Package: ggcheck (via r-universe)

June 3, 2026

Title Inspect 'ggplot2' Plots for Automated Grading in Learning Exercises

Version 0.0.5

Description 'ggcheck' provides functions that inspect 'ggplot2' objects to make it easier for teachers to check that student plots meet expectations. Designed primarily for automated grading via 'gradethis' in interactive 'learnr' tutorials.

License MIT + file LICENSE

URL https://github.com/rstudio/ggcheck

BugReports https://github.com/rstudio/ggcheck/issues

Imports ggplot2, gradethis (>= 0.2.12.9004), purrr, rlang, utils

Suggests testthat

Remotes rstudio/gradethis

Config/Needs/learnr rstudio/learnr, rstudio/gradethis

Config/Needs/website pkgdown, tidyverse/tidytemplate

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Config/pak/sysreqs cmake make libuv1-dev zlib1g-dev

Repository https://rstudio.r-universe.dev

Date/Publication 2023-05-08 19:22:12 UTC

RemoteUrl https://github.com/rstudio/ggcheck

RemoteRef HEAD

RemoteSha 155f77dc6931a48595e4cecdf0a43c1533a77912

Contents

`default_label` *Placeholders for default values*

Description

These functions generate placeholder values.

- `default_label()` can be used as a named argument in `uses_labels()` to check that a label matches the result of `get_default_labels()` with that name.
- `default_param()` can be used as a named argument in `uses_geom_params()` to check that a parameter matched the result of `get_default_params()` with that name.

Usage

```
default_label()

default_param()
```

Value

A placeholder value to be used within `uses_labels()` or `uses_geom_params()`.

Examples

```
require(ggplot2)

p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy, color = trans)) +
  geom_smooth(se = FALSE) +
  labs(title = "My plot", x = "Weight", y = "MPG")

uses_labels(p, x = default_label(), color = default_label())

uses_geom_params(p, "smooth", size = default_param(), se = default_param())
```

---

get_coordinate_system *Which coordinate system does a plot use?*

Description

Which coordinate system does a plot use?

Usage

```
get_coordinate_system(p)
```

Arguments

p	A ggplot2 object

Value

A character string that corresponds to the suffix of a ggplot2 `coord_` function, e.g. `"cartesian"`.

See Also

Other functions for checking coordinate systems: uses_coordinate_system()

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth() +
  coord_polar()
get_coordinate_system(p)
```

---

get_data *Get the data set used by a plot or layer*

Description

get_data returns the data set used by a ggplot object or a single layer extracted from the object with get_geom_layer.

Usage

```
get_data(p, local_only = FALSE)
```

Arguments

p	A ggplot object or a layer extracted from a ggplot object with get_geom_layer.
local_only	TRUE or FALSE. Should get_data onbly return data defined locally in the layer?

Details

When passed a ggplot object (i.e. a plot), get_data will return only the data that has been set globally with ggplot.

When passed a single layer from a plot, the behavior of get_data will depend on the local_only argument passed to .... If local_only = TRUE, get_data will return only the data set, if any, that was defined locally in the function that created the layer. If local_only = FALSE, get_data will return the data used by the layer, whether or not that data was defined globally in ggplot or locally.

Value

A data frame. If no data set is found, get_data returns NULL

See Also

Other functions for checking data: ith_data_is(), ith_data(), uses_data()

Examples

```
require(ggplot2)
d2 <- head(mpg)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(data = d2, color = "red") +
  geom_point()
get_data(p)
get_data(get_geom_layer(p, i = 1))
```

---

get_default_labels *What is the default label for a plot aesthetic?*

Description

What is the default label for a plot aesthetic?

Usage

```
get_default_labels(p, aes = NULL)
```

Arguments

p	A ggplot object
aes	If aes is a character vector, returns only the default labels (based on the plot p) that correspond to the included aesthetics. Defaults to NULL, which returns the default values of all labels.

Value

A named list in which each element is a character string or NULL. Strings are returned for aesthetics with a default value. NULL is returned for aesthetics that do not exist in the plot, or non-aesthetic labels that do not have a default value, like title.

See Also

Other functions for checking labels: get_labels(), uses_labels()

Examples

```
require(ggplot2)

p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class, shape = drv)) +
  geom_smooth() +
  labs(title = "My plot", x = "Weight", y = "MPG", color = NULL)

# Returns the label the ggplot would create by default for an aesthetic
get_default_labels(p, "x")
```

```
get_default_labels(p, c("x", "y"))
get_default_labels(p)

# If an aesthetic does not exist, returns NULL
get_default_labels(p, "size")

# Non-aesthetic labels have no default value, so they also return NULL
get_default_labels(p, "title")
get_default_labels(p, "comment")

# The colo(u)r aesthetic can be matched with or without a u
get_default_labels(p, "color")
get_default_labels(p, "colour")
```

`get_default_params` *What are the default parameters for a plot layer?*

Description

What are the default parameters for a plot layer?

Usage

```
get_default_params(p, geom, params = NULL, i = NULL)
```

Arguments

`p`	A ggplot object
`geom`	A character string found in the suffix of a ggplot2 geom function, e.g. `"point"`.
`params`	A character vector. `get_default_params()` returns the default parameter value with a name matching each string in `params`. If `params` is `NULL` (the default), the default values for all parameters are returned.
`i`	A numerical index, e.g. `1`.

Value

A named list of the same length as `params`, or, if `params` is `NULL`, a named list of default values for all parameters of `geom`.

See Also

Other functions for checking geom parameters: `uses_geom_params()`

Examples

```
require(ggplot2)

p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_smooth(aes(color = class))

# Returns the parameters the ggplot would use by default for a layer
get_default_params(p, "smooth", "linetype")
get_default_params(p, "smooth", c("se", "level"))
get_default_params(p, "smooth")

# If a parameter does not exist, returns NULL
get_default_params(p, "smooth", "shape")

# The colo(u)r aesthetic can be matched with or without a u
get_default_params(p, "smooth", "color")
get_default_params(p, "smooth", "colour")
```

---

get_geom_layer *Isolate a geom layer from a plot*

Description

`get_geom_layer` returns a geom layer from a plot along with the global data sets and aesthetic mappings that the layer may inherit from.

Usage

```
get_geom_layer(p, geom = NULL, i = NULL)
```

Arguments

`p`	A ggplot object
`geom`	A character string found in the suffix of a ggplot2 geom function, e.g. `"point"`.
`i`	A numerical index, e.g. 1.

Details

Users can specify a layer in one of 3 ways:

1. By order of appearance with `i`. The first layer to appear in the plot (the one drawn first, on the bottom) corresponds to `i = 1`.
2. By type of geom with `geom`. `get_geom_layer` will return the first layer that uses the geom.
3. By a combination of `geom` and `i`. `get_geom_layer` will return the ith layer that uses the geom.

Value

An object with class `layer_to_check` to be manipulated further with ggcheck functions.

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(color = "red") +
  geom_point(mapping = aes(color = class)) +
  geom_smooth(se = FALSE)

get_geom_layer(p, i = 1)
get_geom_layer(p, geom = "smooth")
get_geom_layer(p, geom = "point", i = 2)
```

`get_geoms` *List the geoms used by a plot*

Description

`get_geoms` returns a vector of geom names, written as character strings, that describes which geoms in which order are used by a plot.

Usage

```
get_geoms(p)
```

Arguments

p	A ggplot object

Value

A vector of character strings. Each element corresponds to the suffix of a ggplot2 `geom_` function, e.g. `c("point", "line", "smooth")`.

See Also

Other functions for checking geoms: `get_geoms_stats()`, `ith_geom_is()`, `ith_geom_stat()`, `ith_geom()`, `uses_geoms()`

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
get_geoms(p)
```

get_geoms_stats *List the geom and stat combination used by all layers of a plot.*

Description

List the geom and stat combination used by all layers of a plot.

Usage

```
get_geoms_stats(p)
```

Arguments

p	A ggplot object

Value

A list of lists with a GEOM and STAT character. e.g. list(list(GEOM = "point", STAT = "identity"))

See Also

Other functions for checking geoms: `get_geoms()`, `ith_geom_is()`, `ith_geom_stat()`, `ith_geom()`, `uses_geoms()`

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
get_geoms_stats(p)
```

get_labels *List the labels used by a plot*

Description

`get_labels()` returns a named list of labels, written as character strings, indicating which labels are used by a plot.

Usage

```
get_labels(p, aes = NULL)
```

Arguments

p	A ggplot object
aes	If aes is a character vector, returns only the labels corresponding to the included aesthetics. Defaults to NULL, which returns all labels.

Details

Note that get_labels() will return NULL if a label is explicitly set to NULL **or** if a requested aesthetic is not present in the plot.

Value

A named list of character strings.

See Also

Other functions for checking labels: get_default_labels(), uses_labels()

Examples

```
require(ggplot2)

p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth() +
  labs(x = "Weight", y = "MPG", color = NULL)

get_labels(p)
get_labels(p, c("x", "y"))

# The colo(u)r aesthetic can be matched with or without a u
get_labels(p, "color")
get_labels(p, "colour")
```

---

get_mappings *Get aesthetic mappings from a layer or plot*

Description

get_mappings returns the mappings used by a ggplot object or a single layer extracted from the object with get_geom_layer or get_stat_layer.

Usage

```
get_mappings(p, local_only = FALSE)
```

Arguments

p	A ggplot object or a layer extracted from a ggplot object with get_geom_layer or get_stat_layer.
local_only	TRUE or FALSE. Should get_mappings return only the mappings defined locally in a layer. This has no effect when p is a ggplot object.

Details

When passed a ggplot object (i.e. a plot), get_mappings will return only the mappings that have been set globally with ggplot. When passed a single layer from a plot, the behavior of get_mappings will depend on the value of local_only. If local_only = TRUE, get_mappings will return only the mappings defined locally in a layer. When local_only = FALSE, get_mappings will return the combination of global and local methods that will be used to plot a layer.

Value

A list with class uneval, as returned by aes Components of the list are either quosures or constants.

See Also

Other functions for checking mappings: identical_aes(), ith_mappings_use(), ith_mappings(), uses_mappings()

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class))
get_mappings(p)
get_mappings(get_geom_layer(p, i = 1), local_only = FALSE)
```

---

get_stat_layer *Isolate a stat layer from a plot*

Description

get_stat_layer returns a stat layer from a plot along with the global data sets and aesthetic mappings that the layer may inherit from.

Usage

```
get_stat_layer(p, stat = NULL, i = NULL)
```

Arguments

p	A ggplot object
stat	A character string found in the suffix of a ggplot2 stat function, e.g. "bin".
i	A numerical index, e.g. 1.

Details

Users can specify a layer in one of 3 ways:

1. By order of appearance with `i`. The first layer to appear in the plot (the one drawn first, on the bottom) corresponds to `i = 1`.
2. By type of stat with `stat`. `get_stat_layer` will return the first layer that uses the stat
3. By a combination of `stat` and `i`. `get_stat_layer` will return the ith layer that uses the stat

Value

An object with class `layer_to_check` to be manipulated further with ggcheck functions.

Examples

```
require(ggplot2)
p <- ggplot(data = diamonds, aes(price)) +
  stat_bin(bins = 20, binwidth = 500)

get_stat_layer(p, i = 1)
get_stat_layer(p, stat = "bin")
```

---

get_stats *List the stats used by a plot*

Description

`get_stats` returns a vector of stats names, written as character strings, that describes which stats in which order are used by a plot.

Usage

```
get_stats(p)
```

Arguments

`p`	A ggplot object

Value

A vector of character strings. Each element corresponds to the suffix of a ggplot2 `stat_` function, e.g. `c("identity", "smooth")`.

See Also

Other functions for checking stats: `ith_stat_is()`, `ith_stat()`, `uses_stats()`

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
get_stats(p)
```

`gradethis_equal.ggplot` *Compare two* `ggplot`*s to check whether they are equal*

Description

Compare two `ggplot`s to check whether they are equal

Usage

```
## S3 method for class 'ggplot'
gradethis_equal(x, y, ...)
```

Arguments

`x, y`	Two `ggplot` objects to compare
`...`	Unused

Value

A logical value of length one, or an internal gradethis error.

See Also

`gradethis::gradethis_equal()` for the generic function.

Examples

```
library(ggplot2)
library(ggcheck)
library(gradethis)

cty_plot <- ggplot(mpg, aes(x = displ, y = cty)) + geom_point()
hwy_plot <- ggplot(mpg, aes(x = displ, y = cty)) + geom_point()

gradethis_equal(cty_plot, hwy_plot)
gradethis_equal(cty_plot, cty_plot)
```

identical_aes *Are aesthetic mapping specifications "identical"?*

Description

The ggplot2 package uses quosures to record aesthetic mappings. These record both the mapping described as well as the environment in which the mapping was described. As a result, it is difficult to compare mappings created by students in one environment to mappings created on the fly by graders in another environment. `identical_aes` facilitates comparison by ignoring the environments associated with an aesthetic mapping specification. If the two specifications contain identical expressions, e.g. `x = displ`, etc., `identical_aes` returns `TRUE`.

Usage

```
identical_aes(a1, a2)
```

Arguments

a1	The output of aes, perhaps extracted from a ggplot object.
a2	The output of aes, perhaps extracted from a ggplot object.

Value

`TRUE` or `FALSE`

See Also

Other functions for checking mappings: `get_mappings()`, `ith_mappings_use()`, `ith_mappings()`, `uses_mappings()`

is_ggplot *Check if an object is a ggplot*

Description

`is_ggplot()` tests if an object is a ggplot.

`stop_if_not_ggplot()` signals an error if an object is not a ggplot.

`fail_if_not_ggplot()` returns a failing grade if an object is not a ggplot.

Usage

```
is_ggplot(p)

stop_if_not_ggplot(p, message = getOption("ggcheck.error"))

fail_if_not_ggplot(
  p = .result,
  message = getOption("ggcheck.fail"),
  env = parent.frame()
)
```

Arguments

p	An object
message	A message to be displayed if p is not a ggplot object.
env	Environment in which to find .result. Most users of ggcheck will not need to use this argument.

Value

is_ggplot() returns TRUE if p is a ggplot object; otherwise it returns FALSE.

stop_if_not_ggplot() returns an error if p is not a ggplot object; other it invisibly returns NULL.

fail_if_not_ggplot() returns a failing grade if p is not a ggplot object; other it invisibly returns NULL.

Examples

```
require(ggplot2)

p_valid <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point()
is_ggplot(p_valid)
stop_if_not_ggplot(p_valid)
fail_if_not_ggplot(p_valid)

p_invalid <- geom_point()
is_ggplot(p_invalid)
## Not run:
stop_if_not_ggplot(p_invalid)

## End(Not run)
fail_if_not_ggplot(p_valid)
```

`ith_data` *Which data set does the ith layer use?*

Description

`ith_data` returns the data set used by the ith layer.

Usage

```
ith_data(p, i, local_only = FALSE)
```

Arguments

`p`	A ggplot object or a layer extracted from a ggplot object with `get_geom_layer`.
`i`	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on.
`local_only`	`TRUE` or `FALSE`. See the details.

Details

If `local_only = TRUE`, `ith_data` returns the data set, if any, that was defined locally in the function that created the ith layer. If `local_only = FALSE`, `ith_data` returns the data used by the ith layer, whether or not that data was defined globally in `ggplot` or locally.

Functions that use the `ith_` prefix are designed to eliminate the need to call `get_geom_layer` to check a specific layer in a plot, e.g. `p %>% get_geom_layer(geom = "point") %>% get_data()`.

Value

A data frame. If no data set is found, `ith_data` returns `NULL`.

See Also

Other functions for checking data: `get_data()`, `ith_data_is()`, `uses_data()`

Examples

```
require(ggplot2)
d2 <- head(mpg)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(data = d2, color = "red") +
  geom_point()
ith_data(p, i = 1)
```

ith_data_is *Does the ith layer use the correct data set?*

Description

ith_data_is checks whether the student uses the supplied data set for the ith layer of their plot.

Usage

```
ith_data_is(p, data, i, local_only = FALSE)
```

Arguments

p	A ggplot object or a layer extracted from a ggplot object with get_geom_layer.
data	A data frame
i	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on.
local_only	TRUE or FALSE. See the details.

Details

Functions that use the ith_ prefix are designed to eliminate the need to call get_geom_layer to check a specific layer in a plot, e.g. p %>% get_geom_layer(geom = "point") %>% uses_data(mpg).

If local_only = TRUE, ith_data_is will check only the data set, if any, that was defined locally in the function that created the ith layer. If local_only = FALSE, ith_data_is will check the data used by the ith layer, whether or not that data was defined globally in ggplot or locally.

Value

TRUE or FALSE

See Also

Other functions for checking data: get_data(), ith_data(), uses_data()

Examples

```
require(ggplot2)
d2 <- head(mpg)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(data = d2, color = "red") +
  geom_point()
ith_data_is(p, data = head(mpg), i = 1)
```

ith_geom *Which geom is used in the ith layer?*

Description

ith_geom returns the type of geom used by the ith layer.

Usage

```
ith_geom(p, i)
```

Arguments

p	A ggplot object
i	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on.

Value

A character string that corresponds to the suffix of a ggplot2 geom_ function, e.g. "point".

See Also

Other functions for checking geoms: get_geoms_stats(), get_geoms(), ith_geom_is(), ith_geom_stat(), uses_geoms()

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
ith_geom(p, i = 2)
```

ith_geom_is *Is the ith geom what it should be?*

Description

ith_geom_is checks whether the ith layer uses the prescribed type of geom.

Usage

```
ith_geom_is(p, geom, i = 1)
```

Arguments

p	A ggplot object
geom	A character string that corresponds to the suffix of a ggplot2 geom_ function, e.g. "point".
i	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on. ith_geom_is will check the geom used by the ith layer.

Value

TRUE or FALSE

See Also

Other functions for checking geoms: get_geoms_stats(), get_geoms(), ith_geom_stat(), ith_geom(), uses_geoms()

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
ith_geom_is(p, geom = "smooth", i = 2)
```

---

ith_geom_stat *Which geom/stat combination is used in the ith layer?*

Description

ith_geom_stat returns the type of geom used by the ith layer according to a geom/stat combination.

Usage

```
ith_geom_stat(p, i)
```

Arguments

p	A ggplot object
i	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on.

Value

A list of lists with a GEOM and STAT strings, each corresponding to the suffix of a ggplot2 geom_ function (e.g. "point"), and stat_ function (e.g. "identity"). e.g. list(list(GEOM = "point", STAT = "identity"))

See Also

Other functions for checking geoms: get_geoms_stats(), get_geoms(), ith_geom_is(), ith_geom(), uses_geoms()

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
ith_geom_stat(p, i = 2)
```

---

ith_mappings *Return the aesthetic mappings used by the ith layer*

Description

ith_mappings returns the mappings used by a ggplot object or a single layer extracted from the object with get_geom_layer or get_stat_layer.

Usage

```
ith_mappings(p, i, local_only = FALSE)
```

Arguments

p	A ggplot object or a layer extracted from a ggplot object with get_geom_layer or get_stat_layer.
i	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on. ith_mappings_use will check the aesthetics used by the ith layer.
local_only	If TRUE, ith_mappings_use will check only the mappings defined locally in a layer for the presence of mappings. If FALSE, ith_mappings_use will check for mappings in the combination of global and local methods that will be used to plot a layer.

Details

Functions that use the ith_ prefix are designed to eliminate the need to call get_layer to check a specific layer in a plot, e.g. p %>% get_geom_layer(geom = "point") %>% get_mappings().

Value

A list with class uneval, as returned by aes Components of the list are either quosures or constants.

See Also

Other functions for checking mappings: `get_mappings()`, `identical_aes()`, `ith_mappings_use()`, `uses_mappings()`

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
ith_mappings(p, i = 1, local_only = FALSE)
ith_mappings(p, i = 1, local_only = TRUE)
ith_mappings(p, i = 2, local_only = FALSE)
```

---

ith_mappings_use *Does the ith layer use one or more aesthetic mappings?*

Description

`ith_mappings_use` checks whether the student uses the supplied mappings in the ith layer of their plot.

Usage

```
ith_mappings_use(p, mappings, i, local_only = FALSE, exact = FALSE)
```

Arguments

`p`	A ggplot object or a layer extracted from a ggplot object with `get_geom_layer` or `get_stat_layer`.
`mappings`	One or more aesthetic mappings created with `aes`.
`i`	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on. `ith_mappings_use` will check the aesthetics used by the ith layer.
`local_only`	If `TRUE`, `ith_mappings_use` will check only the mappings defined locally in a layer for the presence of `mappings`. If `FALSE`, `ith_mappings_use` will check for `mappings` in the combination of global and local methods that will be used to plot a layer.
`exact`	If `TRUE`, mappings need to be mapped exactly

Details

`ith_mappings_use` ignores whether or not the student supplied additional mappings as well. Functions that use the `ith_` prefix are designed to eliminate the need to call `get_layer` to check a specific layer in a plot, e.g. `p uses_mappings(aes(color = class))`.

Value

A logical value

See Also

Other functions for checking mappings: `get_mappings()`, `identical_aes()`, `ith_mappings()`, `uses_mappings()`

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
ith_mappings_use(p, i = 1, aes(x = displ), local_only = FALSE)
ith_mappings_use(p, i = 1, aes(x = displ), local_only = TRUE)
ith_mappings_use(p, i = 2, aes(x = displ, y = hwy), local_only = FALSE)
```

---

ith_stat *Which stat is used in the ith layer?*

Description

`ith_stat` returns the type of stat used by the ith layer.

Usage

```
ith_stat(p, i)
```

Arguments

p	A ggplot object
i	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on.

Value

A character string that corresponds to the suffix of a ggplot2 `stat_` function, e.g. `"qq"`.

See Also

Other functions for checking stats: `get_stats()`, `ith_stat_is()`, `uses_stats()`

Examples

```
require(ggplot2)
p <- ggplot(data = diamonds, aes(sample = price)) +
  geom_qq()
ith_stat(p, i = 1)
```

ith_stat_is *Is the ith stat what it should be?*

Description

`ith_stat_is` checks whether the ith layer uses the prescribed type of stat

Usage

```
ith_stat_is(p, stat, i = 1)
```

Arguments

`p`	A ggplot object
`stat`	A character string that corresponds to the suffix of a ggplot2 `stat_` function, e.g. `"identity"`.
`i`	A numerical index that corresponds to the first layer of a plot (1), the second layer (2), and so on. `ith_stat_is` will check the stat used by the ith layer.

Value

`TRUE` or `FALSE`

See Also

Other functions for checking stats: `get_stats()`, `ith_stat()`, `uses_stats()`

Examples

```
require(ggplot2)
p <- ggplot(data = diamonds, aes(sample = price)) +
  geom_qq()
ith_stat_is(p, i = 1, "qq")
```

n_layers *How many layers are in a plot?*

Description

How many layers are in a plot?

Usage

```
n_layers(p)
```

Arguments

p	A ggplot object

Value

Numeric. The number of layers.

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
n_layers(p)
```

---

uses_aesthetics *Does a plot use one or more aesthetics?*

Description

`uses_aesthetics` checks whether the student used one or more aesthetics.

Usage

```
uses_aesthetics(p, aesthetics, local_only = FALSE, exact = FALSE)
```

Arguments

p	A ggplot object or a layer extracted from a ggplot object with `get_geom_layer` or `get_stat_layer`..
aesthetics	character vector of variables to check for, e.g. "x" or c("x")
local_only	`TRUE` or `FALSE`. Should `uses_aesthetics` only return mappings defined locally in the layer?
exact	If `TRUE`, variables need to be mapped exactly

Details

By default, `uses_aesthetics` requires that only one of the aesthetics need to be used. Set `exact` to `TRUE` to check if all of the variables have to be matched exactly.

Value

A logical value.

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class))
uses_aesthetics(p, "x")
uses_aesthetics(p, c("x", "y"))
uses_aesthetics(get_geom_layer(p, "point"), c("x", "y", "color"), local_only = TRUE)
uses_aesthetics(get_geom_layer(p, "point"), c("x", "y"), local_only = FALSE)
```

---

uses_coordinate_system

Does a plot use the correct coordinate system?

Description

`uses_coordinate_system` checks whether a plot uses the coordinate system you describe. To describe a coordinate system, use the character string that matches the suffix of the ggplot2 `coord_` function that would make the coordinate system. The default coordinate system for ggplot2 plots is `"cartesian"`.

Usage

```
uses_coordinate_system(p, coordinates)
```

Arguments

`p`	A ggplot2 object
`coordinates`	A character string that corresponds to the suffix of a ggplot2 `coord_` function, e.g. `"cartesian"`.

Value

`TRUE` or `FALSE`

See Also

Other functions for checking coordinate systems: `get_coordinate_system()`

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth() +
  coord_polar()
uses_coordinate_system(p, coordinates = "polar")
```

uses_data *Does a plot or layer use the correct data set?*

Description

uses_data checks whether the data set used by a plot or layer matches the data set provided.

Usage

```
uses_data(p, data, local_only = FALSE)
```

Arguments

p	A ggplot object or a layer extracted from a ggplot object with get_geom_layer.
data	A data frame
local_only	TRUE or FALSE. See the details.

Details

When passed a ggplot object (i.e. a plot), uses_data will check only the data that has been set globally with ggplot.

When passed a single layer from a plot, the behavior of uses_data will depend on the local_only argument passed to .... If local_only = TRUE, uses_data will check only the data set, if any, that was defined locally in the function that created the layer. If local_only = FALSE, uses_data will check the data used by the layer, whether or not that data was defined globally in ggplot or locally.

Value

A data frame.

See Also

Other functions for checking data: get_data(), ith_data_is(), ith_data()

Examples

```
require(ggplot2)
d2 <- head(mpg)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(data = d2, color = "red") +
  geom_point()
uses_data(p, mpg)
uses_data(get_geom_layer(p, i = 1), data = head(mpg))
```

uses_extra_mappings *Does the plot uses extra aesthetic mappings?*

Description

uses_extra_mappings checks if a student's plot contains more than the required aesthetic mappings. Note that we still return TRUE if the student's plot differs from the required aesthetic mappings because they are technically extra mappings from required set. We recommend you use uses_mapping checks for checking required mappings before uses_extra_mappings.

Usage

```
uses_extra_mappings(p, mappings, local_only = FALSE)
```

Arguments

p	A ggplot object or a layer extracted from a ggplot object with get_geom_layer or get_stat_layer.
mappings	One or more aesthetic mappings created with aes.
local_only	If TRUE, uses_extra_mappings will check only the mappings defined locally in a layer for the presence of mappings. If FALSE, uses_extra_mappings will check for mappings in the combination of global and local methods that will be used to plot a layer.

Value

A logical value.

Examples

```
require(ggplot2)
p <- ggplot(data = diamonds, aes(x = cut, sample = price)) +
  geom_qq()
uses_extra_mappings(p, aes(sample = price))
```

uses_geom_params *Does a layer use one of more specific parameters?*

Description

uses_geom_params checks that a plot's geom layer uses a specific parameter.

Usage

```
uses_geom_params(p, geom, ..., params = NULL, i = NULL)

uses_geom_param(p, geom, ..., params = NULL, i = NULL)
```

Arguments

p	A ggplot object
geom	A character string found in the suffix of a ggplot2 geom function, e.g. "point".
...	<dynamic-dots> Named values or character strings. Unnamed arguments will check whether any value was set for that parameter. Named arguments will check whether the parameter with the same name has a matching value. Each argument should have a name matching a ggplot layer parameter. Values may be passed as arguments or as list elements.
params	A named list of geom or stat parameter values, e.g. list(outlier.alpha = 0.01). This list is combined with any inputs to ...
i	A numerical index, e.g. 1.

Details

To specify a specific geom layer, either specify using position using the i index or by using a combination of geom function suffix name and i to check the ith layer that uses the geom.

The params argument accepts a list that contains geom, stat, or aes parameters. This offers flexibility in certain situations where setting a parameter on a geom_ function is actually setting a stat parameter or aes parameter. For example, in geom_histogram(binwidth = 500), the binwidth is a stat parameter, while in geom_histogram(fill = "blue"), the fill is an aes parameter. uses_geom_params will take this into account and check geom, stat, and aes parameters.

Note that uses_geom_params() can detect aes *parameters*, but not aes *mappings*. Parameters are set to static values directly within a layer (e.g. geom_point(color = "blue")), while mappings associate variables in the data with plot aesthetics using aes() (e.g. geom_point(aes(color = class))).

Value

A named logical vector of the same length as the number of inputs to ....

See Also

Other functions for checking geom parameters: get_default_params()

Examples

```
require(ggplot2)

p <- ggplot(data = diamonds, aes(x = cut, y = price)) +
  geom_boxplot(varwidth = TRUE, outlier.alpha = 0.01, fill = "blue")

uses_geom_params(
  p, "boxplot", list(varwidth = TRUE, outlier.alpha = 0.01, fill = "blue")
)

uses_geom_params(
  p, "boxplot", varwidth = TRUE, outlier.alpha = 0.01, fill = "blue"
)
```

```
# Unnamed arguments check that a parameter is set to any value
uses_geom_params(p, "boxplot", "fill")
```

---

uses_geoms *Does a plot use one or more geoms?*

Description

use_geoms tests whether a plot uses one or more geoms created using a geom. If checking for a layer that is created using a stat function, please use uses_stats instead.

Usage

```
uses_geoms(p, geoms, stats = NULL, exact = TRUE)
```

Arguments

p	A ggplot object
geoms	A vector of character strings. Each element should correspond to the suffix of a ggplot2 geom_ function, e.g. c("point", "line", "smooth").
stats	A character vector to optionally check for the stats corresponding to geoms e.g. c("identity", "smooth") if checking c("point", "smooth")
exact	A boolean to indicate whether to use exact matching

Details

By default, the plot must have the exact geoms or geom/stat combinations and in the same order. However, if exact is set to FALSE, the plot geoms or geom/stat combinations do not have to be exact.

Value

TRUE or FALSE

See Also

Other functions for checking geoms: get_geoms_stats(), get_geoms(), ith_geom_is(), ith_geom_stat(), ith_geom()

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
uses_geoms(p, geoms = "point")
uses_geoms(p, geoms = c("point", "smooth"), exact = TRUE)
uses_geoms(p, geoms = c("point", "smooth"), stats = c("identity", "smooth"))
```

uses_labels *Does a plot use one or more labels?*

Description

`uses_labels()` tests whether a plot uses one or more labels.

Usage

```
uses_labels(p, ...)
```

Arguments

p	A ggplot object
...	<dynamic-dots> Character strings. Unnamed arguments will check whether a label exists for that aesthetic. Named arguments will check whether the aesthetic with the same name has a label with a matching value. Each argument should have a matching ggplot aesthetic or label. Strings may be input as individual arguments or as list elements.

Details

Note that `uses_labels()` will match `NULL` if a label is explicitly set to `NULL` **or** if a requested aesthetic is not present in the plot.

Value

A named logical vector of the same length as the number of inputs to `...`.

See Also

Other functions for checking labels: `get_default_labels()`, `get_labels()`

Examples

```
require(ggplot2)

p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class, shape = drv)) +
  geom_smooth() +
  labs(title = "My plot", x = "Weight", y = "MPG", color = NULL)

# Unnamed arguments check if a label is set for the given aesthetic
uses_labels(p, "title", "subtitle", "x", "y")

# The check will return TRUE for labels set to NULL
uses_labels(p, "color")

# The check will return TRUE for aesthetics with default labels
```

```
uses_labels(p, "shape")

# Named arguments check if the label matches an expected value
uses_labels(p, x = "Weight")
uses_labels(p, x = "Weight", y = "MPG", color = NULL)

# You can check for default labels with default_label()
uses_labels(p, shape = default_label(), x = default_label())

# The colo(u)r aesthetic can be matched with or without a u
uses_labels(p, color = NULL)
uses_labels(p, colour = NULL)

# Inputs can be passed from a list, with or without the !!! operator
label_list <- list(x = "Weight", y = "MPG", color = NULL)
uses_labels(p, label_list)
uses_labels(p, !!!label_list)
```

`uses_mappings` *Does a plot or layer use one or more mappings?*

Description

`uses_mappings` checks whether the student used one or more mappings in their plot. By default, `uses_mappings` ignores whether or not the student also supplied additional mappings. Use `uses_extra_mappings` to check if they did. If `exact` is `TRUE`, then all of the mappings have to match exactly.

Usage

```
uses_mappings(p, mappings, local_only = FALSE, exact = FALSE)
```

Arguments

`p`	A ggplot object or a layer extracted from a ggplot object with `get_geom_layer` or `get_stat_layer`.
`mappings`	One or more aesthetic mappings created with `aes`.
`local_only`	If `TRUE`, `uses_mappings` will check only the mappings defined locally in a layer for the presence of `mappings`. If `FALSE`, `uses_mappings` will check for `mappings` in the combination of global and local methods that will be used to plot a layer.
`exact`	If `TRUE`, mappings need to be mapped exactly

Value

A logical value.

See Also

Other functions for checking mappings: `get_mappings()`, `identical_aes()`, `ith_mappings_use()`, `ith_mappings()`

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class))
uses_mappings(p, aes(x = displ))
uses_mappings(get_geom_layer(p, i = 1), aes(x = displ, color = class), local_only = FALSE)
uses_mappings(get_geom_layer(p, i = 1), aes(x = displ, color = class), local_only = TRUE)
uses_mappings(p, aes(x = displ, y = hwy), exact = TRUE)
```

---

uses_stat_param *Does a layer use a specific stat parameter?*

Description

`uses_stat_param` is a mirror function of `uses_geom_param` but instead of checking a plot's geom layer, it checks that a plot's stat layer uses a specific stat parameter.

Usage

```
uses_stat_param(p, stat, params, i = NULL)
```

Arguments

`p`	A ggplot object
`stat`	A character string found in the suffix of a ggplot2 stat function, e.g. `"bin"`.
`params`	A named list of stat or geom parameter values, e.g. `list(bins = 200)`
`i`	A numerical index, e.g. 1.

Details

To specify a specific stat layer, either specify using position using the `i` index or by using a combination of `stat` function suffix name and `i` to check the ith layer that uses the stat.

Value

A boolean

Examples

```
require(ggplot2)
p <- ggplot(diamonds, aes(carat)) +
  stat_bin(bins = 200)
uses_stat_param(p, stat = "bin", params = list(bins = 200))
```

uses_stats *Does a plot use one or more stats?*

Description

uses_stats tests whether a plot uses one or more stats in its layers.

Usage

```
uses_stats(p, stats, geoms = NULL, exact = TRUE)
```

Arguments

p	A ggplot object
stats	A vector of character strings. Each element should correspond to the suffix of a ggplot2 stat_ function, e.g. c("identity", "smooth").
geoms	A character vector to optionally check for the geoms corresponding to stats e.g. c("point", "smooth") if checking c("identity", "smooth")
exact	if TRUE, use exact matching

Details

By default, the plot must have the exact stats or geom/stat combinations and in the same order. However, if exact is set to FALSE, the plot stats or geom/stat combinations do not have to be exact.

Value

TRUE or FALSE

See Also

Other functions for checking stats: get_stats(), ith_stat_is(), ith_stat()

Examples

```
require(ggplot2)
p <- ggplot(data = mpg, mapping = aes(x = displ, y = hwy)) +
  geom_point(mapping = aes(color = class)) +
  geom_smooth()
uses_stats(p, stats = "smooth")
uses_stats(p, stats = c("identity", "smooth"), exact = TRUE)
uses_stats(p, c("smooth", "identity"), geoms = c("smooth", "point"))
```

Index