

# Package ‘PrettyR’

March 23, 2018

**Title** PrettyR

**Version** 0.1.1

**Description** Functions to make pretty graphs and tables. See my web-site <<https://sites.google.com/a/g.clemson.edu/ja-resources>>. or github <<https://github.com/Jadamso>>.

**Depends** R (>= 3.4.1)

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Published** 2018-03-23

**Date** 2018-03-23

**URL** <https://sites.google.com/a/g.clemson.edu/ja-resources>

**RoxygenNote** 6.0.1

**Imports** utils,  
stargazer,  
parallel

**Suggests** formattable

## R topics documented:

bin_mode . . . . .	2
CI_lu . . . . .	2
loess_list . . . . .	3
matJ . . . . .	4
mfx . . . . .	4
mfxi.lm . . . . .	5
mfxlist2vec . . . . .	5
mfxtrim . . . . .	6
polygon_add . . . . .	6
polygon_ci . . . . .	7
polygon_plot . . . . .	7
root . . . . .	8
scatboot . . . . .	8
scatter_gram . . . . .	9
spaghetti_lines . . . . .	10

spaghetti_mean . . . . .	10
spaghetti_plot . . . . .	11
stargazerJ . . . . .	11
stargazerJlist . . . . .	12
Ttest2 . . . . .	13
UROOT . . . . .	14
Wstat . . . . .	14
<b>Index</b>	<b>15</b>

---

bin_mode	<i>Take the Mode of Data after Binning</i>
----------	--

---

## Description

Take the Mode of Data after Binning

## Usage

```
bin_mode(x, breaks = "Sturges")
```

## Arguments

x	a numeric vector
breaks	see hist

## Value

scalar

## Examples

```
bin_mode(runif(100))
```

---

CI_lu	<i>Custom confidence intervals</i>
-------	------------------------------------

---

## Description

Custom confidence intervals

## Usage

```
CI_lu(Pred = NULL, CI = NULL, Fit = NULL, SE = NULL, level = 0.95,
      degf = Inf)
```

**Arguments**

Pred	predict(reg, se.fit = TRUE) object
CI	Compute CI From Regression Prediction Object
Fit	Compute Yhat From Regression Prediction Object
SE	Compute Standard errors
level	confidence level
deg	degrees of freedom in t-distribution

**Details**

Should first try predict(reg, interval = 'confidence', level=level) note deg=inf coef estimates are approx normally distributed

**Value**

data frame of upper and lower confidence bounds

---

loess_list	<i>Run Multiple Loess for Spaghetti Plot</i>
------------	--

---

**Description**

Run Multiple Loess for Spaghetti Plot

**Usage**

```
loess_list(form, splitDF, spag_idname, split = FALSE, parallel = TRUE, ...)
```

**Arguments**

form	regression formula
splitDF	dataframe to regress on
spag_idname	split data frame by this id
split	split up splitDF into list of dataframes?
parallel	use multiple cores
...	arguments passed to loess

**Value**

list of loess outputs

---

matJ	<i>Convert Matrix for stargazerJ</i>
------	--------------------------------------

---

**Description**

Convert Matrix for stargazerJ

**Usage**

```
matJ(Xmat, Xrows = (1:nrow(Xmat))[1:nrow(Xmat)%2 == 0],
      Xcols = 1:ncol(Xmat), Xrd = 0, Xfm = NULL)
```

**Arguments**

Xmat	a matrix of summary regression information
Xrows	which rows to add braces to
Xcols	which columns to add braces to
Xrd	how many digits to round
Xfm	formatC format

**Value**

datatable

---

mfx	<i>Formatting</i>
-----	-------------------

---

**Description**

Formatting

**Usage**

```
mfx(MFXall, trimnames = "PCrast5", varnames = c("HHI", "HHI^2"),
     cnames = c("ACLED", "UCDP"), ROOT = TRUE, nvar = 3, colrep = 2)
```

**Arguments**

nvar	number of variables of interest
colrep	

**Value**

datatable

---

`mfxi.lm`*Run a Regression*

---

**Description**

Run a Regression

**Usage**

```
mfxi.lm(formi, datai, vcv = "standard")
```

**Arguments**

<code>formi</code>	regression formula
<code>datai</code>	data for regression
<code>vcv</code>	type of covariance correction

**Value**

summary table

---

`mfxlist2vec`*Table Formatting*

---

**Description**

Table Formatting

**Usage**

```
mfxlist2vec(trimlist, nvar = 3)
```

**Arguments**

<code>trimlist</code>	object from mfxtrim
<code>nvar</code>	number of variables of interest

**Value**

list of formatted summary tables

---

mfxttrim	<i>Table Summarizing</i>
----------	--------------------------

---

**Description**

Table Summarizing

**Usage**

```
mfxttrim(testlist, rnames, stat = "Std. Error")
```

**Arguments**

testlist	list of lmtest::coefest objects
stat	which statistic to retrieve
varnames	names of variable names

**Value**

list of summary tables

---

polygon_add	<i>Polygon Plot</i>
-------------	---------------------

---

**Description**

Polygon Plot

**Usage**

```
polygon_add(X, ci_lu, col = rgb(0, 0, 0, 0.25), bcol = NA)  
  
polygon_format(X, ci_lu)
```

**Arguments**

X	X matrix from polygon_ci
ci_lu	matrix from polygon_ci
col, bcol	color objects

**Value**

plots polygon

---

`polygon_ci`

*Polygon Plot*

---

### Description

Polygon Plot

### Usage

```
polygon_ci(reg, xname, ...)
```

### Arguments

<code>reg</code>	output from <code>lm()</code> or <code>loess()</code>
<code>xname</code>	name of RHS of regression formula
<code>...</code>	passed to <code>predict()</code>
<code>level</code>	confidence levels

### Value

list with matrix of lower,upper confidence intervals and X variable

---

`polygon_plot`

*Plot the polygon*

---

### Description

Plot the polygon

### Usage

```
polygon_plot(reg, xname, xlb = "X", ylb = "Y", xlm = NULL, ylm = NULL,  
            ttl = NULL, off = FALSE, ...)
```

### Arguments

<code>reg</code>	lm object
<code>xname</code>	name of xvariable of interest
<code>level</code>	confidence interval range

### Value

plots polygon

---

root	<i>Formatting Inputs to Unit Root</i>
------	---------------------------------------

---

### Description

Formatting Inputs to Unit Root

### Usage

```
root(MFX, mfxn1 = "HHI", mfxn2 = "HHI^2", min0 = 0, max1 = 1)
```

### Arguments

MFX	matrix of coefficients
mfxn1, mfxn2	variable names (rows)
min0	minimum of possible values
max1	maximum of possible values

### Details

each column of MFX should provide a different set of estimates

### Value

datatable

---

scatboot	<i>M out of N Bootstrapped Loess Confidence Intervals</i>
----------	---

---

### Description

M out of N Bootstrapped Loess Confidence Intervals

### Usage

```
scatboot(x, y, breps = 100, mfun = function(m) { m^(0.9) },
  confidence = 0.9, family = "gaussian", degree = 2, span = 2/3)
```

### Arguments

x, y	coordinates
breps	number of bootstrap replications
mfun	function to define m out of n subsample
confidence	CI interval
degree, span, family	loess parameters



**Details**

see "<http://content.csbs.utah.edu/~rogers/datanal/R/scatboot.r>"

**Value**

list of loess outputs

---

scatter_gram	<i>Scatter Plot with Histograms</i>
--------------	-------------------------------------

---

**Description**

Scatter Plot with Histograms

**Usage**

```
scatter_gram(X, Y, XBINS = NULL, xbin_scale = function(x) {      x * 2 + 0.5
}, xbks = "Sturges", ybks = "Sturges", col = rgb(0, 0, 0, 0.5),
xlb = "X", ylb = "Y", xrange = NA, yrange = NA, ttl = NULL)
```

**Arguments**

X	vector of values
Y	vector of values
XBINS	bin the X,Y for plotting
xbin_scale	how to scale XBINS
xbks, ybks	how to make histograms
col	color of the plot
xlb, ylb	axis lables
ttl	plot title

**Value**

plots polygon

**Examples**

```
scatter_gram(1:100, runif(100))
```

---

spaghetti_lines	<i>Plot Spaghetti Lines</i>
-----------------	-----------------------------

---

**Description**

Plot Spaghetti Lines

**Usage**

```
spaghetti_lines(YList, Xlist, plot_col = "#00000080", lwd = 0.5, ...)
```

**Arguments**

YList, Xlist	output from PrettyR::spag_loess
plot_col	col in lines()
lwd, ...	passed to lines()

**Details**

a loess line for each list element

**Value**

nothing

---

spaghetti_mean	<i>Plot Average of Spaghetti Lines</i>
----------------	--

---

**Description**

Plot Average of Spaghetti Lines

**Usage**

```
spaghetti_mean(YList, Xlist, plot_col = 1, lwd = 2, ...)
```

**Arguments**

YList, Xlist	output from PrettyR::spag_loess
plot_col, lwd	passed to lines()
...	passed to lines()

---

spaghetti_plot	<i>Spaghetti Plot, a loess line for each list element</i>
----------------	---

---

**Description**

Spaghetti Plot, a loess line for each list element

**Usage**

```
spaghetti_plot(LoList, pdfname, xlb = "X", ylb = "Y", ttl = NULL,
  xln = 3, yln = 3, xlm = NULL, ylm = NULL, plot_col = rgb(0, 0, 0,
  0.5), meanline = FALSE, ...)
```

**Arguments**

LoList	output from PrettyR::spag_loess
pdfname	full name of pdf
xlb, ylb, ttl	axis names
xln, yln	axis lines
xlm, ylm	axis limits
plot_col	color of spaghetti lines
meanline	draw the mean of the spaghettis
...	passed to spaghetti_lines

**Details**

seealso <http://www.ats.ucla.edu/stat/r/faq/spagplot.htm>

---

stargazerJ	<i>Table Formatting</i>
------------	-------------------------

---

**Description**

Table Formatting

**Usage**

```
stargazerJ(Xmat, ..., new.align = NULL, new.title = NULL,
  new.metatitle = NULL, new.tail = NULL, tab.out = NULL,
  new.notes = NULL, pb.tw = 1, new.omit = NULL, out = NULL,
  column.sep.width = "0pt", new.digits = 2, new.digits.extra = 5,
  new.table.placement = "H", new.other = TRUE, new.parbox = NULL)
```

**Arguments**

Xmat	object from matJ
out	NULL
new.parbox	

Details

See also matJ

Value

datatable

Examples

```
MATRIX <- data.frame( diag(4) )
NOTES <- paste0("NOTE", LETTERS, collapse=" ")
new.align <- "l cc @{\\hspace{18pt}} cc"
TITLE <- paste0( "& \\multicolumn{2}{c}{MEOW}",
  "& \\multicolumn{2}{c}{MIX} \\",
  "& Estimate & SE & Estimate & SE")
C3 <- paste0( " Controls &", " G$&$P & & G$&$P & ", " \\ " )
C4 <- paste0( " F.E. &", " C$&$T & & C$&$T & ", " \\ " )

stargazerJ( MATRIX,
  title="Meow Mix Table",
  label="tab:meowmix",
  column.sep.width="-4pt",
  new.notes=NOTES,
  pb.tw=.7,
  new.tail=c(C3, C4),
  new.title=TITLE,
  new.align=new.align)
```

---

stargazerJlist	Table Formatting for Lists
----------------	----------------------------

---

Description

Table Formatting for Lists

Usage

```
stargazerJlist(Xlist, ..., new.align = NULL, new.title = NULL,
  new.metatitle = NULL, new.tail = NULL, tab.out = NULL,
  new.notes = NULL, pb.tw = 1, new.omit = NULL, out = NULL,
  column.sep.width = "0pt", new.digits = 2, new.digits.extra = 5,
  new.table.placement = "H", new.other = TRUE, new.parbox = NULL)
```

Details

different handling in part ## Xmat and Table dimensions

---

Ttest2	<i>Welchs t-test</i>
--------	----------------------

---

## Description

Welchs t-test

Welchs t-test, variant3

## Usage

```
Ttest2(m1, m2, s1, s2, n1, n2, side = 2, m0 = 0)
```

```
Ttest3(m1, m2, s1, s2, n1, n2, m0 = 0, equal.variance = FALSE)
```

## Arguments

m1, m2	the sample means
s1, s2	the sample standard errors
n1, n2	the sample sizes
side	upper, lower, or 2sided
m0	the null value for the difference in means to be tested for. Default is 0.

## Value

a summary of the T-test

## Examples

```
n1 <- 100
x1 <- 1:n1
y1 <- 3*x1+ rnorm(n1, 0, 50)
lm1 <- summary( lm( y1~x1) )
coefs1 <- coef(lm1)[2,1:2]
n2 <- 200
x2 <- 1:n2
y2 <- 2*x2 + rnorm(n2, 0, 50)
lm2 <- summary( lm( y2~x2) )
coefs2 <- coef(lm2)[2,1:2]
Ttest2( coefs1[1], coefs2[1], coefs1[2], coefs2[2], n1, n2 )
```

---

URROOT	<i>Formatting Inputs to Unit Root</i>
--------	---------------------------------------

---

**Description**

Formatting Inputs to Unit Root

**Usage**

URROOT(x, b1, b2, ...)

**Arguments**

x	sequence
b1, b2	coefficients

**Value**

a vector

---

Wstat	<i>Welchs T-test for a Y variable calculated from a window on each side of x0</i>
-------	---

---

**Description**

Welchs T-test for a Y variable calculated from a window on each side of x0

**Usage**

Wstat(wind, dframe, x0, yvar, xvar, corr = 0, ...)

**Arguments**

wind	fraction (percent of data to include)
dframe	data to cut
x0	
xvar	

**Value**

list of loess outputs

# Index

`bin_mode`, [2](#)

`CI_lu`, [2](#)

`loess_list`, [3](#)

`matJ`, [4](#)

`mfx`, [4](#)

`mfxi.lm`, [5](#)

`mfxlist2vec`, [5](#)

`mfxtrim`, [6](#)

`polygon_add`, [6](#)

`polygon_ci`, [7](#)

`polygon_format (polygon_add)`, [6](#)

`polygon_plot`, [7](#)

`root`, [8](#)

`scatboot`, [8](#)

`scatter_gram`, [9](#)

`spaghetti_lines`, [10](#)

`spaghetti_mean`, [10](#)

`spaghetti_plot`, [11](#)

`stargazerJ`, [11](#)

`stargazerJlist`, [12](#)

`t.test2 (Ttest2)`, [13](#)

`Ttest2`, [13](#)

`Ttest3 (Ttest2)`, [13](#)

`UROOT`, [14](#)

`Wstat`, [14](#)