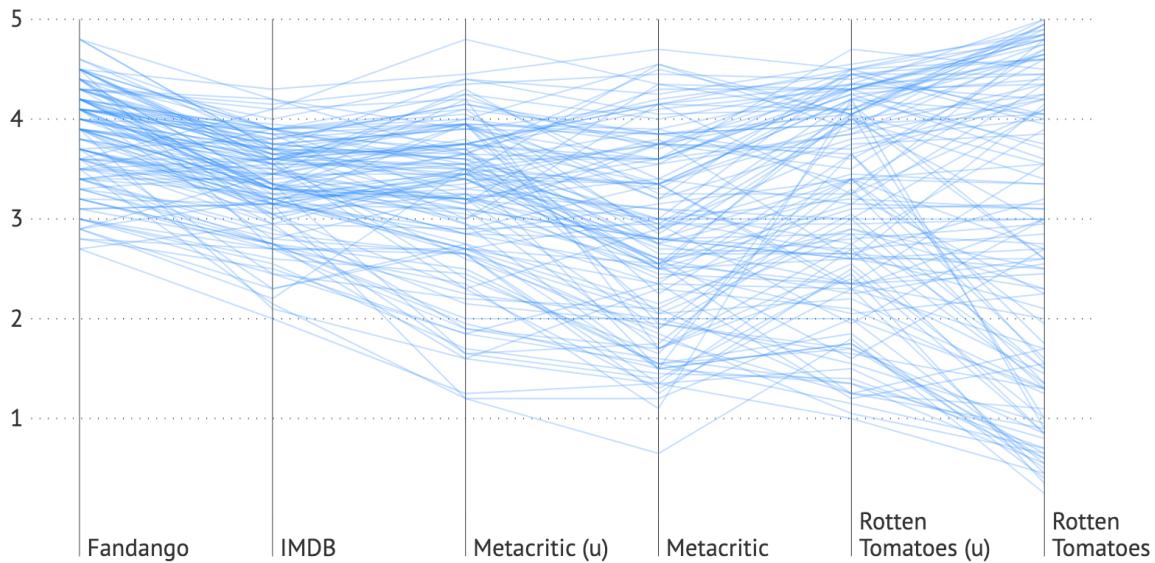


## Station A – Parallel Coordinates

Alternative names: *Parallel plot, Parallel coordinates plot*

**Fadango movie ratings are noticeably more positive than on other sites.**

Based on ratings of movies released in 2015



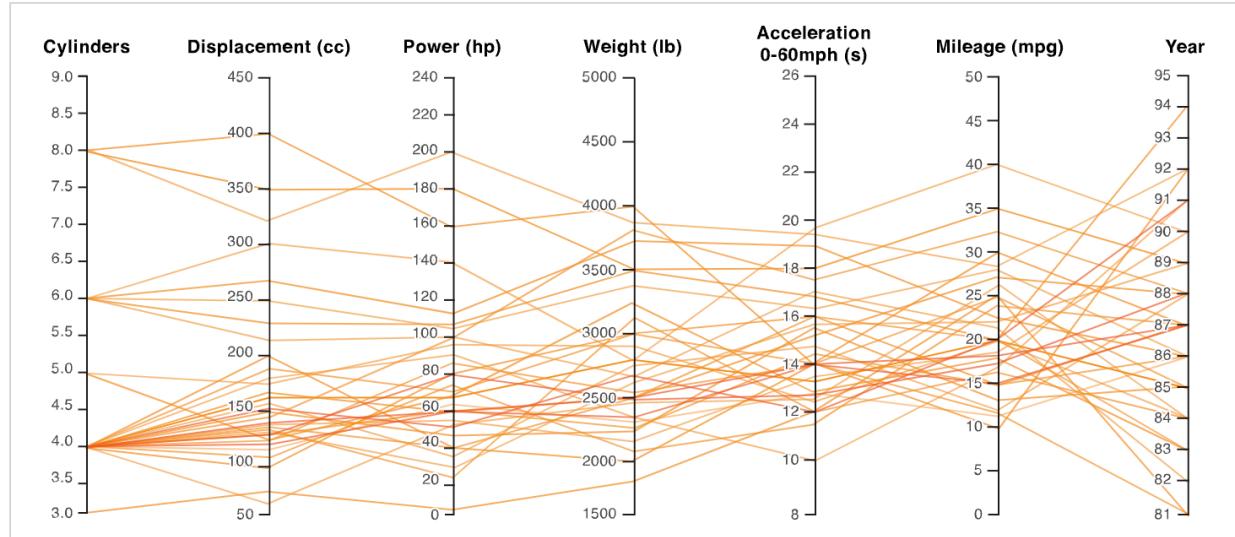
Based on FiveThirtyEight story and data: <https://fivethirtyeight.com/features/fandango-movies-ratings/>

Made with [Datylon](#) - [Edit](#)

The parallel coordinates chart resembles a line chart, but instead of time values, categories are plotted on the horizontal axis. It allows you to plot a multitude of categories/dimensions without compromising the readability in a simple 2d space - all of the dimensions follow the same pattern. A dimension can have both a separate axis or just one of the gridlines if all the dimensions share the same data range. The simplicity of the chart, however, adds some limitations. Maximum two neighboring dimensions relationships can be followed at a time, so the ordering plays a crucial role in this chart.

[source: <https://www.datylon.com/blog/types-of-charts-graphs-examples-data-visualization>]

# Parallel Coordinates Plot



## Description

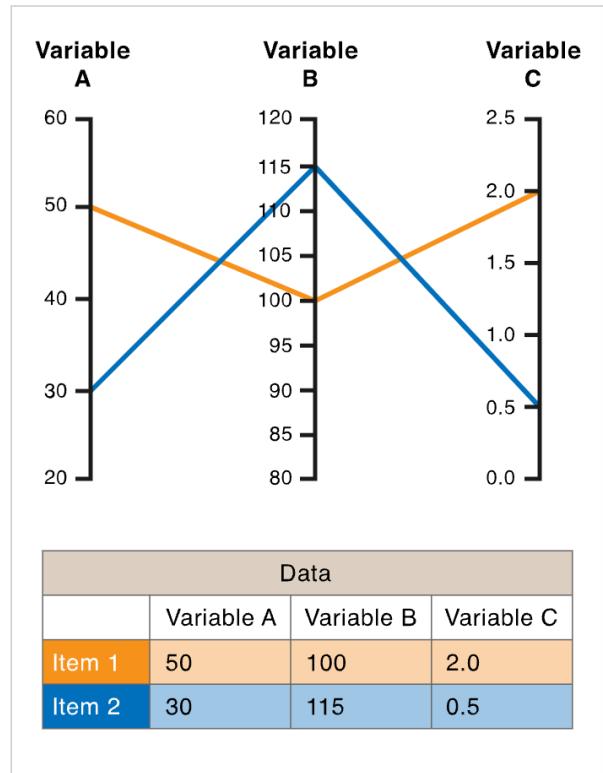
This type of visualisation is used for plotting multivariate, numerical data. Parallel Coordinates Plots are ideal for comparing many variables together and seeing the relationships between them. For example, if you had to compare an array of products with the same attributes (comparing computer or car specs across different models).

In a Parallel Coordinates Plot, each variable is given an axis and all the axes are placed parallel to each other. Each axis can have a different scale, as each variable works off a different unit of measurement, or all the axes can be normalised to keep all the scales uniform. Values are plotted as a series of lines that are connected across all the axes. This means that each line is a collection of points placed on each axis, that have all been connected.

The order the axes are arranged can impact the way how the reader understands the data. One reason for this is that the relationships between adjacent variables are easier to perceive than for non-adjacent variables. So re-ordering the axes can help in discovering patterns or correlations across variables.

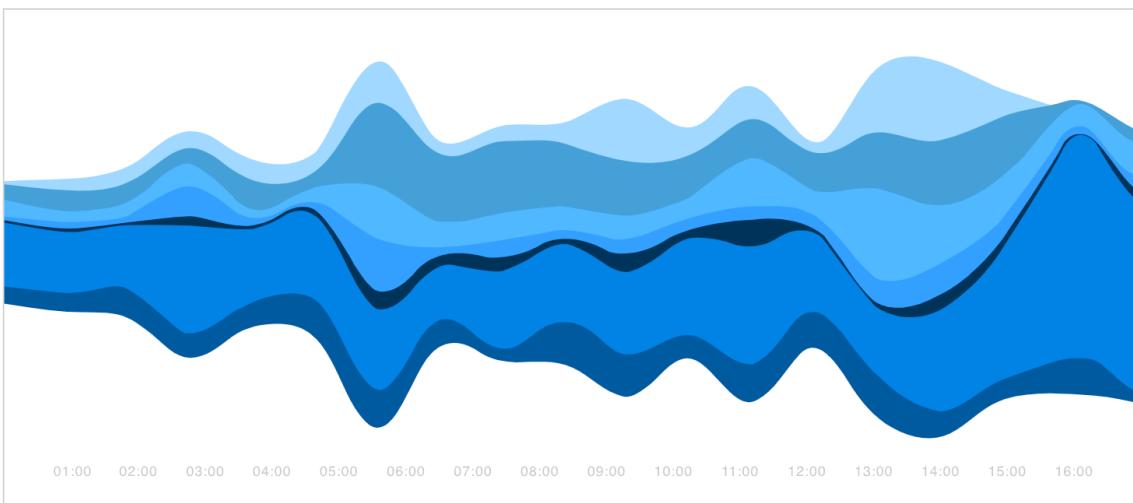
The downside to Parallel Coordinates Plots, is that they can become over-cluttered and therefore, illegible when they're very data-dense. The best way to remedy this problem is through interactivity and a technique known as "brushing". Brushing highlights a selected line or collection of lines while fading out all the others. This allows you to isolate sections of the plot you're interested in while filtering out the noise.

## Anatomy



[source: <https://datavizcatalogue.com/>]

## Station B – Stream Graph



### Description

Also known as a *ThemeRiver*.

This visualisation is a variation of a [Stacked Area Graph](#), but instead of plotting values against a fixed, straight axis, a Stream Graph has values displaced around a varying central baseline. Stream Graphs display the changes in data over time of different categories through flowing, organic shapes that somewhat resemble a river-like stream. This makes Stream Graphs aesthetically pleasing and more engaging.

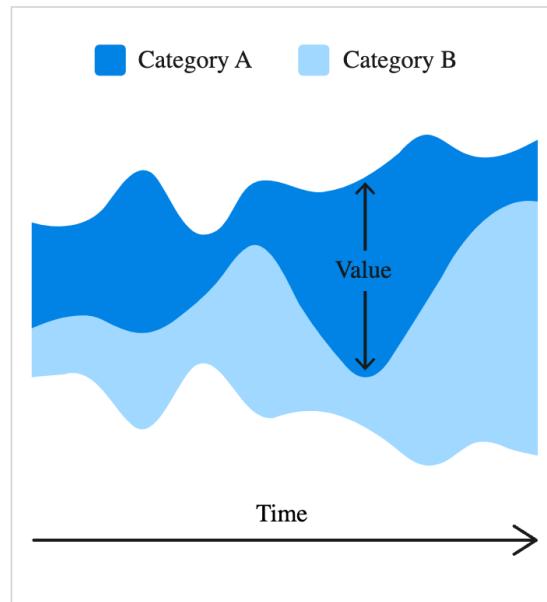
In a Stream Graph, the size of each stream shape is proportional to the values in each category. The axis that a Stream Graph flows parallel to, is used for the timescale. Colour can be used to either distinguish each category or to visualise an additional quantitative variable by varying the colour shade.

Stream Graphs are ideal for displaying high-volume datasets to discover trends and patterns changing over time across a range of different categories. For example, seasonal peaks and troughs in the stream shape can suggest a periodic pattern. A Stream Graph could also be used to visualise the volatility over time for a large group of assets.

The downside to Stream Graphs is that they suffer from legibility issues, as they are often very cluttered with large datasets. The categories with smaller values are often drowned out to make way for categories with much larger values, making it impossible to see all the data. Also, it's impossible to read the exact values visualised in a Stream Graph, as there is no value axis to use as a reference.

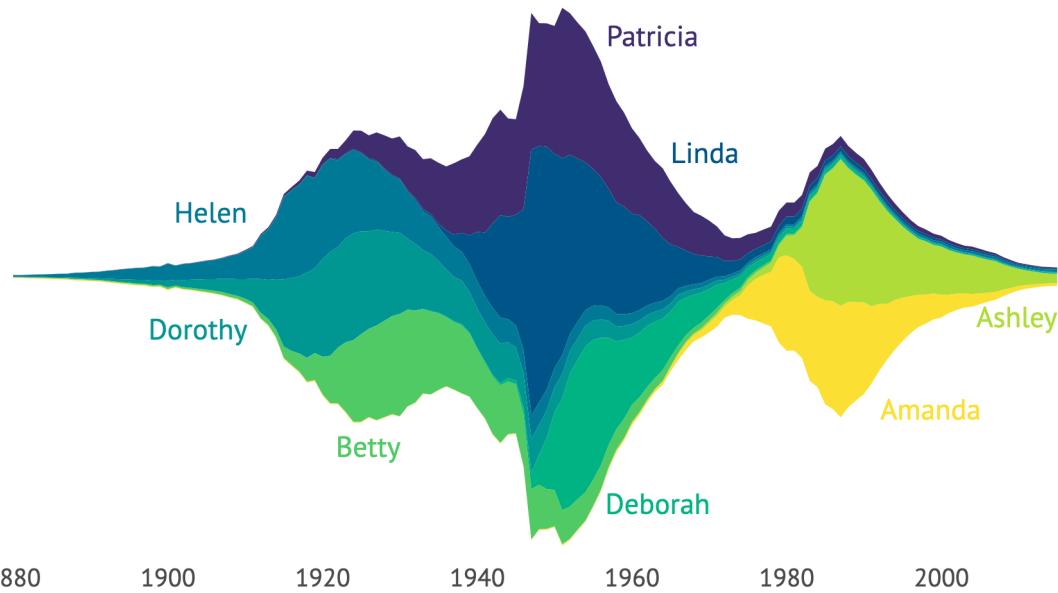
Therefore, Stream Graphs should be reserved for audiences who don't intend to spend much time deciphering the graph and exploring its data. Stream Graphs are better for giving a more general view of the data. They also tend to work significantly better as interactive pieces rather than static or printed graphics.

### Anatomy



[source: <https://datavizcatalogue.com/>]

## Evolution of baby names in US

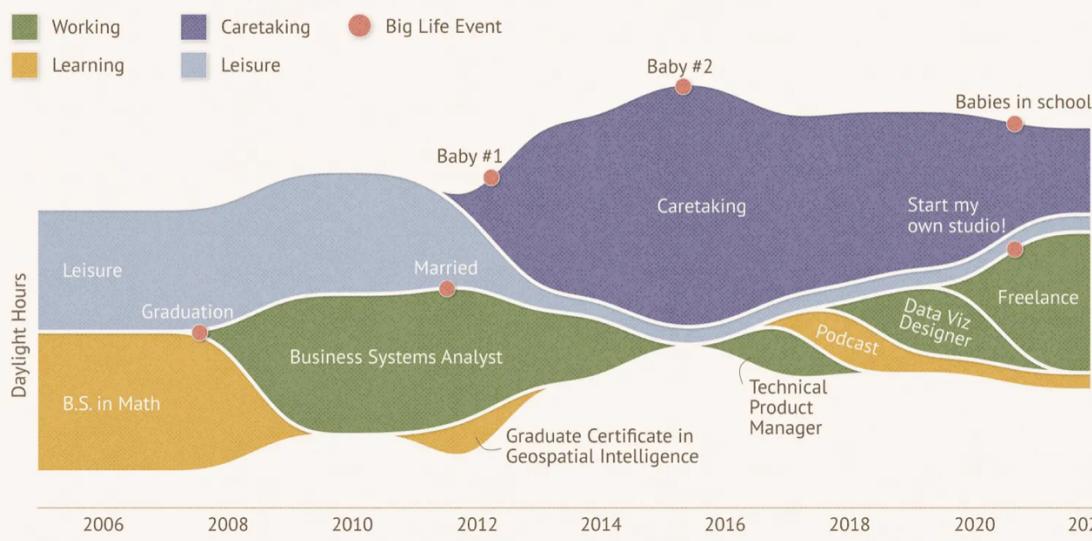


Made with [Datylon](#) - Edit

A stream graph is undoubtedly one of the most beautiful chart types available. This stunning type of chart derives from a stacked area chart, from which it differs by using a central baseline rather than a fixed axis. A stream graph then visualizes different values (compound volumes) around the baseline. This creates a visualization that resembles a river-like stream. The shape of the stream, which consists of peaks and troughs referencing different values over time, can also indicate seasonal patterns. See more similar chart examples [on our inspiration page](#).

[source: <https://www.datylon.com/blog/types-of-charts-graphs-examples-data-visualization>]

## How I've Spent My Time



[source: <https://datavizproject.com/>]

# Station C – Treemap



## Description

Treemaps are an alternative way of visualising the hierarchical structure of a Tree Diagram while also displaying quantities for each category via area size. Each category is assigned a rectangle area with the subcategory rectangles nested inside.

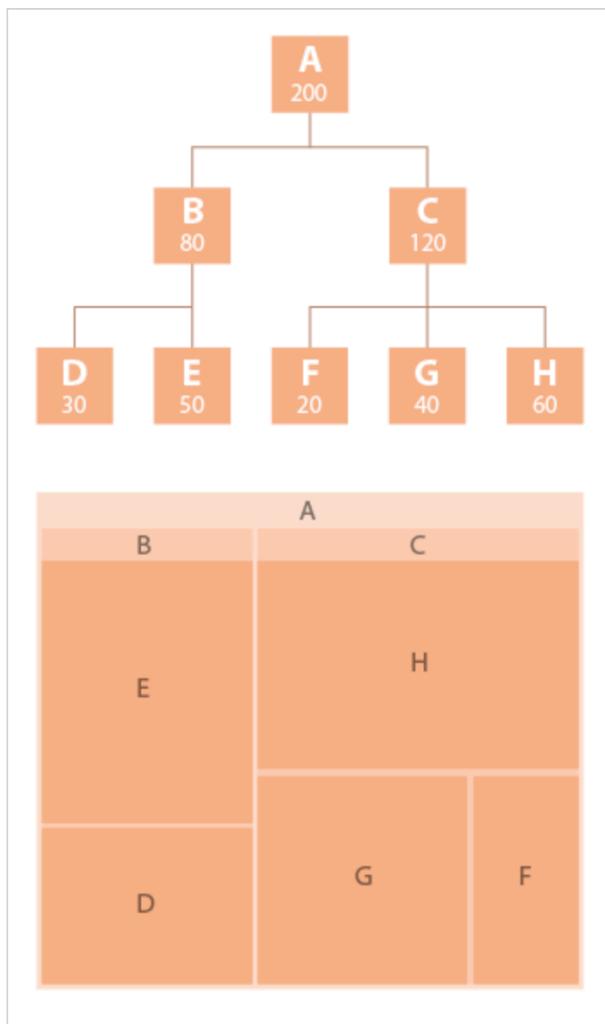
When a quantity is assigned to a category, its area size is in proportion to that quantity and any other quantities within the same parent category in a part-to-whole relationship. Also, the area size of the parent category is the total of its subcategories. If no quantity has been assigned to a subcategory, then its area is divided equally amongst the other subcategories within the parent category.

The way rectangles are divided and ordered into subrectangles depends on the tiling algorithm used. Many tiling algorithms have been developed, but the "squarified algorithm", which keeps each rectangle as square-like as possible is the one commonly used.

Ben Shneiderman originally developed Treemaps as a way of visualising a vast file directory on a computer, without taking up too much space on the screen. This makes Treemaps a more compact and space-efficient option for displaying hierarchies, that can give a quick overview of the hierarcal structure. Treemaps are also great at comparing the proportions between categories via their area size.

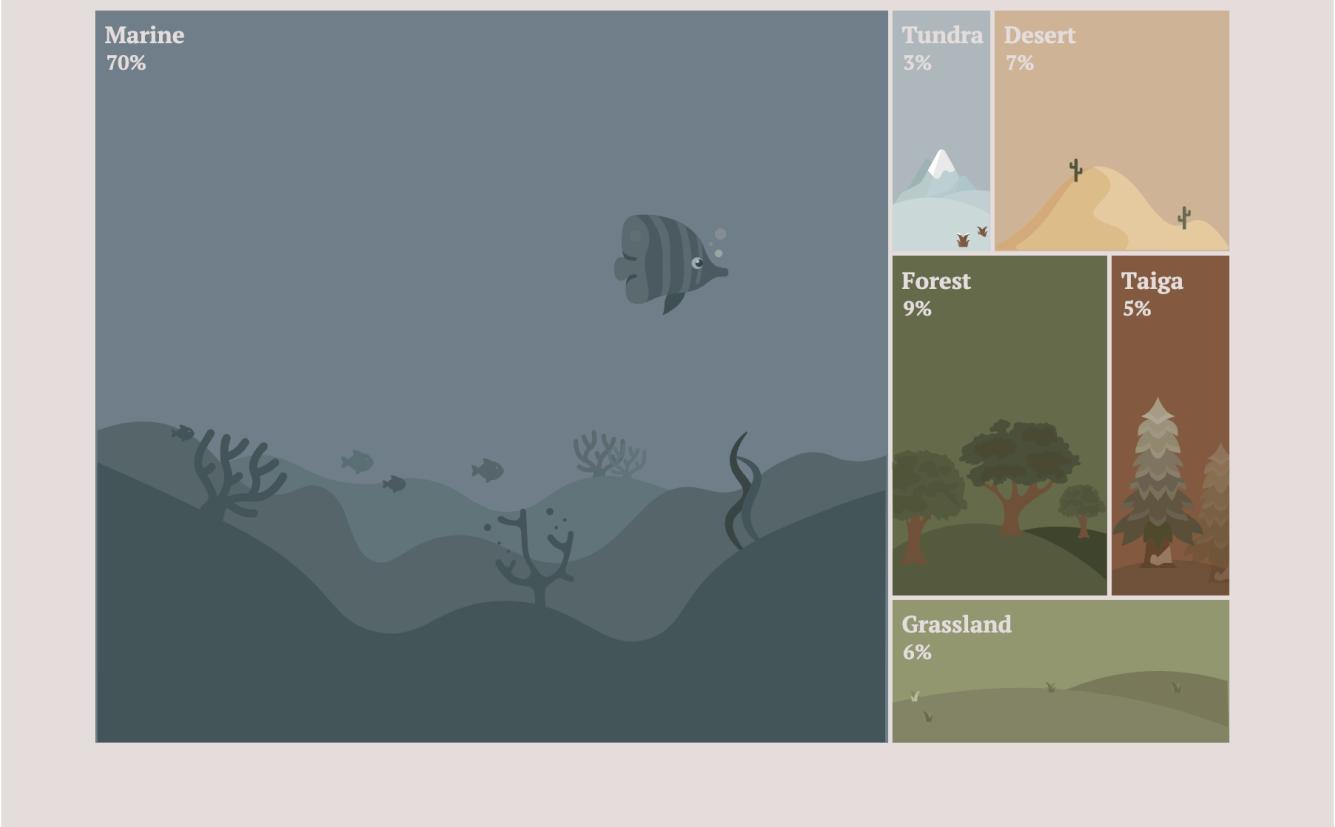
The downside to Treemaps is that they doesn't show the hierachal levels as clearly as other charts that visualise hierachal data (such as a Tree Diagram or Sunburst Diagram).

## Anatomy



[source: <https://datavizcatalogue.com/>]

# Distribution of the six biomes on Earth



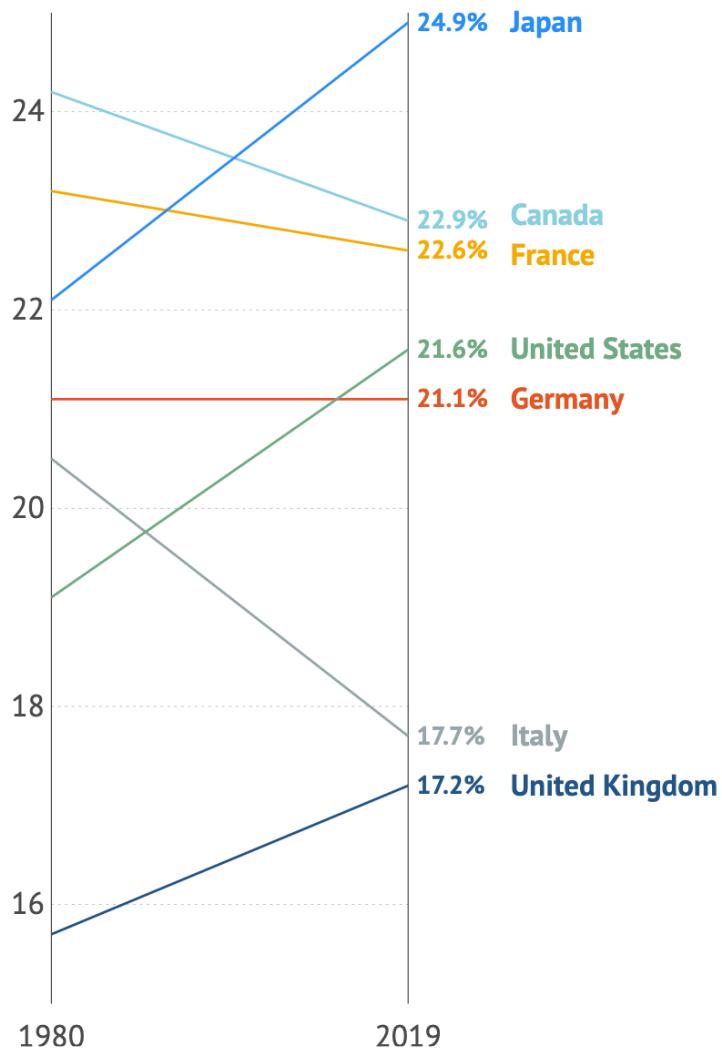
Made with [Datylon](#) - [Edit](#)

Treemap charts come in handy when you are dealing with large numbers of categories with a hierarchical structure. A treemap consists of multiple categories and each category in the treemap is given a rectangle. The categories could be subdivided into smaller rectangles if you are dealing with subcategories in the data. The size of the area of the rectangles communicates the value. Therefore, treemaps are very useful charts in finding relationships fastly, both within and between categories. Another benefit of a treemap is the efficient use of space which makes it easy to show a lot of data at the same time. If you're curious about the history and different features of a treemap chart, you can't miss [the deep dive article](#). We also have a very elaborate [treemap resource page](#) for you to check out before you start making your own treemap.

[source: <https://www.datylon.com/blog/types-of-charts-graphs-examples-data-visualization>]

## Station D – Slope Chart

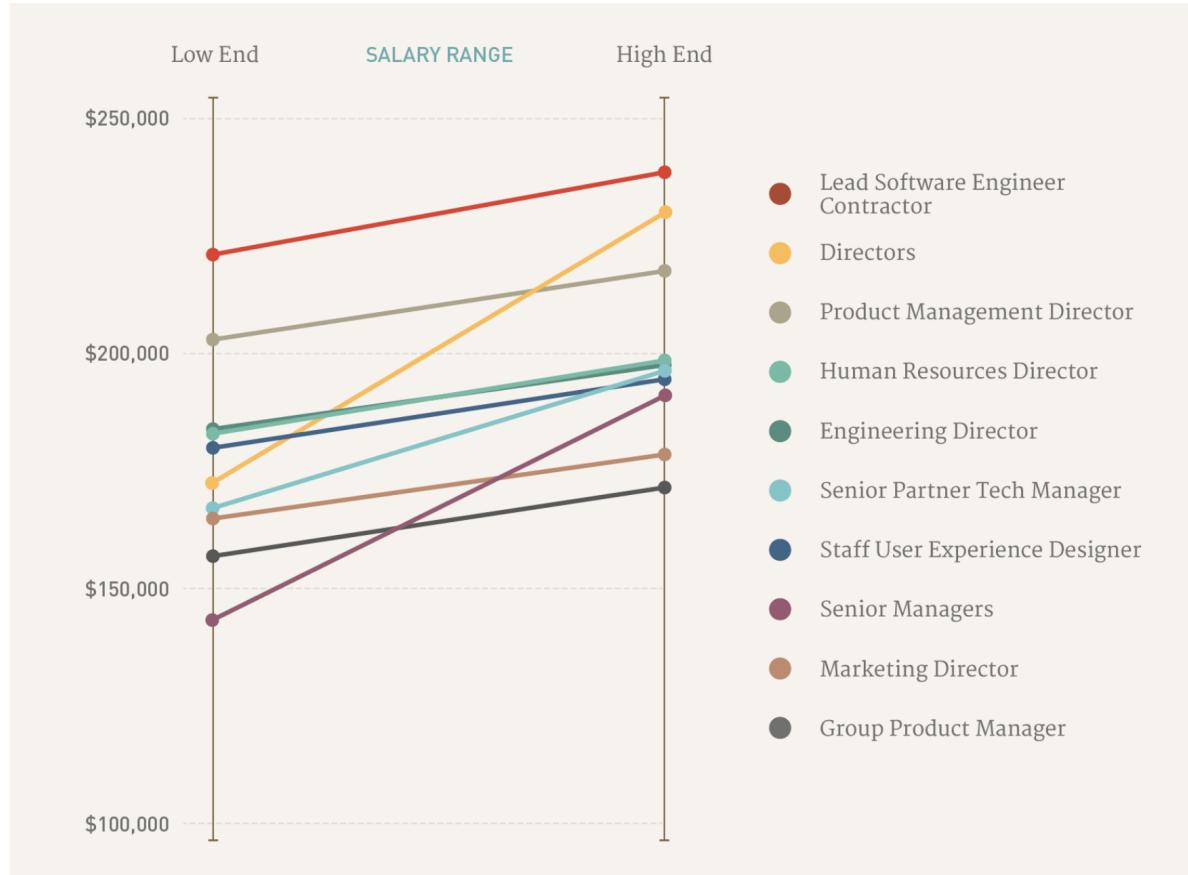
### Changes in total investment % of GDP from 2011 to 2019



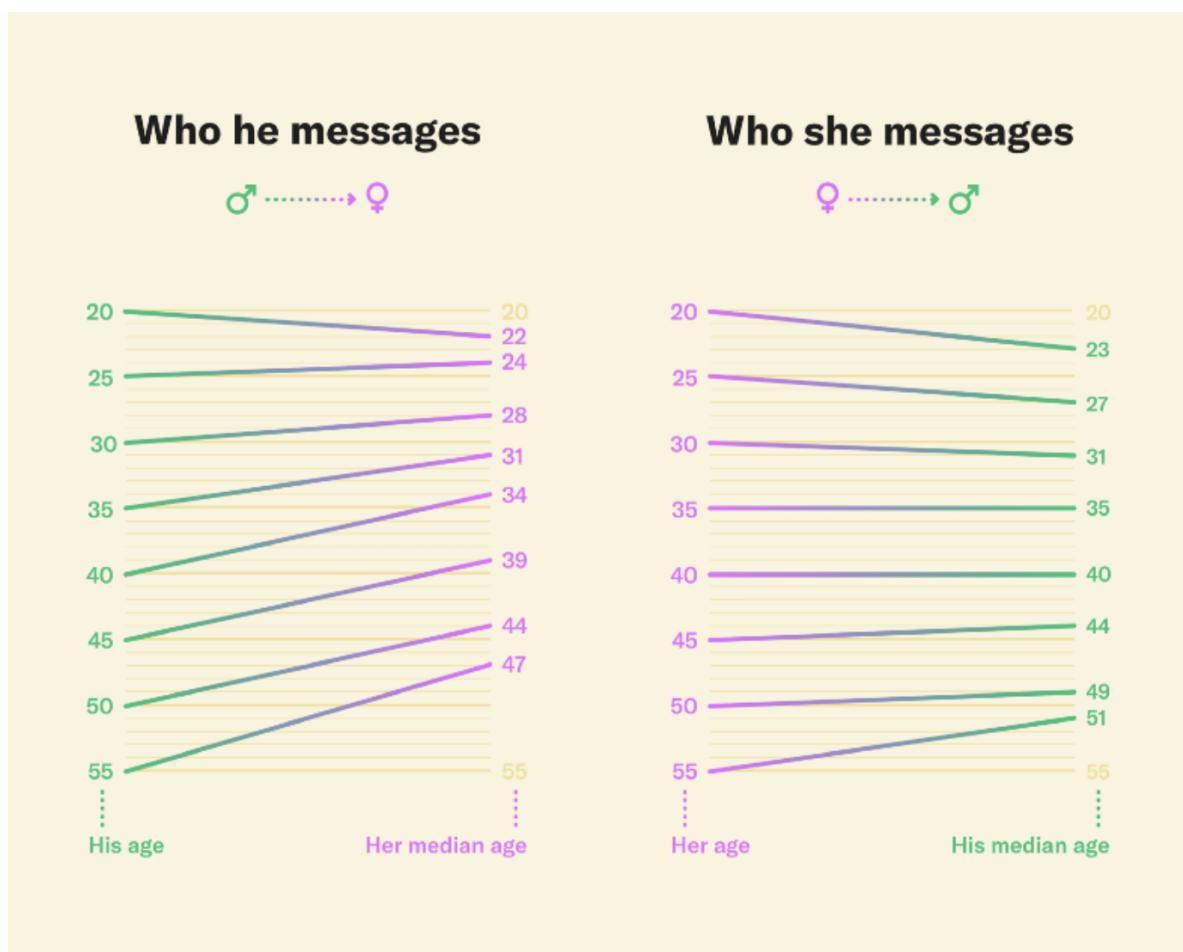
Made with [Datylon](#) - Edit

A slope chart is a chart that emphasizes the evolution between two values by using the angle of the slope to communicate the difference. It can be a change over time or a transition. A slope chart can be a good alternative for a line chart, grouped- or stacked bar chart, if we only have two points in time we want to address. See other slope chart examples [See other slope chart examples on inspiration page](#).

[source: <https://www.datylon.com/blog/types-of-charts-graphs-examples-data-visualization>]



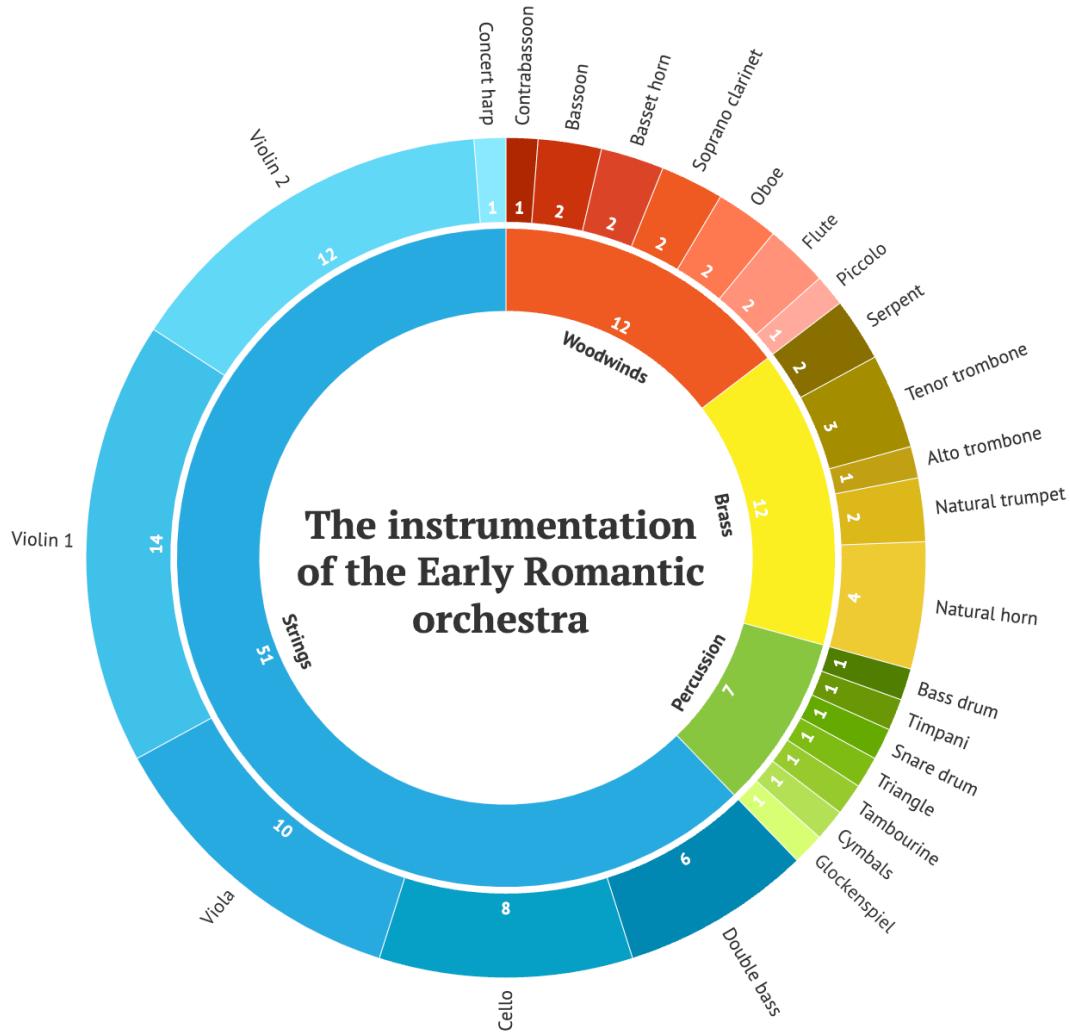
[source: <https://datavizproject.com/>]



[source: <https://datavizproject.com/>]

## Station E – Sunburst Chart

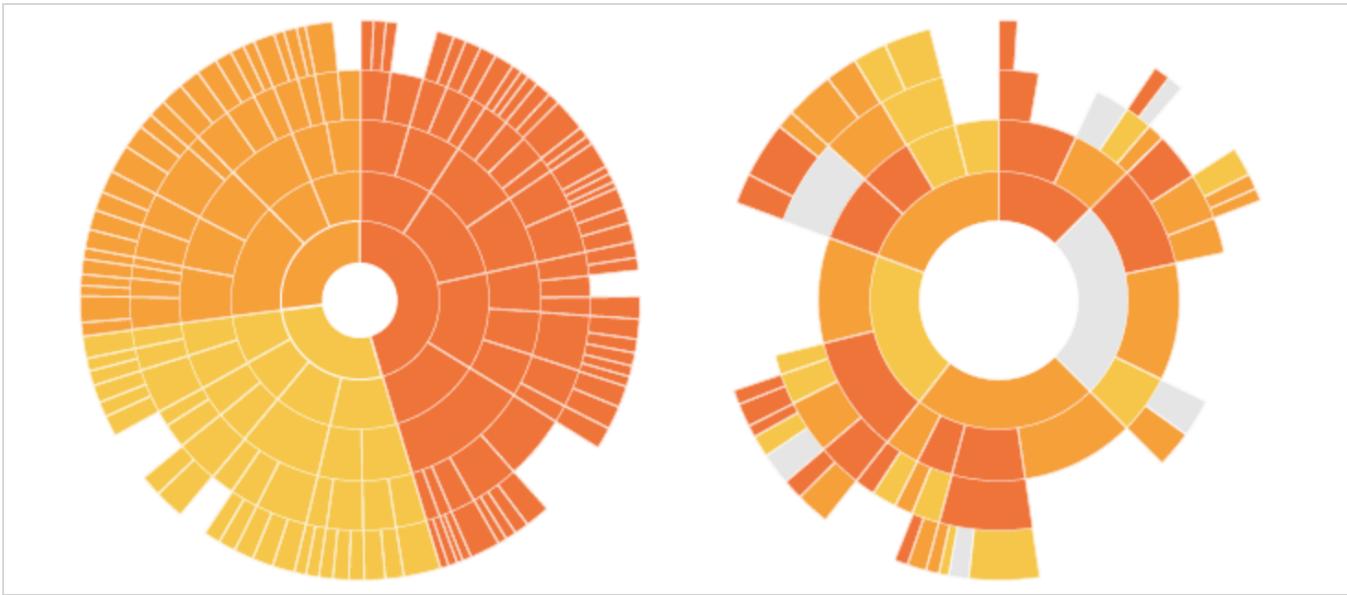
Alternative names: *Multi-level pie chart, Multilayer pie chart, Sunburst graph, Ring chart, Radial treemap*



Made with [Datylon](#) - Edit

A sunburst chart has many names but whatever you call it, it's still a spectacular type of graph. It shows a hierarchical dataset through a series of concentric outward rings. Each of those rings corresponds to a different hierarchy level. The inner circle looks like a donut chart, but each outer ring can be sliced up depending on its relationship to the inner (parent) circle. Sunburst charts are often a good alternative to treemaps, but if you do opt for this type of chart, keep in mind that its radial layout takes more space than a rectangular shape of a treemap.

[source: <https://www.datylon.com/blog/types-of-charts-graphs-examples-data-visualization>]



## Description

Also known as a *Sunburst Chart*, *Ring Chart*, *Multi-level Pie Chart*, *Belt Chart*, *Radial Treemap*.

This type of visualisation shows hierarchy through a series of rings, that are sliced for each category node. Each ring corresponds to a level in the hierarchy, with the central circle representing the root node and the hierarchy moving outwards from it.

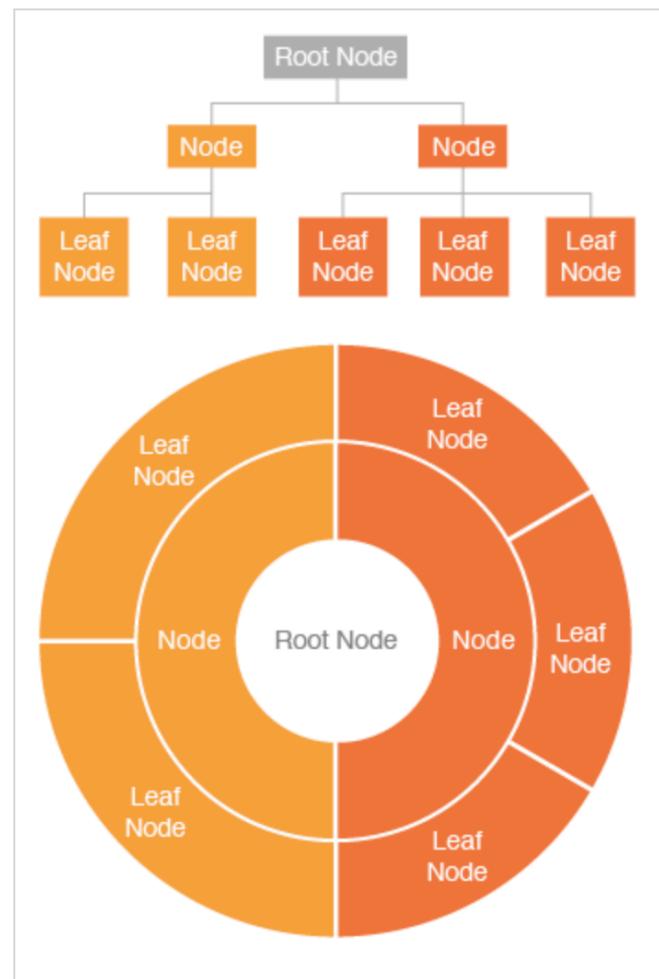
Rings are sliced up and divided based on their hierarchical relationship to the parent slice. The angle of each slice is either divided equally under its parent node or can be made proportional to a value.

Colour can be used to highlight hierachal groupings or specific categories.

## Functions

[Hierarchy](#) [Part-to-a-whole](#)

## Anatomy



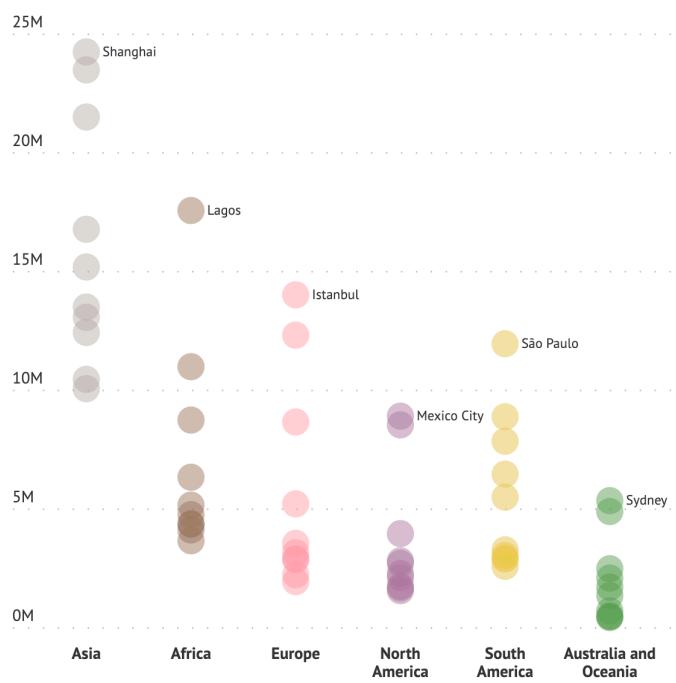
[source: <https://datavizcatalogue.com/>]

# Station F – Strip Plot and Jitter Plot

## Strip plot

Alternative names: *Individual value plot, Single-axis scatter plot*

### The top 10 most populated cities on each continent



Data source: [wikidata.org](https://wikidata.org)

Made with [Datylon](#) - Edit

A strip plot is a type of scatter plot but it only has one categorical and one numerical axis. It is a chart used to illustrate the distribution of many individual one-dimensional values. These values look like dots located along a single (category) axis in this chart. If some of the dots have the same value, they can overlap, creating something that looks like a strip.

[source: <https://www.datylon.com/blog/types-of-charts-graphs-examples-data-visualization>]

## Strip Plot

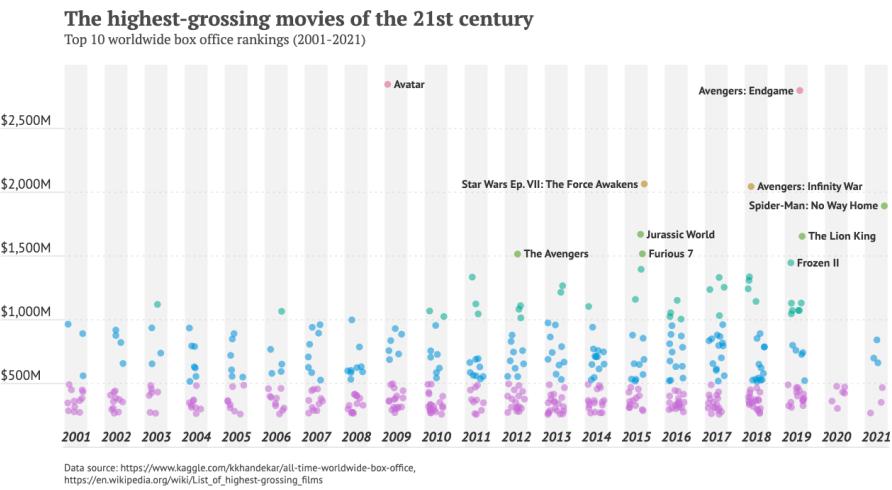
Also called: Individual Value Plot or Single-Axis Scatter Plot



[source: <https://datavizproject.com/>]

# Jitter plot

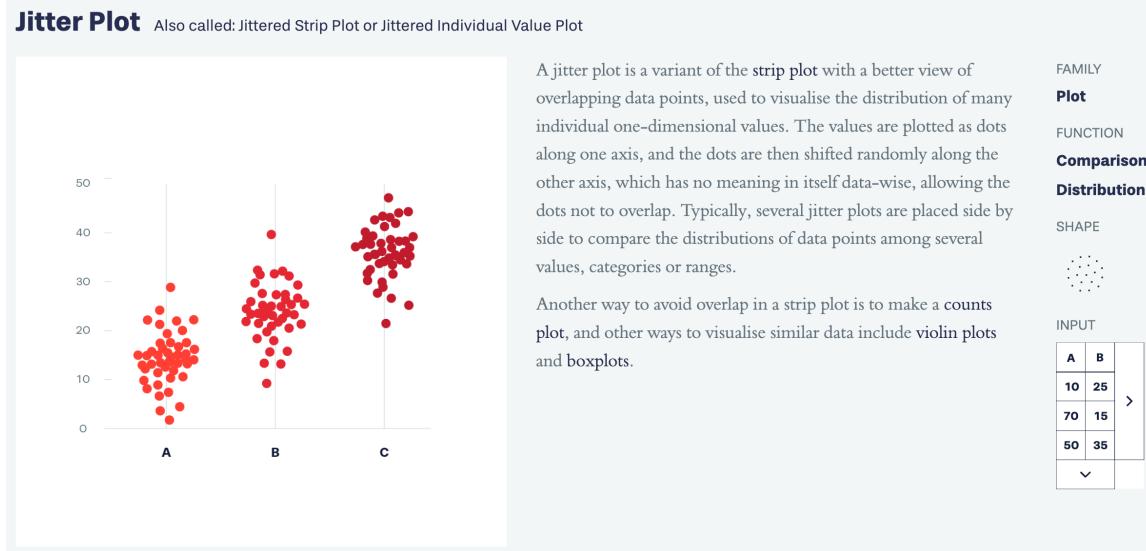
Alternative names: *Jittered strip plot*, *Jittered individual value plot*



Made with Datylon - Edit

A jitter plot is an alternative to a strip plot (see above). It is used to visualize the relationship between a measurement variable and a categorical variable. The main difference from a strip plot is that the dots used in the charts are shifted on the horizontal y-axis, to avoid overlapping (overplotting), which in turn allows avoiding lack of clarity.

[source: <https://www.datylon.com/blog/types-of-charts-graphs-examples-data-visualization>]



[source: <https://datavizproject.com/>]