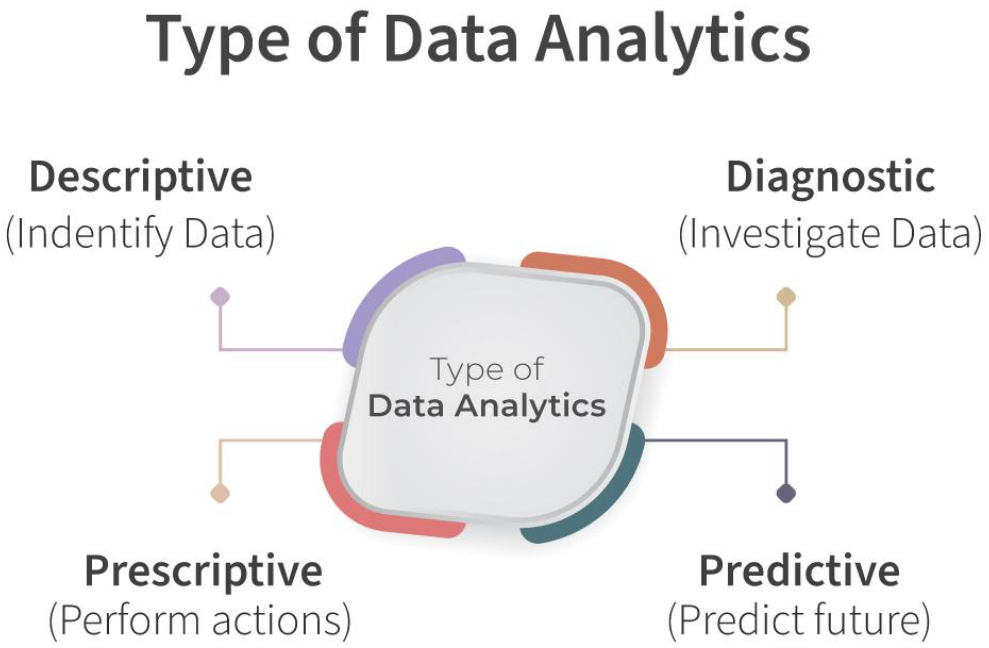
Unit 4
Data Analysis

4.1 Data analytics: Descriptive, diagnostic, predictive and prescriptive analytics
4.2 Exploratory data analysis using descriptive statistics
4.3 Data visualization
4.4 Data visualization techniques
4.5 Principles of effective data visualization
4.6 Feature engineering and other aspects of data manipulation

4.1 Data analytics: Descriptive, diagnostic, predictive and prescriptive analytics

Data Analytics is a process of analyzing raw data to make conclusions and predictions, support decision-making, and uncover patterns and insights.
It involves several types of methods and techniques, classified mainly into descriptive, diagnostic, predictive, and prescriptive analytics.



1. Descriptive Analytics:

Descriptive analysis is considered the beginning point for the analytic journey and often strives to answer questions related to "what happened?". Descriptive analytics is a branch of data analytics that deals with the examination and interpretation of past data to gain insights into what has happened. It involves collecting, summarizing, and presenting historical data in a way to understand patterns, trends, and relationships.
It uses basic statistical techniques such as averages, percentages, and frequency distributions, Data Aggregation, Data visualization etc to present past data.

Example:

- ✓ Sales reports showing monthly revenue trends.
- ✓ Website analytics providing the number of visitors, bounce rates, and page views.

✓ Hospital records summarizing the number of patients treated in a year.

Applications:

1. **Retail:** A retail company wants to analyze its sales over the past year. Descriptive analytics might include:
   - Total sales for each month
   - Average sales per store
   - Total number of products sold
   - Monthly sales growth rates

   This type of analysis will give a clear view of the sales trends and performance over the past year, like identifying the best-selling product or the peak sales season.
2. **Finance:** Summarizing financial statements to assess profitability and expense patterns.
3. **Healthcare:** Generating patient visit reports to track treatment outcomes. Tracking patient admission trends over time.
4. **Social Media:** Measuring follower growth and user engagement metrics like likes, shares, and comments.
5. **Education:** Tracking student performance metrics and attendance over semesters.

2. Diagnostic Analytics:

Diagnostic analytics goes beyond simply describing the data. It seeks to understand why something happened by examining relationships and factors that could have contributed to the observed patterns or trends.

The purpose is to answer the question, "Why did it happen?"

It often involves deeper statistical methods, including correlation analysis, regression analysis, and root cause analysis, to explore relationships between variables and identify potential causes.

Example:

✓ Analyzing why website traffic dropped after a specific campaign.
✓ Investigating the cause of increased churn rates in a subscription service.
✓ Examining factors contributing to a decline in employee productivity.

Applications:

1. **Retail:** Using the retail company's sales data, diagnostic analytics might be used to explore why sales dropped in a particular month. It could involve analyzing:

✓ Whether there was a reduction in marketing spend

- ✓ If inventory shortages occurred
- ✓ Changes in customer buying behavior or competitors' actions

By examining these variables, the company can pinpoint the reasons behind a drop in sales, such as a marketing campaign that didn't perform well or supply chain issues.

2. **Healthcare:** Identifying factors causing an increase in hospital readmission rates.
3. **Manufacturing:** Determining why a specific production line consistently fails quality checks.
4. **Marketing:** Understanding why a recent campaign underperformed.
5. **IT Operations:** Root cause analysis of system outages or performance issues.

3. Predictive Analytics:

Predictive analytics focuses on forecasting future events based on historical data. By applying statistical models and machine learning algorithms, it tries to predict what is likely to happen next.

The purpose is to answer the question, "What is likely to happen?".

Predictive analytics often uses regression models, time series analysis, machine learning models (e.g., decision trees, neural networks), and statistical algorithms to make predictions based on past data.

Examples:

- ✓ Predicting customer purchase behavior using past purchase data.
- ✓ Forecasting energy demand for the upcoming year.
- ✓ Anticipating disease outbreaks based on trends and patterns.

In the context of the retail company, predictive analytics could be used to forecast sales for the upcoming year. For example:

- ✓ Predicting monthly sales based on past trends
- ✓ Estimating future demand for a specific product during holiday seasons
- ✓ Predicting customer churn based on past behavior

This prediction helps the company plan ahead for stock, staffing, and marketing strategies.

Applications:

1. **Retail:** Forecasting inventory needs based on seasonal demand patterns.
2. **Finance:** Predicting stock price movements using historical market data.

3. **Healthcare:** Predicting patient risk for chronic diseases using health records.
4. **Transportation:** Anticipating traffic congestion to optimize routes.
5. **Sports:** Predicting player performance for team selection and strategy development.

4. Prescriptive Analytics:

Prescriptive analytics goes a step further than predictive analytics by recommending actions or strategies to achieve desired outcomes. It not only predicts what might happen but also suggests the best course of action to optimize results.

The purpose is to answer the question, "What should we do about it?".

It involves the use of optimization models, simulations, decision analysis, and machine learning algorithms to suggest the best possible actions. It may also incorporate business rules and constraints to recommend decisions.

Example:

- ✓ Recommending the best supply chain strategy to minimize costs.
- ✓ Optimizing flight schedules to reduce delays and maximize profits.
- ✓ Suggesting personalized product recommendations on e-commerce platforms.

Applications:

1. **Retail:** For the retail company, prescriptive analytics could help determine:
- ✓ How much inventory to order for the next season based on predicted demand
- ✓ The optimal price point to maximize sales or profit
- ✓ The best marketing channels to use based on predicted customer behavior

For example, prescriptive models could suggest adjusting pricing dynamically during peak sales periods or directing resources to the most profitable stores.

2. **Finance:** Recommending portfolio adjustments to maximize returns while minimizing risk.
3. **Healthcare:** Suggesting the best course of treatment based on patient-specific data.
4. **Logistics:** Optimizing delivery routes for cost and time efficiency.
5. **Energy:** Balancing power grids based on real-time consumption data.

4.2 Exploratory data analysis using descriptive statistics:

Exploratory Data Analysis (EDA) is a crucial step in the data analysis process that allows you to understand, summarize, and visualize the key characteristics of a dataset before applying more complex statistical modeling or machine learning techniques. The main objective of EDA is to gain a better understanding of the data's structure, detect any patterns or anomalies, test assumptions, and identify potential relationships between variables.

Measures of Central Tendency:

- ✓ Mean:
  - The mean is most commonly used in datasets with symmetric distributions.
  - It is useful when you want a single summary value that reflects the "center" of the data.
  - The mean is sensitive to outliers (extremely high or low values), which can skew it significantly.
- ✓ Median:
  - The median is ideal for skewed distributions or when outliers are present, as it is less affected by extreme values.
  - The median does not account for the exact values of all the data points, just their position.
  - In very large datasets, finding the median can be more computationally expensive than the mean.
- ✓ Mode:
  - The mode is useful for categorical data or discrete data where you want to identify the most common category or value. It can be used for identifying trends or patterns in a dataset.
  - A dataset might have no mode if all values are unique, or multiple modes, which can make interpretation difficult.
  - It may not provide a comprehensive measure of central tendency in continuous data.

Measures of Dispersion:

Measures of dispersion describe the spread or variability within a dataset. These measures indicate how much individual data points differ from the central tendency (mean, median, or mode) of the data. The primary measures of dispersion include range, variance, standard deviation, and interquartile range (IQR).

- ✓ Range:
  - Simple to compute and provides a quick sense of how spread out the

data is.

- Useful for initial data exploration to understand the extent of variation in the dataset.
- Highly sensitive to outliers as a single extreme value can significantly increase the range.
- Does not provide information about how values are distributed between the minimum and maximum.

✓ Variance:

- Useful when working with statistical models, particularly those that assume normally distributed data.
- Variance is in squared units, making it difficult to interpret directly in the context of the original data.
- Sensitive to extreme values (outliers), as the squared deviations amplify large differences.

✓ Standard Deviation:

- Helps in comparing the dispersion of different datasets with the same unit of measurement.
- Like variance, the standard deviation is sensitive to outliers and may be misleading when the data contains extreme values.

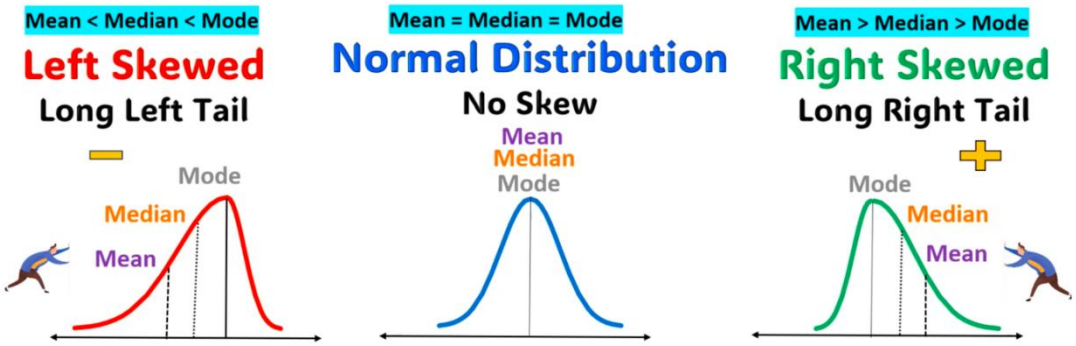
✓ Interquartile Range :

-Use when your data may have outliers or you want to focus on the central spread of the data. The IQR is robust and is useful in situations where the presence of outliers may distort other measures of dispersion.

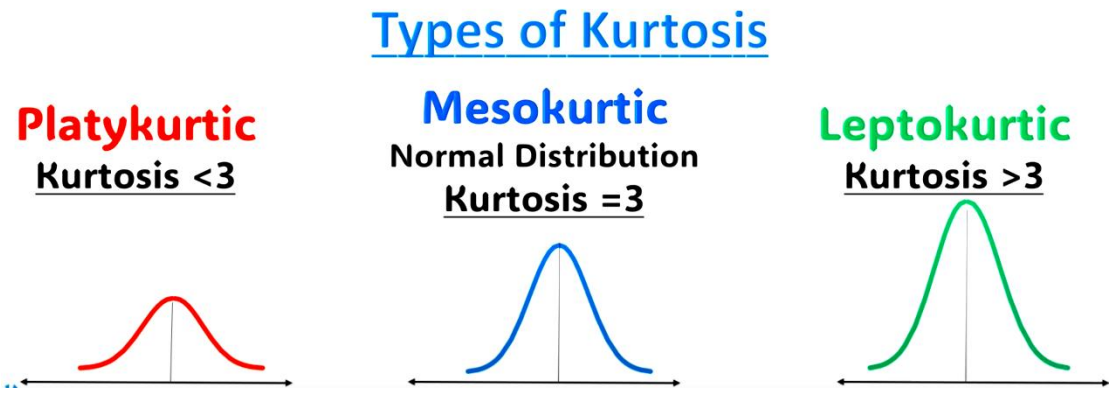
- Does not provide a complete picture of the data's overall spread, as it ignores values outside the interquartile range.

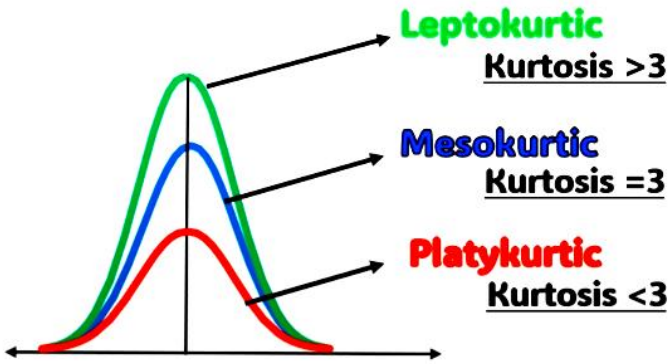
Distribution analysis:

✓ Skewness: Skewness is usually a measure of the symmetry of a dataset and helps to understand how values are spread around them.



✓ Kurtosis: Kurtosis measures the peak or height of data distribution and helps us to understand the shape of frequency distribution.





Histogram

A histogram is a bar chart where X-axis represents intervals (or bins) of data values and Y-axis represents the frequency (or count) of data points within each interval.

Steps to Create a Histogram:

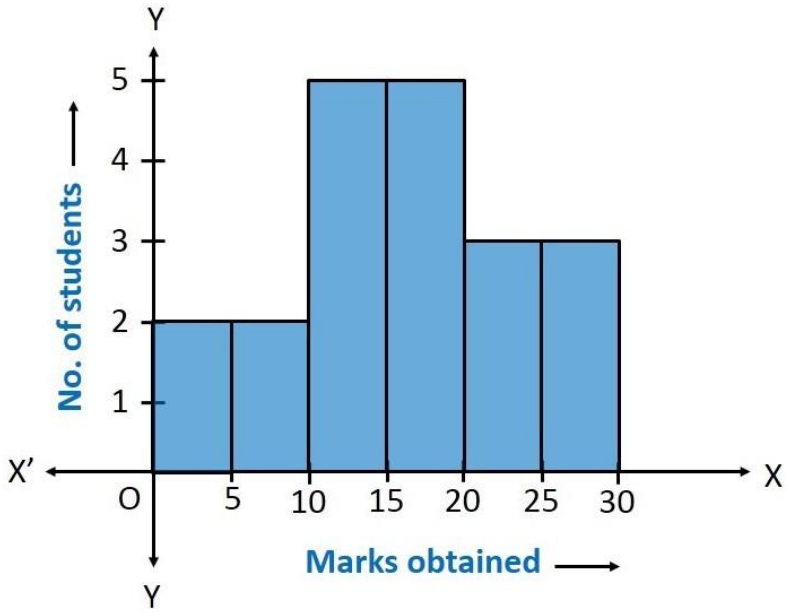
1. Divide the data range into intervals (bins).
2. Count the number of data points in each bin.
3. Plot the bins on the X-axis and their frequencies on the Y-axis.

Interpreting Histograms

1. **Shape of Distribution:**
   Normal (Bell Curve): Symmetrical, peaks at the mean.
   Skewed: Longer tail on one side.
   Uniform: Bars are roughly the same height.
   Bimodal/Multimodal: Multiple peaks indicate clusters in data.
2. **Spread:** Wide histograms suggest high variability, while narrow ones indicate low variability.

3. **Outliers:** Gaps or isolated bars on the ends of the histogram may signify outliers.

Intervals	Frequency
0 - 5	2
5 - 10	2
10 - 15	5
15 - 20	5
20 - 25	3
25 - 30	3



Correlation analysis and data relationships

Correlation analysis is a statistical method used to evaluate the strength and direction of the relationship between two or more variables. It helps identify patterns, dependencies, or connections within data, providing valuable insights for research, decision-making, or modeling

Types of Correlation:

- ✓ **Positive Correlation:** Both variables move in the same direction. For example, there is a positive correlation between smoking and lung cancer.
- ✓ **Negative Correlation:** Variables move in opposite directions. For example, there is a negative correlation between exercise and obesity.
- ✓ **No Correlation:** No relationship between the variables. For example, there is no correlation between shoe size and IQ.

Methods to calculate correlation:

1. **Pearson Correlation Coefficient (r):** Suitable for continuous variables that have a linear relationship.

Formula:

$$r = \frac{n\sum xy - \sum x \sum y}{\sqrt{(n\sum x^2 - (\sum x)^2)(n\sum y^2 - (\sum y)^2)}}$$

where:

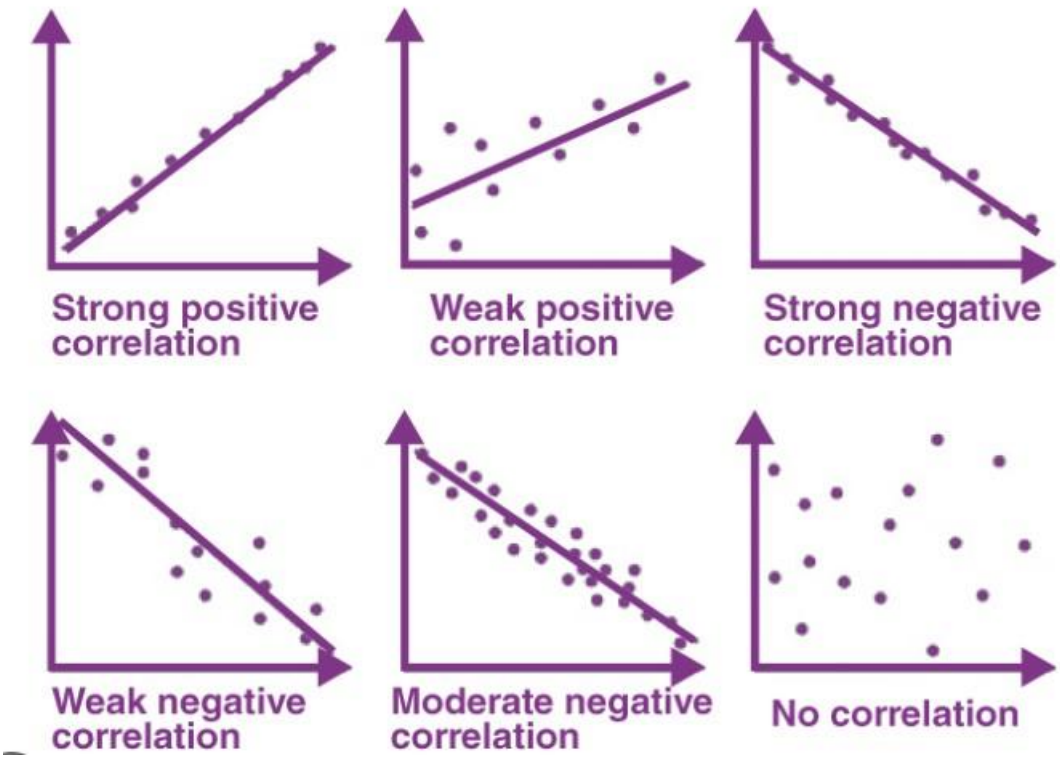
- x and y are the two variables.
- $\sum x, \sum y, \sum xy, \sum x^2, \sum y^2$ represent sums of the variables.
- n is the number of data points.

Interpretation:

r=1: Perfect positive correlation.

r=0: No correlation.

r=−1: Perfect negative correlation.



Assignment: Numerical on **Pearson Correlation Coefficient**

X	1	2	3	4	5
Y	10	20	30	40	50

2. Spearman Rank Correlation Coefficient: Used when variables are ordinal or when there is a non-linear relationship.

Correlation vs. Causation

It is important to note that correlation does not imply causation. Just because two variables are correlated, it does not necessarily mean that one causes the other. There may be a third variable, known as a confounding variable, that is responsible for the correlation.

4.3 Data visualization

Data visualization is the presentation of data in graphical format. Data visualization is a generic term used which describes any attempt to help understanding of data by providing visual representation.
It allows data to be presented in a format that is visually intuitive, making patterns, trends, and outliers easier to identify.
However, the type of visualization selected should be aligned with the nature of the data and the specific goals of the analysis.

- ✓ Categorical Data: Bar Graph, Pie charts, Stacked Bar/Column Chart etc.
- ✓ Quantitative Data: Histograms, Box plot etc.
- ✓ Time Series Data: Line Chart, Area chart etc.
- ✓ Relationships Between Variables: Scatter Plot, Bubble chart
- ✓ Geospatial Data: Heat map, Choropleth Map
- ✓ Multidimensional Data: Radar Chart, Parallel Coordinates Plot

Importance of Data Visualization

Data visualization is crucial because it enables us to make sense of complex and often large datasets.

- ✓ Visualization simplifies this by transforming it into easily interpretable formats, such as charts, graphs, and maps, helping viewers grasp insights quickly.
- ✓ Visual representations make patterns, trends, and outliers more apparent.
- ✓ Faster Decision Making
- ✓ Accessibility and Engagement: Effective visualizations make data accessible to a wide audience, including those without technical backgrounds. It also improves engagement, encouraging users to explore and interact with data.

Types of Data Visualization

Data visualization is used to analyze visually the behavior of the different variables in a dataset, such as a relationship between data points in a variable or the distribution.

1. **Explanatory Data Visualization:** Visualization used to communicate insights clearly and effectively to others, often for reporting, decision-making, and storytelling. It is meant for non-expert audience, having no background knowledge of the subject matter or anyone who needs clear insights from data.

Examples of Explanatory Visualization:

- ✓ Bar Charts: Compare discrete categories (e.g., sales by product category, number of visitors by region).
- ✓ Line Charts: Show trends over time (e.g., revenue growth, stock price changes, monthly sales trends).
- ✓ Pie Charts: Display proportions or percentages (e.g., market share, distribution of budget allocations).
- ✓ Scatter Plots: Visualize relationships between two continuous variables (e.g., age vs. salary, advertising spend vs. conversions).
- ✓ Heat Maps: Show intensity or concentration (e.g., sales by region, customer density, geographical hotspots).
- ✓ Tables and Dashboards: Organize data into structured, digestible formats (e.g., sales performance dashboard, financial summary reports).

Excel, Tableau, Power BI, Google Data Studio, or any visualization tool aimed at producing professional, final reports.

2. **Exploratory Data Visualization:** Visualization used to uncover patterns, relationships, or anomalies in the data that might not be obvious. Often used for data analysis and hypothesis generation. It is meant for data analysts, data scientists, or users who need to discover underlying patterns, trends, or anomalies.

Examples of Exploratory Visualization:

- ✓ Histograms: Show the distribution of a single continuous variable (e.g., distribution of test scores, age distribution).
- ✓ Box Plots: Identify outliers and understand data distributions (e.g., identifying the spread of salary, detection of unusual spending behavior).
- ✓ Correlation Plots: Visualize relationships between two or more continuous variables (e.g., age vs. income, advertising spend vs. conversion rates).

- ✓ Density Plots: Provide insights into the distribution shape of continuous variables (e.g., population density, age distribution in a population).
- ✓ Faceted Plots: Break down data by categories to explore subgroups (e.g., gender-wise sales performance, product-wise trends).
- ✓ Parallel Coordinates Plots: Show relationships between multiple variables at once (e.g., comparing customer attributes like age, income, and purchasing habits).
- ✓ Interactive Dashboards: Enable exploration of data through filters, drill-downs, and sliders to uncover patterns (e.g., exploring customer segmentation, product recommendations, or sales trends).

Python (Pandas, Matplotlib, Seaborn, Plotly), R (ggplot2, d3.js for interactive visualizations), Jupyter notebooks, or software with advanced analytical capabilities like Tableau, Power BI, or custom scripts for interactive exploration.

Infographics and Visualization

Infographics are visual representations that combine data, text, and design to tell a cohesive story or explain complex concepts in an engaging and easy-to-understand format. They often include charts, icons, illustrations, and narrative text. For example, a timeline infographic might showcase the evolution of a technology, using visuals and data points.

On the other hand, visualization refers to transforming raw data into graphical forms like charts, graphs, or maps to explore patterns, trends, and relationships. These data-driven graphical representations, such as scatter plots or heatmaps, are primarily used for data analysis and presenting insights.

Following are the difference among them:

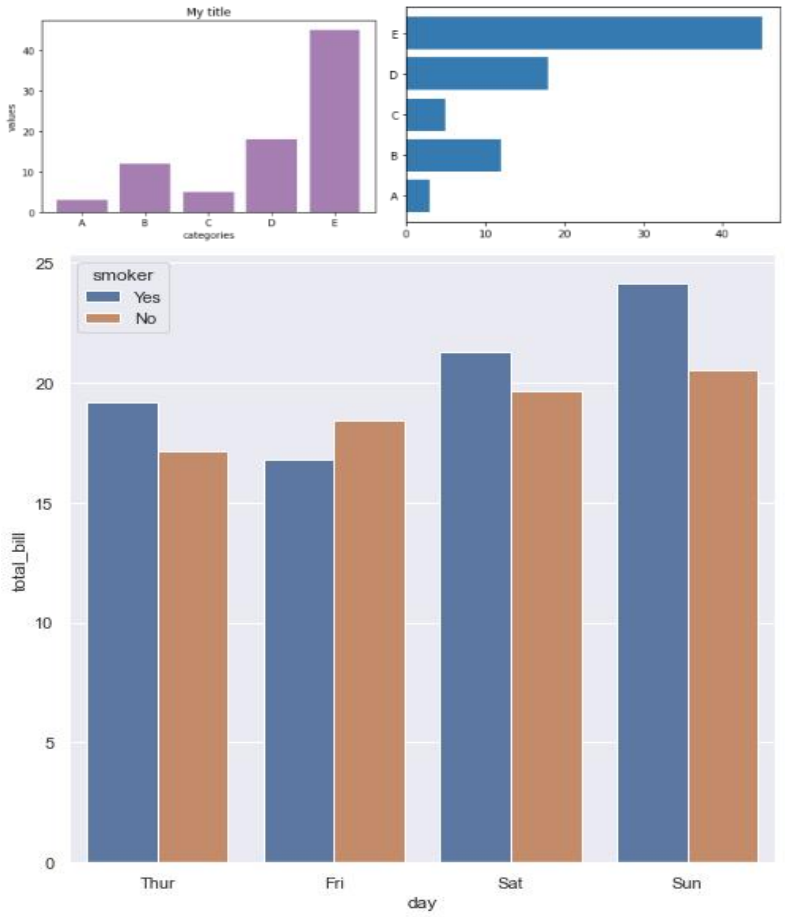
Aspect	Data Visualization	Infographics
Scope	Focuses exclusively on visually representing data, transforming raw information into graphical forms such as charts, graphs, and maps.	Encompasses a broader range of visual communication techniques, including data visualization, text, icons, and images to convey information comprehensively.
Purpose	Aims to simplify complex data sets so that users can rapidly recognize patterns, trends, and insights.	Aim to communicate information, statistics, or expertise clearly and compellingly, frequently combining visual components and text to produce a complete story.
Components	Primarily graphical elements like charts, graphs, diagrams.	Includes data visualization, icons, images, written explanations.
Level of Detail	Emphasizes data patterns and relationships, even with many data points.	Condenses information to provide an overview or highlight key insights.
Engagement	Targets analytical viewers.	Engages a broader audience with visual appeal and storytelling.
Context & Explanation	May lack additional context.	Offers context, explanations, and guidance for better comprehension.
Narrative Element	Lacks a distinct narrative structure.	Integrates data visualization into a larger narrative.
Usage Complexity	Involves complex techniques for precise data representation.	Prioritizes simplicity and clarity for quick understanding.
Outcome	Provides in-depth analysis and insights.	Offers a quick overview for grasping main points and insights.

4.4 Key Data Visualization Techniques

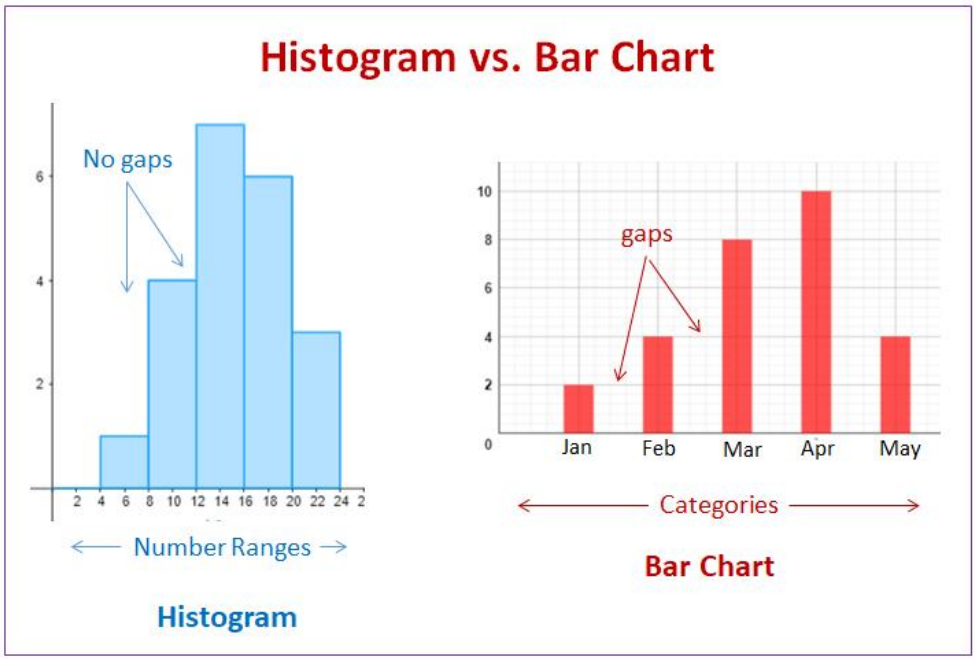
- ✓ **Charts:** Visual representations of data, such as bar charts, line graphs, scatter plots, etc.
- ✓ **Plots:** Graphical representations of data that highlight relationships, distributions, and trends.
- ✓ **Dashboards:** Interactive interfaces combining multiple visualizations to monitor and analyze data effectively.

1. Line plots: One of the most used visualizations, line plots are excellent at tracking the evolution of a variable over time or continuous data. It show relationships between continuous variables, not necessarily time-dependent. X-axis represents one continuous variable and Y-axis represents another continuous variable. e.g., comparing temperature and humidity.

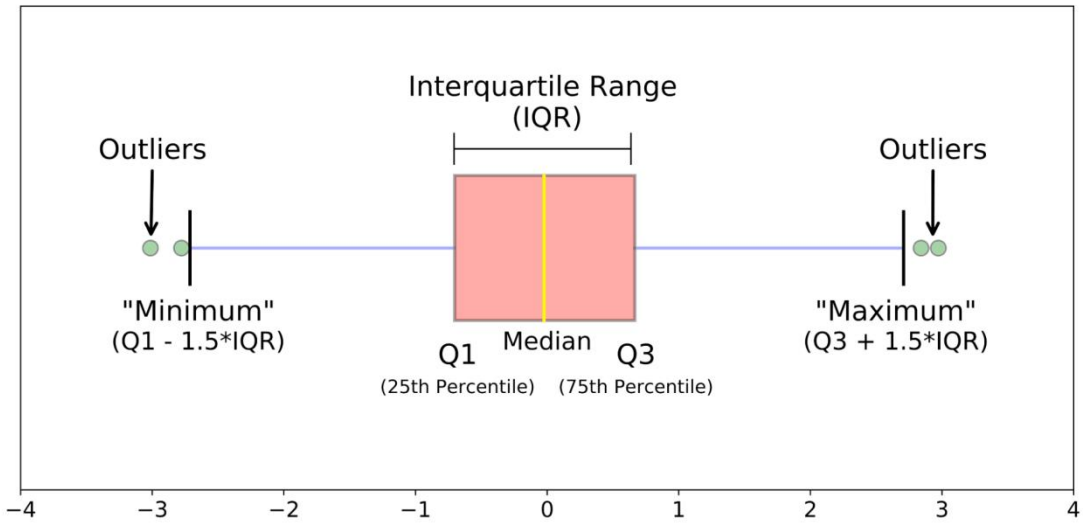
2. Bar plots: Another common visualization—one you'll no doubt be familiar with from school—is the bar chart. Bar charts are a simple but highly effective way of plotting categorical data against discrete values.



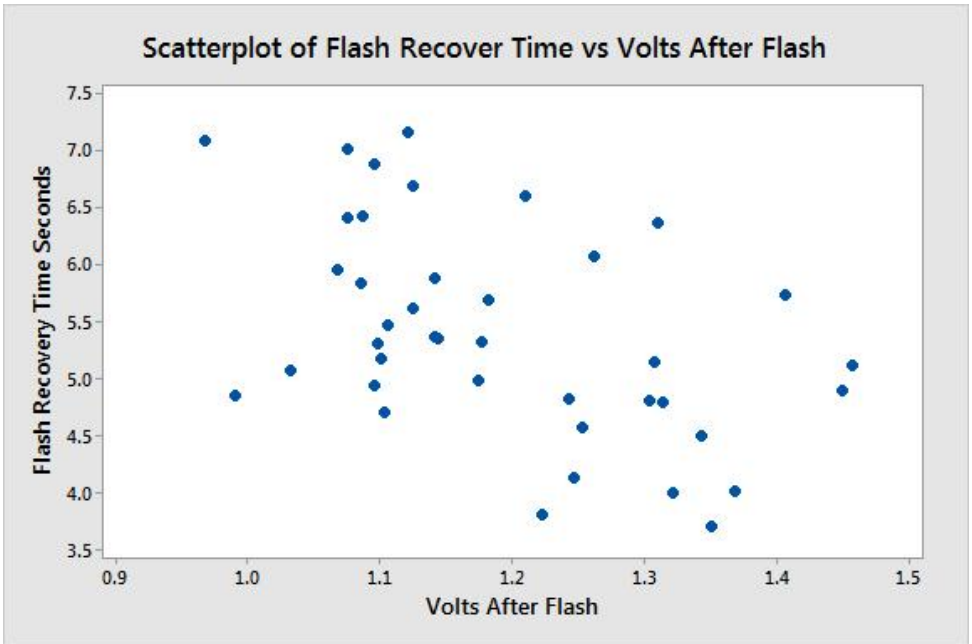
3. Histograms : Used for showing the distribution of a single variable. Good for understanding the frequency of values in intervals.



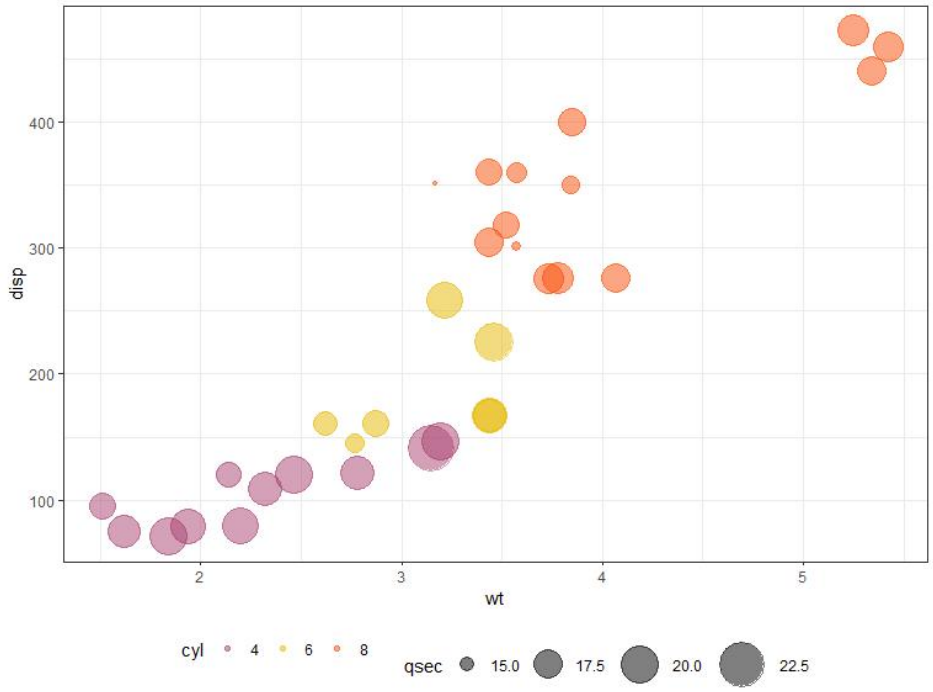
4. Box and whisker plots: This plot is great for showing the distribution, spread, and outliers of continuous data. As it includes, median, Q1, Q3, IQR, Outliers, The upper adjacent value, The lower adjacent value.



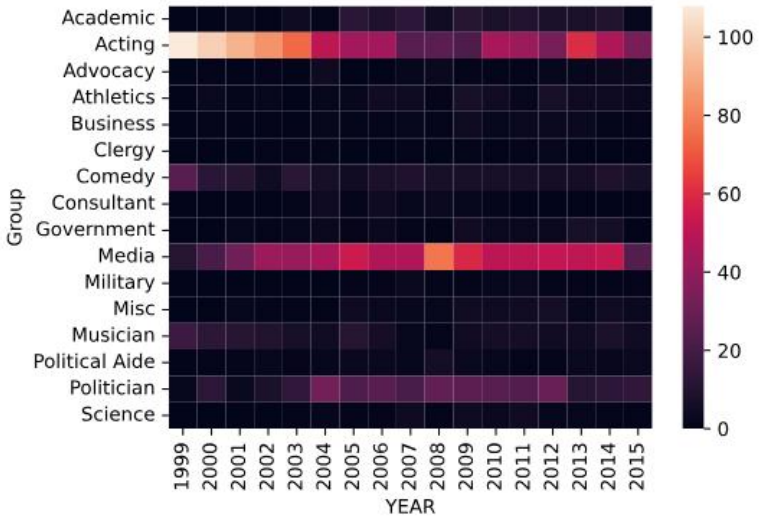
5. Scatter plots: Scatter plots are used to visualize the relationship between two continuous variables. Each point on the plot represents a single data point, and the position of the point on the x and y-axis represents the values of the two variables. It is often used in data exploration to understand the data and quickly surface potential correlations.



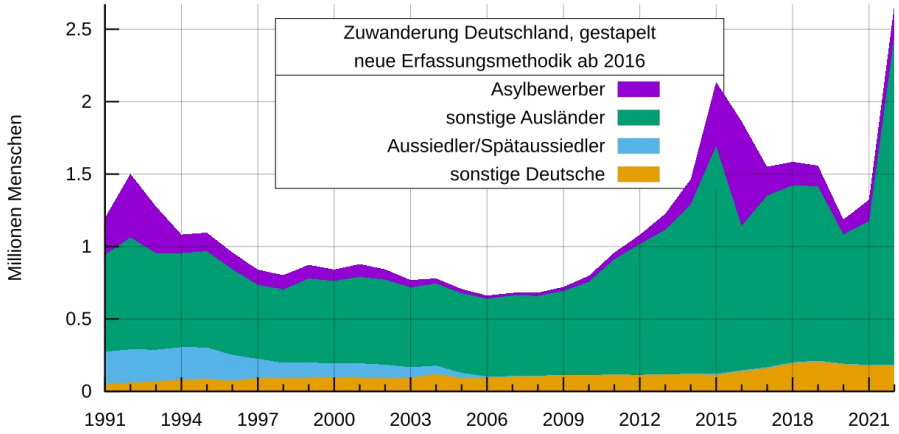
6. Bubble plot: Similar to a scatter plot, but adds a third dimension of information.



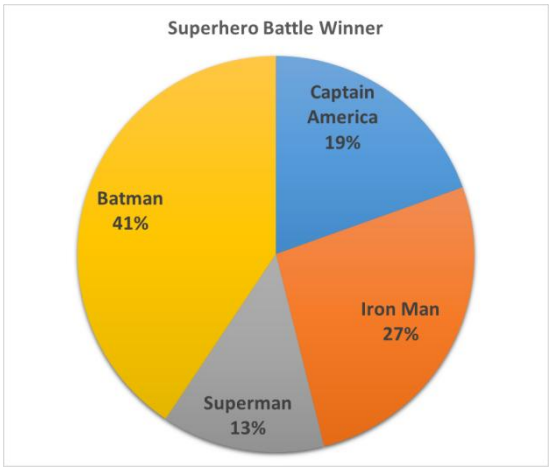
7. Heat maps: A heatmap is a common and beautiful matrix plot that can be used to graphically summarize the relationship between two variables. The degree of correlation between two variables is represented by a color code.



8. Area charts: Similar to a line chart but with shaded areas to emphasize the magnitude of values over time.



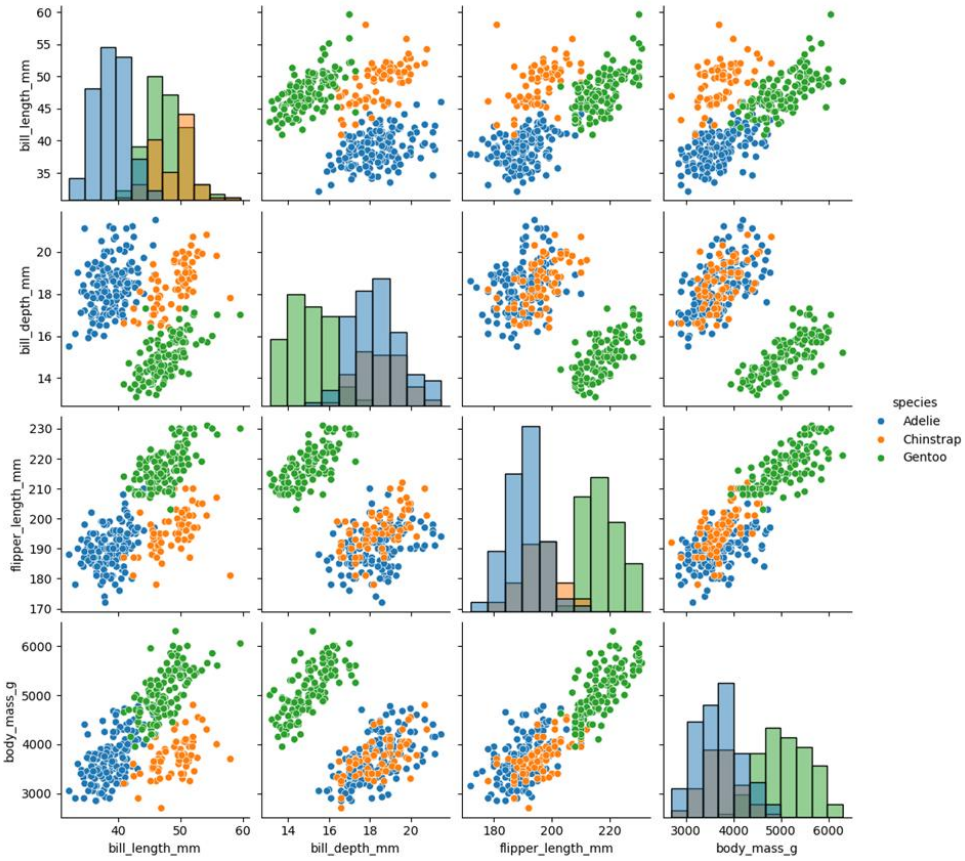
9. Pie Charts: Shows proportions of a whole, often used for categorical data with fewer categories.



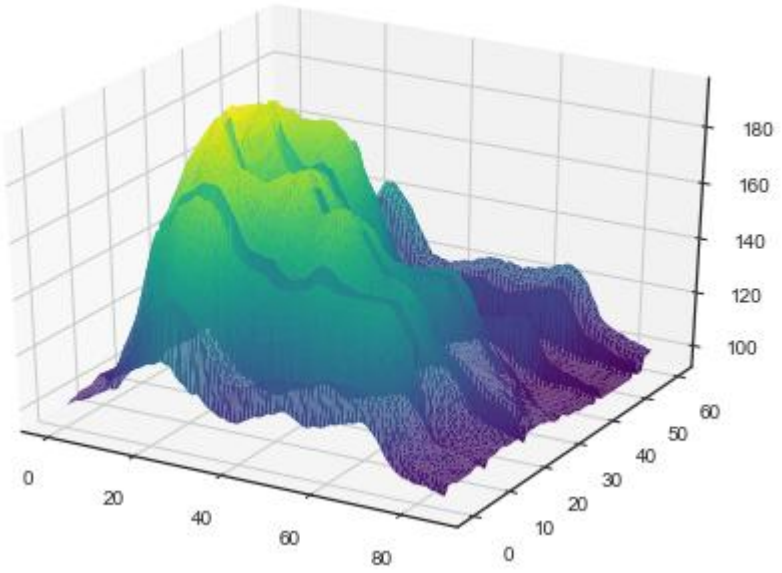
10. Time Series Plot: Time Series Plot: It display data points indexed by time. It uses X-axis for time, Y-axis for value, often with trend lines.

11. Multi-dimensional visualizations (e.g., pair plots, 3D plots):

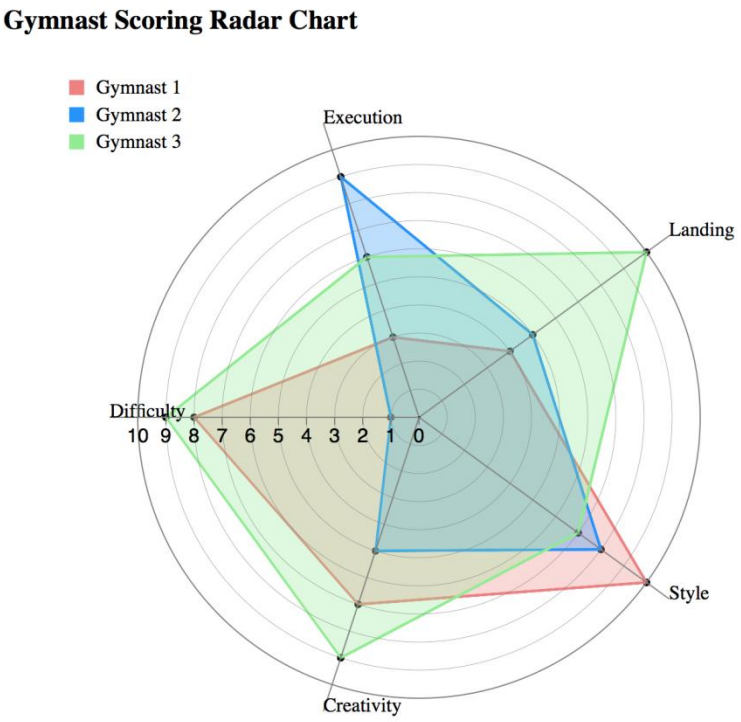
Pair Plots: A pair plot, also known as a scatterplot matrix, is a matrix of graphs that enables the visualization of the relationship between each pair of variables in a dataset. It combines both histogram and scatter plots, providing a unique overview of the dataset's distributions and correlations. The primary purpose of a pair plot is to simplify the initial stages of data analysis by offering a comprehensive snapshot of potential relationships within the data.



3D Plots: A 3D plotting is a way to represent three dimensional data in a graphical format. It allows you to visualize the information in three spatial dimensions, represented as X, Y, and Z coordinates. In 3D plots, data points are not only located on a flat plane but also have depth, creating a more detailed representation of the dataset.

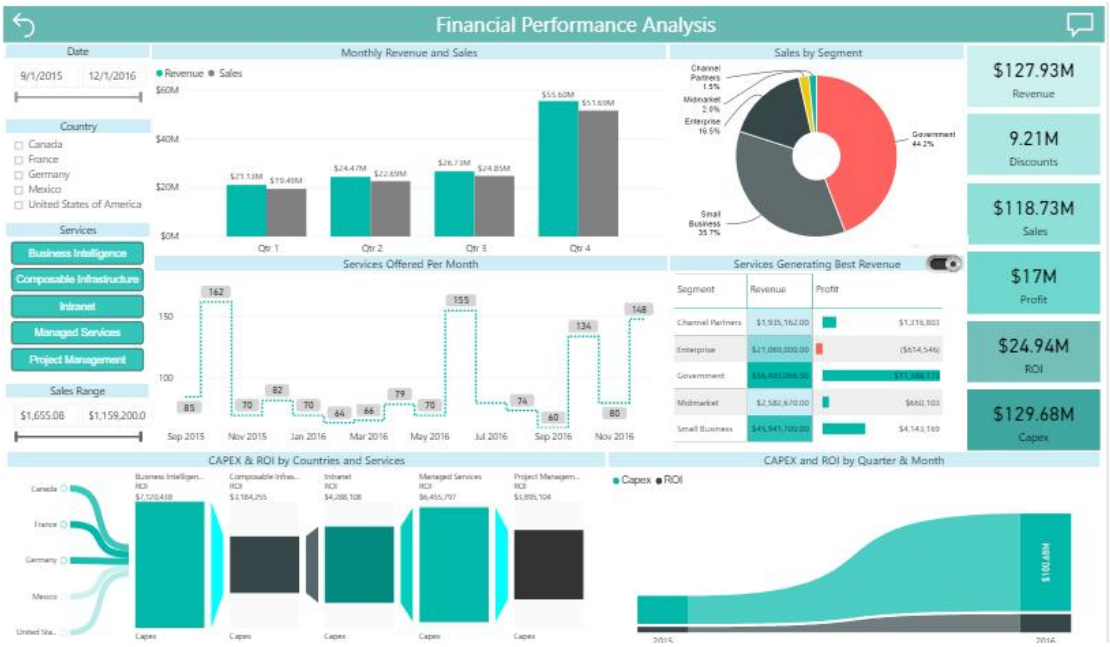


12. **Radar charts:** Used for comparing multiple variables, especially when they are part of a single category.



More Info on: https://careerfoundry.com/en/blog/data-analytics/data-visualization-types/

***DASHBOARD**



Common visualization tools (Matplotlib, Seaborn, Tableau, Power BI, etc.).

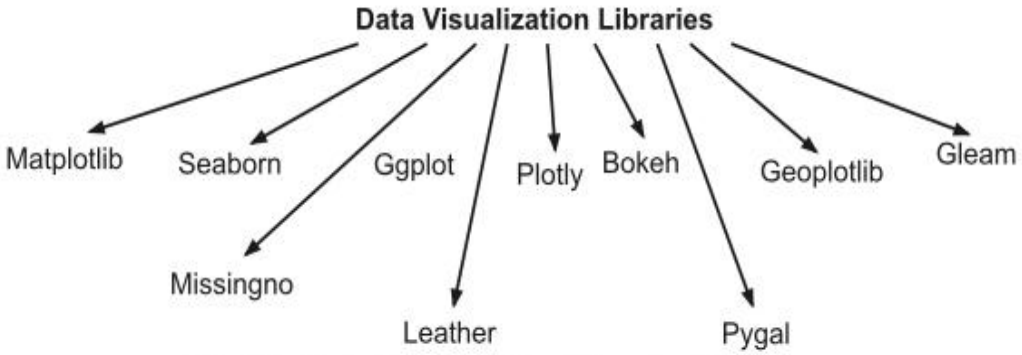


Fig. 4.2: Data Visualization Libraries in Python

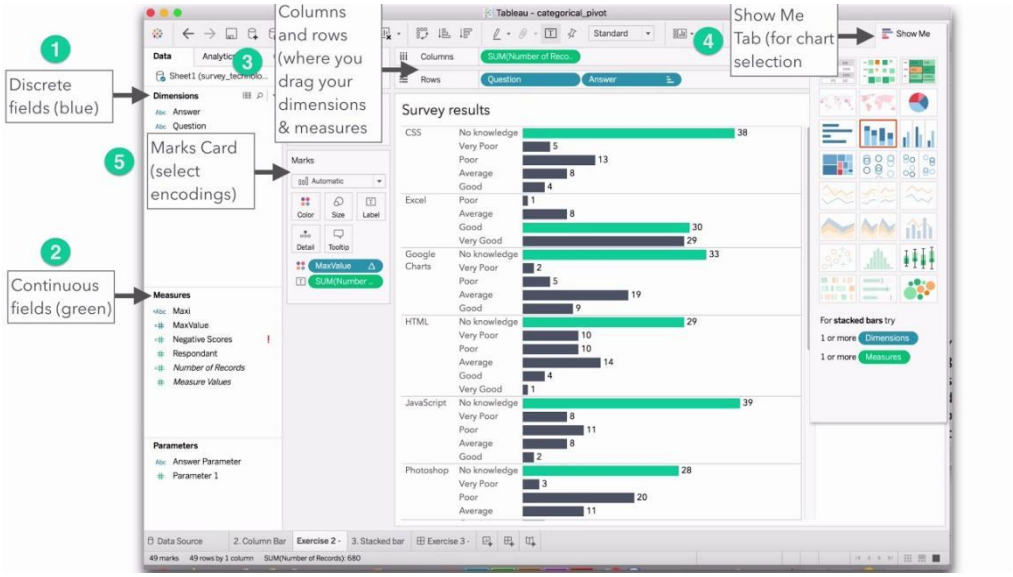
- ✓ **1. Matplotlib:**
  A widely used Python library for creating static, animated, and interactive visualizations.
- ✓ It provides flexibility to create plots like bar charts, line graphs, scatter plots, histograms, etc.
- ✓ It is ideal for creating custom plots and visualizations from scratch, especially when you need fine-grained control over your plots.

2. Seaborn:

- ✓ It is built on top of Matplotlib, Seaborn is a Python library focused on statistical data visualization.
- ✓ It provides attractive and informative plots such as heatmaps, pair plots, box plots, violin plots, and more.
- ✓ It is great for data exploration, statistical analysis, and creating visually appealing, informative plots easily.

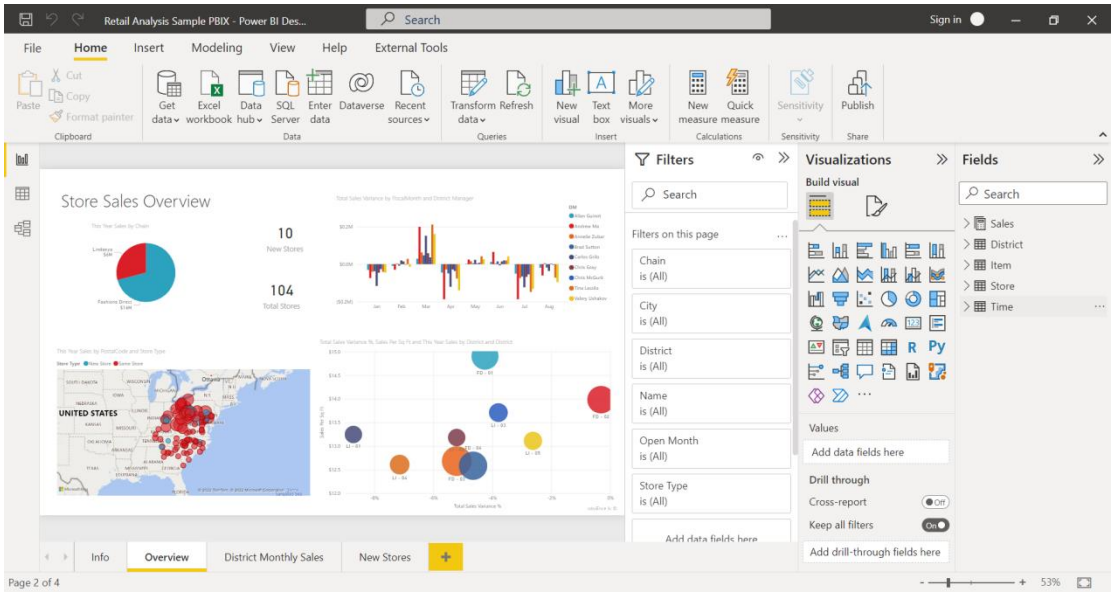
3. Tableau:

- ✓ A powerful, business-oriented data visualization tool known for creating interactive dashboards and reports.
- ✓ It has drag-and-drop interface, data blending from various sources, interactive dashboards, real-time data updates.
- ✓ It is used for business intelligence, large-scale data analysis, and creating interactive reports for sharing with stakeholders.



4. Power BI:

- ✓ A Microsoft tool designed for data analysis, visualization, and business intelligence.
- ✓ It provides interactive dashboards, data modeling, and the ability to connect to various data sources.
- ✓ It is deal for enterprises looking to create interactive, actionable reports and dashboards, especially in combination with other Microsoft tools like Excel and Azure.



5. D3.js (JavaScript):

- ✓ A powerful JavaScript library for producing dynamic and interactive data visualizations.

- ✓ It allows creating complex and interactive visualizations like network graphs, maps, and charts.
- ✓ It is often used for web-based applications, custom interactive visualizations, and large-scale data presentations.

6. Plotly:

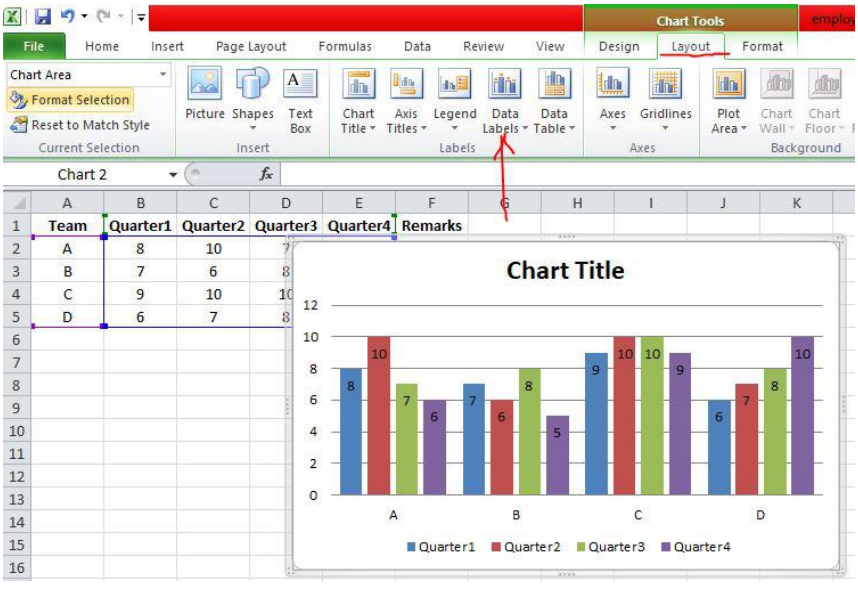
- ✓ A versatile Python library for creating interactive visualizations.
- ✓ It supports a wide range of plots (scatter, line, bar, etc.) with interactive features like zooming, panning, and hover functionality.
- ✓ It is great for creating dynamic, interactive dashboards, especially for web applications.

7. ggplot2:

- ✓ A widely used data visualization library for R, known for creating complex plots with ease.
- ✓ It provides powerful syntax for building visualizations, especially suited for advanced statistical plots.
- ✓ It is often used in R programming for detailed statistical analysis and creating aesthetically pleasing plots.

8. Excel:

- ✓ A widely accessible tool for data visualization and analysis.
- ✓ It offers various chart types like bar, pie, line graphs, and pivot tables(an interactive way to quickly summarize large amounts of data).
- ✓ It is deal for basic to intermediate-level data visualization, often used in business environments for reporting.



Assginment : Graph a box-and-whisker plot for the data values shown.
10, 10, 10, 15, 35, 75, 90, 95, 100, 175, 420, 490, 515, 515, 790
And Comment on result.

4.5 Principles of effective data visualization

Visualizations are means to communicate, and thus, we must ensure that the reader acknowledges the same information we intended to divulge.
The main goal of data visualization is to reduce complexity and provide clarity. Choosing the right data visualization technique is vital for success, but there are many other factors to consider. Here are some key principles to follow for effective data visualization:

1. Know Your Audience:

- Tailor your visualizations to the knowledge level and needs of the audience.
- Keep it simple for general audiences, while offering more details for technical audiences.

2. Clarity and Simplicity:

- Ensure that your visualization is easy to interpret.
- Avoid unnecessary complexity or decoration that can distract from the message.
- Ensure axis labels, legends, and titles are clear and self-explanatory to make the chart easily understandable

3. Data Integrity:

- Ensure that the data is represented truthfully. Avoid distorting the message through improper scaling or misleading visuals.
- Use consistent axes, scales, and units to avoid confusion.

4. Choosing the Right Chart Type:

Select a visualization type that matches the data and the message. For instance:

- ✓ Bar charts for comparing quantities.
- ✓ Line graphs for showing trends over time or continuous data.
- ✓ Scatter plots for understanding relationships.
- ✓ Pie charts for showing proportions (only when there are few categories).

5. Use of color:

- Use color to emphasize important data or patterns, but don't overuse it.
- Ensure color contrast is high enough for readability and accessibility. use color palettes that are distinguishable for everyone.

6. Maintain Consistency:

- Use consistent design elements such as colors, fonts, and chart styles across the visualization to avoid confusion.

- Keep the scale and units consistent throughout to maintain clarity.

7. Context:
- Provide context for the data being visualized. Include relevant labels, titles, and legends that explain what the viewer is seeing.
- Make sure the source of the data is clear and credible.

8. Interactivity(when applicable):
- Interactive visualizations can allow users to explore the data by filtering, zooming, or hovering for additional details, which can enhance understanding.

9. Hierarchical Structure:
- Organize information so that the most important insights are prominent and easy to access.
- Use hierarchy in design, with key metrics and patterns placed in the most visible areas of the visual.

10. Ethical Considerations:
- Visualizations can influence perceptions and decisions. Try to avoid misleading or biased representations. Ethical design ensures transparency.

Visual Encoding (Choosing the right chart for the data)

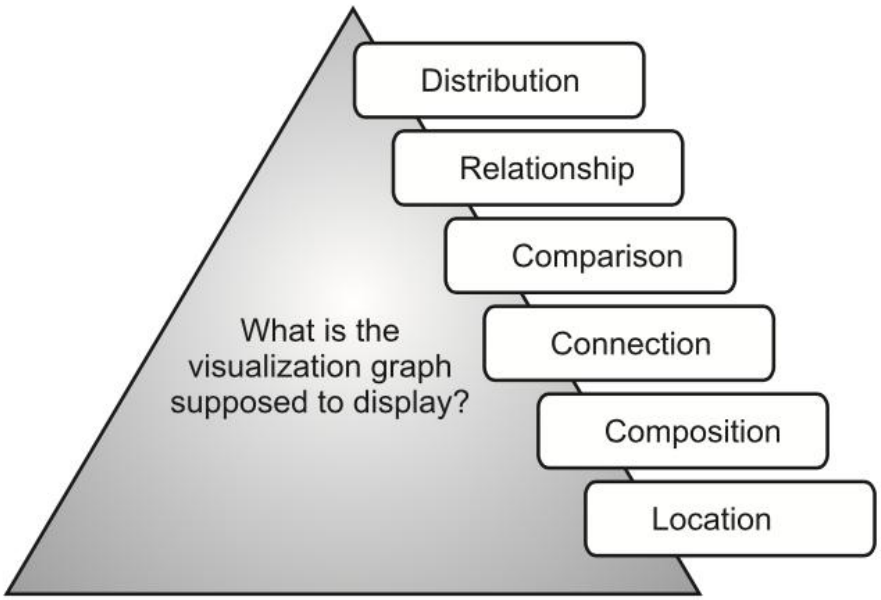
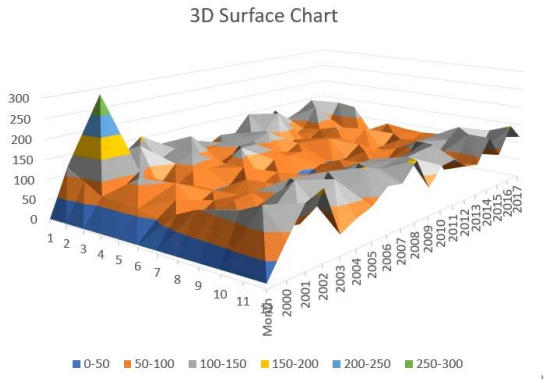


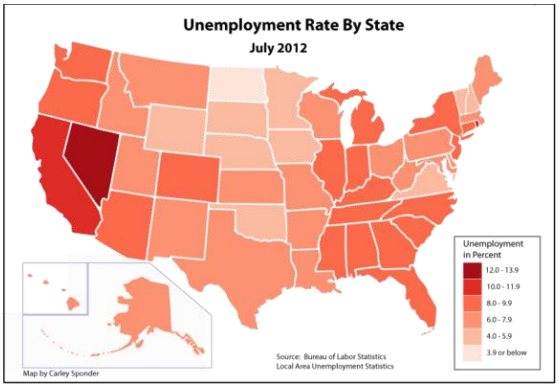
Fig 4.1: Concepts of a Visualization Graph

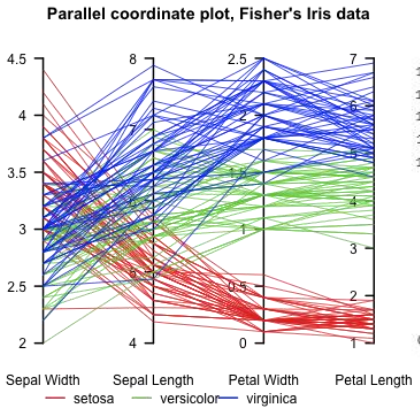
1. **Distribution:** If you want to show how values are distributed
2. **Relationship:** If you want to explore relationships between variables
3. **Comparison:** If you want to compare values across categories/time periods
4. **Composition:** If you want to understand how parts make up a whole
5. **Location:** If you want to visualize spatial data to identify geographical patterns.
6. **Connection:** If you want to show relationships, flows, or linkages between entities.

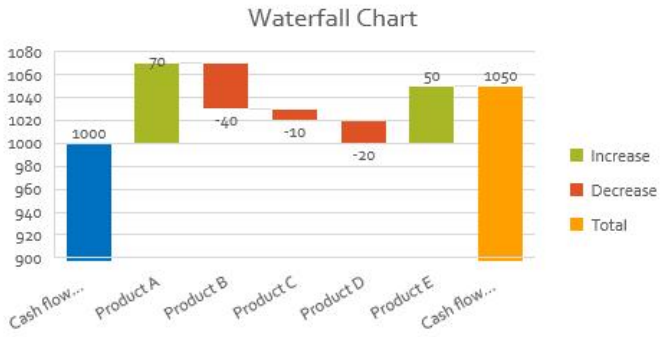
Role of Data Visualization Possible Illustrative Data	Visualization Graph
Distribution	• Scatter chart • 3D Area chart • Histogram
Relationship	• Bubble chart • Scatter chart
Comparison	• Bar chart • Line chart • Column chart • Area chart
Composition	• Pie chart • Waterfall chart • Stacked column chart • Stacked area chart
Location	• Bubble map • Choropleth map • Connection map
Connection	• Matrix chart • Node-link diagram • Word cloud • Alluvial diagram • Tube map

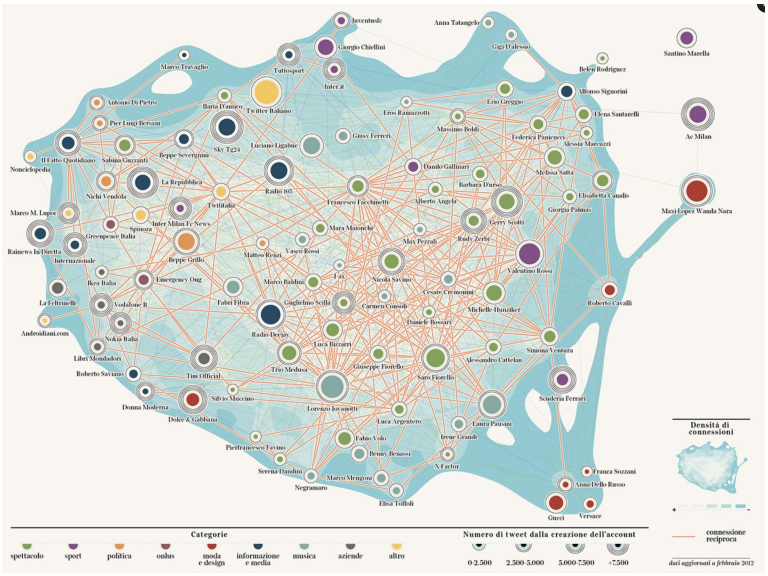
Mapping of the data is based on the visual cues (also called retinal variables) such as location, size, color value, color hue, color saturation, transparency, shape, structure, orientation, and so on.
Based on what type of data, the visualization tools should be effectively chosen to represent data in the visualization graph.











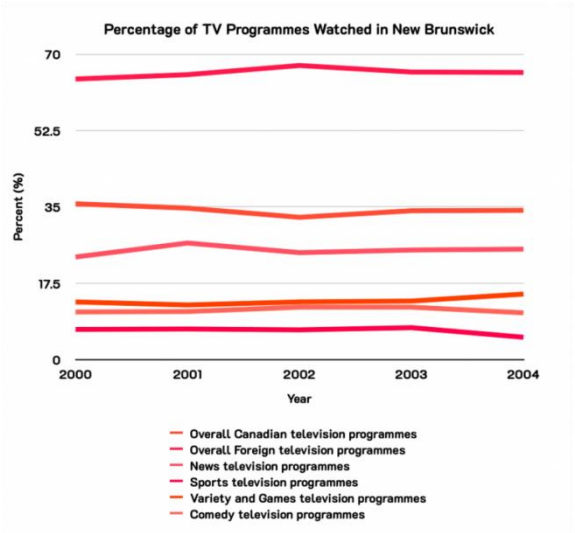
Avoiding misleading visualizations:

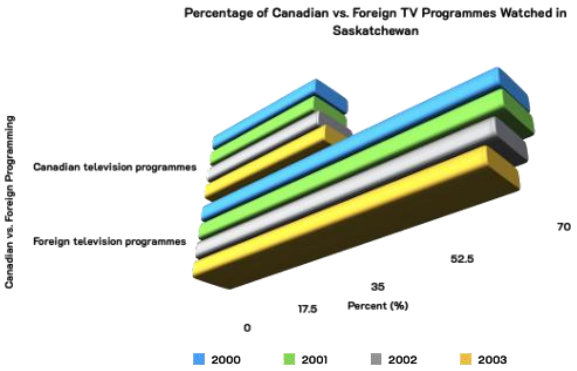
Avoiding misleading visualizations is essential to ensure that your audience interprets your data accurately and fairly. Here are best practices and tips for avoiding common pitfalls in data visualization:

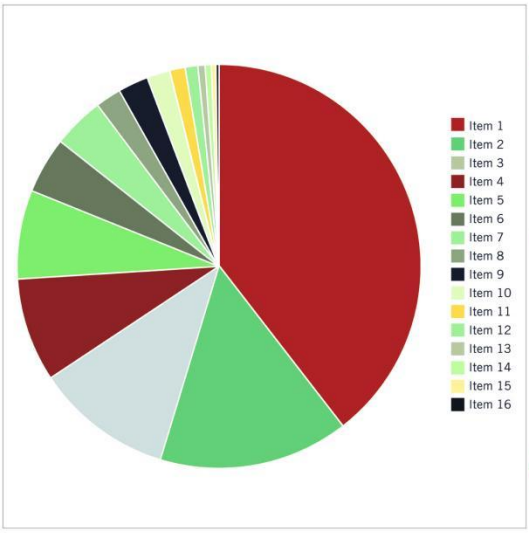
1. Choose the Right Chart Type
- ✓ Avoid Inappropriate Chart Types
- ✓ Consider Data Complexity

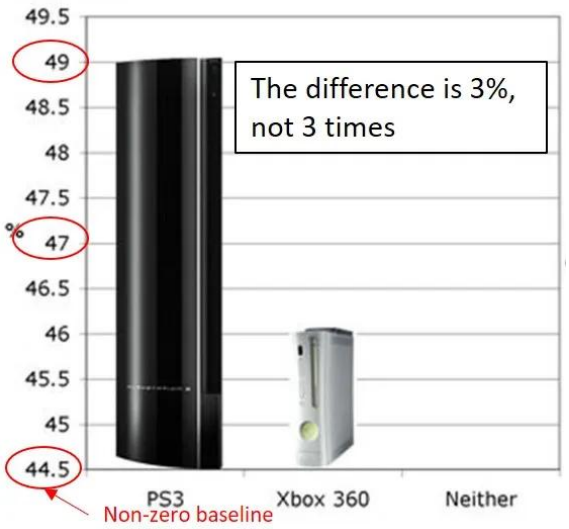
2. Keep Axes Honest(Don't Artificially shorten the Y-axis scale to make changes in the data appear more significant)
3. Avoid Distorting Proportions
4. Avoid Ambiguous Color Choices
5. Be Cautious with Data Aggregation
6. Avoid Cherry-Picking Data(Omitting sources or information to create a more predictable set of results)
7. Don't Misrepresent Trends(Over-smoothing or oversimplifying)i.e Presenting too much or too little information can be misleading
8. Label Clearly
9. Avoid 3D Effects but use wherever necessary
10. Simplify Without Losing Meaning
11. Highlight Key Takeaways

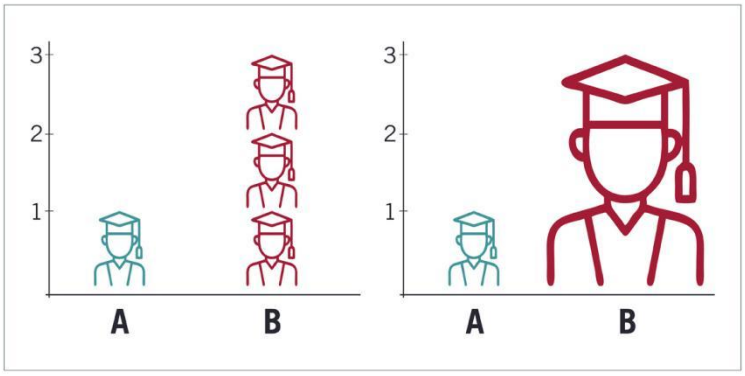
For Reference: https://clauswilke.com/dataviz/index.html






Item 1
Item 2
Item 3
Item 4
Item 5
Item 6
Item 7
Item 8
Item 9
Item 10
Item 11
Item 12
Item 13
Item 14
Item 15
Item 16


49.5
49
48.5
48
47.5
%
47
46.5
46
45.5
45
44.5
The difference is 3%, not 3 times
PS3
Xbox 360
Neither
Non-zero baseline


3
2
1
A
B
3
2
1
A
B

4.6 Feature engineering

Feature engineering refers to the process of transforming raw data into useful representations (features) either to boost model inference, reduce computational footprints, and improve interpretability.

Why is Feature Engineering Important?

✓ Improves Model Performance:

Quality features can significantly enhance the performance of your machine learning models. By creating new features or transforming existing ones, you can help your model learn patterns more effectively.

✓ Reduces Overfitting:

By selecting and engineering features that are most relevant to the task at hand, you can reduce the complexity of your model, thereby reducing the risk of overfitting.

✓ Enhances Interpretability:

Well-engineered features can make your models more interpretable, which is especially important in fields like healthcare and finance where understanding model decisions is crucial.

✓ Facilitates Better Understanding of Data:

The process of feature engineering forces you to dive deep into the data and understand the relationships between different variables, leading to better insights and data-driven decisions.

Feature engineering techniques:

Although there is no universally preferred feature engineering method or pipeline, there are a handful of common tasks used to create features from different data types for different models. Before implementing any of these techniques, however, one must remember to conduct a thorough data analysis to determine both the relevant features and appropriate number of features for addressing a given problem. Additionally, it is best to implement various data cleaning and preprocessing techniques, such as imputation for missing data or missing values, while also addressing outliers that can negatively impact model predictions.

1. Feature transformation: Feature transformation is the process of converting one feature type into another, more readable form for a particular model. This consists of transforming continuous into categorical data, or vice-versa.
   Techniques includes: Binning, One hot Encoding, Target Encoding etc.

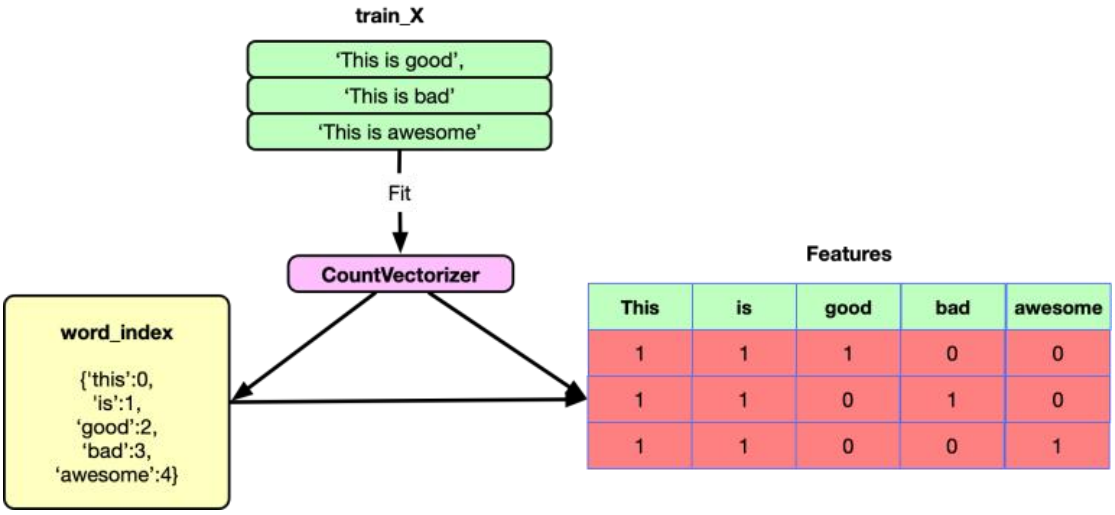
2. Feature extraction: Feature extraction involves transforming raw data into meaningful features that can be used as input to machine learning models. The goal is to derive new features that better represent the problem for the predictive models.

Feature Representation

Techniques for Feature Extraction for Text Data includes:

1. Bag of Words (BoW): Represents text by counting the frequency of each word in the corpus. This is a simple approach that treats text as a set of words without considering word order.
2. Term Frequency-Inverse Document Frequency (TF-IDF): A weighted version of BoW that accounts for how important a word is within a document relative to the entire corpus.
3. Word Embeddings (Word2Vec, GloVe, FastText): Represent words in continuous vector space based on their context. These vectors capture semantic meanings and relationships between words.
4. n-Grams: Extracts sequences of N words (bigrams, trigrams) to capture local dependencies between words.
5. Sentence Embeddings (BERT, GPT, etc.): Represent entire sentences or paragraphs as dense vectors that capture semantic meaning using pre-trained transformer models.

Bag of words:



TF-IDF

$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

TF-IDF

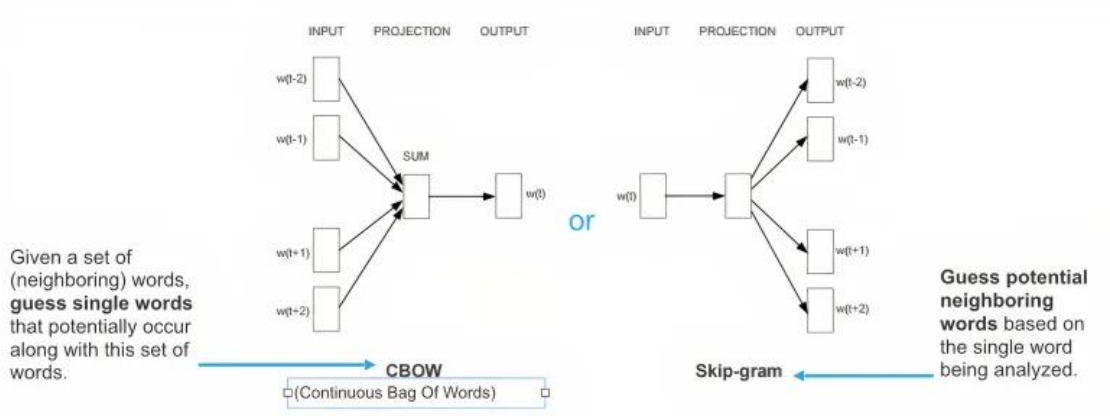
Term x within document y

$tf_{x,y}$ = frequency of x in y

df_x = number of documents containing x

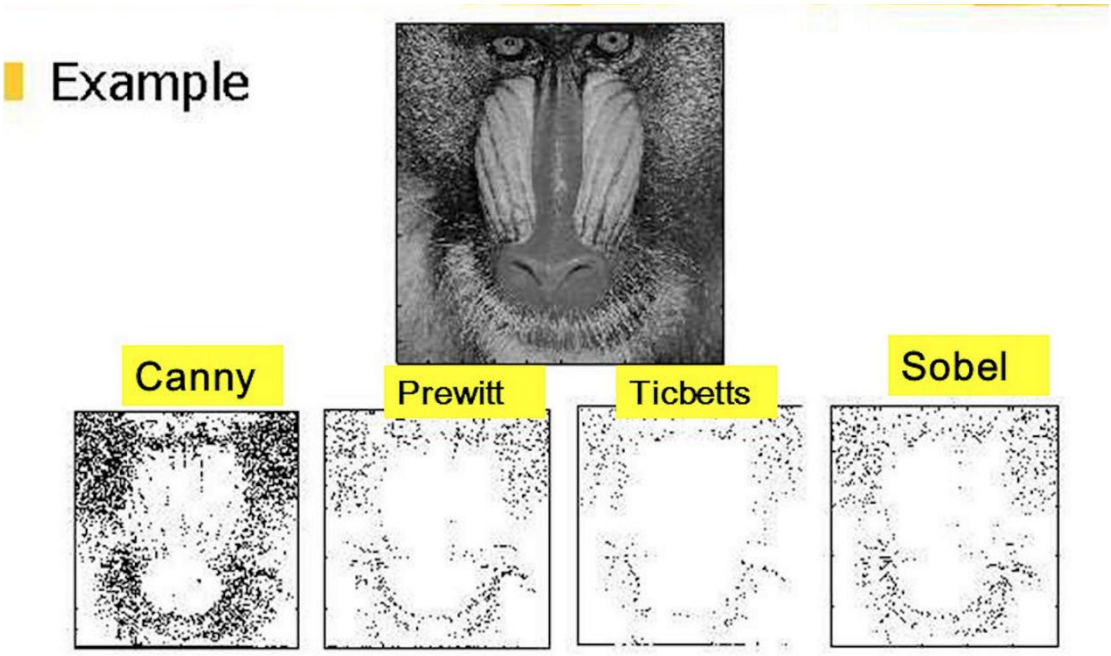
N = total number of documents

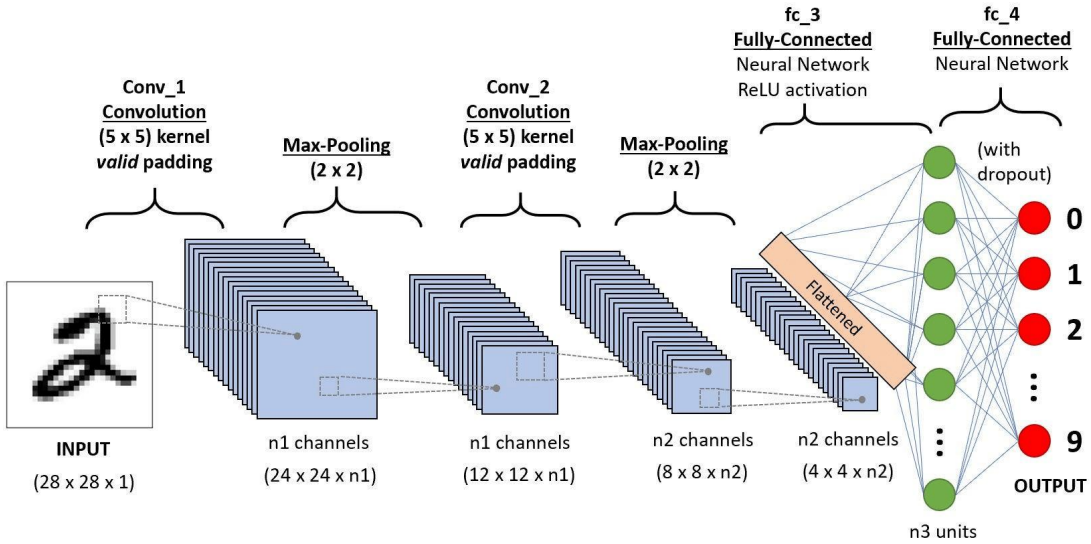
Word2Vec



Techniques for Feature Extraction for Image Data includes:

1. Edge Detection (Sobel, Canny, etc.): Identifies the boundaries of objects in an image by detecting rapid intensity changes.
2. Deep Convolutional Neural Networks (CNNs): Automatically extract hierarchical features (edges, textures, shapes, objects) by using deep networks of convolutions.





Techniques for Feature Extraction for Time-Series Data:

1. Statistical Features: Mean, variance, skewness, kurtosis, etc., computed over time windows.
2. Fourier Transform: Converts time-domain signals into frequency-domain representations, identifying periodic patterns in the data.

and so on.

3. Feature Selection: Feature selection denotes techniques for selecting a subset of the most relevant features to represent a model.

Feature selection techniques can be broadly classified into filter methods, wrapper methods, and embedded methods. These methods differ in how they evaluate the importance of features and the types of models they use.

1. Filter Methods: Filter methods assess the relevance of features based on statistical measures, without using a machine learning model. These methods are generally computationally efficient and easy to implement. Techniques includes: Correlation Coefficient, Chi-Square Test, ANOVA etc.

2. Wrapper Methods: Wrapper methods use a machine learning algorithm to evaluate subsets of features and determine the best feature set by directly assessing model performance. These methods can be computationally expensive as they require training multiple models. Techniques includes:
- Recursive Feature Elimination (RFE) i.e recursively removes the least important features based on the performance of a chosen model.
- Forward Selection: Starts with an empty feature set and adds features one by one. At each step, the feature that improves the model the most is added.
- Backward Elimination: Starts with all features and removes the least important one at each iteration. It continues until the model performance stops improving.

3. Embedded Methods: Embedded methods perform feature selection during the model training process. These methods are computationally efficient since they incorporate feature selection directly into the learning process.
Techniques includes:
- L1 Regularization (Lasso Regression): Lasso (Least Absolute Shrinkage

and Selection Operator) applies L1 regularization to the model, shrinking the coefficients of less important features to zero. This effectively eliminates those features from the model.
- Decision Trees and Random Forests: Tree-based models like Decision Trees, Random Forests, and Gradient Boosting can be used to assess feature importance based on how well they split the data. Features that contribute more to reducing the impurity (e.g., Gini impurity or entropy) are deemed more important.
- Gradient Boosting Machines (GBM): Models like XGBoost and LightGBM also have built-in feature importance based on gradient boosting algorithms. They rank features by their ability to reduce the loss function.

4. Feature Reduction: It is a technique for reducing the dimensionality of the dataset while retaining the most significant information.
   Techniques includes: Principal Component Analysis(PCA), Linear discriminant analysis(LDA) , t-SNE (t-distributed Stochastic Neighbor Embedding) etc.

5. Feature scaling: Feature scaling (sometimes called feature normalization) is a standardization technique to rescale features and limit the impact of large scales on models.
   Technique includes: Min_Max Scaling, Z-score Scaling etc.