* [Chart.js](http://docs.google.com/)
* [Getting Started](http://docs.google.com/getting-started/)
  + [Installation](http://docs.google.com/getting-started/installation.html)
  + [Integration](http://docs.google.com/getting-started/integration.html)
  + [Usage](http://docs.google.com/getting-started/usage.html)
* [General](http://docs.google.com/general/)
  + [Responsive](http://docs.google.com/general/responsive.html)
  + [Pixel Ratio](http://docs.google.com/general/device-pixel-ratio.html)
  + [Interactions](http://docs.google.com/general/interactions/)
    - [Events](http://docs.google.com/general/interactions/events.html)
    - [Modes](http://docs.google.com/general/interactions/modes.html)
  + [Options](http://docs.google.com/general/options.html)
  + [Colors](http://docs.google.com/general/colors.html)
  + [Fonts](http://docs.google.com/general/fonts.html)
* [Configuration](http://docs.google.com/configuration/)
  + [Animations](http://docs.google.com/configuration/animations.html)
  + [Layout](http://docs.google.com/configuration/layout.html)
  + [Legend](http://docs.google.com/configuration/legend.html)
  + [Title](http://docs.google.com/configuration/title.html)
  + [Tooltip](http://docs.google.com/configuration/tooltip.html)
  + [Elements](http://docs.google.com/configuration/elements.html)
* [Charts](http://docs.google.com/)
  + [Line](http://docs.google.com/line.html)
  + [Bar](http://docs.google.com/bar.html)
  + [Radar](http://docs.google.com/radar.html)
  + [Doughnut & Pie](http://docs.google.com/doughnut.html)
  + [Polar Area](http://docs.google.com/polar.html)
  + [Bubble](http://docs.google.com/bubble.html)
  + [Scatter](http://docs.google.com/scatter.html)
  + [Area](http://docs.google.com/area.html)
  + [Mixed](http://docs.google.com/mixed.html)
* [Axes](http://docs.google.com/axes/)
  + [Cartesian](http://docs.google.com/axes/cartesian/)
    - [Category](http://docs.google.com/axes/cartesian/category.html)
    - [Linear](http://docs.google.com/axes/cartesian/linear.html)
    - [Logarithmic](http://docs.google.com/axes/cartesian/logarithmic.html)
    - [Time](http://docs.google.com/axes/cartesian/time.html)
  + [Radial](http://docs.google.com/axes/radial/)
    - [Linear](http://docs.google.com/axes/radial/linear.html)
  + [Labelling](http://docs.google.com/axes/labelling.html)
  + [Styling](http://docs.google.com/axes/styling.html)
* [Developers](http://docs.google.com/developers/)
  + [Chart.js API](http://docs.google.com/developers/api.html)
  + [Updating Charts](http://docs.google.com/developers/updates.html)
  + [Plugins](http://docs.google.com/developers/plugins.html)
  + [New Charts](http://docs.google.com/developers/charts.html)
  + [New Axes](http://docs.google.com/developers/axes.html)
  + [Contributing](http://docs.google.com/developers/contributing.html)
* [Additional Notes](http://docs.google.com/notes/)
  + [Comparison Table](http://docs.google.com/notes/comparison.html)
  + [Popular Extensions](http://docs.google.com/notes/extensions.html)
  + [License](http://docs.google.com/notes/license.html)
* [Published with GitBook](https://www.gitbook.com)

[**Doughnut & Pie**](http://docs.google.com/)

Doughnut and Pie

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 cutoutPercentage. This equates what percentage 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.

## Example Usage

// For a pie chart  
var myPieChart = new Chart(ctx,{  
 type: 'pie',  
 data: data,  
 options: options  
});

// And for a doughnut chart  
var myDoughnutChart = new Chart(ctx, {  
 type: 'doughnut',  
 data: data,  
 options: options  
});

## Dataset Properties

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 colour of a the dataset's arc are generally set this way.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| backgroundColor | Color[] | The fill color of the arcs in the dataset. See [Colors](http://docs.google.com/general/colors.html#colors) |
| borderColor | Color[] | The border color of the arcs in the dataset. See [Colors](http://docs.google.com/general/colors.html#colors) |
| borderWidth | Number[] | The border width of the arcs in the dataset. |
| hoverBackgroundColor | Color[] | The fill colour of the arcs when hovered. |
| hoverBorderColor | Color[] | The stroke colour of the arcs when hovered. |
| hoverBorderWidth | Number[] | The stroke width of the arcs when hovered. |

## Config Options

These are the customisation options specific to Pie & Doughnut charts. These options are merged with the global chart configuration options, and form the options of the chart.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Default | Description |
| cutoutPercentage | Number | 50 - for doughnut, 0 - for pie | The percentage of the chart that is cut out of the middle. |
| rotation | Number | -0.5 \* Math.PI | Starting angle to draw arcs from. |
| circumference | Number | 2 \* Math.PI | 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. |

## Default Options

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

## 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'  
 ]  
};

results matching ""

No results matching ""