

UNIT-VI

Data Visualization

Subject-Data Science



BRACT'S, Vishwakarma Institute of Information Technology, Pune-48

(An Autonomous Institute affiliated to Savitribai Phule Pune University)
(NBA and NAAC accredited, ISO 9001:2015 certified)

- Introduction to Data visualization, Challenges to Big data visualization, Conventional data visualization tools, Techniques for visual data representations.
- Types of data visualization, Visualizing Big Data, Tools used in data visualization.
- Analytical techniques used in Big data visualization

What is Data Visualization

- Data visualization is the process of converting raw data into easily understood pictures of information that enable fast and effective decisions.
- Data visualization is used in software applications to provide an in-built graphical interface.
- It is applied to many areas to enable users to collect useful information from their data for faster, more informed decision making.
- These areas include: Military, private business sectors and scientific research.

Visual Representation of Data

- For exploration, discovery, insight, ..
- Interactive component provides more insight as compared to a static image.
- To communicate information clearly and efficiently, data visualization uses [statistical graphics](#), [plots](#), [information graphics](#) and other tools.
- Numerical data may be encoded using dots, lines, or bars, to visually communicate a quantitative message.

Effective visualization helps users analyze and reason about data and evidence.

- It makes complex data more accessible, understandable and usable. Users may have particular analytical tasks, such as making comparisons or understanding causality, and the design principle of the graphic (i.e., showing comparisons or showing causality) follows the task.
- Tables are generally used where users will look up a specific measurement, while charts of various types are used to show patterns or relationships in the data for one or more variables.

- Data visualization is both an art and a science. It is viewed as a branch of descriptive statistics by some, but also as a grounded theory development tool by others.
- Increased amounts of data created by Internet activity and an expanding number of sensors in the environment are referred to as "big data" or Internet of things.
- Processing, analyzing and communicating this data present ethical and analytical challenges for data visualization.
- The field of data science and practitioners called data scientists help address this challenge.

Overview

- Data visualization refers to the techniques used to communicate data or information by encoding it as visual objects (e.g., points, lines or bars) contained in graphics. The goal is to communicate information clearly and efficiently to users. It is one of the steps in [data analysis](#) or [data science](#).
- According to Friedman (2008) the “main goal of data visualization is to communicate information clearly and effectively through graphical means.” It doesn't mean that data visualization needs to look boring to be functional or extremely sophisticated to look beautiful.

To convey ideas effectively, both visual form and functionality need to go hand in hand, providing insights into a rather sparse and complex data set by communicating its key-aspects in a more natural way.

- Yet designers often fail to achieve a balance between form and function, creating gorgeous data visualizations which fail to serve their main purpose — to communicate information".
- Data visualization is closely related to information graphics, information visualization, scientific visualization, exploratory data analysis and statistical graphics.

What are the benefits of Data Visualization?

- Data visualization allows users to see several different perspectives of the data.
- Data visualization makes it possible to interpret vast amounts of data .
- Data visualization offers the ability to note exceptions in the data.
- Data visualization allows the user to analyze visual patterns in the data.
- Exploring trends within a database through visualization by letting analysts navigate through data and visually orient themselves to the patterns in the data.

Data visualization can help translate data patterns into insights, making it a highly effective decision-making tool.

- Data visualization trains users with the ability to see influences that would otherwise be difficult to find.
- With all the data available, it is difficult to find the shades that can make a difference.
- By simplifying the presentation, Data Visualization can reduce the time and difficulty it takes to move from data to decision making.

Challenges to Big data visualization

- Scalability and dynamics are two major challenges in visual analytics.
 - The visualization-based methods take the challenges presented by the “four Vs” of big data and turn them into following opportunities.
- ✓ *Volume*: The methods are developed to work with an immense number of datasets and enable to derive meaning from large volumes of data.
 - ✓ *Variety*: The methods are developed to combine as many data sources as needed.
 - ✓ *Velocity*: With the methods, businesses can replace batch processing with real-time stream processing.

Value: The methods not only enable users to create attractive infographics and heatmaps, but also create business value by gaining insights from big data.

- Visualization of big data with diversity and heterogeneity (structured, semi-structured, and unstructured) is a big problem.
- **Speed** is the desired factor for the big data analysis.
- Designing a new visualization tool with efficient indexing is not easy in big data.
- Cloud computing and advanced graphical user interface can be merged with the big data for the better management of big data scalability.

Visualization systems must contend with unstructured data forms such as graphs, tables, text, trees, and other metadata. Big data often has unstructured formats.

- Due to bandwidth limitations and power requirements, visualization should move closer to the data to extract meaningful information efficiently.
- Because of the big data size, the need for massive parallelization is a challenge in visualization.
- The challenge in parallel visualization algorithms is decomposing a problem into independent tasks that can be run concurrently.

Effective data visualization is a key part of the discovery process in the era of big data.

- For the challenges of high complexity and high dimensionality in big data, there are different dimensionality reduction methods.
- However, they may not always be applicable.
- The more dimensions are visualized effectively, the higher are the chances of recognizing potentially interesting patterns, correlations, or outliers .



There are also following problems for big data visualization :

- ✓ ***Visual noise:*** Most of the objects in dataset are too relative to each other. Users cannot divide them as separate objects on the screen.
- ✓ ***Information loss:*** Reduction of visible data sets can be used, but leads to information loss.
- ✓ ***Large image perception:*** Data visualization methods are not only limited by aspect ratio and resolution of device, but also by physical perception limits.
- ✓ ***High rate of image change:*** Users observe data and cannot react to the number of data change or its intensity on display.
- ✓ ***High performance requirements:*** It can be hardly noticed in static visualization because of lower visualization speed requirements--high performance requirement.

• Perceptual and interactive scalability are also challenges of big data visualization.

- Querying large data stores can result in high latency, disrupting fluent interaction .
- In Big Data applications, it is difficult to conduct data visualization because of the large size and high dimension of big data.
- Most of current Big Data visualization tools have poor performances in scalability, functionalities, and response time.
- Uncertainty can result in a great challenge to effective uncertainty-aware visualization and arise during a visual analytics process.

- Potential solutions to some challenges or problems about visualization and big data were presented:
- 1. **Meeting the need for speed:** One possible solution is hardware. Increased memory and powerful parallel processing can be used. Another method is putting data in-memory but using a grid computing approach, where many machines are used.
- 2. **Understanding the data:** One solution is to have the proper domain expertise in place.

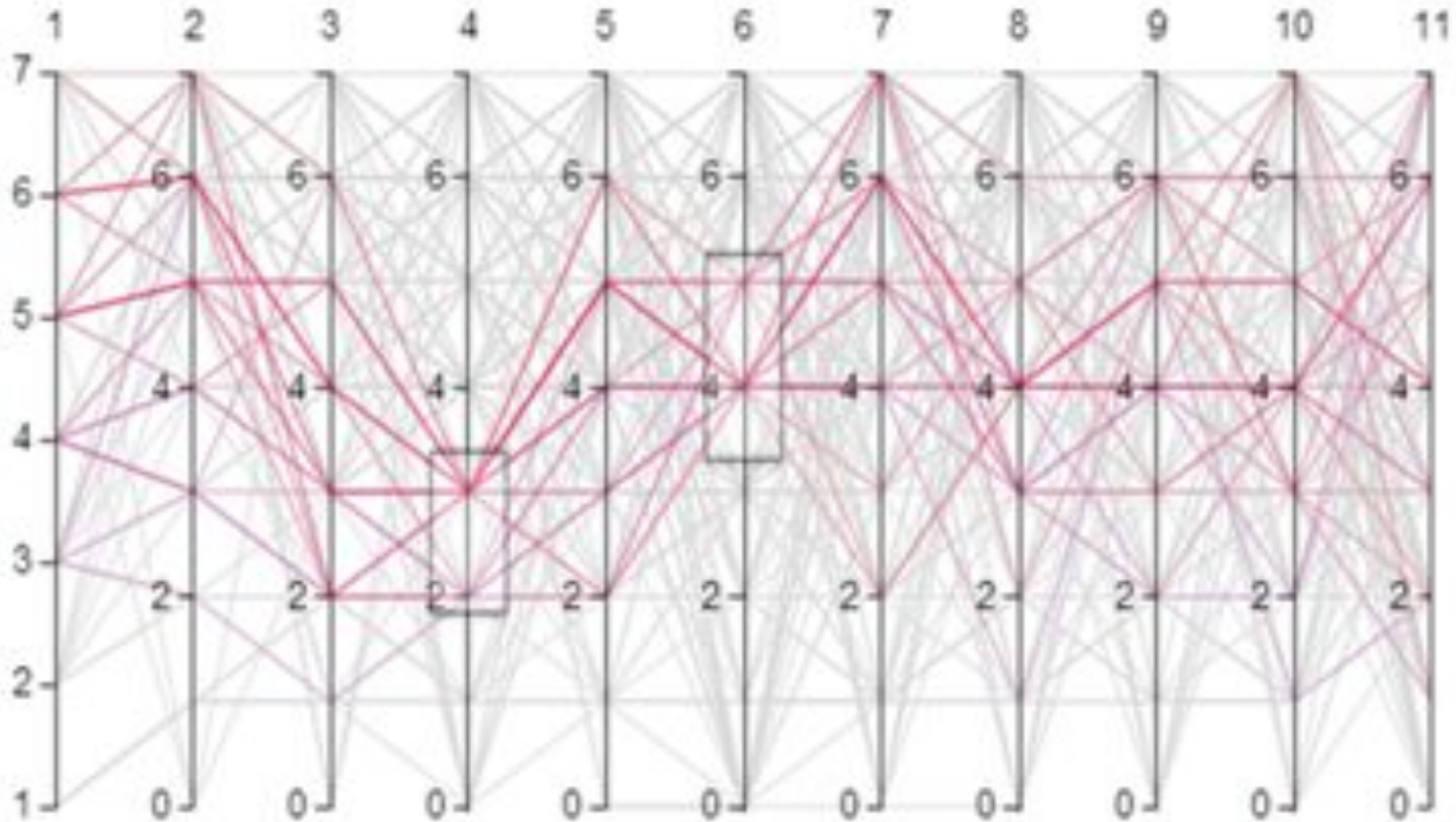
- 3. **Addressing data quality:** It is necessary to ensure the data is clean through the process of data control or information management.
- 4. **Displaying meaningful results:** One way is to cluster data into a higher-level view where smaller groups of data are visible and the data can be effectively visualized.
- 5. **Dealing with outliers:** Possible solutions are to remove the outliers from the data or create a separate chart for the outliers.

Conventional data visualization tools

- Many conventional data visualization methods are often used.
- They are: table, histogram, scatter plot, line chart, bar chart, pie chart, area chart, flow chart, bubble chart, multiple data series or combination of charts, time line, Venn diagram, data flow diagram, and entity relationship diagram, etc.
- In addition, some data visualization methods have been used although they are less known compared the above methods.
- The additional methods are: parallel coordinates, treemap, cone tree, and semantic network, etc.

- Parallel coordinates is used to plot individual data elements across many dimensions. Parallel coordinate is very useful when to display multidimensional data.
- [Figure 1](#) shows parallel coordinates.

Figure 1. Parallel coordinates



- Treemap is an effective method for visualizing hierarchies. The size of each sub-rectangle represents one measure, while color is often used to represent another measure of data.

- [Figure 2](#) shows a treemap of a collection of choices for streaming music and video tracks in a social network community.
- **Cone tree** is another method displaying hierarchical data such as organizational body in three dimensions.
- The branches grow in the form of cone.
- **A semantic network** is a graphical representation of logical relationship between different concepts. It generates directed graph, the combination of nodes or vertices, edges or arcs, and label over each edge.

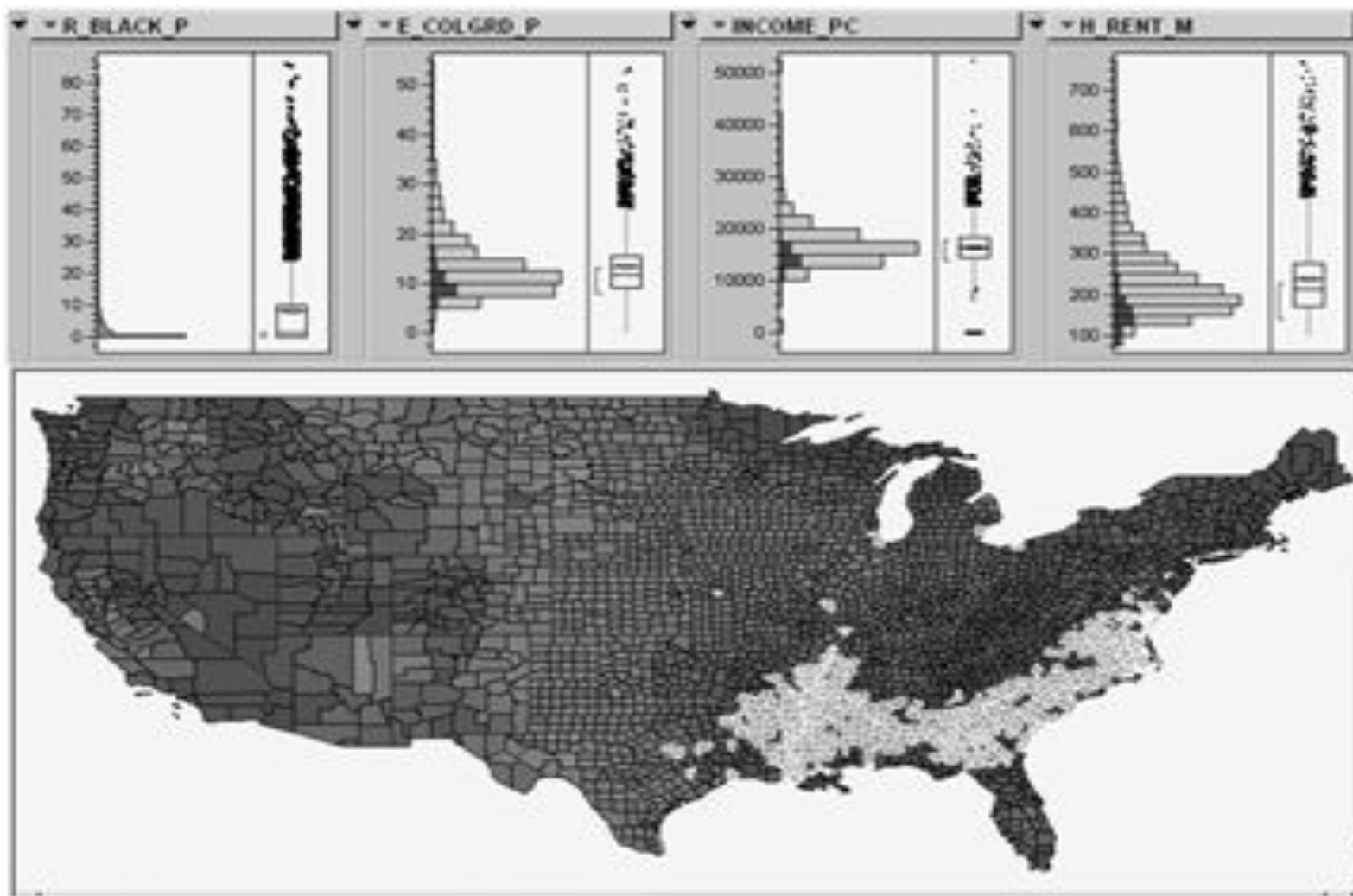
Figure 2. Treemap view of a social network's track selections from a streaming media service



Visualizations are not only static; they can be interactive. Interactive visualization can be performed through approaches such as zooming (zoom in and zoom out), overview and detail, zoom and pan, and focus and context or fish eye.

- **The steps for interactive visualization are as follows :**
- 1. **Selecting:** Interactive selection of data entities or subset or part of whole data or whole data set according to the user interest.
- 2. **Linking:** It is useful for relating information among multiple views. An example is shown in [Figure 3](#).

Figure 3. Interactive brushing and linking between histogram plots (top) and geographic map (bottom) of datasets



- **3. *Filtering*:** It helps users adjust the amount of information for display. It decreases information quantity and focuses on information of interest.
- **4. *Rearranging or Remapping*:** Because the spatial layout is the most important visual mapping, rearranging the spatial layout of the information is very effective in producing different insights.

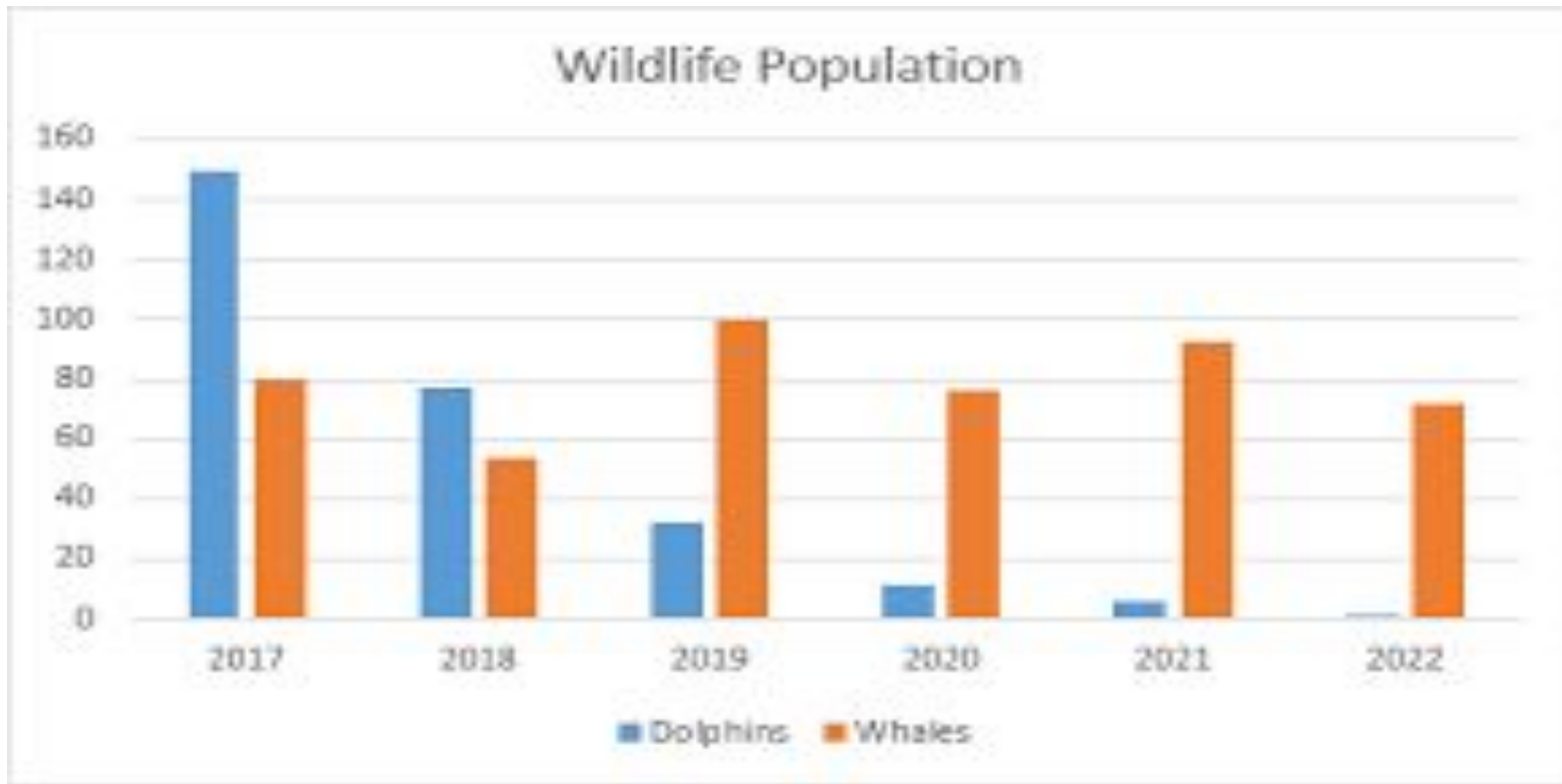
New database technologies and promising Web-based visualization approaches may be vital for reducing the cost of visualization generation and allowing it to help improve the scientific process.

- Because of Web-based linking technologies, visualizations change as data change, which greatly reduces the effort to keep the visualizations timely and up to date.
- These “low-end” visualizations have been often used in business analytics and open government data systems, but they have generally not been used in the scientific process.

Techniques for visual data representations.

(1) Bar Charts

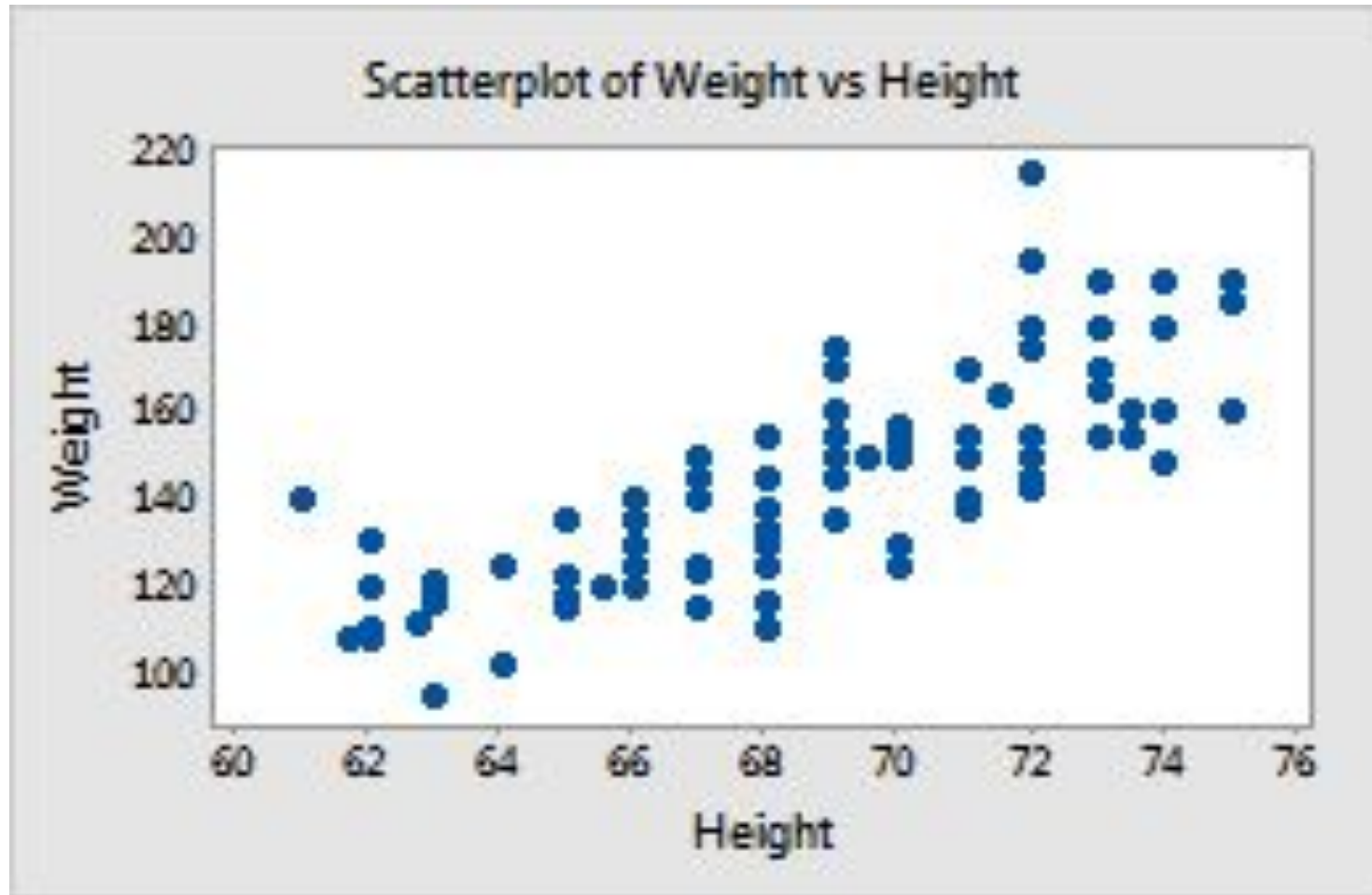
- Bar charts are most commonly used for comparing the quantities of different categories or groups.
- Values of a category are represented using the bars, and they can be configured with either vertical or horizontal bars, with the length or height of each bar representing the value.
- Bar charts can be configured with either vertical or horizontal bars, with the length or height of each bar representing the value.



(2) Scatter Plots

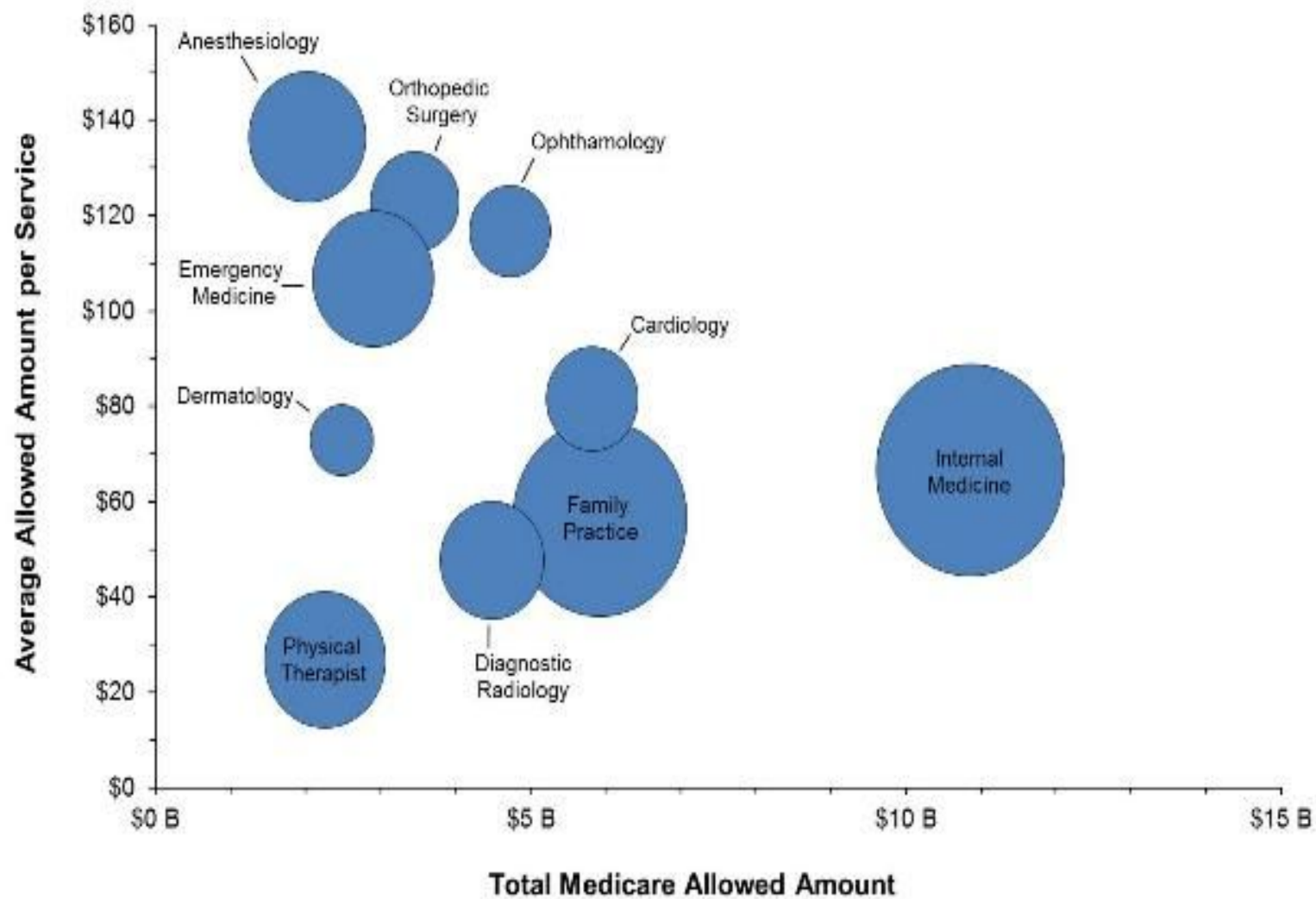
- A scatter plot (or X-Y plot) is a two-dimensional plot that shows the joint variation of two data items.
- In a scatter plot, each marker (symbols such as dots, squares and plus signs) represents an observation.
- The marker position indicates the value for each observation.
- Scatter plots also support grouping. When you assign more than two measures, a scatter plot matrix is produced.
- A scatter plot matrix is a series of scatter plots that displays every possible pairing of the measures that are assigned to the visualization.

- Scatter plots can help you gain a sense of how spread out the data might be or how closely related the data points are.
- They can also quickly identify patterns present in the distribution of the data.



(3)Bubble Plots:

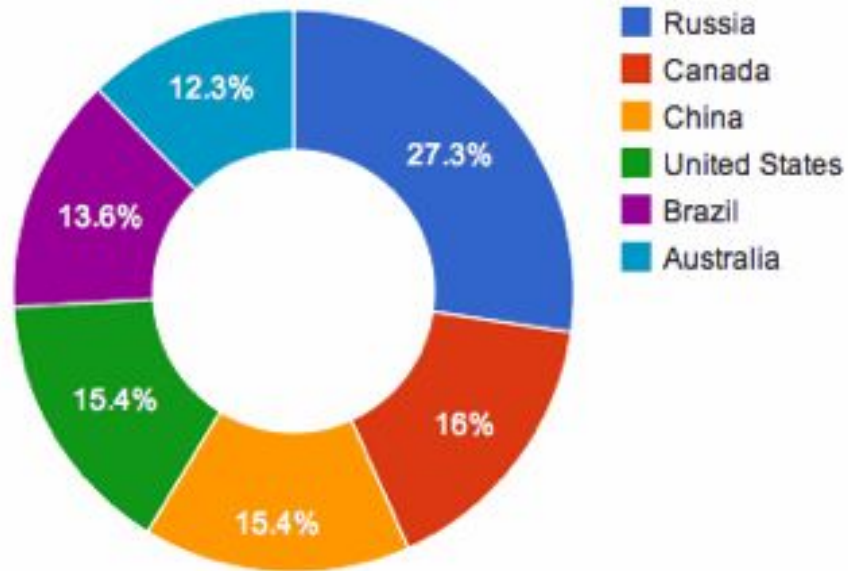
- A Scatter Plot Variation .
- A bubble plot is a variation of a scatter plot in which the markers are replaced with bubbles.
- A bubble plot displays the relationships among at least three measures.
- Two measures are represented by the plot axes. The third measure is represented by the size of the bubbles.
- Each bubble represents an observation.
- Bubble plots are a variation of scatter plots.
- They're especially useful for data sets with dozens to hundreds of values or when the values differ by several orders of magnitude.



(4) Pie and Donut Charts:

- When designing reports or dashboards, another consideration for the effectiveness of a pie or donut chart is the amount of space the chart requires in the sizing of the report.
- Because of their round shape, these charts require extra real estate, so they may be less than ideal when developing dashboards for small screens or mobile devices.
- Other charts (like a bar chart or line chart) may provide a better way to represent the same information in less space.
- Pie and donut charts are most effective when there are limited components and when text and percentages are included to describe the content.

Countries by Area



Types of data visualization

- In general, there are two basic types of data visualization: **exploration**, which helps find a story the data is telling you, and **explanation**, which tells a story to an audience.
- Both types of data visualization must take into account the audience's expectations.
- Within these two basic categories, there are many different ways data can be made visual.

(A)2D Area

- 2D area types of data visualization are usually geospatial, meaning that they relate to the relative position of things on the earth's surface.
- **Cartogram:** A cartogram distorts the geometry or space of a map to convey the information of an alternative variable, such as population or travel time. The two main types are area and distance cartograms.

- **Choropleth:** A choropleth is a map with areas patterned or shaded to represent the measurement of a statistical variable, such as most visited website per country or population density by state.
- **Dot Distribution Map:** A dot distribution or dot density map uses a dot symbol to show the presence of a feature on a map, relying on visual scatter to show spatial pattern.

(B)Temporal

- Temporal visualizations are similar to one-dimensional linear visualizations, but differ because they have a start and finish time and items that may overlap each other.
- **Connected Scatter Plot:** A connected scatter plot is a scatter plot, that displays values of two variables for a set of data, with an added line that connects the data series.

- **Polar Area Diagram:** A polar area diagram is similar to a traditional pie chart, but sectors differ in how far they extend from the center of the circle rather than by the size of their angles.
- **Time Series:** A time series is a sequence of data points typically consisting of successive measurements made over a time interval, such as the number of website visits over a period of several months.

(c) Multidimensional

- Multidimensional data elements are those with two or more dimensions. This category is home to many of the most common types of data visualization.
- **Pie Chart:** A pie or circle chart is divided into sectors to illustrate numerical proportion; the arc length and angle of each sector is proportional to the quantity it represents.

Histogram: A histogram is a data visualization that uses rectangles with heights proportional to the count and widths equal to the “bin size” or range of small intervals.

- **Scatter Plot:** A scatter plot displays values for two variables for a set of data as a collection of points.

(D) Hierarchical

- Hierarchical data sets are orderings of groups in which larger groups encompass sets of smaller groups.
- **Dendrogram:** A dendrogram is a tree diagram used to illustrate an arrangement of clusters produced by hierarchical clustering.

- **Ring Chart:** A ring or sunburst chart is a multilevel pie chart that visualizes hierarchical data with concentric circles.
- **Tree Diagram:** A tree diagram or tree structure represents the hierarchical nature of a structure in graph form. It can be visually represented from top to bottom or left to right.

(E) Network

- Network data visualizations show how data sets are related to one another within a network.
- **Alluvial Diagram:** An alluvial diagram is a type of flow diagram that represents changes in network structure over time.
- **Node-Link Diagram:** A node-link diagram represents nodes as dots and links as line segments to show how a data set is connected.
- **Matrix:** A matrix chart or diagram shows the relationship between two, three, or four groups of information and gives information about said relationship.

Software with the functions of visualization

A lot of big data visualization tools run on the Hadoop platform.

- The common modules in Hadoop are: Hadoop Common, Hadoop Distributed File System (HDFS), Hadoop YARN, and Hadoop MapReduce.
- They analyze big data efficiently, but lack adequate visualization.
- Some software with the functions of visualization and interaction for visualizing data has been developed :

- ***Pentaho:*** It supports the spectrum of BI functions such as analysis, dashboard, enterprise-class reporting, and data mining.
- ***Flare:*** An ActionScript library for creating data visualization that runs in Adobe Flash Player.
- ***JasperReports:*** It has a novel software layer for generating reports from the big data storages.
- ***Dygraphs:*** It is quick and elastic open source JavaScript charting collection that helps discover and understand opaque data sets.
- ***Datameer Analytics Solution and Cloudera:*** Datameer and Cloudera have partnered to make it easier and faster to put Hadoop into production and help users to leverage the power of Hadoop.
- ***Platfora:*** Platfora converts raw big data in Hadoop into interactive data processing engine. It has modular functionality of in-memory data engine.

- **ManyEyes:** It is a visualization tool launched by IBM. Many Eyes is a public website where users can upload data and create interactive visualization.
- **Tableau:** It is a business intelligence (BI) software tool that supports interactive and visual analysis of data. It has an in-memory data engine to accelerate visualization.
- Tableau has three main products to process large-scale datasets, including Tableau Desktop, Tableau Sever, and Tableau Public.
- Tableau also embed Hadoop infrastructure. It uses Hive to structure queries and cache information for in-memory analytics. Caching helps reduce the latency of a Hadoop cluster. Therefore, it can provide an interactive mechanism between users and Big Data applications ^[5].

Big data processing tools can process ZB (zettabytes) and PB (petabytes) data quite naturally, but they often cannot visualize ZB and PB data.

- At present, big data processing tools include Hadoop, High Performance Computing and Communications, Storm, Apache Drill, RapidMiner, and Pentaho BI.

- Data visualization tools include NodeBox, R, Weka, Gephi, Google Chart API, Flot, D3, and Visual.ly, etc. A big data visualization algorithm analysis integrated model based on RHadoop was proposed.
- The integrated model can process ZB and PB data and show valuable results via visualization. The model is suitable for the design of parallel algorithms for ZB and PB data.

Interactive visual cluster analysis is the most intuitive way for discovering clustering patterns.

- The most challenging step is visualizing multidimensional data and allowing users to interactively explore the data and identify clustering structures.
- Optimized star-coordinate visualization models for effective interactive cluster exploration on big data were developed.
- The star-coordinate models are probably the most scalable technique for visualizing large datasets compared with other multidimensional visualization methods such as parallel coordinates and scatter-plot matrix.

Parallel coordinates and scatter-plot matrix are often used for less than ten dimensions, while star coordinates can handle tens of dimensions.

- The star-coordinate visualization can scale up to many points with the help of density-based representation.
- Star-coordinate based cluster visualization does not try to calculate pairwise distances between records; it uses the property of the underlying mapping model to partially keep the distance relationship.
- This is very useful in processing big data.

Direct visualization of big data sources is often not possible or effective. Analytics plays a key role by helping reduce the size and complexity of big data.

- The visualization and analytics can be integrated so that they work best.
- IBM has embedded visualization capabilities into business analytics solutions.
- What makes this possible is the IBM Rapidly Adaptive Visualization Engine (RAVE). RAVE and extensible visualization capabilities help use effective visualization that provides a better understanding of big data.

IBM products, such as IBM® InfoSphere® BigInsights™ and IBM SPSS® Analytic Catalyst, use visualization libraries and RAVE to enable interactive visualizations that can help gain great insight from big data.

- InfoSphere BigInsights is the software that helps analyze and discover business insights hidden in big data.
- SPSS Analytic Catalyst automates big data preparation, chooses proper analytics procedures, and display results via interactive visualization.

The use of immersive virtual reality (VR) platforms for scientific data visualization has been in the process of exploration including software and inexpensive commodity hardware.

- These potentially powerful and innovative tools for multi-dimensional data visualization can provide an easy path to collaborative data visualization. Immersion provides benefits beyond traditional “desktop” visualization tools: it results in a better perception of data scape geometry and more intuitive data understanding.

- Immersive visualization should become one of the foundations to explore the higher dimensionality and abstraction that are attendant with big data.
- The intrinsic human pattern recognition (or visual discovery) skills should be maximized through using emerging technologies.

Tools used in data visualization.

- Tableau is often regarded as the grand master of data visualization software and for good reason.
- Tableau has a very large customer base of 57,000+ accounts across many industries due to its simplicity of use and ability to produce interactive visualizations far beyond those provided by general BI solutions.
- It is particularly well suited to handling the huge and very fast-changing datasets which are used in Big Data operations, including artificial intelligence and machine learning applications, thanks to integration with a large number of advanced database solutions including Hadoop, Amazon AWS, My SQL, SAP and Teradata.

Extensive research and testing has gone into enabling Tableau to create graphics and visualizations as efficiently as possible, and to make them easy for humans to understand.

- It is particularly well suited to handling the huge and very fast-changing datasets which are used in Big Data operations, including artificial intelligence and machine learning applications, thanks to integration with a large number of advanced database solutions including Hadoop, Amazon AWS, My SQL, SAP and Teradata.
- Extensive research and testing has gone into enabling Tableau to create graphics and visualizations as efficiently as possible, and to make them easy for humans to understand.

Qlikview

- Qlik with their Qlikview tool is the other major player in this space and Tableau's biggest competitor. The vendor has over 40,000 customer accounts across over 100 countries, and those that use it frequently cite its highly customizable setup and wide feature range as a key advantage.
- This however can mean that it takes more time to get to grips with and use it to its full potential. In addition to its data visualization capabilities Qlikview offers powerful business intelligence, analytics and enterprise reporting capabilities and I particularly like the clean and clutter-free user interface.

- Qlikview is commonly used alongside its sister package, QlikSense, which handles data exploration and discovery.
- There is also a strong community and there are plenty of third-party resources available online to help new users understand how to integrate it in their projects.

- This is a very widely-used, JavaScript-based charting and visualization package that has established itself as one of the leaders in the paid-for market.
- It can produce 90 different chart types and integrates with a large number of platforms and frameworks giving a great deal of flexibility.
- One feature that has helped make FusionCharts very popular is that rather than having to start each new visualization from scratch, users can pick from a range of “live” example templates, simply plugging in their own data sources as needed.

- Like FusionCharts this also requires a licence for commercial use, although it can be used freely as a trial, non-commercial or for personal use. Its website claims that it is used by 72 of the world's 100 largest companies and it is often chosen when a fast and flexible solution must be rolled out, with a minimum need for specialist data visualization training before it can be put to work.
- A key to its success has been its focus on cross-browser support, meaning anyone can view and run its interactive visualizations, which is not always true with newer platforms.

Datawrapper is increasingly becoming a popular choice, particularly among media organizations which frequently use it to create charts and present statistics. It has a simple, clear interface that makes it very easy to upload csv data and create straightforward charts, and also maps, that can quickly be embedded into reports.

Plotly enables more complex and sophisticated visualizations, thanks to its integration with analytics-oriented programming languages such as Python, R and Matlab. It is built on top of the open source d3.js visualization libraries for JavaScript, but this commercial package (with a free non-commercial licence available) adds layers of user-friendliness and support as well as inbuilt support for APIs such as Salesforce.

- Sisense provides a full stack analytics platform but its visualization capabilities provide a simple-to-use drag and drop interface which allow charts and more complex graphics, as well as interactive visualizations, to be created with a minimum of hassle.
- It enables multiple sources of data to be gathered into one easily accessed repositories where it can be queried through dashboards instantaneously, even across Big Data-sized sets. Dashboards can then be shared across organizations ensuring even non technically-minded staff can find the answers they need to their problems.

Data Visualization

- Data visualization allows us to interpret data
- It allows us to play with various parameters and its impact on overall outcome or prediction
- To provide more insight
- Exploratory tool for data scientist

Types of Visualization

- Scientific Visualization
 - Structural Data – Seismic, Medical
- Information Visualization –
 - No inherent structure – News, stock market, top grossing movies, facebook connections
- Visual Analytics – Use visualization to understand and synthesize large amounts of multimodal data – audio, video, text, images, networks of people ..

Why visualize data?

- Observe the patterns
- Identify extreme values that could be anomalies
- Easy interpretation
- To provide requires and crisp solution/outcome to management or higher authority
- Incorporate visualization principles to build an interactive visualization of your own data

Types of Plots

- Scatterplot
- Histogram
- Barplot
- Box and whiskers plot
- Pair wise plots

Popular Tools and Software

- Excel
- Python
- R
- Tableau

What is a scatter plot?

- A scatter plot is a set of points that represents the values obtained for two different variables plotted on a horizontal and vertical axes

When to use scatter plots?

- Scatter plots are used to convey the relationship between two numerical variables
- Scatter plots are sometimes called correlation plots because they show how two variables are correlated





Importing data into Spyder

- Importing necessary libraries

```
import pandas as pd
```

← 'pandas' library to work with dataframes

```
import numpy as np
```

← 'numpy' library to do numerical operations

```
import matplotlib.pyplot as plt
```



'matplotlib' library to do visualization

- Importing data

```
cars_data = pd.read_csv('Toyota.csv', index_col=0,  
                        na_values=["??", "????"])
```


Variable explorer		
Name	Type	Size
cars_data	DataFrame	(1436, 10)

- Removing missing values from the dataframe

```
cars_data.dropna(axis = 0, inplace=True)
```

Variable explorer		
Name	Type	Size
cars_data	DataFrame	(1096, 10)

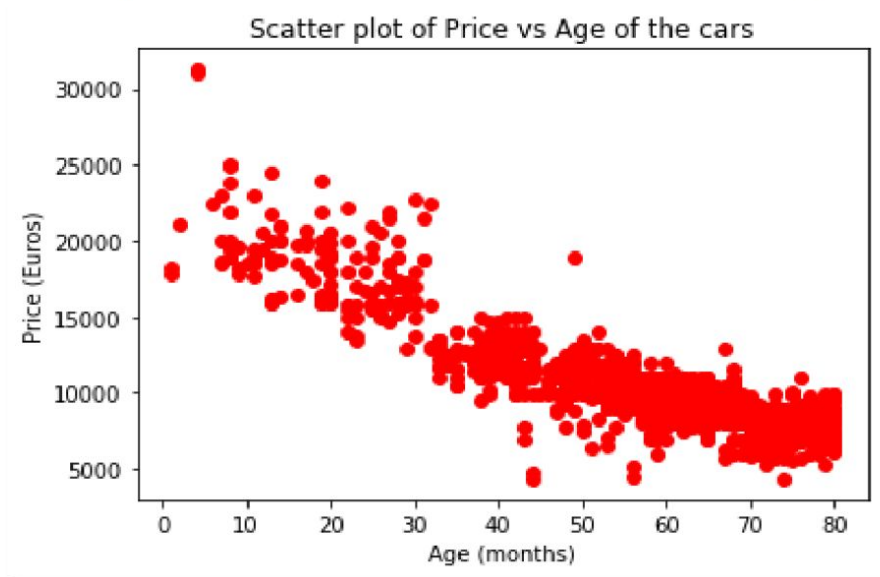
Scatter plot



```
plt.scatter(xcars_data['Age'], ycars_data['Price'], c='red')  
plt.title('Scatter plot of Price vs Age of the cars')  
plt.xlabel('Age (months)')  
plt.ylabel('Price (Euros)')  
plt.show()
```

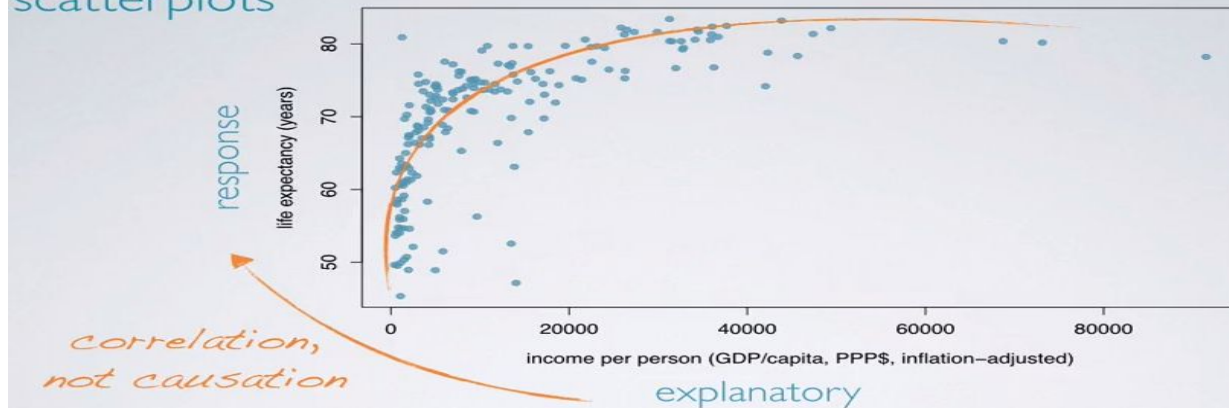
Scatter Plot

- The price of the car decreases as age of the car increases

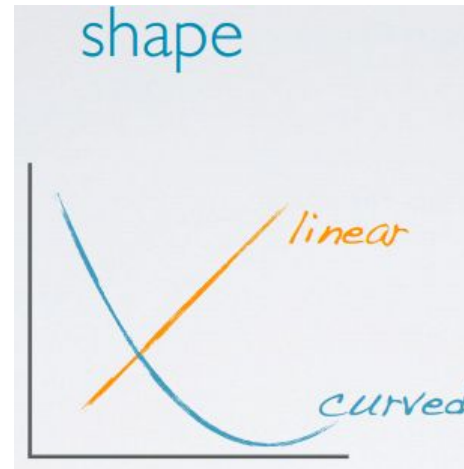


Visualizing Numerical data

Scatter Plot: A common tool for
visualizing the relationship between



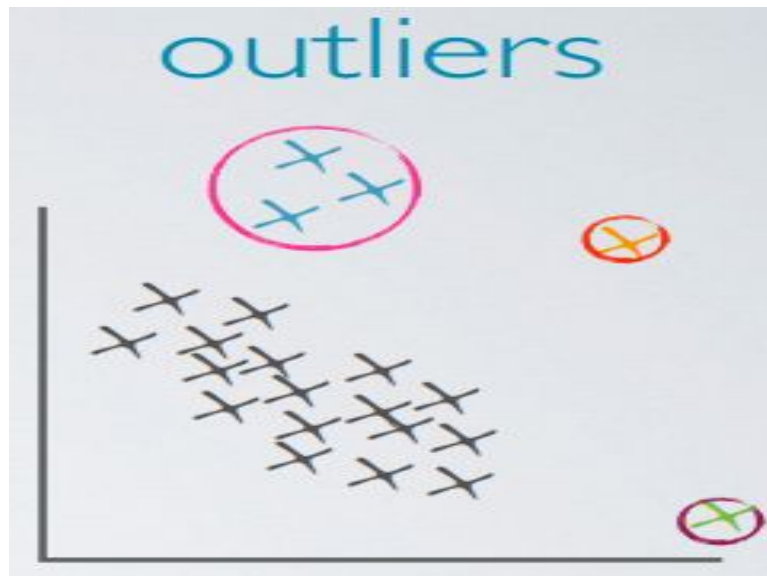
- The shape of the relationship:
- Is it **linear**;
- Or **non-linear**;

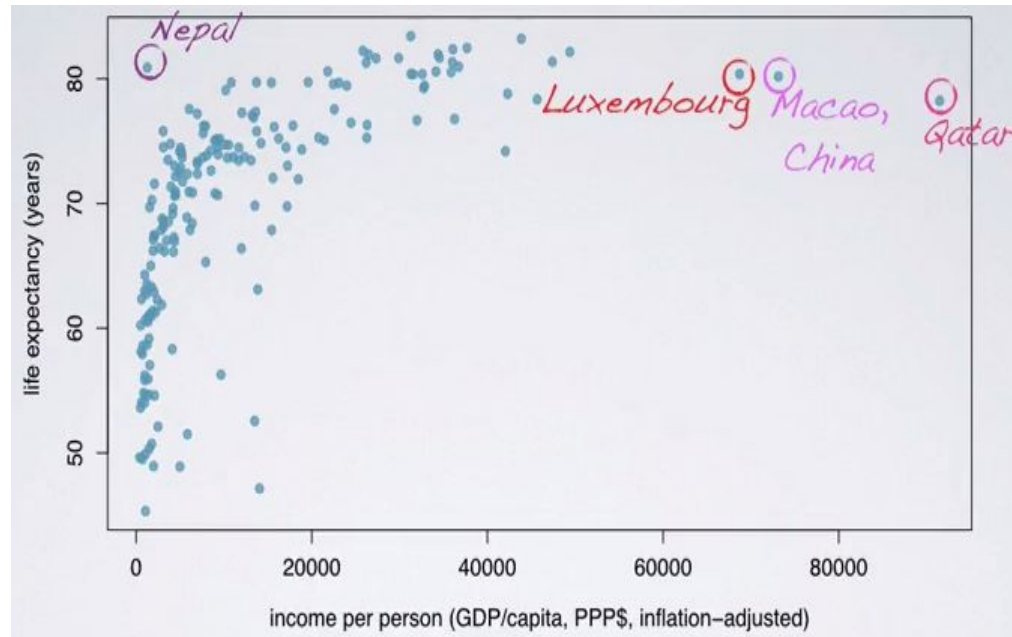


- The strength of the relationship:
- **Strong** indicated by little scatter?
- Or **weak**, indicated by lots of scatter?

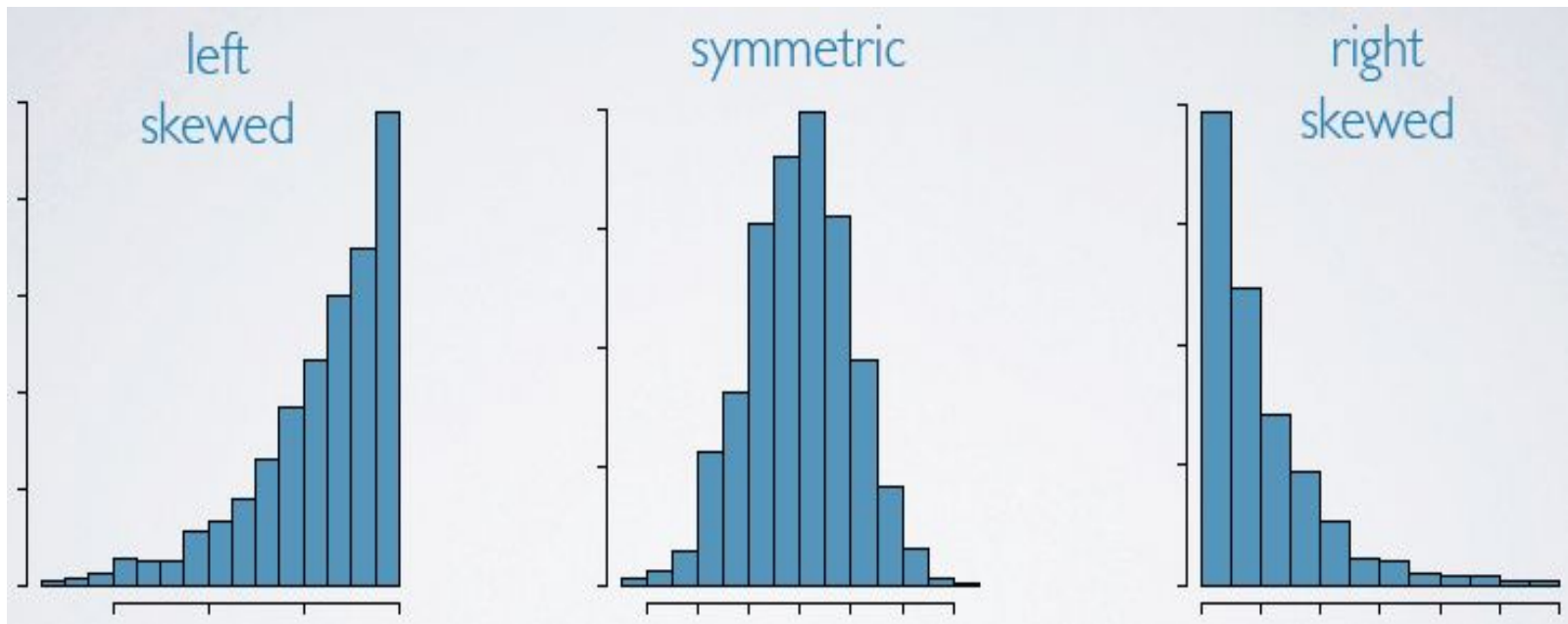


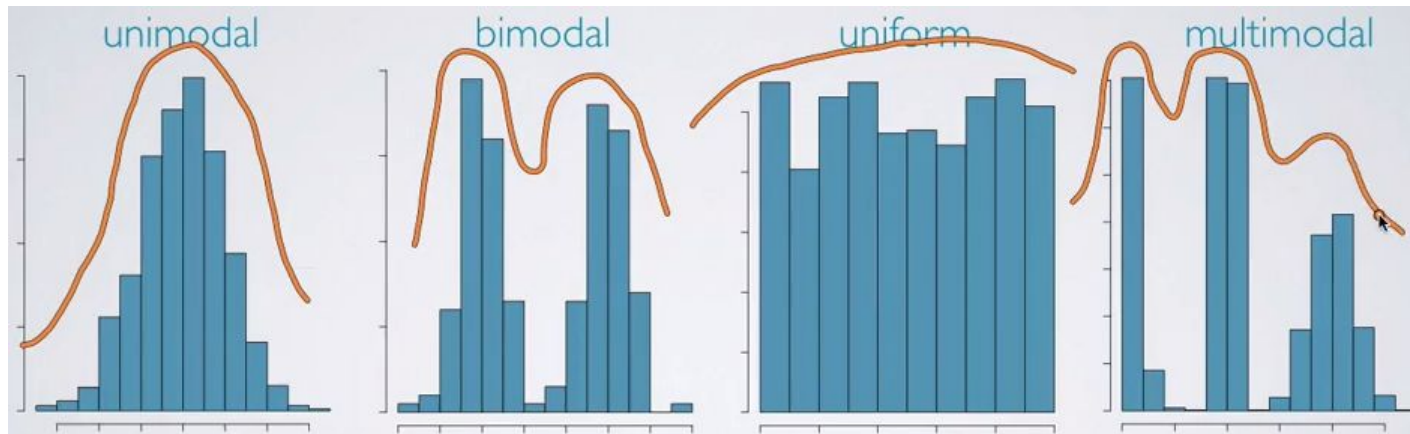
Outliers Detection





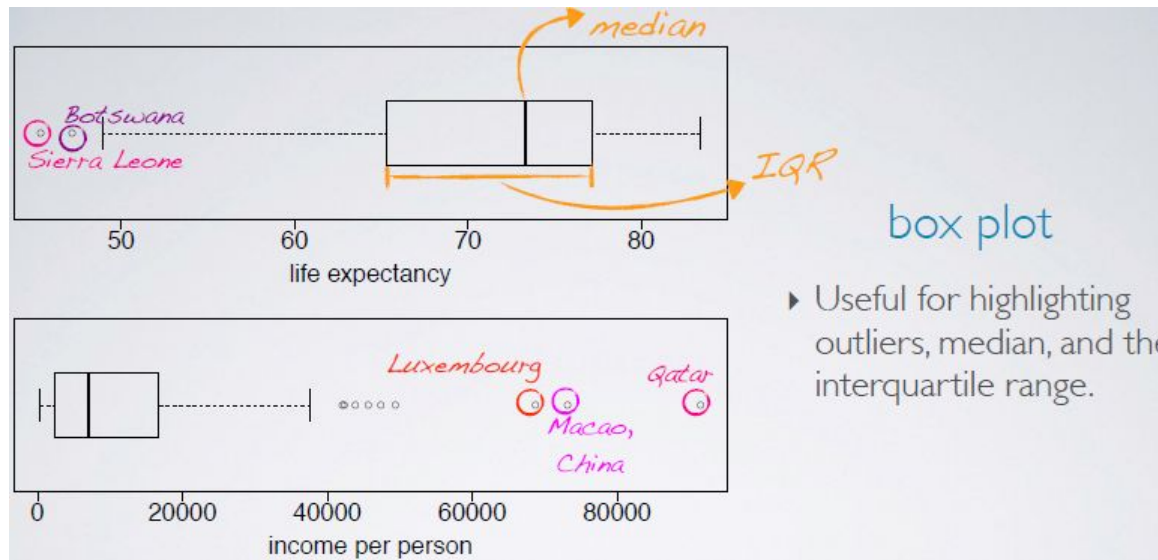
Visualization





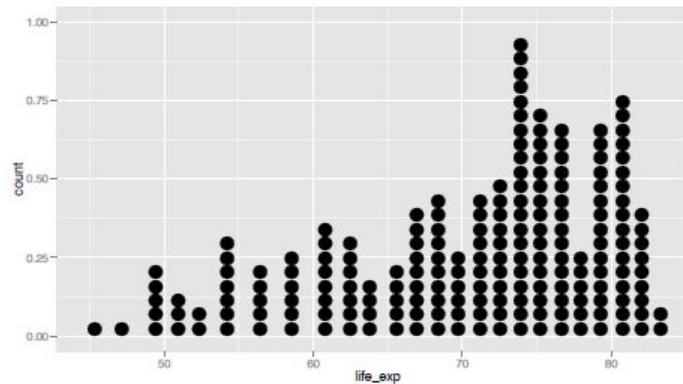
Single Mode prediction, 1 or 2 predictions, Continuous and uniform data prediction , More than 2 predictions

Box plot

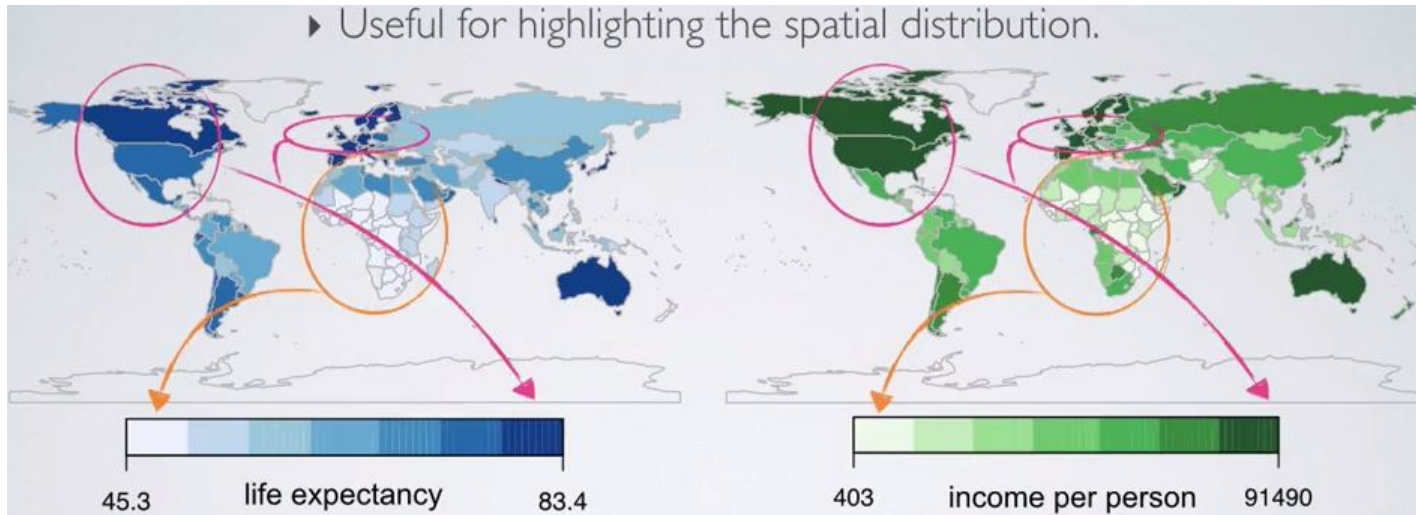


Dot PLOT

- A dot plot is useful especially when individual values are of interest.
- However, as the sample size increases, the dot plot may get too busy.



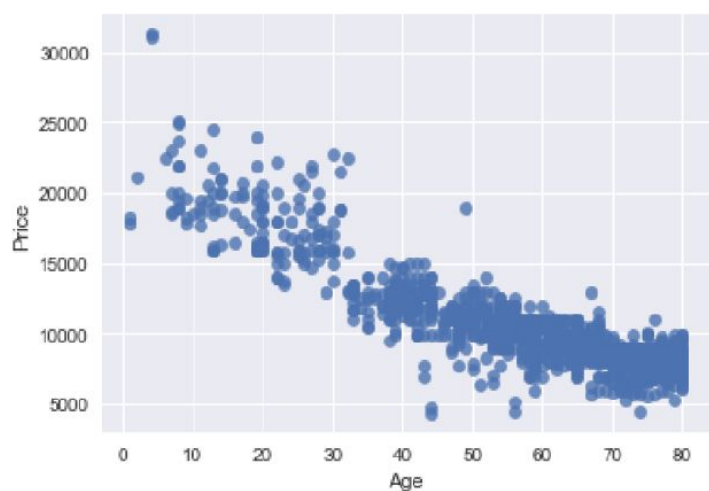
► Useful for highlighting the spatial distribution.



Scatter plot

- Scatter plot of *Price vs Age* without the regression fit line

```
sns.regplot(x=cars_data['Age'], y=cars_data['Price'],  
            fit_reg=False)
```

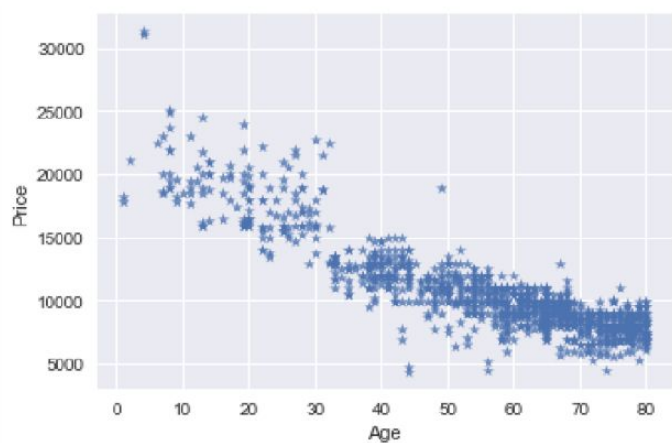


Python for Data Science

Scatter plot

- Scatter plot of *Price vs Age* by customizing the appearance of markers

```
sns.regplot(x=cars_data['Age'], y=cars_data['Price'],  
            marker="*", fit_reg=False)
```



Python for Data Science

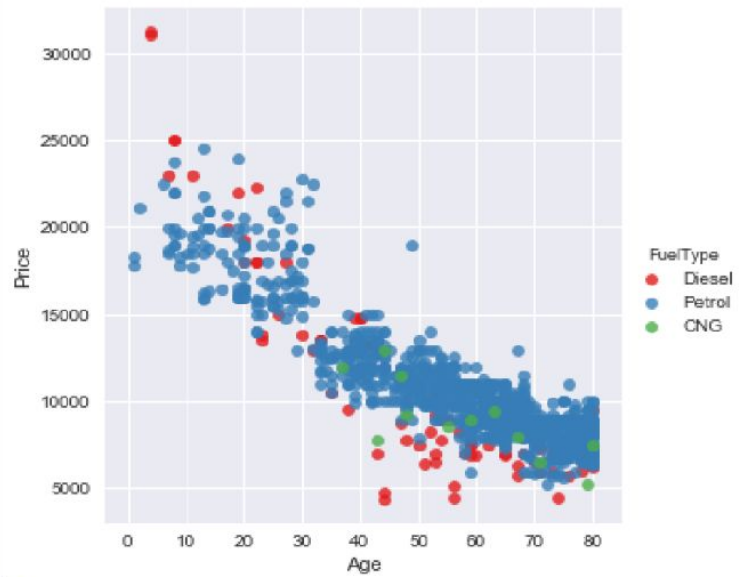
Scatter plot

- Scatter plot of *Price vs Age* by *FuelType*
- Using **hue** parameter, including another variable to show the fuel types categories with different colors

```
sns.lmplot(x='Age', y='Price', data=cars_data,  
           fit_reg=False, hue='FuelType',  
           legend=True, palette="Set1")
```

Scatter plot

- Scatter plot of *Price vs Age by FuelType*



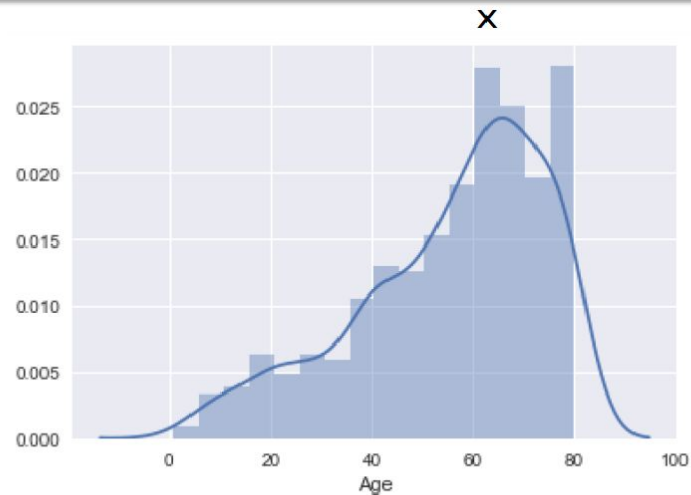
Similarly, custom the appearance of the markers using

- transparency
- shape
- size

Histogram

- Histogram with default kernel density estimate

```
sns.distplot(cars_data['Age'] )
```



Python for Data Science

Histogram

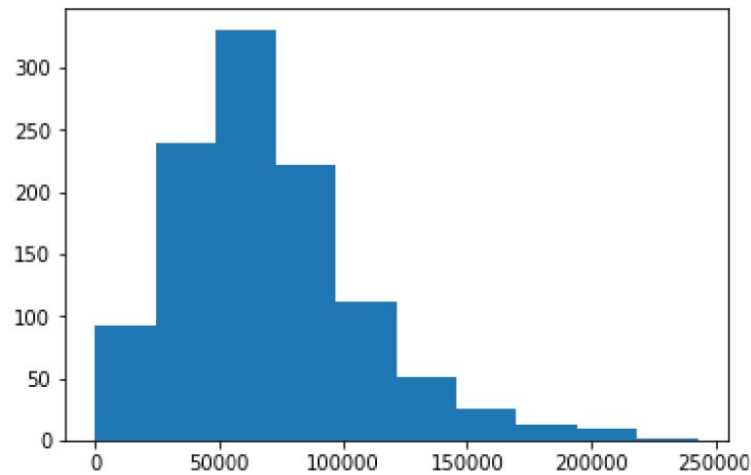
What is a histogram?

- It is a graphical representation of data using bars of different heights
- Histogram groups numbers into ranges and the height of each bar depicts the frequency of each range or bin

When to use histograms?

- To represent the frequency distribution of numerical variables

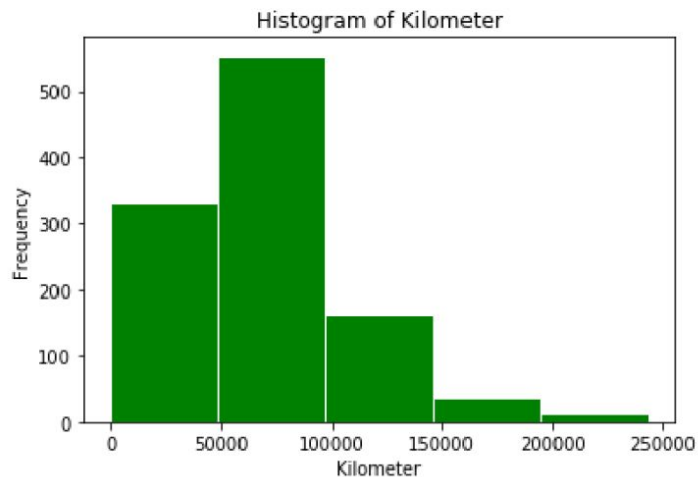
`plt.hist(xcars_data['KM'])` → Histogram with default arguments



Histogram

```
plt.hist(cars_data['KM'],  
         color = 'green',  
         edgecolor = 'white',  
         bins = 5)  
  
plt.title('Histogram of Kilometer')  
plt.xlabel('Kilometer')  
plt.ylabel('Frequency')  
  
plt.show()
```

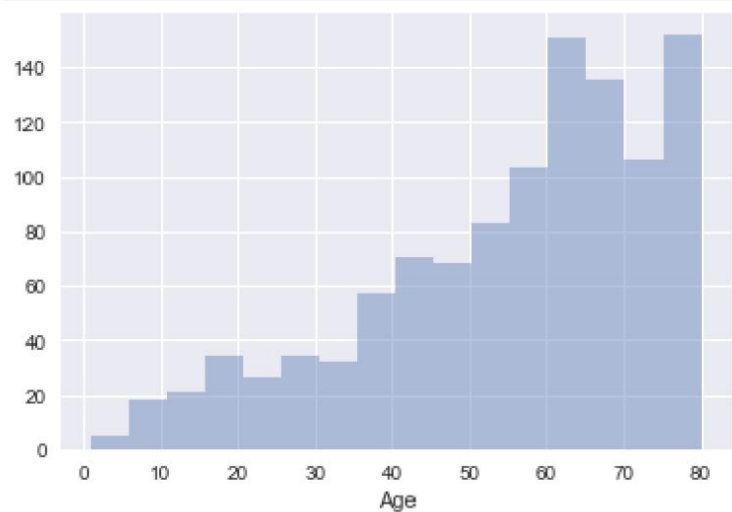
- Frequency distribution of kilometre of the cars shows that most of the cars have travelled between 50000 – 100000 km and there are only few cars with more distance travelled



Histogram

- Histogram without kernel density estimate

```
sns.distplot(cars_data['Age'], kde=False)
```



Python for Data Science

What is a bar plot?

- A bar plot is a plot that presents categorical data with rectangular bars with lengths proportional to the counts that they represent

When to use bar plot?

- To represent the frequency distribution of categorical variables
- A bar diagram makes it easy to compare sets of data between different groups

Bar plot

```
counts    = [979, 120, 12]  
fuelType  = ('Petrol', 'Diesel', 'CNG')  
index     = np.arange(len(fuelType))
```

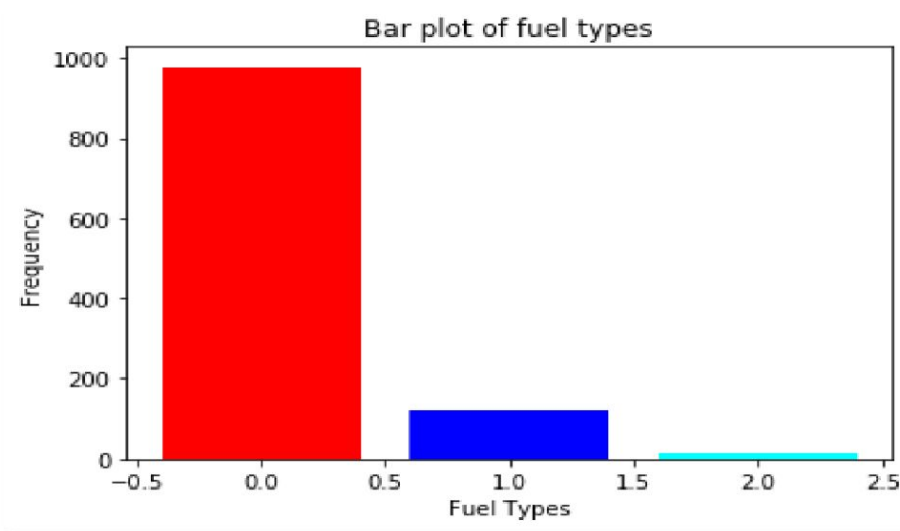
x height of the bars

↓ ↓

```
plt.bar(index, counts, color=['red', 'blue', 'cyan'])  
plt.title('Bar plot of fuel types')  
plt.xlabel('Fuel Types')  
plt.ylabel('Frequency')  
plt.show()
```

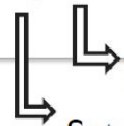
Bar plot

- Frequency distribution of fuel type



```
counts    = [979, 120, 12]  
fuelType  = ('Petrol', 'Diesel', 'CNG')  
index     = np.arange(len(fuelType))
```

```
plt.bar(index, counts, color=['red', 'blue', 'cyan'])  
plt.title('Bar plot of fuel types')  
plt.xlabel('Fuel Types')  
plt.ylabel('Frequency')  
plt.xticks(index, fuelType, rotation = 90)  
plt.show()
```



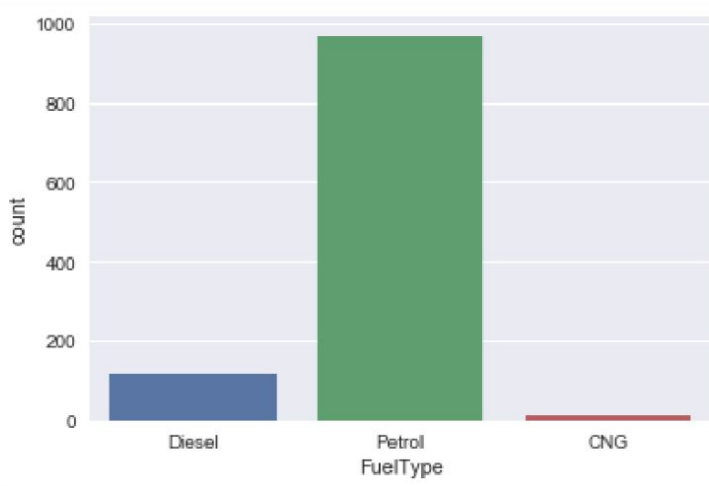
Set the labels of the xticks

Set the location of the xticks

Bar plot

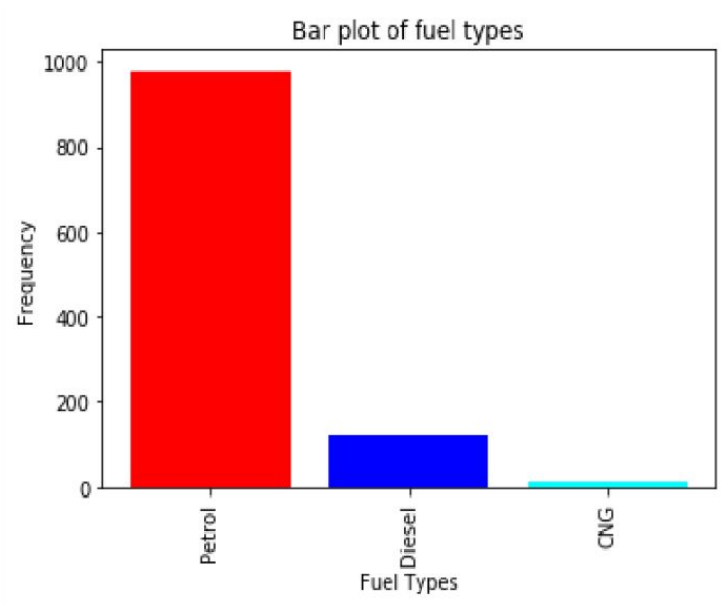
- Frequency distribution of fuel type of the cars

```
sns.countplot(x="FuelType", data=cars_data)
```



Python for Data Science

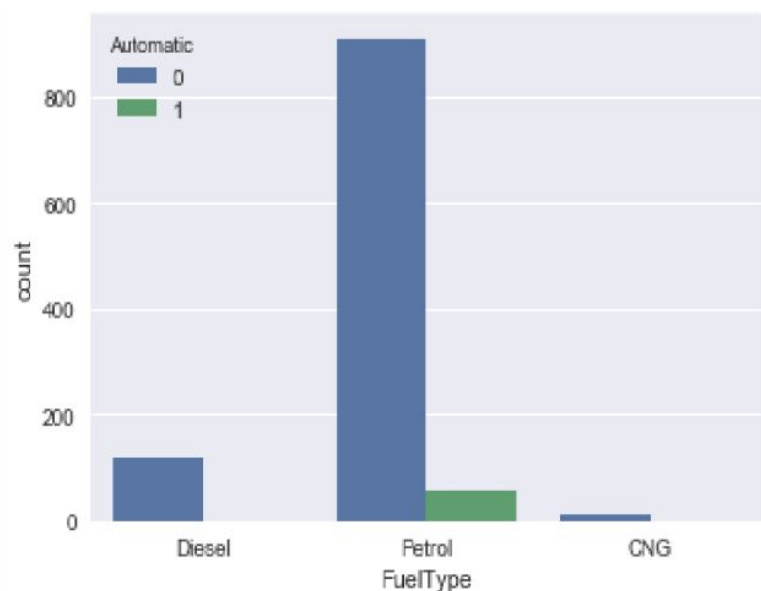
- Bar plot of fuel type shows that most of the cars have petrol as fuel type



Grouped bar plot

- Grouped bar plot of *FuelType* and *Automatic*

```
sns.countplot(x="FuelType", data=cars_data, hue = "Automatic")
```



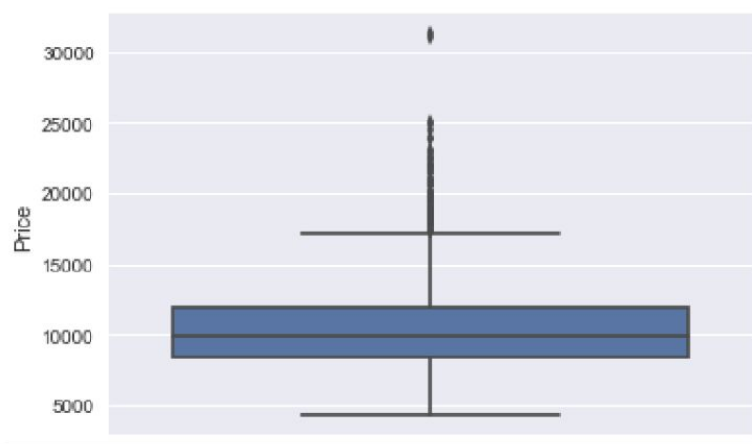
```
pd.crosstab(index = cars_data['Automatic'],
             columns = cars_data2['FuelType'],
             dropna = True)
```

```
Out[5]:
FuelType  CNG  Diesel  Petrol
Automatic
0          15     144   1104
1           0       0     73
```


Box and whiskers plot – numerical variable

- Box and whiskers plot of *Price* to visually interpret the five-number summary

```
sns.boxplot(y=cars_data["Price"] )
```

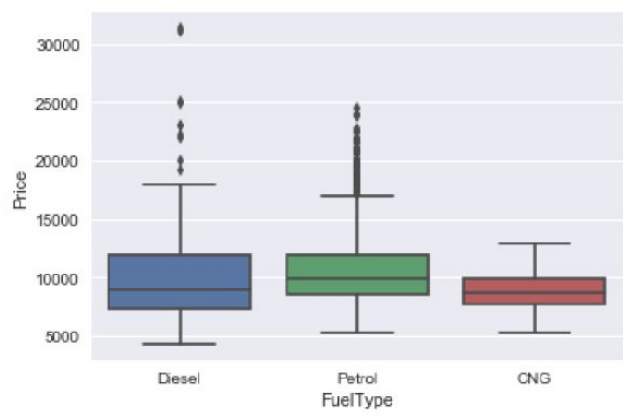


Python for Data Science

Box and whiskers plot

- Box and whiskers plot for numerical vs categorical variable
- Price of the cars for various fuel types

```
sns.boxplot(x = cars_data['FuelType'], y = cars_data["Price"])
```

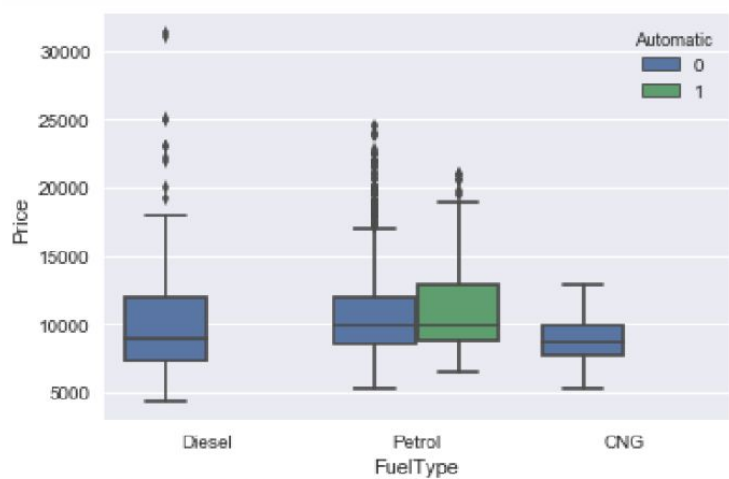


Python for Data Science

Grouped box and whiskers plot

- Grouped box and whiskers plot of *Price* vs *FuelType* and *Automatic*

```
sns.boxplot(x = "FuelType", y = cars_data["Price"],  
            hue = "Automatic", data = cars_data)
```

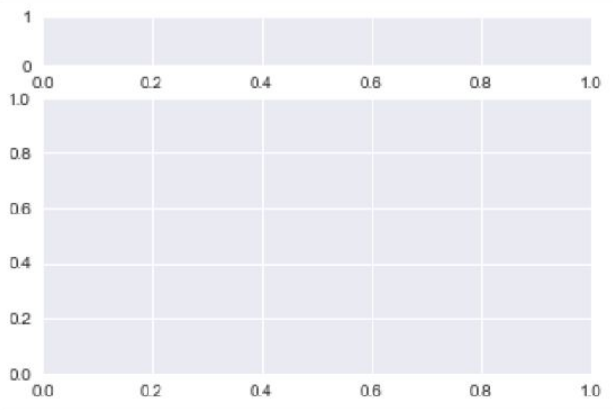


Python for Data Science

Box-whiskers plot and Histogram

- Let's plot box-whiskers plot and histogram on the same window
- Split the plotting window into 2 parts

```
f,(ax_box, ax_hist)=plt.subplots(2, gridspec_kw={"height_ratios": (.15, .85)})
```

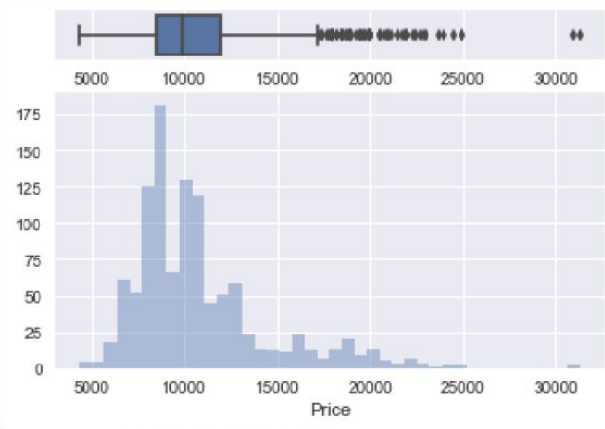


Box-whiskers plot and Histogram

- Now, add create two plots

```
sns.boxplot(cars_data["Price"] , ax=ax_box)
```

```
sns.distplot(cars_data["Price"], ax=ax_hist, kde = False)
```



Python for Data Science

Pairwise relationship using scatter plot and histogram

Code:

```
sns.pairplot(cars_data, kind="scatter", hue="FuelType")  
plt.show()
```