

# Package ‘myplots.rb’

January 25, 2024

**Title** The Package Contains Useful Plotting Functions

**Version** 0.0.0.9000

## Description

This package contains a function for plotting influence statistics and a quick scatterplot function.

**License** GPL (>= 3)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.1

**Depends** ggplot2,  
R (>= 3.5)

## R topics documented:

ggraph . . . . .	1
influence_plots . . . . .	2

<b>Index</b>	<b>3</b>
--------------	----------

---

ggraph	<i>Create a quick scatter plot in ggplot.</i>
--------	---

---

## Description

This will graph two given vectors in a ggplot-style scatter plot with the x-axis labeled "x" and the y-axis labeled "y".

## Usage

```
ggraph(x, y, point_color = "black", point_size = 1.5, point_shape = 19)
```

## Arguments

x	This is the first vector to be plotted.
y	This is the first vector to be plotted.
point_color	This is the color of the points that will be plotted.
point_size	This is the size of the points that will be plotted. The default is size 1.5.
point_shape	This is the shape of the points that will be plotted. The default is 19: a filled circle.

**Value**

This function returns a ggplot scatter plot object.

**Examples**

```
## Create a scatter plot of y vs x.  
x <- rnorm(100)  
y <- x + rnorm(100, 0, 0.3)  
ggraph(x, y)
```

---

influence_plots	<i>Influence Plots</i>
-----------------	------------------------

---

**Description**

This will compute many common residual and influence plots to check adequacy of a given model.

**Usage**

```
influence_plots(model)
```

**Arguments**

model	This is a lm or glm (with binomial family) object
-------	---

**Value**

This function returns plots for the jackknife (externally studentized) residuals vs index, jackknife residuals vs fitted values, leverage values vs index, Cook's distance vs index, DfFits vs index, and all DfBetas vs index.

**Examples**

```
## Randomly generate correlated variables x and y and then create the  
## influence plots for them  
x <- rnorm(100)  
y <- x + rnorm(100, 0, 0.3)  
influence_plots(lm(y ~ x))
```

# Index

ggraph, [1](#)

influence\_plots, [2](#)