Reference version 2.2.1.9000

Plot basics

All ggplot2 plots with a call to **ggplot() (../reference/ggplot.html)**, supplying default data and aesthethic mappings, specified by **aes() (../reference/aes.html)**. You then add layers, scales, coords and facets with `+`. To save a plot to disk, use **ggsave() (../reference/ggsave.html)**.

ggplot (ggplot.html)	Create a new ggplot
aes (aes.html)	Construct aesthetic mappings
+.gg (gg-add.html)	Add components to a plot
ggsave (ggsave.html)	Save a ggplot (or other grid object) with sensible defaults
qplot (qplot.html) quickplot (qplot.html)	Quick plot

Layer: geoms

A layer combines data, aesthetic mapping, a geom (geometric object), a stat (statistical transformation), and a position adjustment. Typically, you will create layers using a `geom_` function, overriding the default position and stat if needed.

geom_abline (geom_abline.html) **geom_hline (geom_abline.html)** **geom_vline (geom_abline.html)**	Reference lines: horizontal, vertical, and diagonal
geom_bar (geom_bar.html) geom_col (geom_bar.html) stat_count (geom_bar.html)	Bars charts
geom_bin2d (geom_bin2d.html) **stat_bin_2d (geom_bin2d.html)**	Heatmap of 2d bin counts
geom_blank (geom_blank.html)	Draw nothing

geom_boxplot (geom_boxplot.html) **stat_boxplot (geom_boxplot.html)**	A box and whiskers plot (in the style of Tukey)
geom_contour (geom_contour.html) **stat_contour (geom_contour.html)**	2d contours of a 3d surface
geom_count (geom_count.html) **stat_sum (geom_count.html)**	Count overlapping points
geom_density_2d (geom_density_2d.html) **stat_density_2d (geom_density_2d.html)**	Contours of a 2d density estimate
geom_density (geom_density.html) **stat_density (geom_density.html)**	Smoothed density estimates
geom_dotplot (geom_dotplot.html)	Dot plot
geom_errorbarh (geom_errorbarh.html)	Horizontal error bars
geom_hex (geom_hex.html) **stat_bin_hex (geom_hex.html)**	Hexagonal heatmap of 2d bin counts
geom_freqpoly (geom_histogram.html) **geom_histogram (geom_histogram.html)** **stat_bin (geom_histogram.html)**	Histograms and frequency polygons
geom_jitter (geom_jitter.html)	Jittered points
geom_crossbar (geom_linerange.html) **geom_errorbar (geom_linerange.html)** **geom_linerange (geom_linerange.html)** **geom_pointrange (geom_linerange.html)**	Vertical intervals: lines, crossbars & errorbars
geom_map (geom_map.html)	Polygons from a reference map

geom_path (geom_path.html) **geom_line (geom_path.html)** **geom_step (geom_path.html)**	Connect observations
geom_point (geom_point.html)	Points
geom_polygon (geom_polygon.html)	Polygons
geom_qq (geom_qq.html) stat_qq (geom_qq.html)	A quantile-quantile plot
geom_quantile (geom_quantile.html) **stat_quantile (geom_quantile.html)**	Quantile regression
geom_ribbon (geom_ribbon.html) **geom_area (geom_ribbon.html)**	Ribbons and area plots
geom_rug (geom_rug.html)	Rug plots in the margins
geom_segment (geom_segment.html) **geom_curve (geom_segment.html)**	Line segments and curves
geom_smooth (geom_smooth.html) **stat_smooth (geom_smooth.html)**	Smoothed conditional means
geom_spoke (geom_spoke.html)	Line segments parameterised by location, direction and distance
geom_label (geom_text.html) **geom_text (geom_text.html)**	Text
geom_raster (geom_tile.html) **geom_rect (geom_tile.html)** **geom_tile (geom_tile.html)**	Rectangles
geom_violin (geom_violin.html) **stat_ydensity (geom_violin.html)**	Violin plot
stat_sf (ggsf.html) geom_sf (ggsf.html) coord_sf (ggsf.html)	Visualise sf objects

Layer: stats

A handful of layers are more easily specified with a `stat_` function, drawing attention to the statistical transformation rather than the visual appearance.

stat_ecdf (stat_ecdf.html)	Compute empirical cumulative distribution

stat_ellipse (stat_ellipse.html)	Compute normal confidence ellipses
stat_function (stat_function.html)	Compute function for each x value
stat_identity (stat_identity.html)	Leave data as is
stat_summary_2d (stat_summary_2d.html) **stat_summary_hex (stat_summary_2d.html)**	Bin and summarise in 2d (rectangle & hexagons)
stat_summary_bin (stat_summary.html) **stat_summary (stat_summary.html)**	Summarise y values at unique/binned x
stat_unique (stat_unique.html)	Remove duplicates

Layer: position adjustment

All layers have a position adjustment that resolves overlapping geoms. Override the default by using the `position` argument to the `geom_` or `stat_` function.

position_dodge (position_dodge.html)	Dodge overlapping objects side-to-side
position_identity (position_identity.html)	Don't adjust position
position_jitter (position_jitter.html)	Jitter points to avoid overplotting
position_jitterdodge (position_jitterdodge.html)	Simultaneously dodge and jitter
position_nudge (position_nudge.html)	Nudge points a fixed distance
position_stack (position_stack.html) **position_fill (position_stack.html)**	Stack overlapping objects on top of each another

Layer: annotations

Annotation are special types of layer than don't inherit global settings from the plot. Rhey are used to add fixed reference data to plot.

geom_abline (geom_abline.html) **geom_hline (geom_abline.html)** **geom_vline (geom_abline.html)**	Reference lines: horizontal, vertical, and diagonal
annotate (annotate.html)	Create an annotation layer
annotation_custom (annotation_custom.html)	Annotation: Custom grob
annotation_logticks (annotation_logticks.html)	Annotation: log tick marks
annotation_map (annotation_map.html)	Annotation: a maps
annotation_raster (annotation_raster.html)	Annotation: high-performance rectangular tiling
borders (borders.html)	Create a layer of map borders

Aesthetics

The following help topics give a broad overview of some of the ways you can use each aesthetic

aes_colour_fill_alpha (aes_colour_fill_alpha.html)	Colour related aesthetics: colour, fill and alpha
aes_group_order (aes_group_order.html)	Aesthetics: grouping
aes_linetype_size_shape (aes_linetype_size_shape.html)	Differentiation related aesthetics: linetype, size, shape
aes_position (aes_position.html)	Position related aesthetics: x, y, xmin, xmax, ymin, ymax, xend, yend

Scales

Scales control the details of how data values are translated to visual properties. Override the default scales to tweak details like the axis labels or legend keys, or to use a completely different translation from data to aesthetic. **labs() (../reference/labs.html)** and **lims() (../reference/lims.html)** are convenient helpers for the most common adjustments to the labels and limits.

labs (labs.html) xlab (labs.html) ylab (labs.html) ggtitle (labs.html)	Modify axis, legend, and plot labels
lims (lims.html) xlim (lims.html) ylim (lims.html)	Set scale limits
expand_limits (expand_limits.html)	Expand the plot limits, using data
scale_alpha (scale_alpha.html) scale_alpha_continuous (scale_alpha.html) scale_alpha_discrete (scale_alpha.html)	Alpha transparency scales
scale_colour_brewer (scale_brewer.html) scale_fill_brewer (scale_brewer.html) scale_colour_distiller (scale_brewer.html) scale_fill_distiller (scale_brewer.html)	Sequential, diverging and qualitative colour scales from colorbrewer.org

scale_x_continuous (scale_continuous.html) scale_y_continuous (scale_continuous.html) scale_x_log10 (scale_continuous.html) scale_y_log10 (scale_continuous.html) scale_x_reverse (scale_continuous.html) scale_y_reverse (scale_continuous.html) scale_x_sqrt (scale_continuous.html) scale_y_sqrt (scale_continuous.html)	Position scales for continuous data (x & y)
scale_x_date (scale_date.html) scale_y_date (scale_date.html) scale_x_datetime (scale_date.html) scale_y_datetime (scale_date.html) scale_x_time (scale_date.html) scale_y_time (scale_date.html)	Position scales for date/time data
scale_x_discrete (scale_discrete.html) scale_y_discrete (scale_discrete.html)	Position scales for discrete data
scale_colour_gradient (scale_gradient.html) scale_fill_gradient (scale_gradient.html) scale_colour_gradient2 (scale_gradient.html) scale_fill_gradient2 (scale_gradient.html) scale_colour_gradientn (scale_gradient.html) scale_fill_gradientn (scale_gradient.html)	Gradient colour scales

scale_colour_grey (scale_grey.html) scale_fill_grey (scale_grey.html)	Sequential grey colour scales
scale_colour_hue (scale_hue.html) scale_fill_hue (scale_hue.html)	Evenly spaced colours for discrete data
scale_colour_identity (scale_identity.html) scale_fill_identity (scale_identity.html) scale_shape_identity (scale_identity.html) scale_linetype_identity (scale_identity.html) scale_alpha_identity (scale_identity.html) scale_size_identity (scale_identity.html)	Use values without scaling
scale_linetype (scale_linetype.html) scale_linetype_continuous (scale_linetype.html) scale_linetype_discrete (scale_linetype.html)	Scale for line patterns
scale_colour_manual (scale_manual.html) scale_fill_manual (scale_manual.html) scale_size_manual (scale_manual.html) scale_shape_manual (scale_manual.html) scale_linetype_manual (scale_manual.html) scale_alpha_manual (scale_manual.html)	Create your own discrete scale
scale_shape (scale_shape.html)	Scales for shapes, aka glyphs

scale_radius (scale_size.html)	Scales for area or radius
scale_size (scale_size.html)	
scale_size_area (scale_size.html)	

Guides: axes and legends

The guides (the axes and legends) help readers interpret your plots. Guides are mostly controlled via the scale (e.g. with the `limits`, `breaks`, and `labels` arguments), but sometimes you will need additional cover over the guide apperance. Use **guides() (../reference/guides.html)** or the `guide` argument to individual scales along with **guide_colourbar() (../reference/guide_colourbar.html)** or **guide_legend() (../reference/guide_legend.html)**.

guide_colourbar (guide_colourbar.html)	Continuous colour bar guide
guide_colorbar (guide_colourbar.html)	
guide_legend (guide_legend.html)	Legend guide
guides (guides.html)	Set guides for each scale
sec_axis (sec_axis.html) dup_axis (sec_axis.html) derive (sec_axis.html)	Specify a secondary axis

Facetting

Facetting generates small multiples, each displaying a different subset of the data. Facets are an alternative to aesthetics for displaying additional discrete variables.

facet_grid (facet_grid.html)	Lay out panels in a grid
facet_wrap (facet_wrap.html)	Wrap a 1d ribbon of panels into 2d

Facetting: labels

These functions provide a flexible toolkit for controlling the display of the “strip” labels on facets.

labeller (labeller.html)	Construct labelling specification

label_value (labellers.html)	Useful labeller functions
label_both (labellers.html)	
label_context (labellers.html)	
label_parsed (labellers.html)	
label_wrap_gen (labellers.html)	
label_bquote (label_bquote.html)	Label with mathematical expressions

Coordinate systems

The coordinate system determines how the `x` and `y` aesthetics combine to position elements in the plot. The default coordinate system is Cartesian (**coord_cartesian() (../reference/coord_cartesian.html)**), which can be tweaked with **coord_map() (../reference/coord_map.html)** , **coord_fixed() (../reference/coord_fixed.html)** , **coord_flip() (../reference/coord_flip.html)** , and **coord_trans() (../reference/coord_trans.html)** , or completely replaced with **coord_polar() (../reference/coord_polar.html)** .

coord_cartesian (coord_cartesian.html)	Cartesian coordinates
coord_fixed (coord_fixed.html)	Cartesian coordinates with fixed "aspect ratio"
coord_flip (coord_flip.html)	Cartesian coordinates with x and y flipped
coord_map (coord_map.html)	Map projections
coord_quickmap (coord_map.html)	
coord_polar (coord_polar.html)	Polar coordinates
coord_trans (coord_trans.html)	Transformed Cartesian coordinate system

Themes

Themes control the display of all non-data elements of the plot. You can override all settings with a complete theme like **theme_bw() (../reference/ggtheme.html)** , or choose to tweak individual settings by using **theme() (../reference/theme.html)** and the `element_` functions. Use **theme_set() (../reference/theme_get.html)** to modify the active theme, affecting all future plots.

theme (theme.html)	Modify components of a theme

theme_grey (ggtheme.html)	Complete themes
theme_gray (ggtheme.html)	
theme_bw (ggtheme.html)	
theme_linedraw (ggtheme.html)	
theme_light (ggtheme.html)	
theme_dark (ggtheme.html)	
theme_minimal (ggtheme.html)	
theme_classic (ggtheme.html)	
theme_void (ggtheme.html)	
theme_test (ggtheme.html)	
theme_get (theme_get.html)	Get, set, and modify the active theme
theme_set (theme_get.html)	
theme_update (theme_get.html)	
theme_replace (theme_get.html)	
%+replace% (theme_get.html)	
margin (element.html)	Theme elements
element_blank (element.html)	
element_rect (element.html)	
element_line (element.html)	
element_text (element.html) rel (element.html)	

Programming with ggplot2

These functions provides tools to help you program with ggplot2, creating functions and for-loops that generate plots for you.

aes_ (aes_.html) aes_string (aes_.html) aes_q (aes_.html)	Define aesthetic mappings programatically
print (print.ggplot.html) plot (print.ggplot.html)	Explicitly draw plot

Extending ggplot2

To create your own geoms, stats, scales, and facets, you'll need to learn a bit about the object oriented system that ggplot2 uses. Start by reading **vignette("extending-ggplot2") (../articles/extending-ggplot2.html)** then consult these functions for more details.

ggproto (ggproto.html)	Create a new ggproto object
ggproto_parent (ggproto.html)	
is.ggproto (ggproto.html)	
print (print.ggproto.html) format (print.ggproto.html)	Format or print a ggproto object

Vector helpers

ggplot2 also provides a handful of helpers that are useful for creating visualisations.

cut_interval (cut_interval.html)	Discretise numeric data into categorical
cut_number (cut_interval.html)	
cut_width (cut_interval.html)	
mean_cl_boot (hmisc.html)	A selection of summary functions from Hmisc
mean_cl_normal (hmisc.html)	
mean_sdl (hmisc.html)	
median_hilow (hmisc.html)	
mean_se (mean_se.html)	Calculate mean and standard error
resolution (resolution.html)	Compute the "resolution" of a numeric vector

Data

ggplot2 comes with a selection of built-in datasets that are used in examples to illustrate various visualisation challenges.

diamonds (diamonds.html)	Prices of 50,000 round cut diamonds
economics (economics.html)	US economic time series
faithfuld (faithfuld.html)	2d density estimate of Old Faithful data
midwest (midwest.html)	Midwest demographics
mpg (mpg.html)	Fuel economy data from 1999 and 2008 for 38 popular models of car
msleep (msleep.html)	An updated and expanded version of the mammals sleep dataset
presidential (presidential.html)	Terms of 11 presidents from Eisenhower to Obama
seals (seals.html)	Vector field of seal movements
txhousing (txhousing.html)	Housing sales in TX

luv_colours (luv_colours.html)	`colors()` in Luv space

Autoplot and fortify

autoplot() (../reference/autoplot.html) is an extension mechansim for ggplot2 it provides a way for package authors to add methods that work like the base `plot()` function, generating useful default plots with little user interaction. **fortify() (../reference/fortify.html)** turns objects into tidy data frames: it has largely been superceded by the broom package (https://github.com/dgrtwo/broom).

autoplot (autoplot.html)	Create a complete ggplot appropriate to a particular data type
autolayer (autolayer.html)	Create a ggplot layer appropriate to a particular data type
fortify (fortify.html)	Fortify a model with data.
map_data (map_data.html)	Create a data frame of map data

ggplot2 is a part of the **tidyverse**, an ecosystem of packages designed with common APIs and a shared philosophy. Learn more at tidyverse.org (http://tidyverse.org).

Developed by Hadley Wickham (http://hadley.nz), Winston Chang.
Site built by pkgdown (http://hadley.github.io/pkgdown/).