Cheatography

R Cheat Sheet Cheat Sheet

by Jeffrey M. Pisklak (non_human_entity) via cheatography.com/127433/cs/24819/

Operators

`=` `<-`	Assigns a value to an object
`>`	greater than
`<`	less than
`>=`	greater than or equal to
`<=`	less than or equal to
`==`	exactly equal to
`!=`	not equal to
`!x`	not x
`x | y`	x OR y
`x & y`	x AND y
`|>`	Sends something (e.g., dataframe or function output) to something else. i.e., it is an elegant way to nest tidyverse functions.

Example:

```
library(tidyverse)
msleep |>
   fil ter (co nse rvation == 'domes tic ated') |>
     sum mar ise(m = mean(b rainwt, na.rm = TRUE),
                 s = sd(bra inwt, na.rm = TRUE),
                 med = median (br ainwt, na.rm = TRUE)
                 )
```

Note that `summar ise()` is nested within `filter()`, which is using info from the `msleep` dataframe.

Basic R Functions

Access a function's help file

```
?[function name]
```

Load a csv file

```
read.csv( "  fi l e na  me.c  s v ", header = TRUE )
```

Install a library

```
insta  ll.p  a c k ag  es(  " l ibrary name")
```

Load an installed library

```
libra  ry(  li brary name)
```

Resize images in Google Collab

```
optio  ns(  re p r.p  lot.width = x, repr.p  lo t.h eight = y)
```

Return the amount of values in x

```
lengt h(x)
```

Return the number of rows in a dataframe

```
nrow(d ata frame)
```

Return the absolute value(s) in x

```
abs(x)
```

Return the sum of all the values in x

```
sum(x)
```

Return the square-root of the value(s) in x

```
sqrt(x)
```

By **Jeffrey M. Pisklak**
(non_human_entity)
cheatography.com/non-human-entity/

Published 28th December, 2023.
Last updated 22nd September, 2025.

Cheatography

R Cheat Sheet Cheat Sheet

by Jeffrey M. Pisklak (non_human_entity) via cheatography.com/127433/cs/24819/

Basic R Functions (cont)

Return the mean of the values in x with optional arguments for trimming and removing NAs

```
mean(x, tr = 0, na.rm = FALSE)
```

Return the median of the values in x with optional arguments removing NAs

```
median(x, na.rm = FALSE)
```

Return the sample standard deviation of values in x with optional argument for removing NAs

```
sd(x, na.rm = FALSE)
```

Return the sample variance of values in x with optional argument for removing NAs

```
var(x, na.rm = FALSE)
```

Return the quartiles for x with optional argument for removing NAs

```
quanti le(x, na.rm = FALSE)
```

Sort the values of x into ascending order

```
sort(x)
```

Compute the median absolute deviation of x with optional argument to remove NAs

```
mad(x, na.rm = FALSE)
```

Find NA values in x (returns TRUE/FALSE)

```
is.na(x)
```

Paste things together into a single string

```
paste(x, y, z, sep = "  ")
```

Create a table of counts

Examples:

```
table(x)
table(x, y)
```

Data Frames

Create a new data frame

```
col_1 <- c("A ", " B", " C")
col_2 <- c(21, 22, NA)
new_df <- data.f ram e(c ol_1, col_2)
```

Add a column

```
new_df $col_3 <- c(51, 52, 53)
```

Select a specific value (e.g., 52 = row 2, column 3)

```
new_df[2, 3]
```

Select a series of values (e.g., all of row 2)

```
new_df[2, c(1,2,3)]
```

or

```
new_df[2, ]
```

Select an entire column (e.g., column 2)

```
new_df $col_2
```

or

```
new_df[ , 2]
```

Isolate values that are not NAs

```
new_df $co l_2 [!i s.n a(n ew_ df$ col_2)]
```

Filter Function

Used to select specific observations from a dataframe according to a rule you specify.

```
filter (da taf rame, subset rule)
```

Example 1:

```
filter (he igh tData, Father < 60.1 | Father > 75.3)
```

Example 2:

```
heightData |> filter (Father < 60.1 | Father > 75.3)
```

Cheatography

R Cheat Sheet Cheat Sheet

by Jeffrey M. Pisklak (non_human_entity) via cheatography.com/127433/cs/24819/

Subset Function

Used to select specific observations from a dataframe according to a rule you specify.

```
subset (da taf rame, subset rule, select = ("co lumns to keep"))
```

Example:

```
outliers <- subset (he igh tData, Father < 60.1 | Father > 75.3, select = c("F ath er"))
```

Library Functions

library(tidyverse) or **library(dplyr)**

Aggregate data sets into a new dataframe.

For example . . .

```
msleep %>%
  group_by(vore, conser vation) %>%
  summarise(m = mean(b rainwt, na.rm = TRUE),
            s = sd(bra inwt, na.rm = TRUE)
            )
```

library(rcompanion)

Calculates lambda for Tukey's ladder of powers

```
transf orm Tuk ey(x, plotit = FALSE, return Lambda = TRUE)
```

library(WRS2)

Winsorized *variance* of x

```
winvar(x, tr = .2)
```

Distribution Functions

Return the the corresponding *quantile* for a given cumulative probability

Normal Distribution

```
qnorm  (pr  ob a b ility, mean, sd)
```

T Distribution

```
qt(pr  oba  bi lity, df, lower.t ail)
```

F Distribution

```
qf(pro bab ility, df1, df2, lower.t ail)
```

Chi-Square Distribution

```
qchisq (pr oba bility, df, lower.t ail)
```

Return the the corresponding cumulative *probability* for a given quantile.

Normal Distribution

```
pnorm  (qu antile, mean, sd)
```

T Distribution

```
pt(qua ntile, df, lower.t ail)
```

F Distribution

```
pf(qua ntile, df1, df2, lower.t ail)
```

Chi-Square Distribution

```
pchisq (qu antile, df, lower.t ail)
```

Note:

- z-scores and t-scores (e.g. critical T and test statistics) are types of quantiles.

- The calculations are all performed from left to right by default unless you specify lower.tail = FALSE).

Plotting: library(ggplot2)

Histogram

```
ggplot (da taF rame, aes(x = Dep_Var)) +
            geo m_h ist ogr am( colour = " bla ck",
                                    fill = " whi te")
```

Plotting: library(ggplot2) (cont)

Density Plot

```
ggplot (da taF rame, aes(x = Dep_Var)) +
            geo m_d ens ity (colour = " bla ck",
                                    fill = " pin k",
                                        adjust = 1)
```

Boxplot - for one sample

```
ggplot (da taF rame, aes(y = Dep_Var,)) +
            geo m_b oxp lot()
```

Boxplot - for two or more samples

```
ggplot (da taF rame, aes(x = Indep_Var, y = Dep_Var)) +
            geo m_b oxp lot()
```

Barplot with errorbars

```
ggplot (pl otData, aes(x = Indep_Var, y = Dep_Var,
                                     fill = Indep_ Var)) +
    geom_b ar(stat = " ide nti ty", colour = " bla ck") +
    geom_e rro rba r(a es(ymin = bottom _va lues,
                                          ymax = top_va lues),
                                    width = .25)
```

Q-Q Plot For two independent samples

Remove + facet_wrap() for a single sample

```
ggplot (da taF rame, aes(sample = Dep_Var)) +
                stat_qq() +
                stat_q q_l ine() +
              facet_ wrap(~ Indep_Var)
```

Line Plot of Means with Two Predictors

```
ggplot (pl otData, aes(x = Predic torA, y = Means, group = Predic torB, colour = Predic torB)) +
     geom_l ine (po sition = positi on_ dod ge( width = 0.4)) +
     geom_p oin t(p osition = positi on_ dod ge( width = 0.4))
```

Scatterplot with Regression Line

```
ggplot (da taf rame, aes(x = predictor, y = response)) +
     geom_p oint() +
     geom_a bli ne( int ercept = b0, slope = b1)
```

Note:

Indep_Var = Independent Variable

Dep_Var = Dependent Variable

plotData = Dataframe of aggregated values

R Style Guide (from the Tidyverse)

https://style.tidyverse.org/