Data Visualization

Using Python

What is Data Visualization?

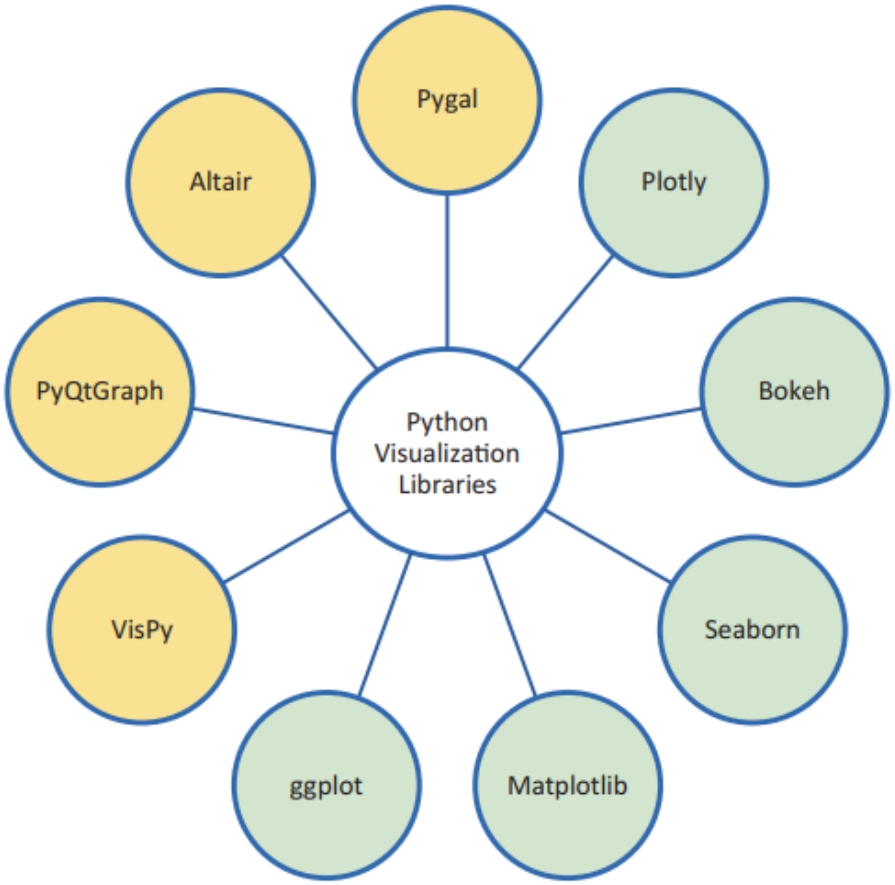
- Data visualization is basically graphical representation of information and data.
- Using Graphs, Curves, Maps, Charts, we can easily understand trends, outliers, and pattern of the data.

- Data visualization plays very vital role in different domain. In the world of Big Data, the Data visualization is essential to analyze massive amount of information, and make data driven decisions.

Advantages of Data Visualization

- Data visualization provides us a quick, clear understanding of the information. We can visualize large amount of data in an understandable and coherent way.
- It helps us to identify emerging trends and act quickly based on what we see. Trends makes more sense when they are represented graphically.
- This helps us to identify relationships and patterns within the digital assets.
- If the visualization is available, then any one can understand this easily. Some other person who do not analyze the data, can easily understand by seeing the visualization.

Data Visualization Libraries in Python



Data visualization libraries

Types of Data Visualization

There are different data visualization techniques available. These are like the following -

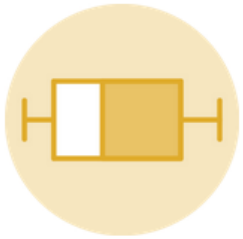
- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

Visualization Type: Distribution

- In this type of Plot, it shows how data spread out over an interval or is grouped. There are four types of Distribution Plots

Violin Plot

Density Plot



Box Plot

Histogram

- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

Visualization Type: Correlation

- Six types of Correlation Plots

Scatter Plot

Bubble Plot

2D Density Plot

Connected Scatter Plot

Heatmap

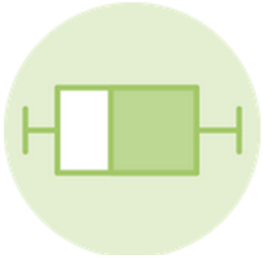
Correlogram

- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

Visualization Type: Ranking

- Six types of Ranking Plots

Bar Plot



Box Plot

Parallel Plot

Lollypop Plot

Wordcloud

Spider

- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

Visualization Type: Part of a Whole

- Six types of Part of a Whole Plots

Stacked Bar Plot

Venn Diagram

Pie Plot

Tree Plot

Doughnut Plot

Tree Diagram

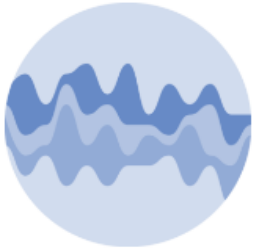
- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

Visualization Type: Evolution

- Five types of Evolution Plots

Line Plot

Stacked
Area Plot



Stream
Chart

Area Plot

Parallel Plot

- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

Visualization Type: Maps

- Four types of Maps Plot

Map

Connection Map

Choropleth Map

Bubble Map

- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

Visualization Type: Flow

- Three types of Flow Plot

Chord Diagram

Network Chart

Sankey Diagram

- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

Some other visualization types

- Some six other visualization techniques

Animation

Data Art

3D Plot

Cheat Sheet

Color

Bad Chart

- Distribution
- Correlation
- Ranking
- Part of a Whole
- Evolution
- Maps
- Flow
- Others

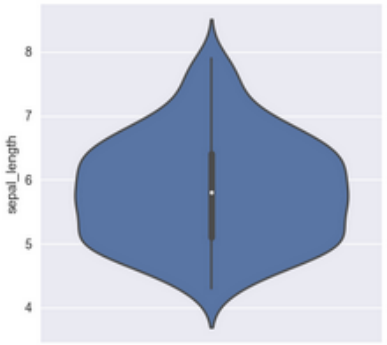
Distribution Plots

Descriptions and Examples

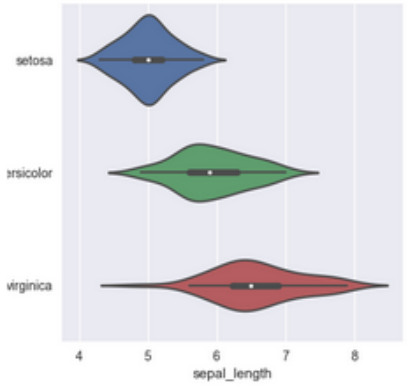
Violin Plot

- Violin Plots allow to visualize the distribution of a numeric variable for one or several groups.
- It is really close from a Box Plot, but allows a deeper understanding of the density.
- Violins are particularly adapted when the amount of data is huge and showing individual observations gets impossible.
- Seaborn is particularly adapted to realize them through its violin function.
- Violin Plots are a really convenient way to show the data and would probably deserve more attention compared to Box Plot that can sometimes hide features of the data.

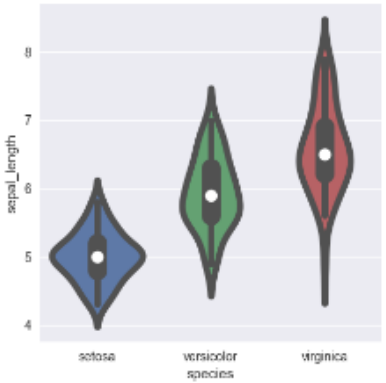
Violin Plot Examples (Seaborn)



Input format for Violin Plot

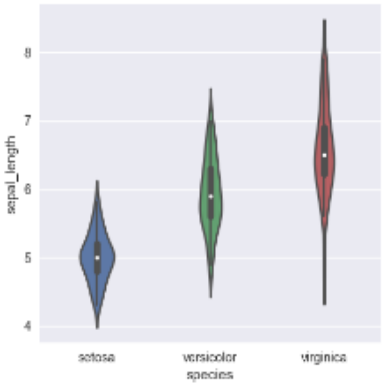


Horizontal Violin Plot

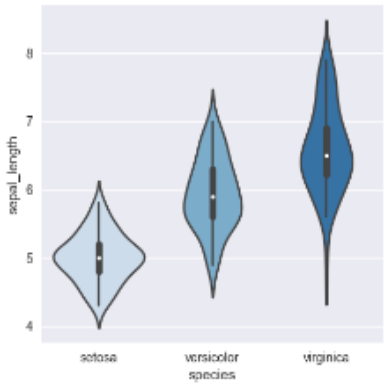


Violin Plot Linewidth

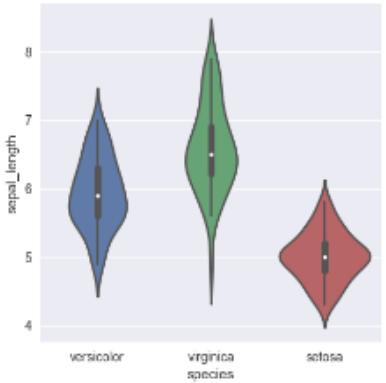
Violin Plot Examples (Seaborn)



Control Violin Plot Width

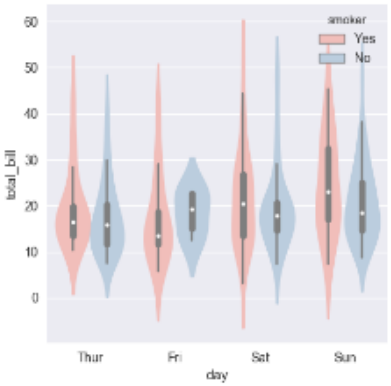


Control Violin Plot Color

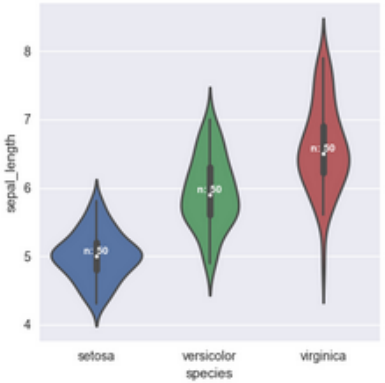


Control Order of Groups on Violin Plot

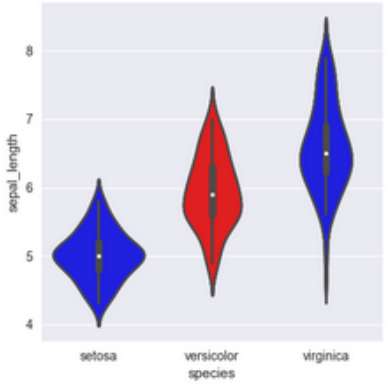
Violin Plot Examples (Seaborn)



Grouped Violin Plot



Show Number of Observations on Violin Plot

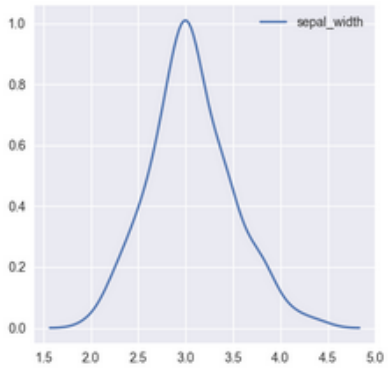


Control Violin Plot Color

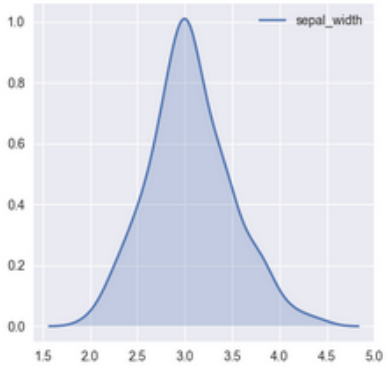
Density Plot

- A Density Plot shows the distribution of a numerical variable.
- It takes only set of numeric values as input. It is really close to a Histogram.
- Since it is a really common dataviz technique, most of the dataviz libraries allow to draw it. Note that it is highly recommended to play with the bandwidth argument in order not to miss a specific pattern in the data.
- Note that you can compare the distribution of several variables Plotting them on the same axis, using faceting or through a Joy Plot.

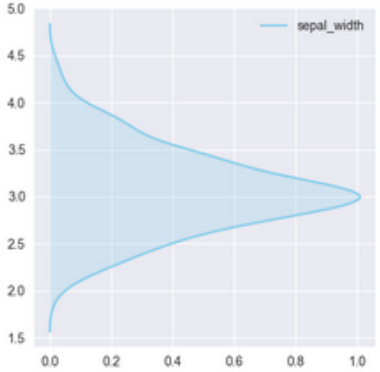
Density Plot Examples (Seaborn)



Basic Density Plot

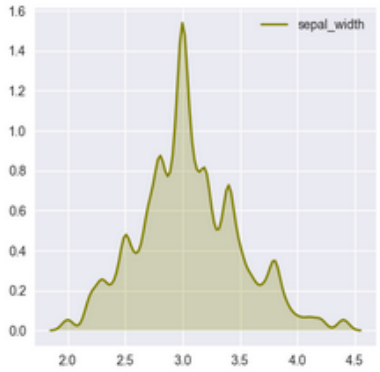


Density Plot with Shade

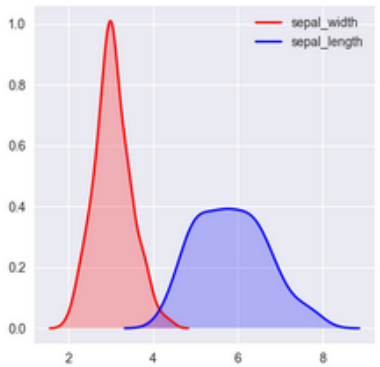


Horizontal Density Plot

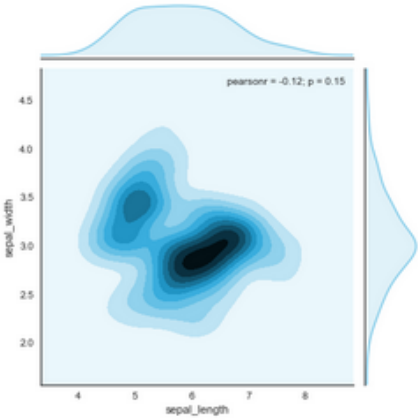
Density Plot Examples (Seaborn)



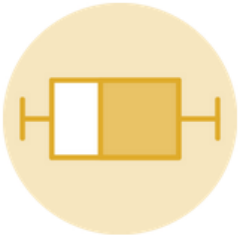
Control Bandwidth of Density Plot



Density Plot for Several Variables

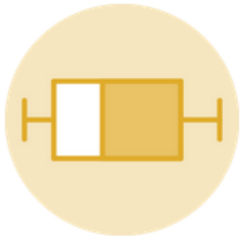


Custom Color of Marginal Plot

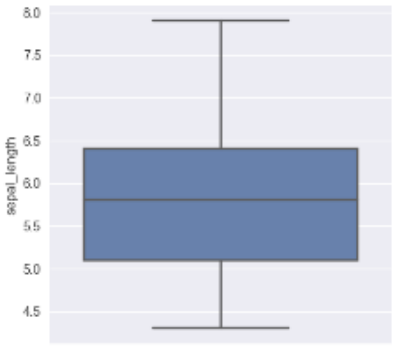


Box Plot

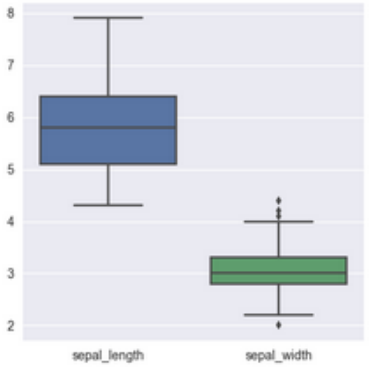
- Box Plot is probably one of the most common type of graphic. It gives a nice summary of one or several numeric variables.
- The line that divides the box into 2 parts represents the median of the data. The end of the box shows the upper and lower quartiles.
- The extreme lines shows the highest and lowest value excluding outliers.
- Note that Box Plot hide the number of values existing behind the variable. Thus, it is highly advised to print the number of observation, add unique observation with jitter or use a Violin Plot if you have many observations.



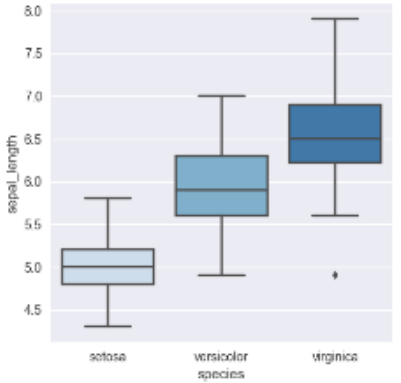
Box Plot Examples (Seaborn)



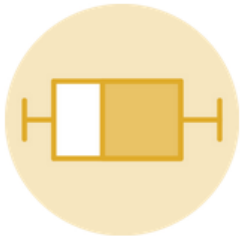
Basic Box Plot and Input Format



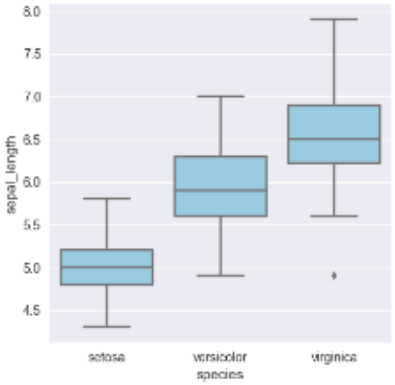
Basic Box Plot and Input Format



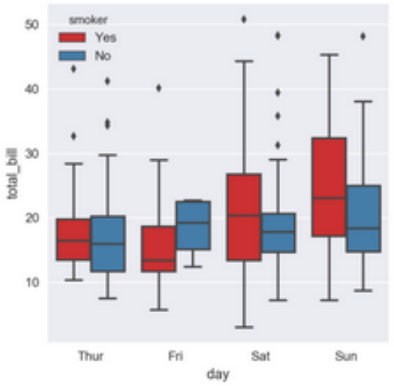
Color Palette on Box Plot



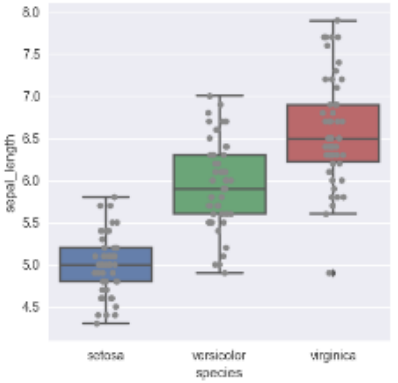
Box Plot Examples (Seaborn)



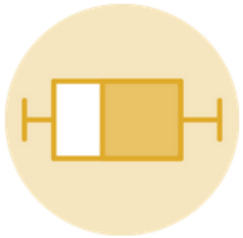
Uniform Color on Seaborn Box Plot



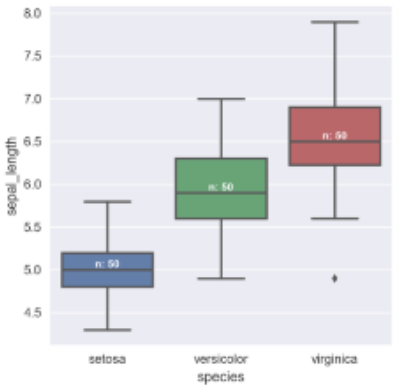
Grouped Box Plot



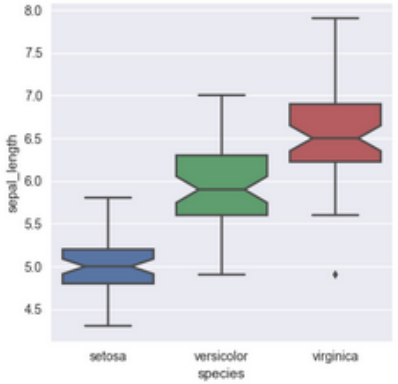
Box Plot with Jitter



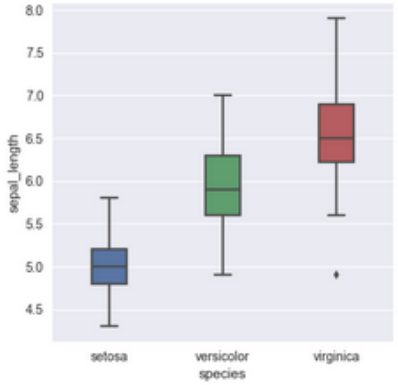
Box Plot Examples (Seaborn)



Show Number of Observation on Box Plot



Add Notch to Seaborn Box Plot

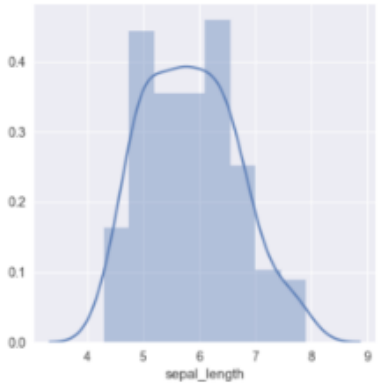


Control Width of Seaborn Box Plot

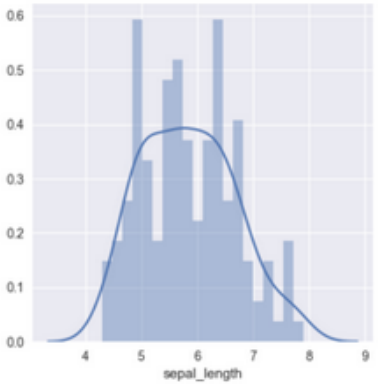
Histogram

- An Histogram is an accurate graphical representation of the distribution of numerical data.
- It takes as input one numerical variable only. The variable is cut into several bins, and the number of observation per bin is represented by the height of the bar.
- Note that the shape of the Histogram can be really different following the number of bins you set. Thus, try different values before taking any conclusion.
- Note that it is really close to Density Plot. To compare several distributions, add them on the same axis or use Violin Plots.

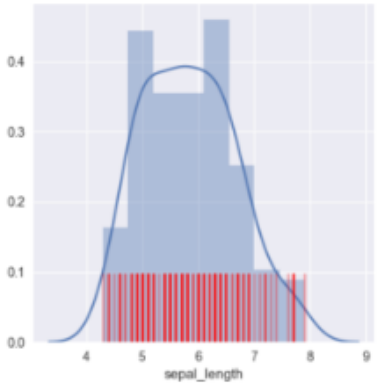
Histogram Examples (Seaborn)



Basic Histogram

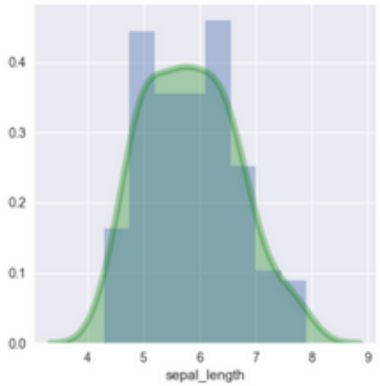


Control Bins on Seaborn Histogram

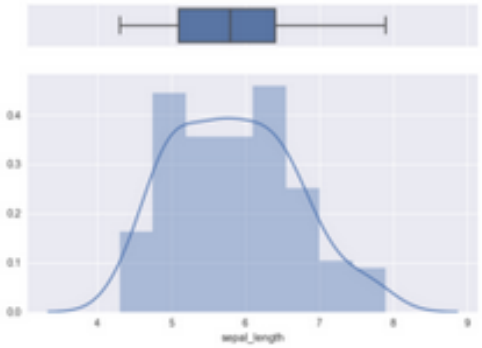


Custom Rug

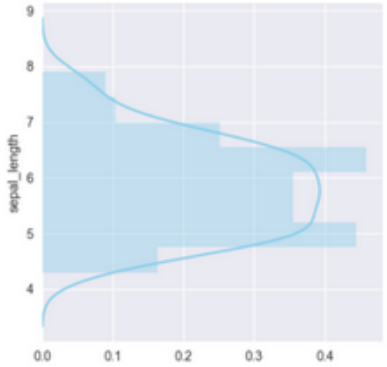
Histogram Examples (Seaborn)



Custom Density

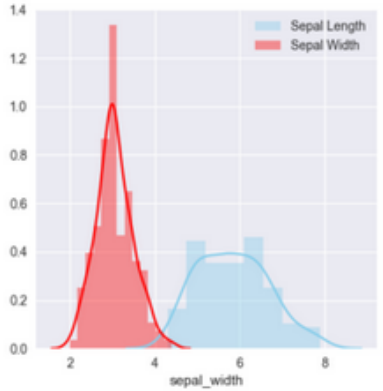


Histogram with Box Plot on Top

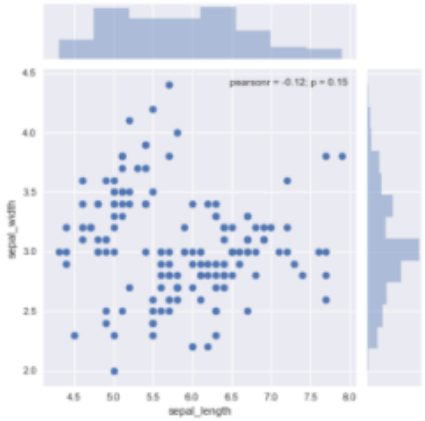


Vertical Histogram

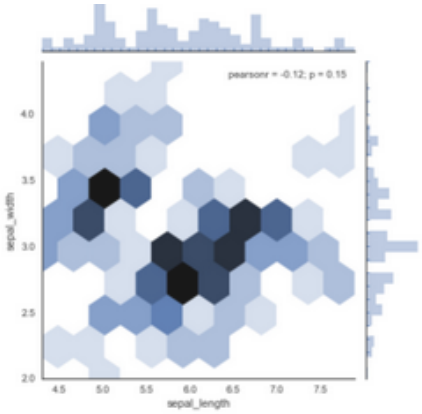
Histogram Examples (Seaborn)



Histogram with Several Variables



Default Marginal Plot



Custom Marginal Area

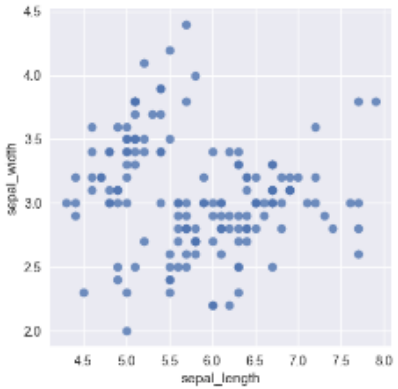
Correlation Plots

Descriptions and Examples

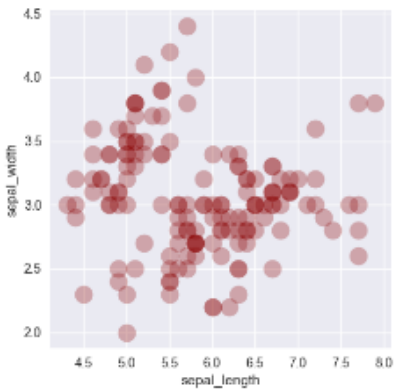
Scatter Plot

- A Scatter Plot displays the value of 2 sets of data on 2 dimensions. Each dot represents an observation. The position on the X (horizontal) and Y (vertical) axis represents the values of the 2 variables.
- It is really useful to study the relationship between both variables. It is common to provide even more information using colors or shapes (to show groups, or a third variable).
- It is also possible to map another variable to the size of each dot, what makes a bubble Plot. If you have many dots and struggle with over Plotting, consider using 2D Density Plot.

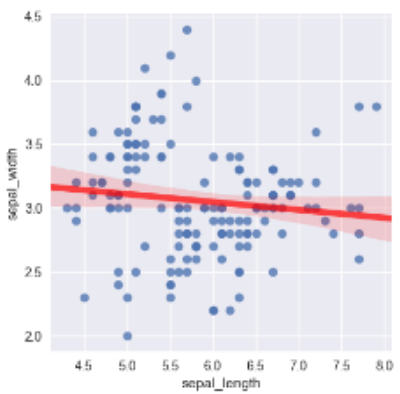
Scatter Plot Examples (Seaborn)



Basic Scatter Plot with Seaborn

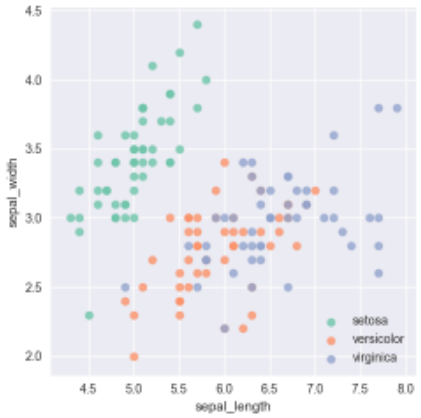


Change Marker Color

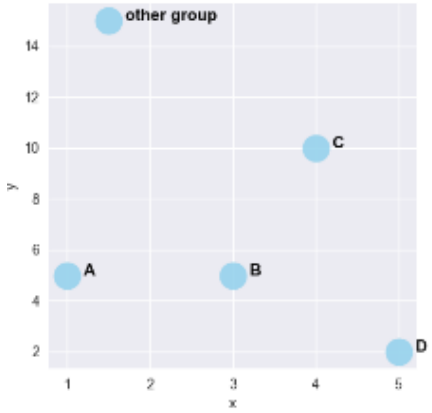


Custom Regression Fit

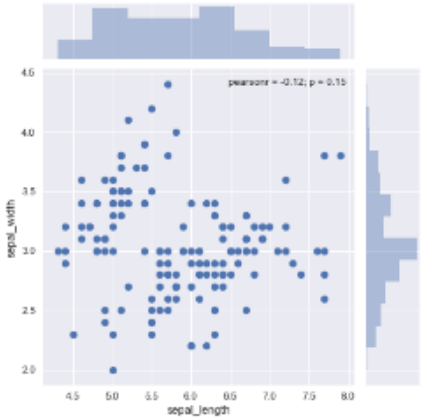
Scatter Plot Examples (Seaborn)



Use Categorical Variable for Color



Use Loop to Add Text Annotation

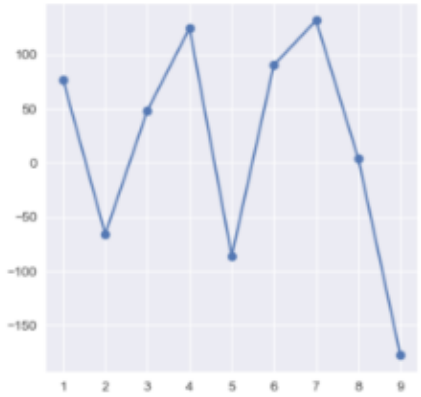


Default Marginal Plot

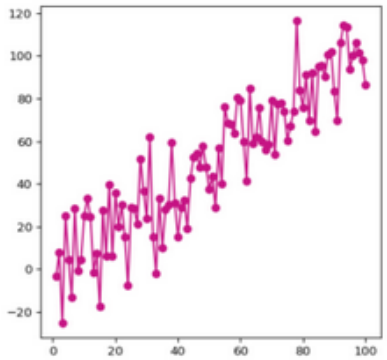
Connected Scatter Plot

- A connected Scatter Plot is really close from a Scatter Plot, except that dots are linked one to each other with lines. This means that the values of your X axis must be ordered to make this type of representation useful. Thus, connected scatter Plot are often used for time series where the X axis represents time.
- If you want to fill the area under the line you will get an Area Chart.

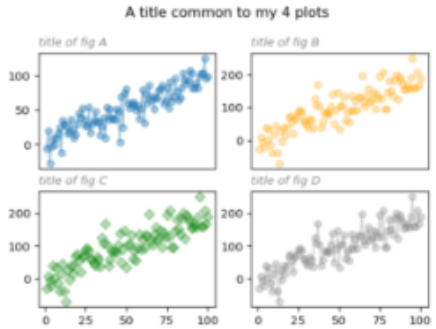
Connected Scatter Plot Examples (MatPlotLib, Seaborn)



Basic Connected Scatter Plot

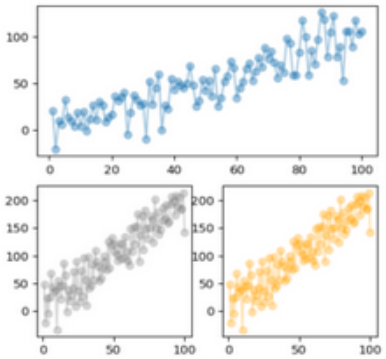


MatPlotLib Style

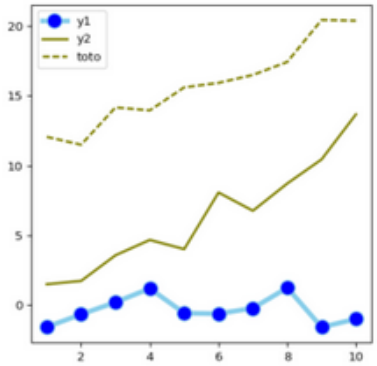


Main Margin for SubPlots

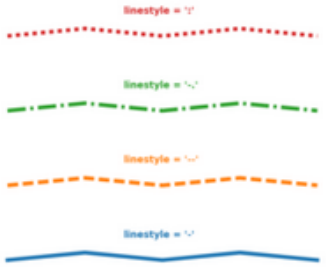
Connected Scatter Plot Examples (MatPlotLib, Seaborn)



Custom Proportion on SubPlot



Line Chart with Several Lines

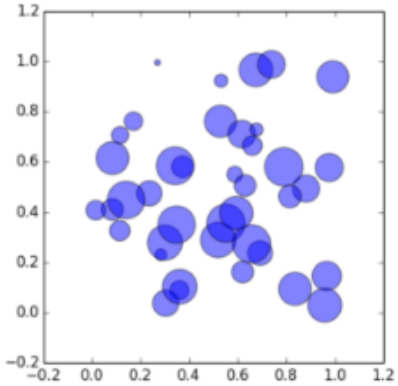


Customize Line Chart

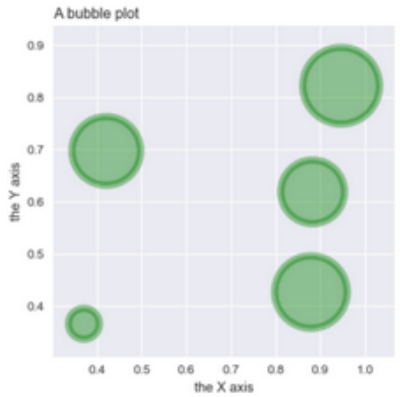
Bubble Plot

- A Bubble Plot is a Scatter Plot where a third dimension is added: the value of an additional variable is represented through the size of the dots.
- You need 3 numerical variables as input: one is represented by the X axis, one by the Y axis, and one by the size.
- Do not forget to provide a legend to make possible the link between the size and the value. Note that too many bubble make the chart hard to read.
- This type of representation is usually not recommended for big amount of data.
- The area of the circles must be proportional to the value, not to the radius, to avoid exaggerate the variation in your data.

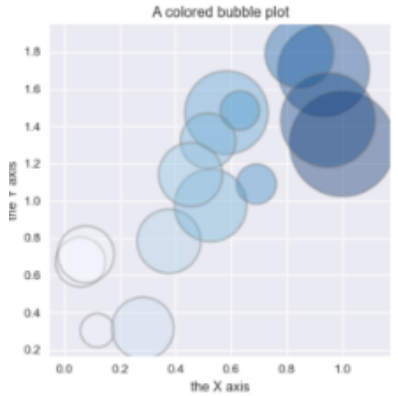
Bubble Plot Examples (MatPlotLib)



Basic Bubble Plot



Bubble with Seaborn style

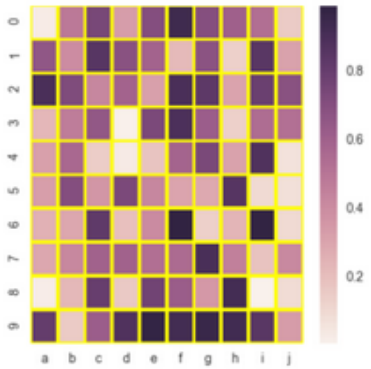


Map a color to bubble

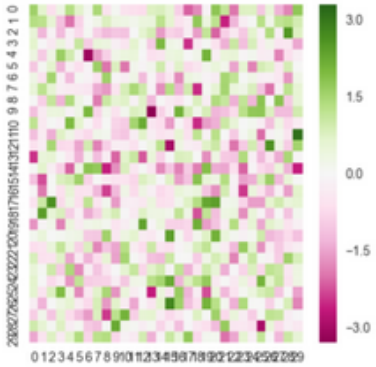
Heatmap

- A Heat Map (or Heatmap) is a graphical representation of data where the individual values contained in a matrix are represented as colors.
- It is a bit like looking a data table from above. It is really useful to display a general view of numerical data, not to extract specific data point. It is quite straight forward to make a Heat Map.

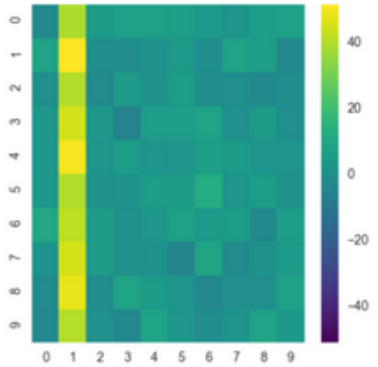
Heatmap Examples (Seaborn)



Custom Seaborn Heatmap



Control Heatmap color

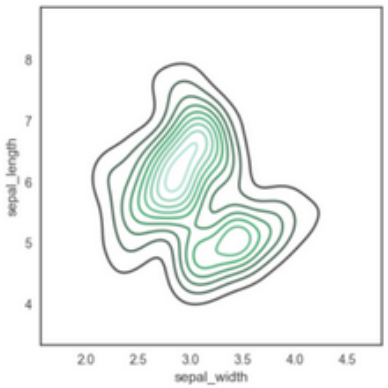


Column normalization on Heatmap

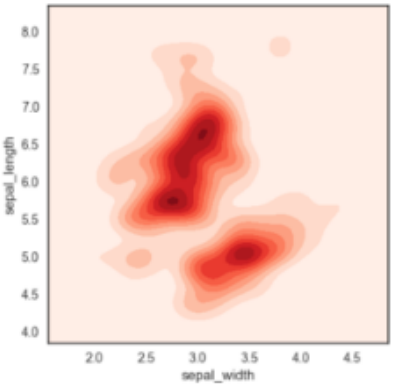
2D Density Plot

- A 2D Density Plot or 2D Histogram is an extension of the well known Histogram. It shows the distribution of values in a data set across the range of two quantitative variables.
- It is really useful to avoid over Plotting in a Scatter Plot. If you have too many dots, the 2D Density Plot counts the number of observations within a particular area of the 2D space. This specific area can be a square or a hexagon (hexbin).
- You can also estimate a 2D kernel density estimation and represent it with contours.

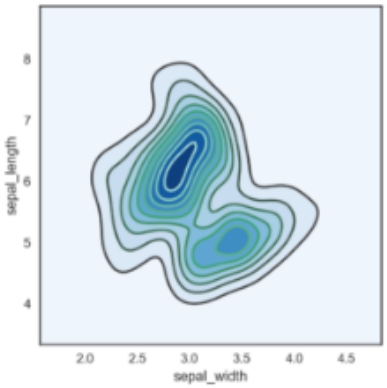
2D Density Plot Examples



Contour Plot with Seaborn

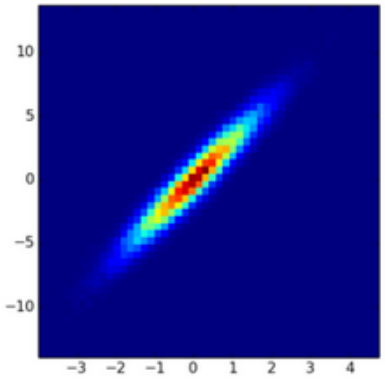


Density Plot with Seaborn

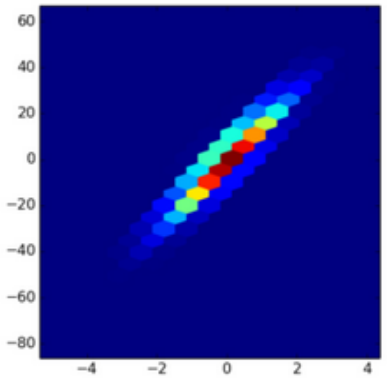


Contour Plot with Seaborn

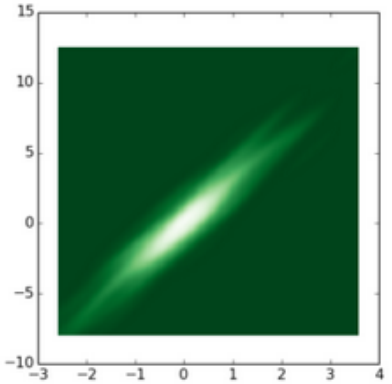
2D Density Plot Examples



Adjust Bin size of 2D Histogram

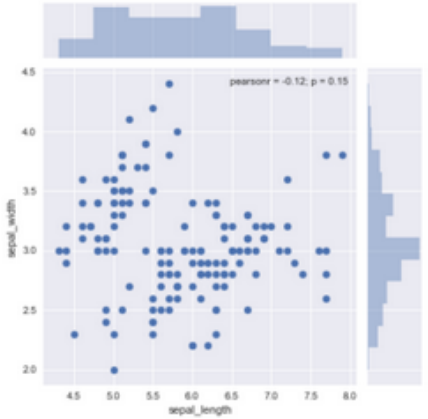


Hexbin Plot with MatPlotLib

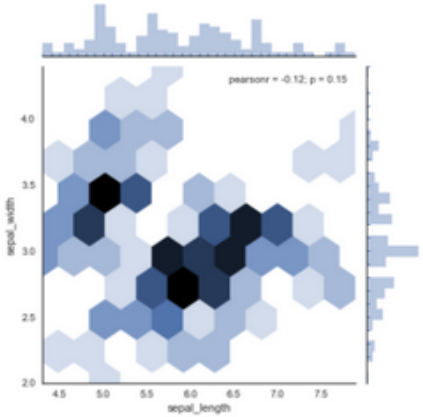


Color of 2D Density Plot

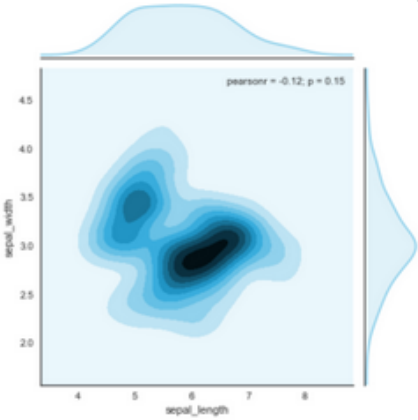
2D Density Plot Examples



Default Marginal Plot



Custom Marginal Area

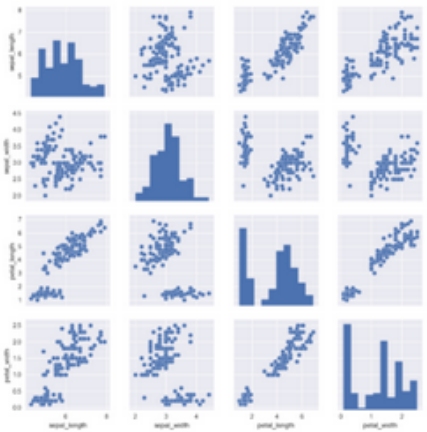


Custom Color of Marginal Plot

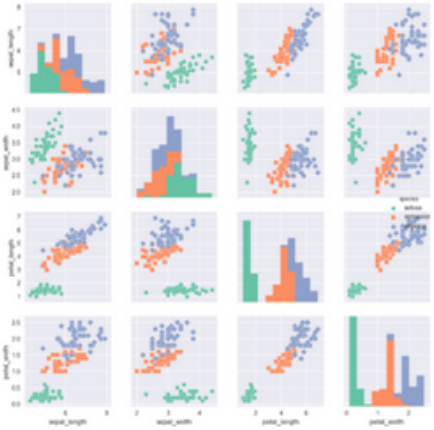
Correlogram Plot

- A Correlogram or correlation matrix allows to analyze the relationship between each pair of numerical variables of a matrix.
- The correlation between each pair of variable is visualize through a Scatter Plot, or a symbol that represents the correlation (bubble, line, number..). The diagonal represents the distribution of each variable, using an Histogram or a Density Plot.
- This technique is widely used for exploratory analysis since it avoids to make hundreds of Plots to observe a matrix.

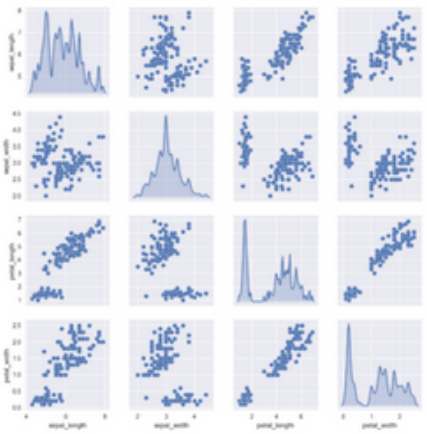
Correlogram Plot Examples (Seaborn)



Start Simple Correlogram



Represent Groups on Correlogram



Custom Distribution

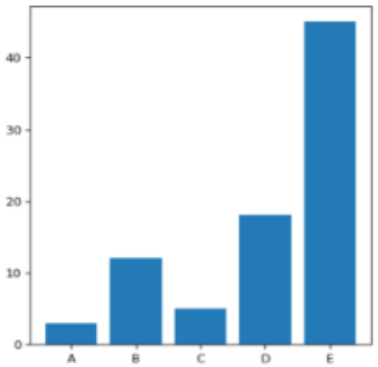
Ranking Plots

Descriptions and Examples

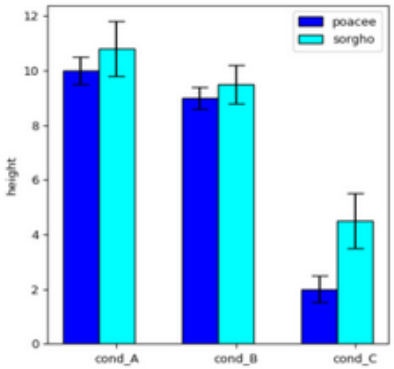
Bar Plot

- A Bar Plot (or Barchart) is one of the most common type of Plot. It shows the relationship between a numerical variable and a categorical variable. For example, you can display the height of several individuals using bar chart.
- Barcharts are often confounded with Histograms, which is highly different. (It has only a numerical variable as input and shows its distribution).
- A common mistake is to use Bar Plots to represent the average value of each group. If you have several values per group, showing only the average dissimulate a part of the information. In this case, consider doing a Box Plot or a Violin Plot.

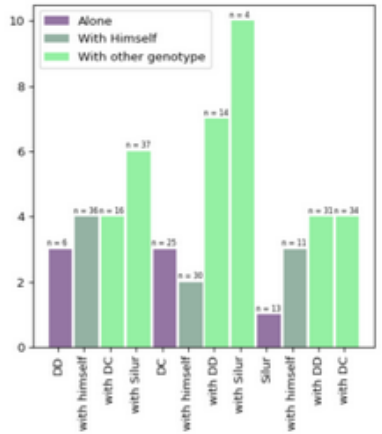
Bar-Plot Examples (MatPlotLib)



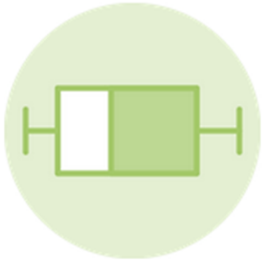
Basic Bar Plot



Add Error Bars



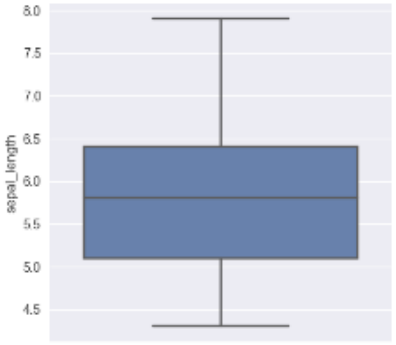
Show Number of Observations One Mar Plot



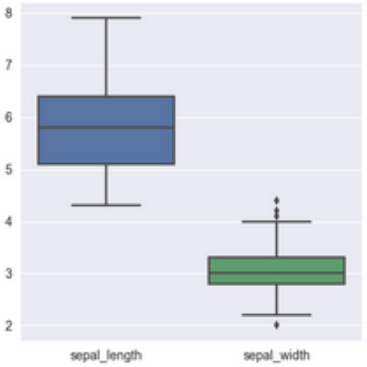
Box Plot

- Box Plot is probably one of the most common type of graphic. It gives a nice summary of one or several numeric variables.
- The line that divides the box into 2 parts represents the median of the data. The end of the box shows the upper and lower quartiles.
- The extreme lines shows the highest and lowest value excluding outliers.
- Note that Box Plot hide the number of values existing behind the variable. Thus, it is highly advised to print the number of observation, add unique observation with jitter or use a Violin Plot if you have many observations.

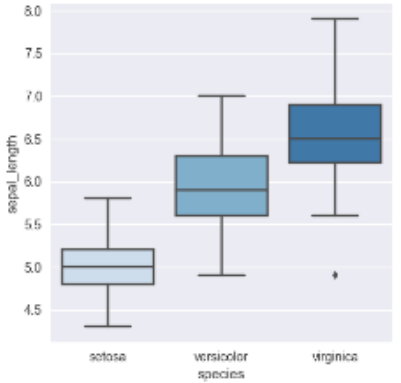
Box Plot Examples (Seaborn)



Basic Box Plot and input Format

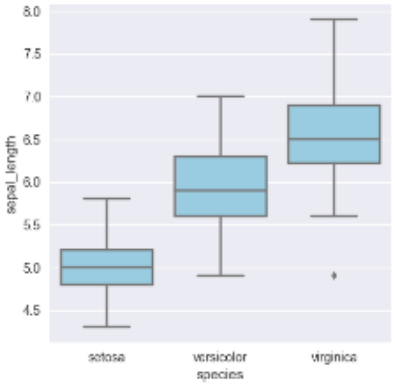


Basic Box Plot and input Format

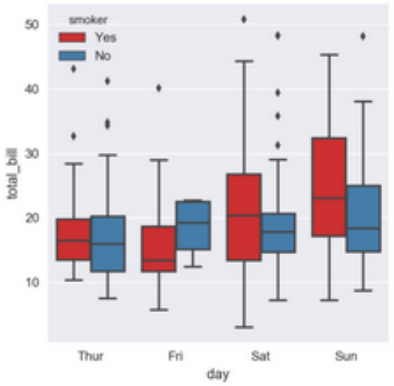


Color Palette on Box Plot

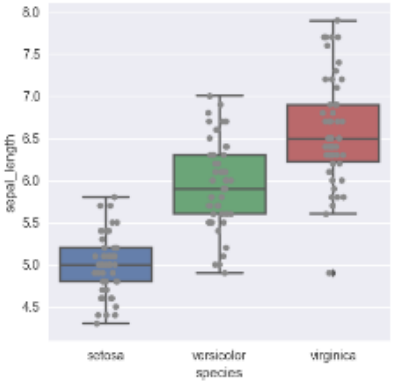
Box Plot Examples (Seaborn)



Uniform Color on Seaborn Box Plot

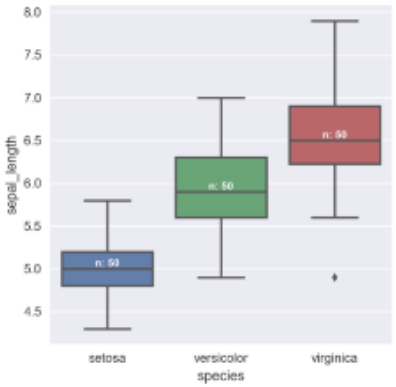


Grouped Box Plot

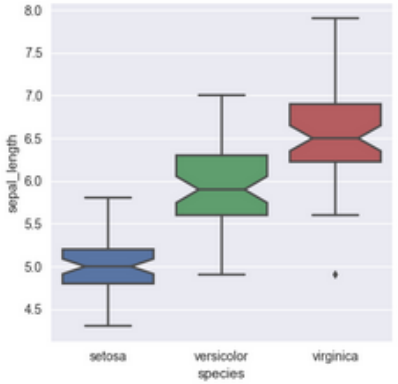


Box Plot with Jitter

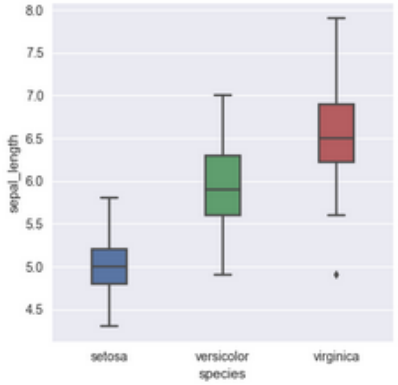
Box Plot Examples (Seaborn)



Show Number of Observation on Box Plot



Add Notch to Seaborn Box Plot

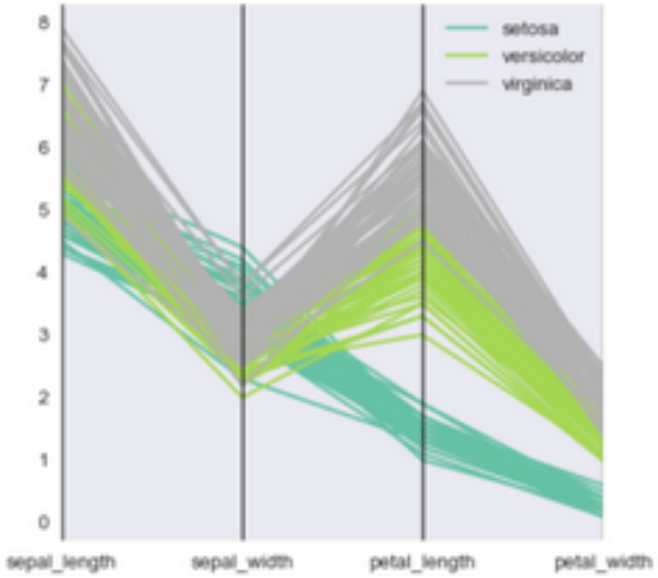


Control Width of Seaborn Box Plot

Parallel Plot

- Parallel Plot or Parallel Coordinates Plots allow to compare the feature of several individual observations on a set of numerical variables. Each vertical bar represents a variable and usually has its own scale. (The units can even be different!). Values are then Plotted as series of lines connected across each axis.
- Playing with color can then be used to represent different groups of individual, or highlight a specific one. That should allow to detect interesting patterns!

Parallel Plot Examples (Pandas)

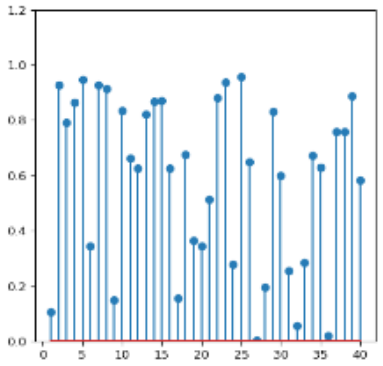


Parallel Plot with Pandas

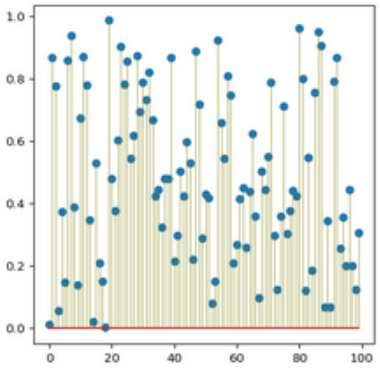
Lollipop Plot

- A Lollipop Plot is an hybrid between a scatter Plot and a Bar Plot. It shows the relationship between a numerical variable and another variable, numerical OR categorical.
- Cleveland dot Plots allows to compare the values of 2 numerical values for each group. Note that Lollipop Plot can be done using the specific stem() function, or using the hline() and vline() functions.

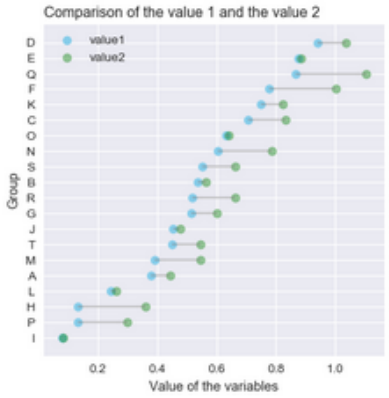
Lollypop Plot Examples (MatPlotLib)



Basic Lollipop Plot



Custom Stem of Lollipop

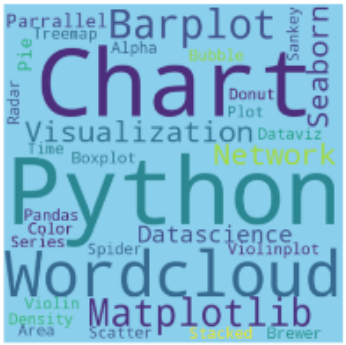


Lollipop with 2 Groups

Wordcloud Plot

- A Wordcloud (or Tag cloud) is a visual representation of text data. It displays a list of words, the importance of each being shown with font size or color.
- This format is useful for quickly perceiving the most prominent terms. Python is totally adapted to draw this kind of representation, thanks to the Wordcloud library developed by Andreas Mueller.

Wordcloud Plot Examples (Wordcloud library basics)



Basic Wordcloud

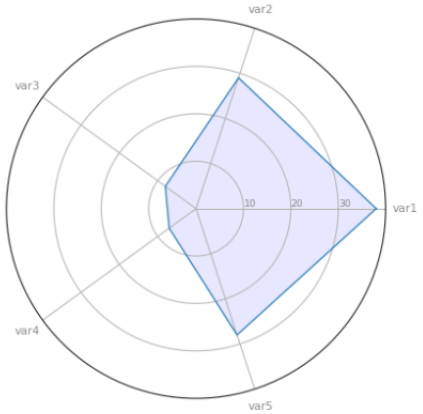
Change words color in Wordcloud

Wordcloud with specific shape

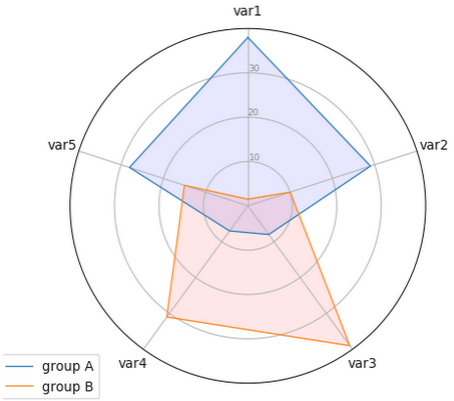
Radar Chart

- A Radar Chart or Spider Plot or Polar chart or Web chart allows to study the feature of one or several individuals for several numerical variables. It is possible to represent several individuals on the same graph but be careful, the chart can quickly become unreadable.
- We can try to use faceting: display as many chart as the number of individual, it makes easy to compare the shape of each.
- Python does not have any built in function to create Radar Chart. Thus, you have to be courageous and dive into the code. Note that Radar Chart can make hard to read values, so often a simple MarPlot or Parallel Plot is advised.

Radar Chart Examples (MatPlotLib)



Basic Radar Chart



Several group on the same Radar Chart

Faceting and Radar Chart

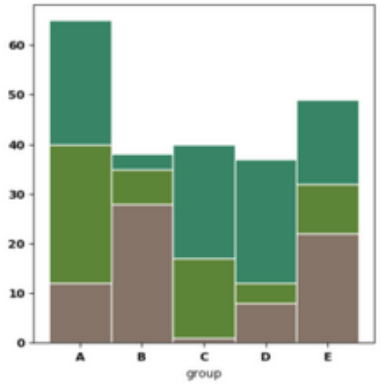
Part of a Whole Plots

Descriptions and Examples

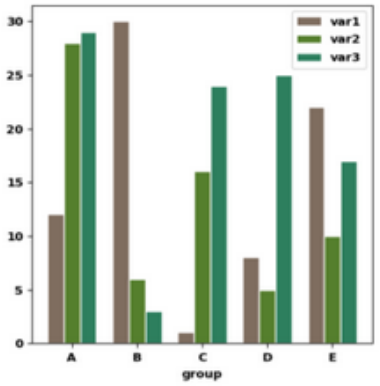
Stacked Bar Plot

- This section display grouped Barcharts, stacked Barcharts and percent stacked Barcharts. This 3 types of Bar Plot variation have the same objective.
- It displays a numerical value for several entities, organized into groups and subgroups. A grouped Bar Plot display the subgroups one beside each other, whereas the stacked ones display them on top of each other.
- The percent variation normalize the data to make in sort the value of each group is 100. It allows the compare the importance of each subgroups in each group more effectively. Note that faceting can be a good alternative.

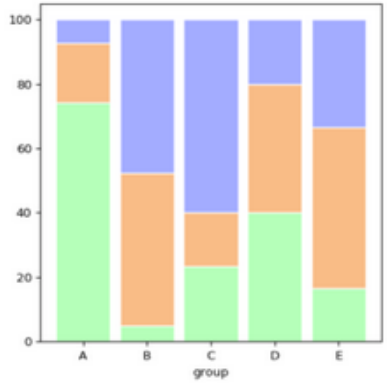
Stacked Bar-Plot Examples (MatPlotLib)



Stacked Barchart



Grouped Bar Plot

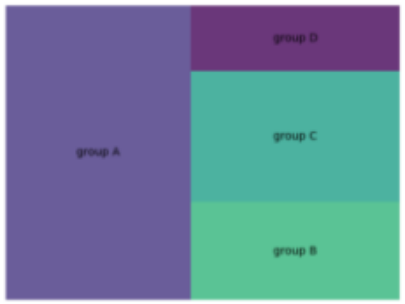


Percent Stacked Bar Chart

Tree Plot

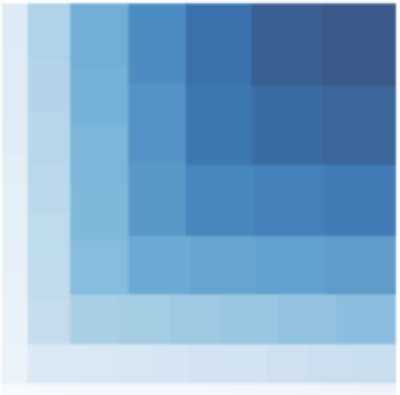
- Treemaps display hierarchical data as a set of nested rectangles. Each group is represented by a rectangle, which area is proportional to its value.
- Using color schemes, it is possible to represent several dimensions: groups, subgroups…
- Treemaps have the advantage to make efficient use of space, what makes them useful to represent a big amount of data. Using python, treemaps can be made through the squarify library. This library implements the tilling algorithm that keeps each rectangle as square as possible

Tree Plot Examples (Squarify)



Basic treemap

Custom treemap color



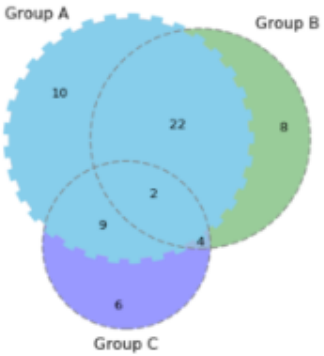
Treemap with color mapped to importance

Venn Diagram

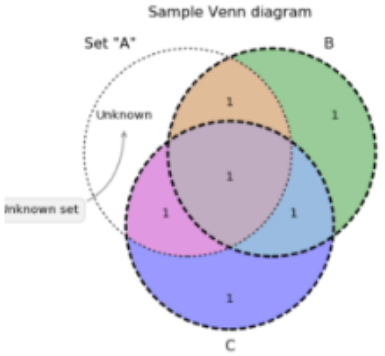
- A Venn Diagram (also called primary diagram, set diagram or logic diagram) is a diagram that shows all possible logical relations between a finite collection of different sets.
- Each set is represented by a circle. The circle size represents the importance of the group. The groups are usually overlapping: the size of the overlap represents the intersection between both groups.
- It is unadvised to make Venn Diagram with more than 3 groups because it would become hard to read. In python, Venn Diagram are realized using the venn2 and venn3 function of the MatPlotLib library according to the number of group you have. See here for module installation.

Venn Diagram Examples (MatPlotLib)

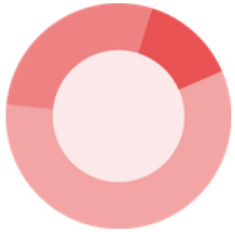
Venn Diagram with 3 groups



Custom a circle on Venn

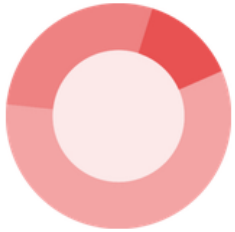


Elaborated Venn Diagram



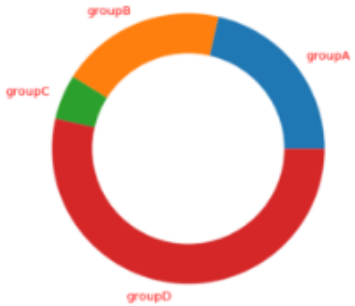
Donut Plot

- A Donut Chart is essentially a Pie Chart with an area of the center cut out.
- However, it is much more appreciated on a data viz point of view, as explained in data-to-viz.com.
- You can do it with python and the MatPlotLib library. Its construction relies on the use of the plt.Pie function. Then, add a circle at the center or play with the radius and width arguments.



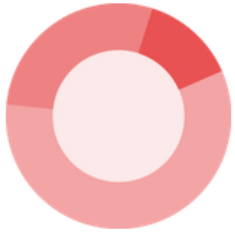
Donut Plot Examples (MatPlotLib)

Basic donut Plot

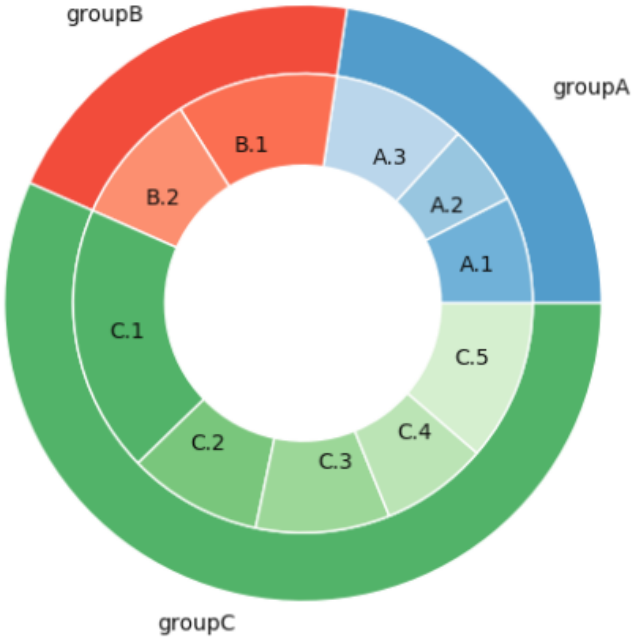


custom labels of donut Plot

Custom wedges of donut Plot



Donut Plot Examples (MatPlotLib)

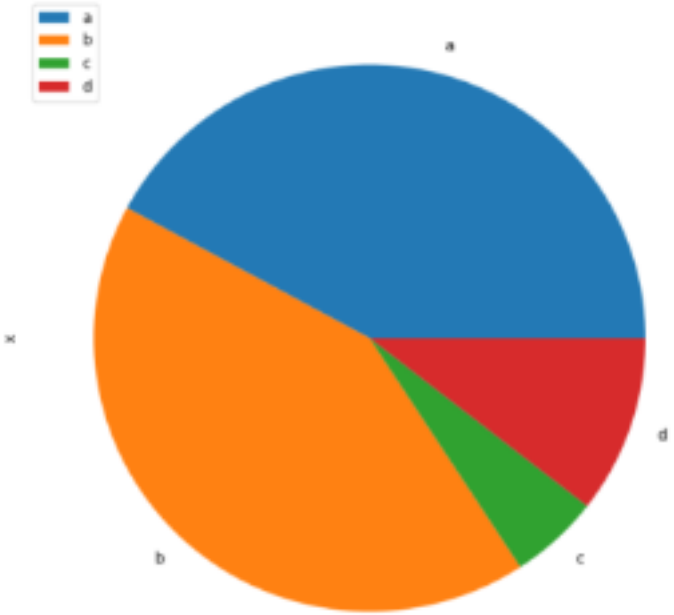


Donut with subgroups

Pie Plot

- Pie Chart is probably one of the most common type of chart. It is a circular graphic which is divided into slices to illustrate numerical proportion.
- The point of a Pie Chart is to show the relationship of parts out of a whole.
- You have to keep in mind that Pie Chart is easily the worst way to convey information ever developed in the history of data visualization. Thus, it must be avoided and replaced with Bar Plot most of the time. Indeed, it is really difficult for a human to make the difference between the size of similar slices.
- 3D Pie Plot is even worse since it distorts reality.

Pie Plot Examples (Pandas)

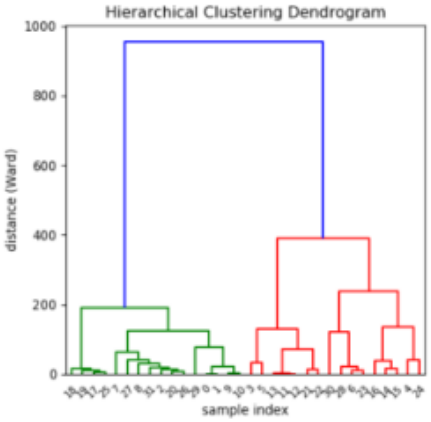


Pandas piePlot

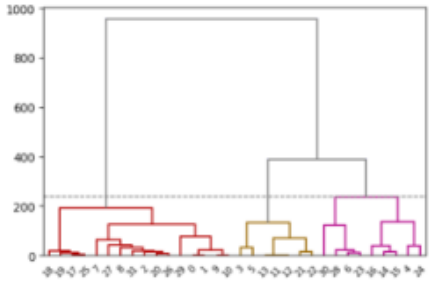
Tree Plot

- A Dendrogram or tree diagram allows to illustrate the hierarchical organization of several entities. For example, we often use it to make family trees. It is constituted of a root node, which give birth to several nodes that ends by giving leaf nodes (the bottom of the tree).
- Dendrogram can be made with 2 types of dataset.
  - a numeric matrix where several variables describe the features of individuals. We can then calculate the distance between individuals and clustering them.
  - A hierarchical dataset where the relationship between entities is provided directly.

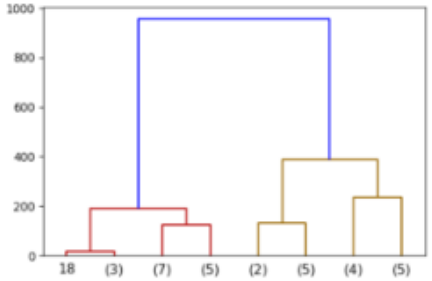
Tree Plot Examples (Seaborn)



Dendrogram Basics

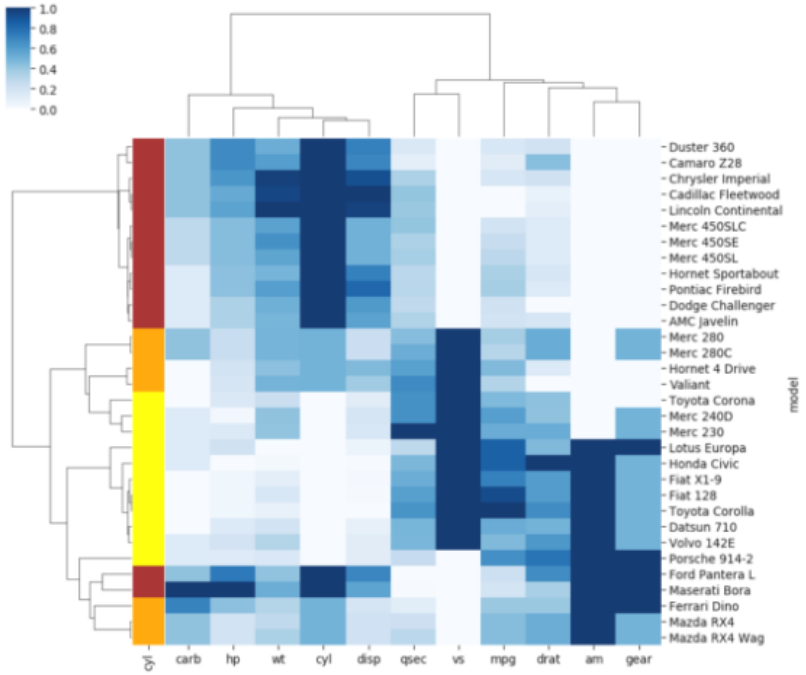


Dendrogram color



Truncated dendrogram

Tree Plot Examples (Seaborn)



Dendrogram with Heat Map and colored leaves

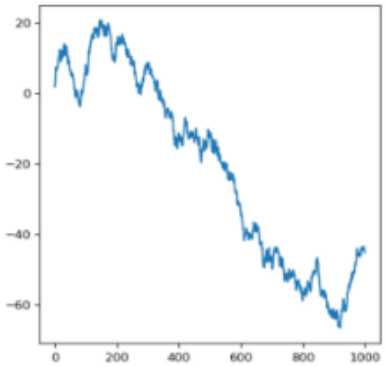
Evolution Plots

Descriptions and Examples

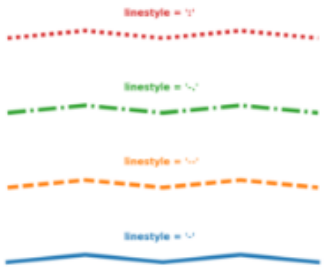
Line Plot

- A Line Chart or line graph is a type of chart which displays information as a series of data points called ‘markers’ connected by straight line segments.
- It is a basic type of chart common in many fields.
- It is similar to a scatter Plot except that the measurement points are ordered (typically by their x-axis value) and joined with straight line segments.
- A Line Chart is often used to visualize a trend in data over intervals of time – a time series – thus the line is often drawn chronologically. In these cases they are known as run charts.

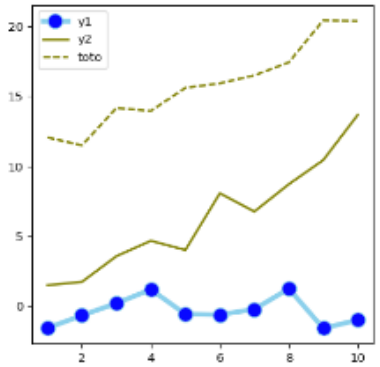
Line Plot Examples (MatPlotLib)



Basic Line Chart

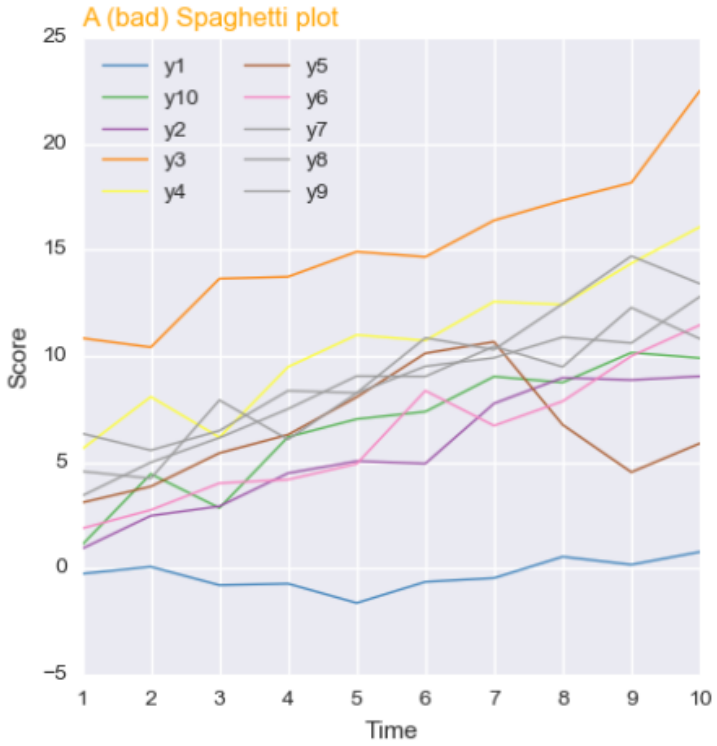


Customize Line Chart



Line Chart with several Lines

Line Plot Examples (MatPlotLib)

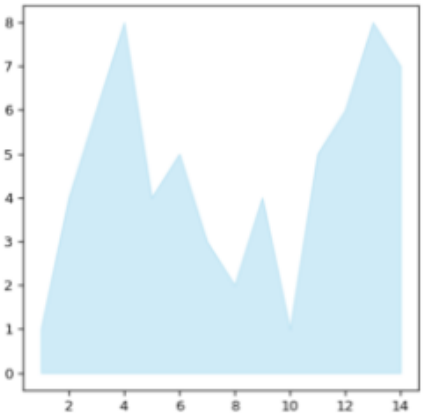


Spaghetti Plot

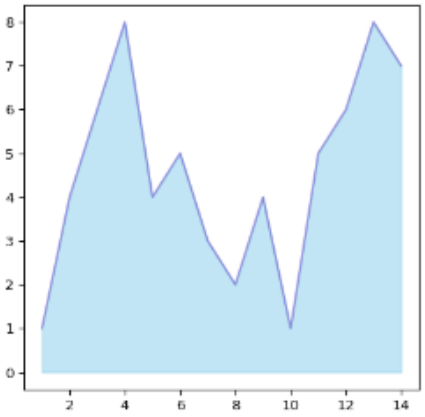
Area Plot

- An Area Chart is really similar to a Line Chart, except that the area between the x axis and the line is filled in with color or shading.
- It represents the evolution of a numerical variable following another numerical variable. If you want to represent this evolution for several groups in the same time, you are probably interested by Stacked Area Chart, where every groups are displayed one of top of each other.
- Note that an interesting alternative is probably to use faceting. In python, Area Chart can be done using the fillbetween function of MatPlotLib. The stackPlot function could work as well, but it is more adapted for Stacked Area Charts.

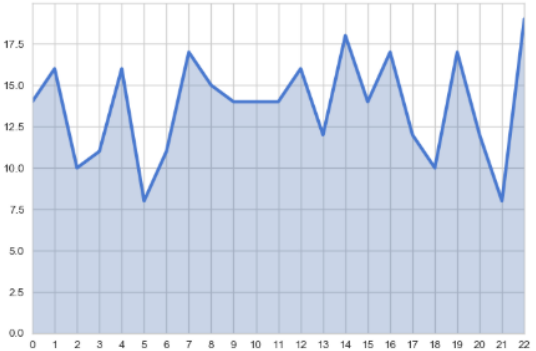
Area Plot Examples (MatPlotLib)



Basic Area Chart



Add line to Area Chart

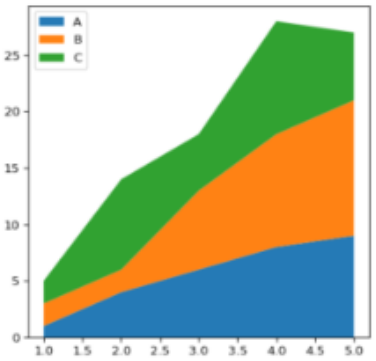


Area Chart with white grid

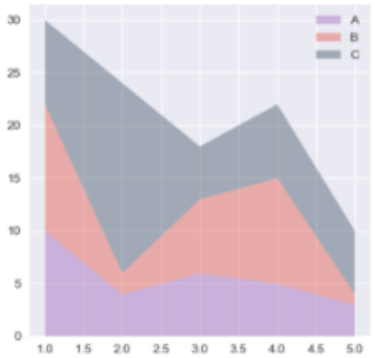
Stacked Area Plot

- A Stacked Area Chart is the extension of a basic Area Chart to display the evolution of the value of several groups on the same graphic. The values of each group are displayed on top of each other.
- It allows to check on the same figure the evolution of both the total of a numeric variable, and the importance of each group. If only the relative importance of each group interests you, you should probably draw a percent Stacked Area Chart.
- Note that this chart becomes hard to read if too many groups are displayed and if the patterns are really different between groups. In this case, think about using faceting instead.

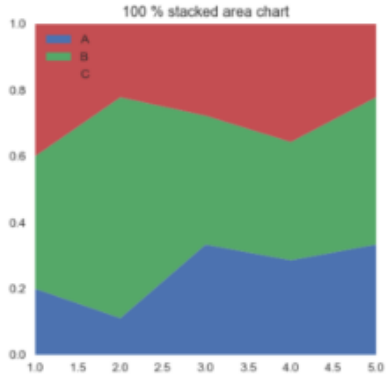
Stacked Area Plot Examples (MatPlotLib)



Basic Stacked Area Chart

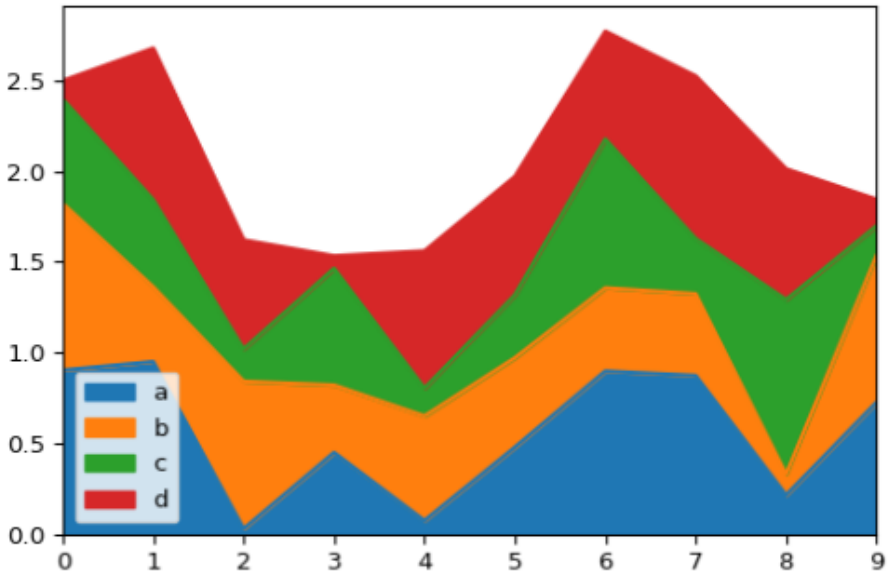


Play with baseline



Percent Stacked Area Chart

Stacked Area Plot Examples (MatPlotLib)

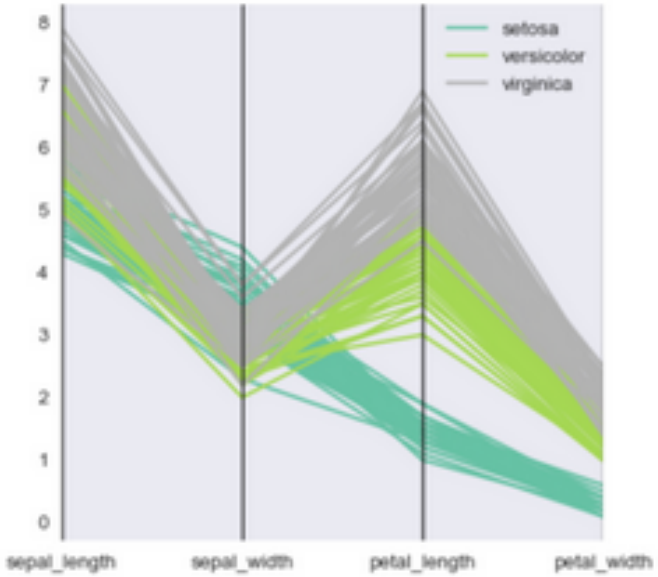


Pandas Stacked Area Chart

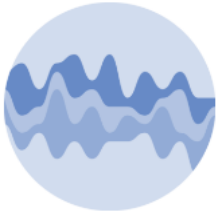
Parallel Plot

- Parallel Plot or Parallel Coordinates Plots allow to compare the feature of several individual observations on a set of numerical variables. Each vertical bar represents a variable and usually has its own scale. (The units can even be different!). Values are then Plotted as series of lines connected across each axis.
- Playing with color can then be used to represent different groups of individual, or highlight a specific one. That should allow to detect interesting patterns!

Parallel Plot Examples (Panda)

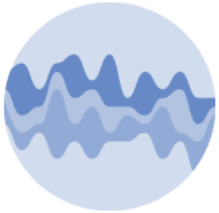


Parallel Plot with Pandas

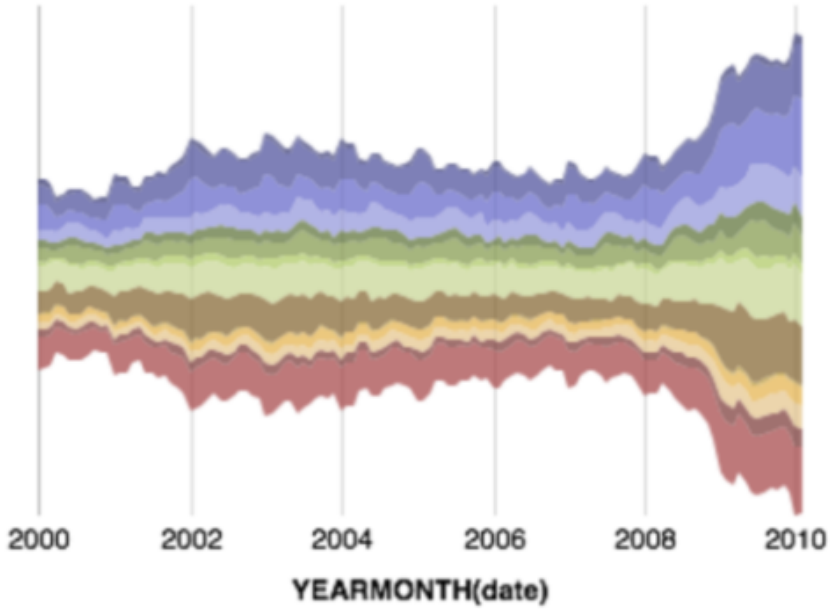


Stream Chart

- A Stream Graph is really close from a Stacked Area Chart. It displays the evolution of a numerical value (Y axis) following another numerical value (X axis). This evolution is represented for several groups, all with a distinct color.
- Contrary to a Stacked Area, there is no corner: edges are rounded what gives this nice impression of flow. Stream chart get really useful when displayed in an interactive mode: highlighting a group gives you directly an insight of its evolution.



Stream Chart Examples



Stream Chart with Altair

Maps Plots

Descriptions and Examples

Map Plot

- This page is dedicated to the creation of background map with python. Three other sections exist for chloropleth, connection and bubble map. It displays many example using different projections, colors, boundaries and geographical positions.
- To make a map, you need the information of the shape of your countries, regions or whatever zone.
- Basemap allows to make high quality static maps, and GmPlot allows to make google map style interactive maps. A lot of tools currently exist in python, but I hope you will be able to fit most of your needs with these 2 tools.

Map Plot Examples (Folium)

Most basic background map

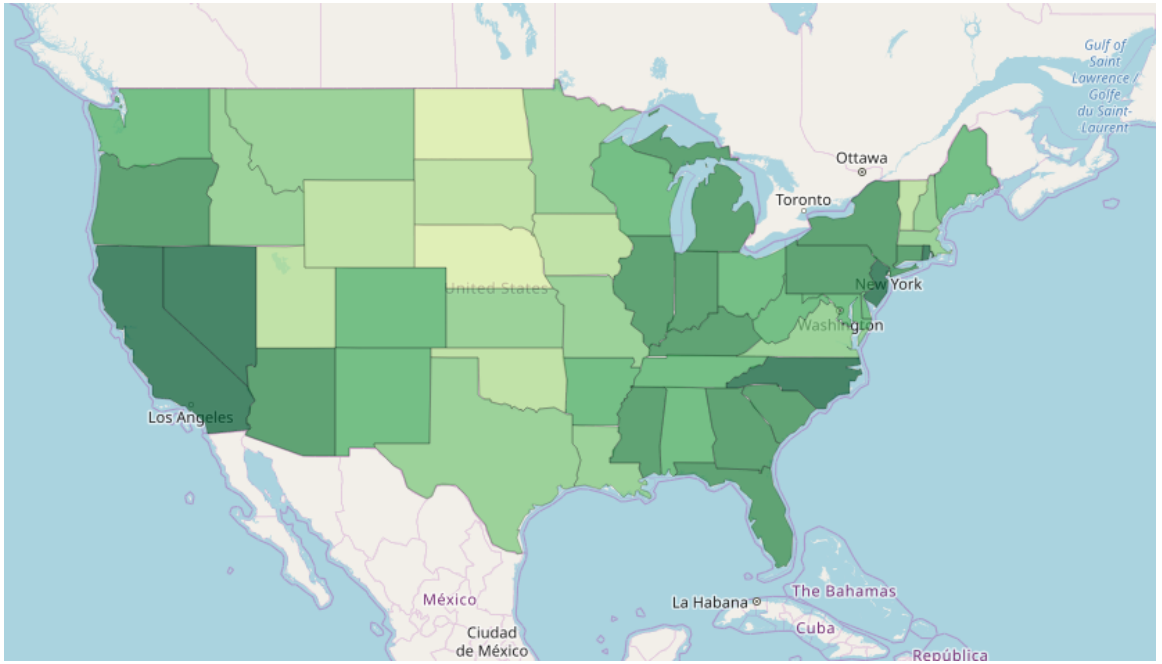
Set bounding

Load a background

Choropleth Map Plot

- A choropleth map displays divided geographical areas or regions that are colored, shaded or patterned in relation to a data variable.
- It allows to study how a variable evaluate along a territory.
- Note that several pitfalls are commonly observed in choropleth map. The number of data point in each area remains unknown. If the variable you represent is linked to the area, you have to normalize your variable first.

Choropleth Map Plot Examples (Folium)

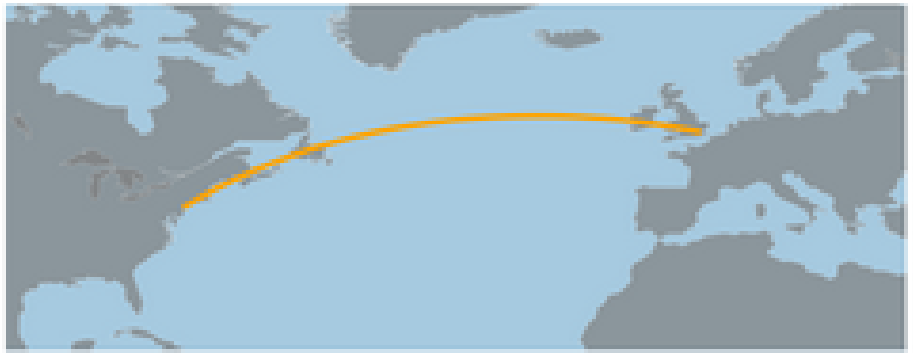


Basic choropleth map with python

Connection Map Plot

- A connection map allows to show the connection between several positions on a map.
- The link between 2 places can be drawn with a straight line, or more commonly by representing the ‘great circle‘: the shortest route between them.
- Knowing that the earth is a sphere, this results in rounded lines that give a really pleasant look to the map. Python is an awesome tool to draw such a map due to the drawgreatcircle function of the basemap library.

Connection Map Plot Examples (Basemap)



A connection line

Bubble Map Plot

- This section is dedicated to maps on which we add markers. These markers can be bubble with a size relative to a numeric value: in this case we call it a bubble map.
- It is relatively easy to add markers on your map with the basemap library.
- Once your map background is drawn, you can simply use the usual MatPlotLib functions to add a Plot in top of it.
- Markers can be customized in term of shape, size and color. Even more info can be displayed in a interactive map where people can click markers.

Bubble Map Plot Examples (Folium)

Add markers on your map

Flow Plots

Descriptions and Examples

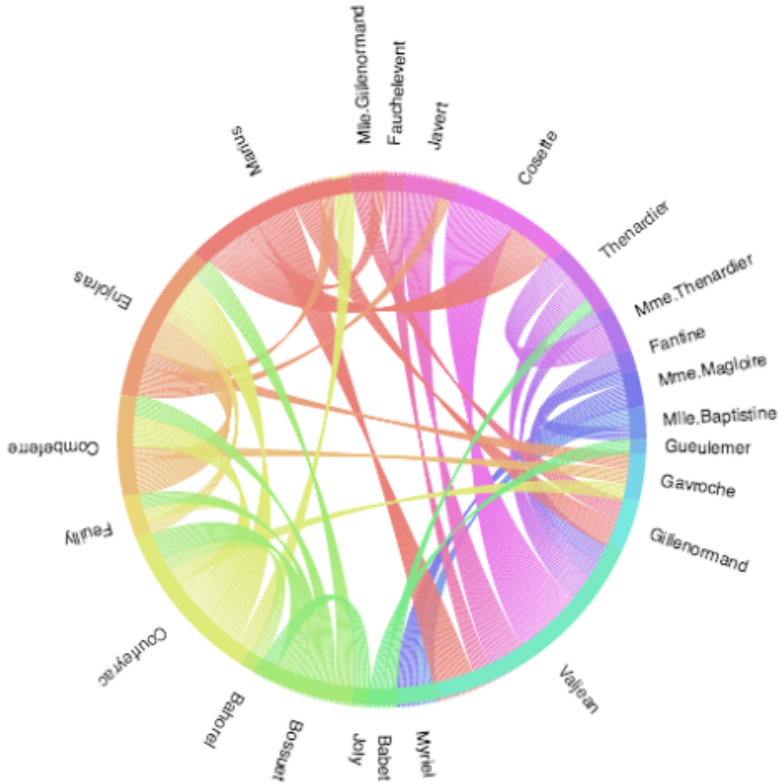
Chord Diagram

Descriptions and Examples

Chord Diagram

- Chord diagrams allow to visualize flows between several entities. Each entity is represented by a fragment on the outside of the circle. Then, arcs are drawn between each entities.
- The size of the arc is proportional to the importance of the flow.
- Unfortunately, there is currently no specific library allowing to make proper chord diagram in python. It is possible to make it using Plotly, but it takes a lot effort. Bokeh is another option, but the result is not entirely satisfying. It is hard not to propose you to use R and the awesome circlize package!

Chord Diagram Examples

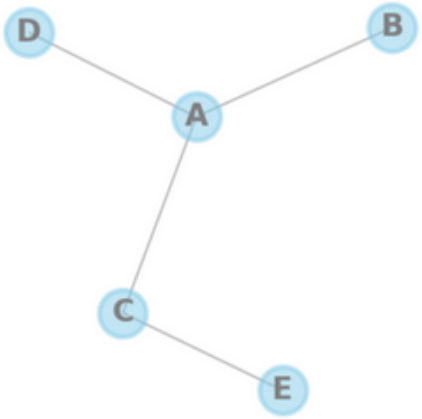


Chord diagram with Bokeh

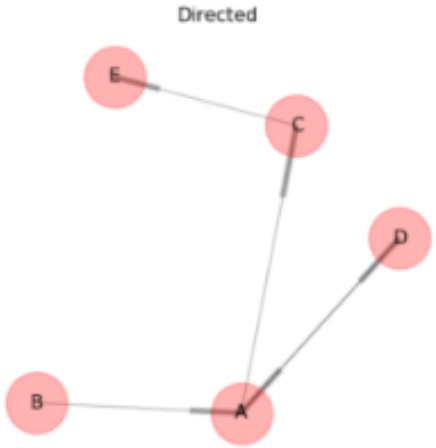
Network Chart

- Network diagrams (or chart, or graph) show interconnections between a set of entities. Each entity is represented by a Node (or vertices).
- Connection between nodes are represented through links (or edges). The theory and realization of network is a large field of research in itself.

Network Chart Examples (NetworkX)



Custom network look

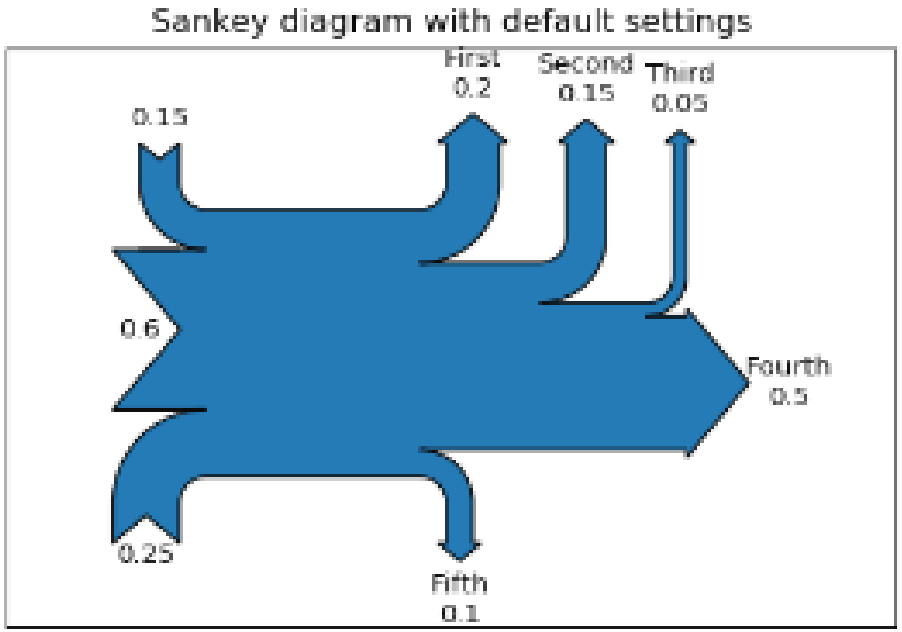


Directed or UN-directed network

Sankey Diagram

- A Sankey Diagram is a visualization technique that allows to display flows. Several entities are represented by rectangles or text, and linked with arrow that have a width proportional to the importance of the flow.
- The use of colors is recommended to make the distinction between groups easier.
- In python, the MatPlotLib library proposes a specific module allowing to realize basic Sankey diagrams.

Sankey Diagram Examples (MatPlotLib)



Sankey with MatPlotLib

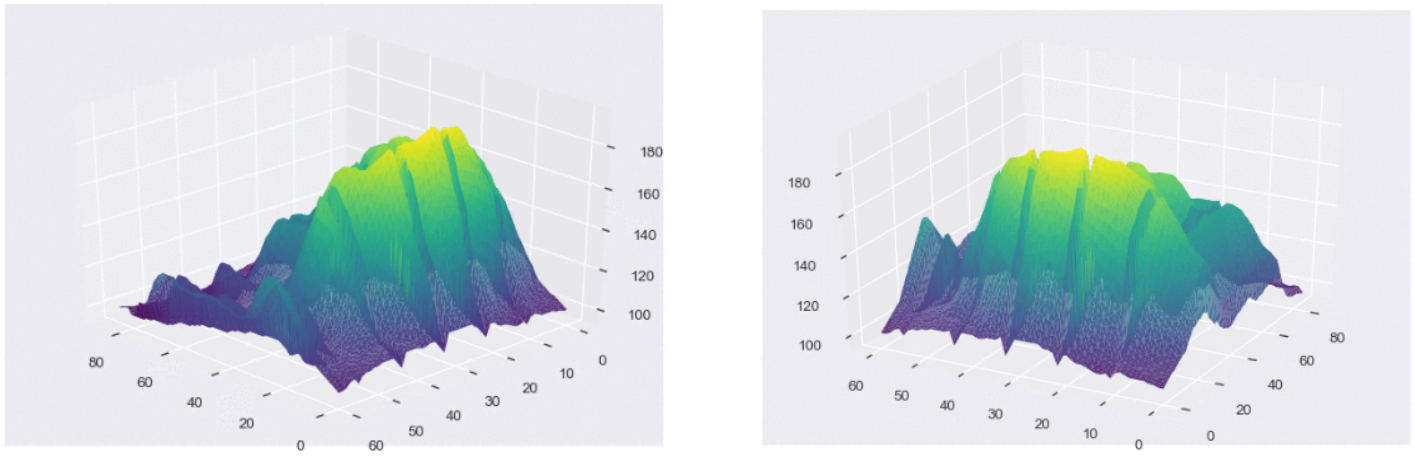
Some Other Visualization Types

Descriptions and Examples

Animation

- An animated graphic is a graphic that moves, often in a GIF or movie format. It can be really useful to describe the evolution of something, or to denote the difference between two states (going from one to another and backward).

Animated Diagrams Example (Image Magick)

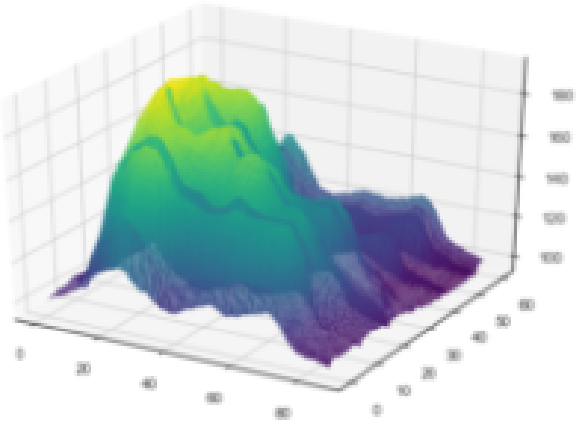


Animated Plot

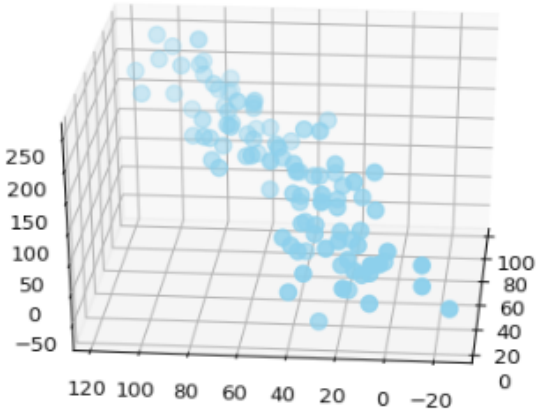
3D Plot

- Python allows to realize 3D graphics thanks to the mPlot3d toolkit of the MatPlotLib library.
- However, be really careful with the use of 3D Plots. Even if it can produce some nice looking results, it is often not the most understandable way to represent data. Moreover, the quality of the 3D chart made with python are currently limited.
- 3D graphics can be good for Scatter Plot where you have 3 numerical values, and especially if you want to show a difference between groups (like for PCA). Another good utilization is for surface Plots.

3D Plot Example



3D surface Plot



3D Scatter Plot