 [Chart.js](http://docs.google.com/docs/3.9.1/)

[Home](http://docs.google.com/docs/3.9.1/)

[API](http://docs.google.com/docs/3.9.1/api/)

[Samples](http://docs.google.com/docs/3.9.1/samples/)

Ecosystem Ecosystem

* [Awesome (opens new window)](https://github.com/chartjs/awesome)
* [Slack (opens new window)](https://chartjs-slack.herokuapp.com/)
* [Stack Overflow (opens new window)](https://stackoverflow.com/questions/tagged/chart.js)

[GitHub (opens new window)](https://github.com/chartjs/Chart.js)

[Home](http://docs.google.com/docs/3.9.1/)

[API](http://docs.google.com/docs/3.9.1/api/)

[Samples](http://docs.google.com/docs/3.9.1/samples/)

Ecosystem Ecosystem

* [Awesome (opens new window)](https://github.com/chartjs/awesome)
* [Slack (opens new window)](https://chartjs-slack.herokuapp.com/)
* [Stack Overflow (opens new window)](https://stackoverflow.com/questions/tagged/chart.js)

[GitHub (opens new window)](https://github.com/chartjs/Chart.js)

* [Chart.js](http://docs.google.com/docs/3.9.1/)
* Getting Started
* General
* Configuration
* Chart Types
  + [Area Chart](http://docs.google.com/docs/3.9.1/charts/area.html)
  + [Bar Chart](http://docs.google.com/docs/3.9.1/charts/bar.html)
  + [Bubble Chart](http://docs.google.com/docs/3.9.1/charts/bubble.html)
  + [Doughnut and Pie Charts](http://docs.google.com/docs/3.9.1/charts/doughnut.html)
  + [Line Chart](http://docs.google.com/docs/3.9.1/charts/line.html)
  + [Mixed Chart Types](http://docs.google.com/docs/3.9.1/charts/mixed.html)
  + [Polar Area Chart](http://docs.google.com/docs/3.9.1/charts/polar.html)
  + [Radar Chart](http://docs.google.com/docs/3.9.1/charts/radar.html)
  + [Scatter Chart](http://docs.google.com/docs/3.9.1/charts/scatter.html)
* Axes
* Developers

[**#**](#gjdgxs) Doughnut and Pie Charts

Pie and doughnut charts are probably the most commonly used charts. They are divided into segments, the arc of each segment shows the proportional value of each piece of data.

They are excellent at showing the relational proportions between data.

Pie and doughnut charts are effectively the same class in Chart.js, but have one different default value - their cutout. This equates to what portion of the inner should be cut out. This defaults to 0 for pie charts, and '50%' for doughnuts.

They are also registered under two aliases in the Chart core. Other than their different default value, and different alias, they are exactly the same.

config setup

const config = { type: 'doughnut', data: data, };

const config = {  
 type: 'doughnut',  
 data: data,  
};

const data = { labels: [ 'Red', 'Blue', 'Yellow' ], datasets: [{ label: 'My First Dataset', data: [300, 50, 100], backgroundColor: [ 'rgb(255, 99, 132)', 'rgb(54, 162, 235)', 'rgb(255, 205, 86)' ], hoverOffset: 4 }] };

const data = {  
 labels: [  
 'Red',  
 'Blue',  
 'Yellow'  
 ],  
 datasets: [{  
 label: 'My First Dataset',  
 data: [300, 50, 100],  
 backgroundColor: [  
 'rgb(255, 99, 132)',  
 'rgb(54, 162, 235)',  
 'rgb(255, 205, 86)'  
 ],  
 hoverOffset: 4  
 }]  
};

config setup

const config = { type: 'pie', data: data, };

const config = {  
 type: 'pie',  
 data: data,  
};

const data = { labels: [ 'Red', 'Blue', 'Yellow' ], datasets: [{ label: 'My First Dataset', data: [300, 50, 100], backgroundColor: [ 'rgb(255, 99, 132)', 'rgb(54, 162, 235)', 'rgb(255, 205, 86)' ], hoverOffset: 4 }] };

const data = {  
 labels: [  
 'Red',  
 'Blue',  
 'Yellow'  
 ],  
 datasets: [{  
 label: 'My First Dataset',  
 data: [300, 50, 100],  
 backgroundColor: [  
 'rgb(255, 99, 132)',  
 'rgb(54, 162, 235)',  
 'rgb(255, 205, 86)'  
 ],  
 hoverOffset: 4  
 }]  
};

## [**#**](#30j0zll) Dataset Properties

Namespaces:

* data.datasets[index] - options for this dataset only
* options.datasets.doughnut - options for all doughnut datasets
* options.datasets.pie - options for all pie datasets
* options.elements.arc - options for all [arc elements](http://docs.google.com/docs/3.9.1/configuration/elements.html#arc-configuration)
* options - options for the whole chart

The doughnut/pie chart allows a number of properties to be specified for each dataset. These are used to set display properties for a specific dataset. For example, the colours of the dataset's arcs are generally set this way.

| Name | Type | [Scriptable](http://docs.google.com/docs/3.9.1/general/options.html#scriptable-options) | [Indexable](http://docs.google.com/docs/3.9.1/general/options.html#indexable-options) | Default |
| --- | --- | --- | --- | --- |
| [backgroundColor](#1fob9te) | [Color](http://docs.google.com/docs/3.9.1/general/colors.html) | Yes | Yes | 'rgba(0, 0, 0, 0.1)' |
| [borderAlign](#3znysh7) | 'center'|'inner' | Yes | Yes | 'center' |
| [borderColor](#1fob9te) | [Color](http://docs.google.com/docs/3.9.1/general/colors.html) | Yes | Yes | '#fff' |
| [borderJoinStyle](#1fob9te) | 'round'|'bevel'|'miter' | Yes | Yes | undefined |
| [borderRadius](#2et92p0) | number|object | Yes | Yes | 0 |
| [borderWidth](#1fob9te) | number | Yes | Yes | 2 |
| [circumference](#tyjcwt) | number | - | - | undefined |
| [clip](#tyjcwt) | number|object|false | - | - | undefined |
| [data](#3dy6vkm) | number[] | - | - | **required** |
| [hoverBackgroundColor](#1t3h5sf) | [Color](http://docs.google.com/docs/3.9.1/general/colors.html) | Yes | Yes | undefined |
| [hoverBorderColor](#1t3h5sf) | [Color](http://docs.google.com/docs/3.9.1/general/colors.html) | Yes | Yes | undefined |
| [hoverBorderJoinStyle](#1t3h5sf) | 'round'|'bevel'|'miter' | Yes | Yes | undefined |
| [hoverBorderWidth](#1t3h5sf) | number | Yes | Yes | undefined |
| [hoverOffset](#1t3h5sf) | number | Yes | Yes | 0 |
| [offset](#1fob9te) | number | Yes | Yes | 0 |
| [rotation](#tyjcwt) | number | - | - | undefined |
| [spacing](#1fob9te) | number | - | - | 0 |
| [weight](#1fob9te) | number | - | - | 1 |

All these values, if undefined, fallback to the scopes described in [option resolution](http://docs.google.com/general/options)

### [**#**](#tyjcwt) General

| Name | Description |
| --- | --- |
| circumference | Per-dataset override for the sweep that the arcs cover |
| clip | How to clip relative to chartArea. Positive value allows overflow, negative value clips that many pixels inside chartArea. 0 = clip at chartArea. Clipping can also be configured per side: clip: {left: 5, top: false, right: -2, bottom: 0} |
| rotation | Per-dataset override for the starting angle to draw arcs from |

### [**#**](#1fob9te) Styling

The style of each arc can be controlled with the following properties:

| Name | Description |
| --- | --- |
| backgroundColor | arc background color. |
| borderColor | arc border color. |
| borderJoinStyle | arc border join style. See [MDN (opens new window)](https://developer.mozilla.org/en-US/docs/Web/API/CanvasRenderingContext2D/lineJoin). |
| borderWidth | arc border width (in pixels). |
| offset | arc offset (in pixels). |
| spacing | Fixed arc offset (in pixels). Similar to offset but applies to all arcs. |
| weight | The relative thickness of the dataset. Providing a value for weight will cause the pie or doughnut dataset to be drawn with a thickness relative to the sum of all the dataset weight values. |

All these values, if undefined, fallback to the associated [elements.arc.\*](http://docs.google.com/docs/3.9.1/configuration/elements.html#arc-configuration) options.

### [**#**](#3znysh7) Border Alignment

The following values are supported for borderAlign.

* 'center' (default)
* 'inner'

When 'center' is set, the borders of arcs next to each other will overlap. When 'inner' is set, it is guaranteed that all borders will not overlap.

### [**#**](#2et92p0) Border Radius

If this value is a number, it is applied to all corners of the arc (outerStart, outerEnd, innerStart, innerRight). If this value is an object, the outerStart property defines the outer-start corner's border radius. Similarly, the outerEnd, innerStart, and innerEnd properties can also be specified.

### [**#**](#1t3h5sf) Interactions

The interaction with each arc can be controlled with the following properties:

| Name | Description |
| --- | --- |
| hoverBackgroundColor | arc background color when hovered. |
| hoverBorderColor | arc border color when hovered. |
| hoverBorderJoinStyle | arc border join style when hovered. See [MDN (opens new window)](https://developer.mozilla.org/en-US/docs/Web/API/CanvasRenderingContext2D/lineJoin). |
| hoverBorderWidth | arc border width when hovered (in pixels). |
| hoverOffset | arc offset when hovered (in pixels). |

All these values, if undefined, fallback to the associated [elements.arc.\*](http://docs.google.com/docs/3.9.1/configuration/elements.html#arc-configuration) options.

## [**#**](#4d34og8) Config Options

These are the customisation options specific to Pie & Doughnut charts. These options are looked up on access, and form together with the global chart configuration the options of the chart.

| Name | Type | Default | Description |
| --- | --- | --- | --- |
| cutout | number|string | 50% - for doughnut, 0 - for pie | The portion of the chart that is cut out of the middle. If string and ending with '%', percentage of the chart radius. number is considered to be pixels. |
| radius | number|string | 100% | The outer radius of the chart. If string and ending with '%', percentage of the maximum radius. number is considered to be pixels. |
| rotation | number | 0 | Starting angle to draw arcs from. |
| circumference | number | 360 | Sweep to allow arcs to cover. |
| animation.animateRotate | boolean | true | If true, the chart will animate in with a rotation animation. This property is in the options.animation object. |
| animation.animateScale | boolean | false | If true, will animate scaling the chart from the center outwards. |

## [**#**](#2s8eyo1) Default Options

We can also change these default values for each Doughnut type that is created, this object is available at Chart.overrides.doughnut. Pie charts also have a clone of these defaults available to change at Chart.overrides.pie, with the only difference being cutout being set to 0.

## [**#**](#3dy6vkm) Data Structure

For a pie chart, datasets need to contain an array of data points. The data points should be a number, Chart.js will total all of the numbers and calculate the relative proportion of each.

You also need to specify an array of labels so that tooltips appear correctly.

data = {  
 datasets: [{  
 data: [10, 20, 30]  
 }],  
 // These labels appear in the legend and in the tooltips when hovering different arcs  
 labels: [  
 'Red',  
 'Yellow',  
 'Blue'  
 ]  
};

Last Updated: 8/3/2022, 12:46:38 PM

←  [Bubble Chart](http://docs.google.com/docs/3.9.1/charts/bubble.html)   [Line Chart](http://docs.google.com/docs/3.9.1/charts/line.html)  →