 [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
* Axes
  + [Axes](http://docs.google.com/docs/3.9.1/axes/)
  + Cartesian
  + Radial
  + [Labeling Axes](http://docs.google.com/docs/3.9.1/axes/labelling.html)
  + [Styling](http://docs.google.com/docs/3.9.1/axes/styling.html)
* Developers

[**#**](#gjdgxs) Axes

Axes are an integral part of a chart. They are used to determine how data maps to a pixel value on the chart. In a cartesian chart, there is 1 or more X-axis and 1 or more Y-axis to map points onto the 2-dimensional canvas. These axes are known as ['cartesian axes'](http://docs.google.com/docs/3.9.1/axes/cartesian/).

In a radial chart, such as a radar chart or a polar area chart, there is a single axis that maps points in the angular and radial directions. These are known as ['radial axes'](http://docs.google.com/docs/3.9.1/axes/radial/).

Scales in Chart.js >v2.0 are significantly more powerful, but also different than those of v1.0.

* Multiple X & Y axes are supported.
* A built-in label auto-skip feature detects would-be overlapping ticks and labels and removes every nth label to keep things displaying normally.
* Scale titles are supported.
* New scale types can be extended without writing an entirely new chart type.

## [**#**](#30j0zll) Default scales

The default scaleId's for carterian charts are 'x' and 'y'. For radial charts: 'r'. Each dataset is mapped to a scale for each axis (x, y or r) it requires. The scaleId's that a dataset is mapped to, is determined by the xAxisID, yAxisID or rAxisID. If the ID for an axis is not specified, first scale for that axis is used. If no scale for an axis is found, a new scale is created.

Some examples:

The following chart will have 'x' and 'y' scales:

let chart = new Chart(ctx, {  
 type: 'line'  
});

The following chart will have scales 'x' and 'myScale':

let chart = new Chart(ctx, {  
 type: 'bar',  
 data: {  
 datasets: [{  
 data: [1, 2, 3]  
 }]  
 },  
 options: {  
 scales: {  
 myScale: {  
 type: 'logarithmic',  
 position: 'right', // `axis` is determined by the position as `'y'`  
 }  
 }  
 }  
});

The following chart will have scales 'xAxis' and 'yAxis':

let chart = new Chart(ctx, {  
 type: 'bar',  
 data: {  
 datasets: [{  
 yAxisID: 'yAxis'  
 }]  
 },  
 options: {  
 scales: {  
 xAxis: {  
 // The axis for this scale is determined from the first letter of the id as `'x'`  
 // It is recommended to specify `position` and / or `axis` explicitly.  
 type: 'time',  
 }  
 }  
 }  
});

The following chart will have 'r' scale:

let chart = new Chart(ctx, {  
 type: 'radar'  
});

The following chart will have 'myScale' scale:

let chart = new Chart(ctx, {  
 type: 'radar',  
 scales: {  
 myScale: {  
 axis: 'r'  
 }  
 }  
});

## [**#**](#1fob9te) Common Configuration

Note

These are only the common options supported by all axes. Please see specific axis documentation for all of the available options for that axis.

### [**#**](#3znysh7) Common options to all axes

Namespace: options.scales[scaleId]

| Name | Type | Default | Description |
| --- | --- | --- | --- |
| type | string |  | Type of scale being employed. Custom scales can be created and registered with a string key. This allows changing the type of an axis for a chart. |
| alignToPixels | boolean | false | Align pixel values to device pixels. |
| backgroundColor | [Color](http://docs.google.com/docs/3.9.1/general/colors.html) |  | Background color of the scale area. |
| display | boolean|string | true | Controls the axis global visibility (visible when true, hidden when false). When display: 'auto', the axis is visible only if at least one associated dataset is visible. |
| grid | object |  | Grid line configuration. [more...](http://docs.google.com/docs/3.9.1/axes/styling.html#grid-line-configuration) |
| min | number |  | User defined minimum number for the scale, overrides minimum value from data. [more...](http://docs.google.com/docs/3.9.1/axes/#axis-range-settings) |
| max | number |  | User defined maximum number for the scale, overrides maximum value from data. [more...](http://docs.google.com/docs/3.9.1/axes/#axis-range-settings) |
| reverse | boolean | false | Reverse the scale. |
| stacked | boolean|string | false | Should the data be stacked. [more...](http://docs.google.com/docs/3.9.1/axes/#stacking) |
| suggestedMax | number |  | Adjustment used when calculating the maximum data value. [more...](http://docs.google.com/docs/3.9.1/axes/#axis-range-settings) |
| suggestedMin | number |  | Adjustment used when calculating the minimum data value. [more...](http://docs.google.com/docs/3.9.1/axes/#axis-range-settings) |
| ticks | object |  | Tick configuration. [more...](http://docs.google.com/docs/3.9.1/axes/#tick-configuration) |
| weight | number | 0 | The weight used to sort the axis. Higher weights are further away from the chart area. |

## [**#**](#2et92p0) Tick Configuration

Note

These are only the common tick options supported by all axes. Please see specific axis documentation for all of the available tick options for that axis.

### [**#**](#tyjcwt) Common tick options to all axes

Namespace: options.scales[scaleId].ticks

| Name | Type | Scriptable | Default | Description |
| --- | --- | --- | --- | --- |
| backdropColor | [Color](http://docs.google.com/docs/3.9.1/general/colors.html) | Yes | 'rgba(255, 255, 255, 0.75)' | Color of label backdrops. |
| backdropPadding | [Padding](http://docs.google.com/docs/3.9.1/general/padding.html) |  | 2 | Padding of label backdrop. |
| callback | function |  |  | Returns the string representation of the tick value as it should be displayed on the chart. See [callback](http://docs.google.com/docs/3.9.1/axes/labelling.html#creating-custom-tick-formats). |
| display | boolean |  | true | If true, show tick labels. |
| color | [Color](http://docs.google.com/docs/3.9.1/general/colors.html) | Yes | Chart.defaults.color | Color of ticks. |
| font | Font | Yes | Chart.defaults.font | See [Fonts](http://docs.google.com/docs/3.9.1/general/fonts.html) |
| major | object |  | {} | [Major ticks configuration](http://docs.google.com/docs/3.9.1/axes/styling.html#major-tick-configuration). |
| padding | number |  | 3 | Sets the offset of the tick labels from the axis |
| showLabelBackdrop | boolean | Yes | true for radial scale, false otherwise | If true, draw a background behind the tick labels. |
| textStrokeColor | [Color](http://docs.google.com/docs/3.9.1/general/colors.html) | Yes | `` | The color of the stroke around the text. |
| textStrokeWidth | number | Yes | 0 | Stroke width around the text. |
| z | number |  | 0 | z-index of tick layer. Useful when ticks are drawn on chart area. Values <= 0 are drawn under datasets, > 0 on top. |

## [**#**](#3dy6vkm) Axis Range Settings

Given the number of axis range settings, it is important to understand how they all interact with each other.

The suggestedMax and suggestedMin settings only change the data values that are used to scale the axis. These are useful for extending the range of the axis while maintaining the auto fit behaviour.

let minDataValue = Math.min(mostNegativeValue, options.suggestedMin);  
let maxDataValue = Math.max(mostPositiveValue, options.suggestedMax);

In this example, the largest positive value is 50, but the data maximum is expanded out to 100. However, because the lowest data value is below the suggestedMin setting, it is ignored.

let chart = new Chart(ctx, {  
 type: 'line',  
 data: {  
 datasets: [{  
 label: 'First dataset',  
 data: [0, 20, 40, 50]  
 }],  
 labels: ['January', 'February', 'March', 'April']  
 },  
 options: {  
 scales: {  
 y: {  
 suggestedMin: 50,  
 suggestedMax: 100  
 }  
 }  
 }  
});

In contrast to the suggested\* settings, the min and max settings set explicit ends to the axes. When these are set, some data points may not be visible.

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

By default data is not stacked. If the stacked option of the value scale (y-axis on horizontal chart) is true, positive and negative values are stacked separately. Additionally a stack option can be defined per dataset to further divide into stack groups [more...](http://docs.google.com/docs/3.9.1/general/data-structures/#dataset-configuration). For some charts, you might want to stack positive and negative values together. That can be achieved by specifying stacked: 'single'.

## [**#**](#4d34og8) Callbacks

There are a number of config callbacks that can be used to change parameters in the scale at different points in the update process. The options are supplied at the top level of the axis options.

Namespace: options.scales[scaleId]

| Name | Arguments | Description |
| --- | --- | --- |
| beforeUpdate | axis | Callback called before the update process starts. |
| beforeSetDimensions | axis | Callback that runs before dimensions are set. |
| afterSetDimensions | axis | Callback that runs after dimensions are set. |
| beforeDataLimits | axis | Callback that runs before data limits are determined. |
| afterDataLimits | axis | Callback that runs after data limits are determined. |
| beforeBuildTicks | axis | Callback that runs before ticks are created. |
| afterBuildTicks | axis | Callback that runs after ticks are created. Useful for filtering ticks. |
| beforeTickToLabelConversion | axis | Callback that runs before ticks are converted into strings. |
| afterTickToLabelConversion | axis | Callback that runs after ticks are converted into strings. |
| beforeCalculateLabelRotation | axis | Callback that runs before tick rotation is determined. |
| afterCalculateLabelRotation | axis | Callback that runs after tick rotation is determined. |
| beforeFit | axis | Callback that runs before the scale fits to the canvas. |
| afterFit | axis | Callback that runs after the scale fits to the canvas. |
| afterUpdate | axis | Callback that runs at the end of the update process. |

### [**#**](#2s8eyo1) Updating Axis Defaults

The default configuration for a scale can be easily changed. All you need to do is set the new options to Chart.defaults.scales[type].

For example, to set the minimum value of 0 for all linear scales, you would do the following. Any linear scales created after this time would now have a minimum of 0.

Chart.defaults.scales.linear.min = 0;

## [**#**](#17dp8vu) Creating New Axes

To create a new axis, see the [developer docs](http://docs.google.com/docs/3.9.1/developers/axes.html).

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

←  [Scatter Chart](http://docs.google.com/docs/3.9.1/charts/scatter.html)   [Cartesian Axes](http://docs.google.com/docs/3.9.1/axes/cartesian/)  →