

R Functions List (+ Examples) | All Basic Commands of the R Programming Language

There are thousands and thousands of functions in the [R programming language](#) available – And every day more commands are added to the [Cran homepage](#).

To bring some light into the dark of the R jungle, I'll provide you in the following with a (very incomplete) list of some of the **most popular and useful R functions**.

For many of these functions, I have created **tutorials with quick examples**. Just click on the function you are interested in, in order to get more information.

Let me know in the comments, in case you are interested in any particular function. I'll try my best to publish a tutorial for this function as soon as possible.



List of R Commands & Functions

- [abline](#) – Add straight lines to plot.
- [abs](#) – Compute the absolute value of a numeric data object.
- [addmargins](#) – Put margins on tables or arrays.
- [addNA](#) – Turn NA values into a factor level.
- [aggregate](#) – Compute summary statistics of subgroups of a data set.
- [alist](#) – Create a list object containing function arguments.
- [all.equal](#) – Test if two objects are nearly equal.
- [all_equal \[dplyr\]](#) – Compare two data frames.

- all_groups_identical [groupdata2] – Test if groupings of two factors are identical.
- all – Check whether all values of a logical vector are TRUE.
- anti_join [dplyr] – Anti join two data frames.
- any – Check whether any values of a logical vector are TRUE.
- apply – Apply function by rows or columns of a data frame.
- approx – Return a list of values that linearly interpolate given data points.
- approxfun – Return a function that performs linear interpolation.
- apropos – Return character vector with names of objects that contain the input.
- arrange [dplyr] – Order data frames and tibbles.
- as.data.frame – Coerce object to data frame.
- as.Date – Convert character strings to Date class.
- as.double – Convert integer to double class.
- as.factor – Convert a data object to the class factor.
- as.function – Convert an object to a function.
- as.name – Convert character string to name class object.
- as.numeric – Convert a data object to the class numeric.
- as.ordered – Create ordered factor.
- as.POSIXct – Manipulate date and time objects.
- as.POSIXlt – Manipulate date and time objects.
- as.single – Convert double-precision vector to single.
- as.tbl [dplyr] – Convert data.frame to tibble.
- attach – Give access to variables of a data.frame.
- attr – Return or set a specific attribute of a data object.
- attributes – Return or set all attributes of a data object.
- ave – Calculate averages.
- axis – Create user-defined axis ticks.
- barplot – Create a barplot.
- beep [beepR] – Play sound at the end of an R script.
- between [dplyr] – Test whether a numeric value falls into a specified range.
- bind_cols [dplyr] – Bind columns and vectors.

- bind_rows [dplyr] – Bind rows of data sets.
- body – Print and change the body of a function.
- box – Draw a border around a plot.
- boxplot – Create a boxplot.
- break – Break for-loop in R.
- browser – Inspect the currently used environment.
- call – Create objects of the class call.
- case_when [dplyr] – Distinguish between cases based on logical conditions.
- casefold – Translate character to lower or upper case.
- cases [memisc] – Distinguish between cases based on logical conditions.
- cat – Return character string in readable format.
- cbind – Combine vectors, matrices and/or data frames by column.
- ceiling – Round numeric up to the next higher integer.
- charmatch – Return position of first match between two data objects.
- chartr – Translate characters of string to lower and/or upper case.
- chol – Conduct a Choleski decomposition.
- choose.files – Interactively choose a file.
- clip – Specify the clipping region of a graphic.
- coalesce [dplyr] – Replace missing values.
- colMeans – Compute the mean of each column of a numeric data frame, matrix or array.
- colMedians [robustbase] – Compute the median of each column of a numeric matrix.
- colnames – Retrieve or set the column names of a data frame or matrix.
- colSums – Compute the sum of each column of a numeric data frame, matrix or array.
- combine [dplyr] – Combine values.
- combn [combinat] – Generate all possible combinations of vector object.
- comment – Query or set a comment attribute.
- complete.cases – Return a logical vector that indicates which cases have no missing values.
- cor – Calculate correlation coefficients.
- crossprod – Calculate matrix cross products.
- cumall [dplyr] – Apply cumulative version of all Function.

- cumany[dplyr] – Apply cumulative version of any Function.
- cume_dist[dplyr] – Rank a vector.
- cummax – Calculate cumulative maxima.
- cummean[dplyr] – Apply cumulative version of mean Function.
- cummin – Calculate cumulative minima.
- cumprod – Calculate cumulative products.
- cumsum – Calculate the cumulative sum of the elements of a numeric vector.
- cut – Convert numeric values into factorial ranges.
- data.frame – Create data frames.
- date – Return the current system date and time.
- dbern – Return corresponding value of bernoulli PDF.
- dbeta – Return corresponding value of beta density.
- dbinom – Return corresponding value of binomial density.
- dcast[data.table] – Reshape data.tables.
- dcauchy – Return corresponding value of cauchy density.
- dchisq – Return corresponding value of chi-square PDF.
- dense_rank[dplyr] – Rank a vector.
- density – Draw Kernel Density Plot.
- deparse – Convert an expression to the character class.
- deriv – Compute the derivative of an expression.
- det – Calculate the determinant of a matrix.
- detach – Remove the attachment of a data.frame or unload a package.
- determinant – Calculate the modulus of the determinant on the logarithm scale.
- dexp – Return corresponding value of exponential density.
- df – Return corresponding value of F PDF.
- dgamma – Return corresponding value of gamma density.
- dgeom – Return corresponding value of geometric PDF.
- dhyper – Return corresponding value of hypergeometric PDF.
- diag – Modify the diagonal of a matrix.
- diff – Compute difference between pairs of consecutive elements of a vector.

- difftime – Calculate the time difference of two date or time objects.
- digits [TeachingDemos] – Add straight lines to plot.
- dim – Return the dimension (e.g. the number of columns and rows) of a matrix, array or data frame.
- dimnames – Get and set the dimnames of a data object.
- dir – Return a character vector of file and/or folder names within a directory.
- dist – Compute distance matrices.
- distCosine [geosphere] – Calculate the Law of Cosines great circle distance.
- distHaversine [geosphere] – Calculate the Haversine great circle distance.
- distinct [dplyr] – Retain only unique rows of data set.
- distMeeus [geosphere] – Calculate the Meeus great circle distance.
- distRhumb [geosphere] – Calculate a distance along a rhumb line.
- distVincentyEllipsoid [geosphere] – Calculate the Vincenty Ellipsoid great circle distance.
- distVincentySphere [geosphere] – Calculate the Vincenty Sphere great circle distance.
- dlnorm – Return corresponding value of log normal PDF.
- dlogis – Return corresponding value of logistic PDF.
- dnbinom – Return corresponding value of negative binomial density.
- dnorm – Create standard normal distribution.
- do.call – Execute function by its name and a list of corresponding arguments.
- download.file – Download file from internet.
- dpois – Return value of poisson density.
- drop – Remove redundant dimension information.
- droplevels – Remove unused levels of a factor.
- dsignrank – Return corresponding PDF value of wilcoxon signedrank statistic.
- dt – Return corresponding value of Student t PDF.
- dtukey – Return corresponding value of studentized range PDF.
- dunif – Return corresponding value of uniform PDF.
- uplicated – Determine duplicate elements.
- dweibull – Return corresponding value of weibull density.
- dwilcox – Return corresponding PDF value of wilcoxon rank sum statistic.
- ecdf – Compute the Empirical Cumulative Distribution Function (ECDF).

- eval – Evaluate an expression and return the result.
- everything [tidyselect] – Select all variables of a data set.
- exists – Check whether an object is defined in the R environment.
- exp – Compute exponentials.
- expand.grid – Get all combinations of two variables.
- expression – Create an R object of the expression class.
- extract [tidyr] – Split the columns of a data set.
- file.choose – Interactively choose a file.
- file.copy – Copy files between directories.
- file.path – Construct the path to a file from components in a platform-independent way.
- file_path_sans_ext – Get file name without extension.
- filter [dplyr] – Subset rows by logical condition.
- find – Return location where objects of a given name can be found.
- first [dplyr] – Extract first value from vector.
- fivenum – Calculate the Tukey five-number summaries.
- floor – Round numeric down to the next lower integer.
- format – Encode data objects into common formats.
- formatC – Format numbers.
- full_join [dplyr] – Full join two data frames.
- gap.plot [plotrix] – Break axis of plot.
- gather [tidyr] – Reshape data frame from wide to long format in R.
- gc – Collect garbage to clean up memory.
- geom_bar – Create a barplot.
- geom_boxplot [ggplot2] – Create a boxplot.
- geom_curve [ggplot2] – Add curve to ggplot2 plot.
- geom_histogram [ggplot2] – Draw a histogram.
- geom_line [ggplot2] – Draw a line graph.
- geom_segment [ggplot2] – Add line segment to ggplot2 plot.
- geometric.mean [psych] – Compute the geometric mean.
- get – Search and call a data object.

- get0 – Call an existing data object or return an alternative value.
- getwd – Return filepath of the current working directory.
- ggpairs [GGally] – Return a matrix of plots based on a given data set.
- gl – Generate a factor variable.
- gregexpr – Search for match of certain character pattern.
- grep – Search for match of certain character pattern and return indices.
- grepl – Search for match of certain character pattern and return logical.
- grid – Add a grid to a Base R plot.
- gsub – Replace all matches in character string.
- harmonic.mean [psych] – Compute the harmonic mean.
- hasName – Check whether a data frame object contains a certain variable name.
- head – Return the first part of a data object.
- heatmap – Create heatmap in Base R.
- hist – Create histogram.
- identical – Test objects for exact equality.
- identity – Return a data object.
- if_else [dplyr] – Conditionally select an element.
- ifelse – Select elements conditionally.
- inner_join [dplyr] – Inner join two data frames.
- integrate – Integrate a user-defined function.
- interactive – Check if R runs interactively.
- intersect – Return the intersection of two data objects.
- invisible – Change the print mode to of a data object to invisible.
- IQR – Compute the Interquartile Range.
- is.data.frame – Test if object is a data frame.
- is.double – Test whether a data object has the double class.
- is.function – Test if an object is a function.
- is.na – Return a logical vector or matrix indicating which elements are missing.
- is.name – Test whether a data object has the class name.
- is.nan – Return a logical vector or matrix indicating which elements are not a number.

- is.null – Return a logical value indicating whether a data object is of the data class NULL.
- is.ordered – Test if data object has ordered factor class.
- is.primitive – Test if an object is a primitive function.
- is.tbl [dplyr] – Check if data object is a tibble.
- is.unsorted – Check whether an input is unsorted.
- ISOdate – Convert numeric values to date and time objects.
- ISOdatetime – Convert numeric values to date and time objects.
- isprime [matlab] – Check for prime number.
- jitter – Add noise to a numeric vector.
- julian – Return the number of days between two date objects.
- lag [dplyr] – Lag a vector.
- lapply – Apply function to all list elements.
- last [dplyr] – Extract last value from vector.
- last_col [tidyselect] – Select the last variable of a data set.
- layout – Specify complex plot arrangements.
- lead [dplyr] – Lead a vector.
- left_join [dplyr] – Left join two data frames.
- length – Return the length of data objects such as vectors or lists.
- libPath – Get or set directory path of installed packages.
- license – Get the license terms under which R is distributed.
- list.dirs – Obtain list of directories.
- list.files – List files with specific extension in working directory.
- load – Load RData workspace file into R.
- log – Compute logarithms of numeric values.
- log10 – Compute logarithms of numeric values with base of 10.
- log2 – Compute logarithms of numeric values with base of 2.
- lower.tri – Modify the lower triangular part of a matrix.
- lowess – Compute the lowess smoother.
- ls – Return the names of data objects in an environment.
- mad – Compute the median absolute deviation.

- make.names – Create syntactically valid names.
- mapply – Apply function to multiple data groups.
- match.call – Match arguments of a function.
- match.fun – Extract the syntax of a function.
- match – Return position of first match between two data objects.
- max – Compute the maximum value of a vector or column.
- mean – Compute the arithmetic mean.
- median – Compute the median.
- melt [reshape2] – Reshape data frame from wide to long format in R.
- merge – Merge data frames.
- message – Generate diagnostic message.
- mget – Search and call several data objects.
- min_rank [dplyr] – Rank a vector.
- min – Compute the minimum value of a vector or column.
- missing – Check if a value was set as an argument to a function.
- Mode [DescTools] – Compute the mode.
- months – Return the corresponding month of a date.
- mtext – Write text to the margins of a graphic.
- mutate [dplyr] – Add new variable to data set.
- n.readLines [reader] – Skip ahead in a file and read text lines.
- na.omit – Remove incomplete cases of a data object.
- na_if [dplyr] – Replace NA values.
- names – Return or set the names of a data object.
- nchar – Return the number of elements (e.g. letters) of a character object.
- ncol – Return the number of columns of a matrix or data frame.
- near [dplyr] – Compare numeric vectors.
- next – Skip certain parts of for-loop in R.
- norm – Compute the norm of a matrix.
- nrow – Return the number of rows of a matrix or data frame.
- nth [dplyr] – Extract certain values from vector.

- [ntile \[dplyr\]](#) – Rank a vector.
- [object.size](#) – Report the space allocated for an object.
- [objects](#) – Return the names of data objects in an environment.
- [OlsonNames](#) – Get a list of known time zones.
- [optim](#) – Apply general-purpose optimization.
- [optimize](#) – Perform one dimensional optimization.
- [order](#) – Return position of each input element in ascending or descending order.
- [ordered](#) – Create ordered factor.
- [outer](#) – Apply a function to two arrays.
- [packageVersion](#) – Find out the currently loaded version of an R package.
- [pairs](#) – Return a plot matrix consisting of scatterplots for each variable-combination of a data frame.
- [par](#) – Set or Query graphical parameters.
- [parse](#) – Convert a character to the expression class.
- [paste](#) – Concatenate inputs to string.
- [paste0](#) – Concatenate inputs to string with given separator.
- [pbern](#) – Return corresponding value of bernoulli CDF.
- [pbeta](#) – Return corresponding value of beta cumulative distribution function.
- [pbinom](#) – Return corresponding value of binomial cumulative distribution function.
- [pcauchy](#) – Return corresponding value of cauchy cumulative distribution function.
- [pchisq](#) – Return corresponding value of chi-square CDF.
- [percent \[formattable\]](#) – Format number as percentage.
- [percent \[scales\]](#) – Format number as percentage.
- [percent_rank \[dplyr\]](#) – Rank a vector.
- [permn \[combinat\]](#) – Generate all possible permutatins of vector object.
- [pexp](#) – Return corresponding value of exponential cumulative distribution function.
- [pf](#) – Return corresponding value of F CDF.
- [pgamma](#) – Return corresponding value of gamma cumulative distribution function.
- [pgeom](#) – Return corresponding value of geometric CDF.
- [phyper](#) – Return corresponding value of hypergeometric CDF.
- [pivot_longer \[tidyr\]](#) – Reshape data from wide to long format.

- `pivot_wider` [`tidyr`] – Reshape data from long to wide format.
- `plnorm` – Return corresponding value of log normal CDF.
- `plogis` – Return corresponding value of logistic CDF.
- `plot` – Draw a scatterplot or density plot.
- `pmatch` – Return position of first match between two data objects.
- `pmax` – Return the parallel maxima of two or more vectors.
- `pmin` – Return the parallel minima of two or more vectors.
- `pnbinom` – Return corresponding value of negative binomial cumulative distribution function.
- `pnorm` – Return value of distribution function.
- `polygon` – Draw a polygon to a plot.
- `ppois` – Return value of poisson cumulative distribution function.
- `pretty` – Compute a sequence of equally spaced round values.
- `print` – Return data object to the R (or RStudio) console.
- `prop.table` – Create a proportions table.
- `psignrank` – Return corresponding CDF value of wilcoxon signedrank statistic.
- `pt` – Return corresponding value of Student t CDF.
- `ptukey` – Return corresponding value of studentized range CDF.
- `pull` [`dplyr`] – Extract columns of data frame or tibble.
- `punif` – Return corresponding value of uniform CDF.
- `pweibull` – Return corresponding value of weibull CDF.
- `pwilcox` – Return corresponding CDF value of wilcoxon rank sum statistic.
- `qbern` – Return corresponding value of bernoulli quantile function.
- `qbeta` – Return corresponding value of beta quantile function.
- `qbinom` – Return corresponding value of binomial quantile function.
- `qcauchy` – Return corresponding value of cauchy quantile function.
- `qchisq` – Return corresponding value of chi-square quantile function.
- `qexp` – Return corresponding value of exponential quantile function.
- `qf` – Return corresponding value of F quantile function.
- `qgamma` – Return corresponding value of gamma quantile function.
- `qgeom` – Return corresponding value of geometric quantile function.

- qhyper – Return corresponding value of hypergeometric quantile function.
- qlnorm – Return corresponding value of log normal quantile function.
- qlogis – Return corresponding value of logistic quantile function.
- qnbinom – Return corresponding value of negative binomial quantile function.
- qnorm – Return value of quantile function.
- qpois – Return value of poisson quantile function.
- qqline – Add theoretical line to QQplot.
- qqnorm – Create a normal QQplot.
- qqplot – Create QQplot of two data sets.
- qr – Perform a QR matrix decomposition.
- qsignrank – Return corresponding quantile function value of wilcoxon signedrank statistic.
- qt – Return corresponding value of Student t quantile function.
- qtukey – Return corresponding value of studentized range quantile function.
- quantile – Compute sample quantiles.
- quarters – Return the corresponding quarter of a date.
- quit – Terminate an R session.
- qunif – Return corresponding value of uniform quantile function.
- qweibull – Return corresponding value of weibull quantile function.
- qwilcox – Return corresponding quantile function value of wilcoxon rank sum statistic.
- R.Version – Get current R version.
- range – Get the minimum and maximum of a data object.
- rank – Return ranking position of input.
- rbern – Return bernoulli distributed random number.
- rbeta – Draw random number from beta density.
- rbind.fill [plyr] – Row-bind matrices or data frames with different columns.
- rbind – Combine vectors, matrices and/or data frames by row.
- rbinom – Draw random number from binomial density.
- rcauchy – Draw random number from cauchy density.
- rchisq – Return chi-square distributed random number.
- read.xlsx [openxlsx Package] – Read Excel File in R.

- read.xlsx [xlsx Package] – Read Excel File in R.
- read_excel [readxl Package] – Read Excel File in R.
- readline – Interactively read a line from the terminal.
- readLines – Read text lines from an input file.
- readRDS – Load RData workspace file into R.
- recode [dplyr] – Replace certain values of a vector.
- recode_factor [dplyr] – Replace certain values of a factor vector.
- regexec – Search for match of certain character pattern.
- regexpr – Search for match of certain character pattern.
- remove – Clear specific data object from R workspace.
- rename [dplyr] – Rename variables of data frame or tibble.
- reorder – Reorder Levels of Factor.
- rep – Replicate elements of vectors and lists.
- replace – Exchanges values in data object.
- replicate – Perform a repeated evaluation of an expression.
- return – Return output of user-defined R function.
- rev – Return a reversed version of vectors or other data objects.
- rexp – Draw random number from exponential density.
- rf – Return F distributed random number.
- rgamma – Draw random number from gamma density.
- rgeom – Return geometrically distributed random number.
- rhyper – Return hypergeometrically distributed random number.
- right_join [dplyr] – Right join two data frames.
- rlnorm – Return log normally distributed random number.
- rlogis – Return logistically distributed random number.
- rm – Clear specific data object from R workspace.
- rnbinom – Draw random number from negative binomial density.
- rnorm – Draw normally distributed random number.
- rollmax [zoo] – Calculate moving maximum.
- rollmean [zoo] – Calculate moving average.

- rollmedian [zoo] – Calculate moving median.
- rollsum [zoo] – Calculate moving sum.
- round_any [plyr] – Round to multiple of any number.
- round – Round numeric to specified number of decimal places.
- row.names – Get or set row names of data frame or matrix.
- row_number [dplyr] – Rank a vector.
- row – Return row indices and labels.
- rowMeans – Compute the mean of each row of a numeric data frame, matrix or array.
- rowMedians [robustbase] – Compute the median of each row of a numeric matrix.
- rowSums – Compute the sum of each row of a numeric data frame, matrix or array.
- rpois – Draw random value of poisson density.
- rsignrank – Return random number according to wilcoxon signedrank statistic.
- rt – Return Student t distributed random number.
- rtukey – Return studentized range distributed random number.
- runif – Return uniformly distributed random number.
- rweibull – Draw random value of weibull density.
- rwilcox – Return random number according to wilcoxon rank sum statistic.
- sample.int – Draw set of random integers in range.
- sample_frac [dplyr] – Take a sample of a certain fraction of a data set.
- sample_n [dplyr] – Take a sample of N cases of a data set.
- sample – Take a random sample or permutation of a data object.
- sapply – Apply function to all list elements and convert output to vector.
- save.image – Save global R environment as RData workspace file.
- save – Save R data objects as RData workspace file.
- saveRDS – Save single R data object as RData workspace file.
- scale_colour_brewer [ggplot2] – Change color palette in ggplot2 plot.
- scale_fill_brewer [ggplot2] – Change palette of filling colors in ggplot2 plot.
- scale – Standardize data.
- scan – Read data into vector or list from file or R console.
- sd – Compute standard deviation.

- search – Return a list of attached packages and objects.
- searchpaths – Return a list of paths to attached packages and data objects.
- segments – Draw a line segment between two pairs of points.
- select [dplyr] – Select variables of data frame or tibble.
- semi_join [dplyr] – Semi join two data frames.
- seq – Generate sequence of numeric values.
- set.seed – Set a random seed.
- setdiff – Identify which elements of a data object X are not existent in a data object Y.
- setnames [data.table] – Change names of a data.frame or data.table by reference.
- setNames [stats] – Set names of a data object and return the object.
- setwd – Specify new working directory.
- shapiro.test – Perform Shapiro-Wilk normality test.
- sign – Return the signs of numeric elements.
- signif – Round numeric to specified number of digits.
- single – Create a single-precision vector.
- sink – Export R output to external file.
- slice [dplyr] – Extract certain rows of data set.
- solve – Solve system of equations.
- sort.int – Return input in ascending or descending order.
- sort.list – Return position of each element of a single input.
- sort – Return input in ascending or descending order.
- split – Divide data into groups.
- sprintf – Return character containing a formatted combination of input values.
- sqrt – Compute the square root of a numeric data object.
- stat_qq [ggplot2] – Create a QQplot.
- stat_qq_line [ggplot2] – Add theoretical line to QQplot.
- std.error [plotrix] – Compute standard error in R.
- stop – Generate error message and stop executing the current R code.
- stopifnot – Check if all expressions are TRUE.
- str_c [stringr] – Combine several inputs into a single character vector.

- `str_count [stringr]` – Count number of matches in a character string.
- `str_detect [stringr]` – Detect presence or absence of pattern in character string.
- `str_dup [stringr]` – Duplicate and concatenate strings within a character vector.
- `str_ends [stringr]` – Detect the presence or absence of a pattern at the end of a character string.
- `str_extract [stringr]` – Extract matching patterns from character string .
- `str_flatten [stringr]` – Flatten a character string.
- `str_length [stringr]` – Return the length of a string.
- `str_locate [stringr]` – Locate the first position of patterns in a character string.
- `str_locate_all [stringr]` – Locate the position of patterns in a character string.
- `str_match [stringr]` – Extract first matched groups from a string.
- `str_match_all [stringr]` – Extract matched groups from a string.
- `str_order [stringr]` – Order a vector of character strings.
- `str_remove [stringr]` – Remove first matched patterns in a character string.
- `str_remove_all [stringr]` – Remove all matched patterns in a character string.
- `str_replace [stringr]` – Replace first matched patterns in character strings.
- `str_replace_all [stringr]` – Replace all matched patterns in character strings.
- `str_replace_na [stringr]` – Replace NA values in a character string.
- `str_sort [stringr]` – Sort a vector of character strings.
- `str_split [stringr]` – Split up a string into pieces.
- `str_split_fixed [stringr]` – Split up a string into a fixed amount of pieces.
- `str_starts [stringr]` – Detect the presence or absence of a pattern at the beginning of a character string.
- `str_sub [stringr]` – Extract or replace substring from character string.
- `str_subset [stringr]` – Extract character strings matching a pattern.
- `str_trim [stringr]` – Trim whitespace from a character string.
- `str_trunc [stringr]` – Truncate character strings.
- `str_which [stringr]` – Identify positions in a character string matching a pattern.
- `str_wrap [stringr]` – Wrap a character string into nicely formatted paragraphs.
- `str` – Compactly display the structure of a data object.
- `strftime` – Convert time objects to characters.

- strptime – Convert characters to time objects.
- strsplit – Split elements of a character string.
- structure – Set additional attributes of a data object.
- sub – Replace first match in character string.
- substr – Extract or replace a substring in a character vector.
- substring – Extract or replace a substring in a character vector.
- sum – Compute the sum of a numeric input vector.
- summary – Compute summary statistics of data and model objects.
- suppressWarnings – Suppress warning messages temporarily.
- sweep – Apply an operation (e.g. + or -) to a matrix by row or column.
- switch – Run a specific code block and return its result.
- Sys.chmod – Change file permissions.
- Sys.Date – Return the current system date.
- Sys.getenv – Obtain the values of environment variables.
- Sys.getlocale – Query aspects of the locale.
- Sys.getpid – Get the process ID of an R session.
- Sys.glob – Conduct a wildcard expansion on file paths.
- Sys.info – Return system and user information.
- Sys.localeconv – Find details about the numerical and monetary representations in the current locale.
- Sys.setFileTime – Change the time of a file.
- Sys.setlocale – Set aspects of the locale.
- Sys.time – Return the current system time.
- Sys.timezone – Get the name of the current time zone.
- system.file – Find the names of system files.
- t – Transpose data frame.
- tail – Return the last part of a data object.
- tapply – Apply function to multiple data groups.
- tcrossprod – Calculate cross product of transpose of matrix.
- text – Draw text elements to plots.

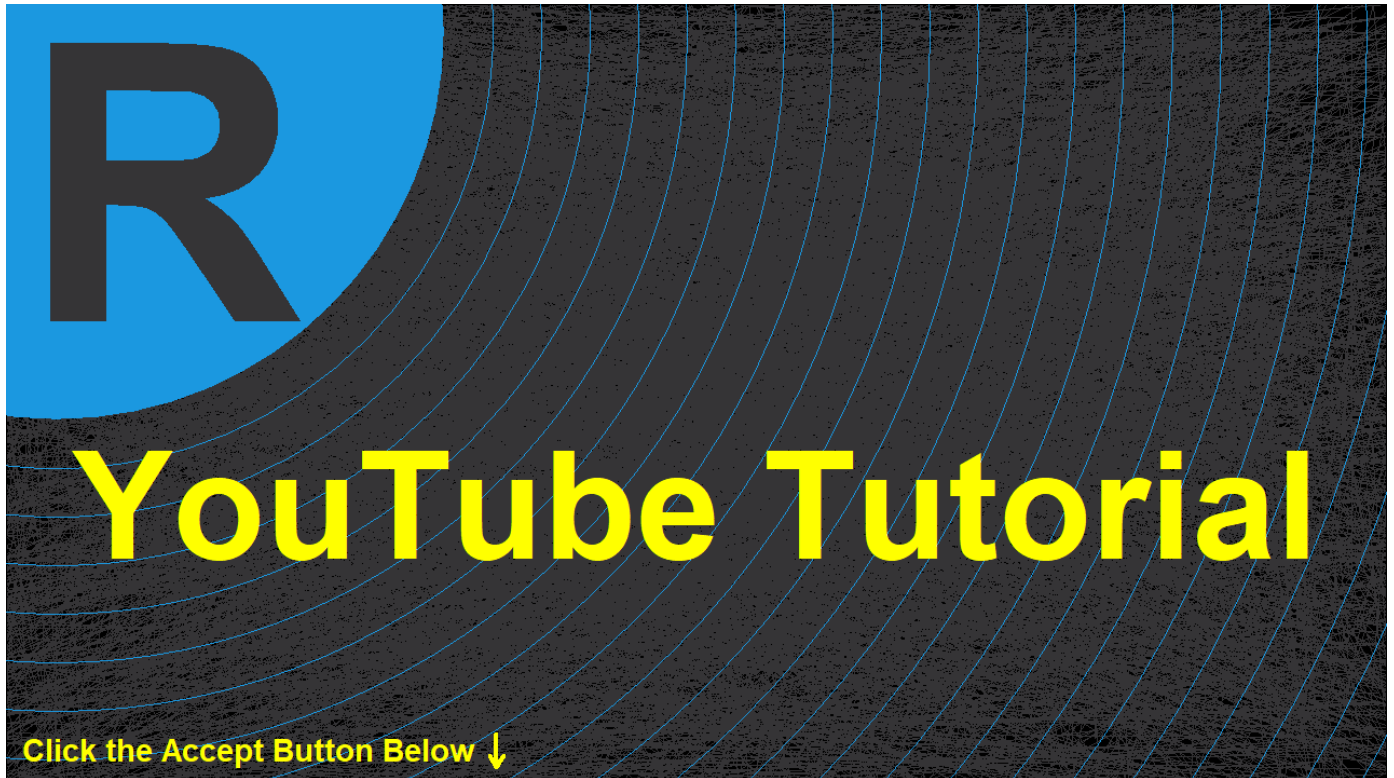
- textxy – Add text labels to a plot.
- theme_bw [ggplot2] – Change theme of ggplot2 Plot to theme_bw.
- theme_classic [ggplot2] – Change theme of ggplot2 Plot to theme_classic.
- theme_dark [ggplot2] – Change theme of ggplot2 Plot to theme_dark.
- theme_economist [ggthemes] – Change theme of ggplot2 Plot to theme_economist.
- theme_gray [ggplot2] – Change theme of ggplot2 Plot to theme_gray.
- theme_light [ggplot2] – Change theme of ggplot2 Plot to theme_light.
- theme_linedraw [ggplot2] – Change theme of ggplot2 Plot to theme_linedraw.
- theme_minimal [ggplot2] – Change theme of ggplot2 Plot to theme_minimal.
- theme_test [ggplot2] – Change theme of ggplot2 Plot to theme_test.
- theme_void [ggplot2] – Change theme of ggplot2 Plot to theme_void.
- tolower – Translate character to lower case.
- top_frac [dplyr] – Extract certain fraction at the top of data.
- top_n [dplyr] – Extract first n rows of data.
- toupper – Translate character to upper case.
- tracemem – Return memory address of data object.
- transform – Modify data frames easily.
- transmute [dplyr] – Create new variable based on data set.
- trimws – Trim Leading and Trailing Whitespace.
- trunc – Cut off decimal places of a numeric input.
- tryCatch – Check expressions for error and warning messages.
- type.convert – Convert data object to appropriate class.
- tz [lubridate] – Set and query the time zone of a date and time object.
- unclass – Manipulate Data Classes.
- union – Return values that appear in at least one of two data objects.
- unique – Remove duplicates from data objects.
- uniroot – Find a one dimensional root.
- unit [grid] – Create data object with the class unit.
- unlist – Convert a list to a single vector.
- unsplit – Reverse output of split function.

- upper.tri – Modify the lower triangular part of a matrix.
- url.show – Display a text URL.
- vapply – Apply function to all list elements and convert output to vector.
- var – Compute sample variance.
- View – Open a spreadsheet-style data viewer.
- warning – Generate warning message.
- weekdays – Return the corresponding weekday of a date.
- weighted.mean – Compute the weighted arithmetic mean.
- weightedMean [matrixStats] – Compute the weighted arithmetic mean.
- which – Return indices of certain values based on logical condition.
- which.max – Determine location of first maximum.
- which.min – Determine location of first minimum.
- window – Extract a time series subset.
- with – Evaluates expression in environment constructed based on data frame.
- within – Evaluates expression in environment and modify data frame.
- write – Write data to files.
- write_clip – Copy data to the clipboard.
- write.xlsx [xlsx] – Export data frame from R to Excel workbook.
- wt.mean [SDMTools] – Compute the weighted arithmetic mean.
- x11 – Start a graphics device driver for the X Window System.
- xtfrm – Produce numeric vector which sorts in same order as input.
- xyplot [lattice] – Create scatterplot.

Basic R Functions – Video Tutorial for the Most Common Commands

Are you looking for an overview of the most common R functions? Have a look at the following Video of the UTSSC YouTube channel. In the video, the speaker gives an introduction to a list of some basic R commands.

Please accept YouTube cookies to play this video. By accepting you will be accessing content from YouTube, a service provided by an external third party.



[YouTube privacy policy](#)

If you accept this notice, your choice will be saved and the page will refresh.

Accept YouTube Content

Subscribe to the Statistics Globe Newsletter

Get regular updates on the latest tutorials, offers & news at Statistics Globe.
I hate spam & you may opt out anytime: [Privacy Policy](#).

61 Comments. Leave new

**Kothandaraman Narasimhan**

October 30, 2019 3:37 pm

Hi Joachim, the compilation is getting better day by day. Small suggestion, in the list of R Commands & Functions can you please add Tidyverse grammar such as “pull” etc.

Thank you

Nara

Reply

**Joachim**

October 30, 2019 3:45 pm

Hey Nara,

That's great to hear, thank you! 😊 I am planning to do a series on tidyverse functions soon. I'll keep you updated on that.

Regards,

Joachim

Reply

**Joachim**

November 29, 2019 5:31 pm

Hey Nara,

As promised, I have produced a series of tutorials on the dplyr package. You can either search on this page for [dplyr] or you can check out this page: <https://statisticsglobe.com/dplyr-r-package>

Regards,

Joachim

Reply



Oreste Bruno

May 4, 2020 12:44 pm

Muchas gracias por todo el material que publican. Un gran trabajo. Felicitaciones

Reply



Joachim

May 4, 2020 1:39 pm

Muchas gracias Oreste! Nice to hear your positive feedback!

Reply



Ali Soltani

August 30, 2020 2:34 pm

Hi,

It would be nice to add the “subset” function to the list.

Reply



Joachim

August 31, 2020 7:10 am

Hi Ali,

Thank you for the suggestion!

I'm planning to do a series with new R-function tutorials starting this week. I'll also include a tutorial on the subset function to this series.

You may follow me on Twitter to get regular updates on the newest R-function tutorials:

<https://twitter.com/JoachimSchork>

Regards,

Joachim

Reply

**Cessar**

September 2, 2020 7:13 pm

This is great! Thank you. Could you include the switch command and information on how to write functions within functions lines of code (if not done already; I did not check for that)?

Reply

**Joachim**

September 3, 2020 6:19 am

Hey Cessar,

Thanks a lot for the nice feedback!

Actually, I have created a tutorial on the switch function just a few days ago and haven't added it to the list yet. You can find it here: <https://statisticsglobe.com/switch-function-in-r/>

Could you elaborate what you mean with "how to write functions within functions lines of code"? I'm not sure if I got what you are looking for. 😊

Regards,

Joachim

Reply

**Cristo**

October 20, 2020 8:12 pm

I am still wondering why anyone would take the effort to try to use an archaic programme like R. A command prompt? We are living in 2020 and not 1980. Why not use Excel? For anyone not used to typing commands and programming it is a disaster which leads to frustration also caused by thousands of add-ons which is the reason why copying commands in tutorials and examples does not work because you just haven't downloaded the right add-on.

Reply

**Joachim**

October 21, 2020 5:54 am

Hi Cristo,

Thank you for your comment and your thoughts.

In my opinion, there are numerous reasons for both R and Excel and I think discussing this here in detail would be beyond the scope of this article.

However, you can find many discussions that are highlighting the pros and cons of both programmes:

<https://www.northeastern.edu/graduate/blog/r-vs-excel/>

<https://www.rforexcelusers.com/excel-vs-r-when-to-use-what/>

<https://www.quora.com/What-really-differentiates-R-from-Excel>

Personally, I prefer R since I like to control everything my software is doing in a script. One thing I don't like in Excel is that it often formats data automatically without telling the user (e.g. numeric data to dates).

However, as I said I think both softwares have pros and cons and in the end this heavily depends on personal taste.

Regards,

Joachim

Reply



Denny Fernandez

February 15, 2021 2:34 pm

Hola Joachim. I just want to add on your comment to Cristo: you have to decide if you want to be limited on what to do, but in an easy way (Excel) versus doing what you really want to do, with some extra work in R (not much extra work thanks to people like Joachim). Best.

Reply



Joachim

February 15, 2021 4:12 pm

Hola Denny,

Thanks a lot for these additional thoughts. I definitely agree with what you said.

Also, thank you very much for the kind words! 😊

Regards,

Joachim

Reply



Rajesh

November 26, 2020 7:36 pm

Great tutorial thank you for sharing

Reply



Joachim

November 27, 2020 6:49 am

Thank you for the kind words Rajesh!

Reply



Simon

November 27, 2020 5:25 pm

Hi!

This side is very nice and helpful! Really good job, specially on the examples.

Maybe you can link the functions. F.e. when you use the mean function in your aggregate example you can link it – so it gets easy to click through your list. By now this is not a big problem but it might get if you include others from different packages. It also might create a more connected learning.

But nevertheless really love your page!!

Greetz

Simon

Reply



Joachim

November 28, 2020 8:03 am

Hey Simon,

Thanks a lot for the kind words and the feedback!

In fact, I already try to include links to other functions in case I'm referring to them in a tutorial. It is sometimes difficult to add links to older tutorials when publishing a tutorial about a new function

though.

I have added a link to the mean function within the text of the aggregate tutorial. However, please let me know in case you spot other missing links! 😊

Thanks again!

Joachim

Reply



Vijeyachander

February 18, 2021 8:05 am

Hey Joachim, Very useful content.

Have one query, is there any library or package where we can perform goodness of fit test and estimates its parameter for the following distribution

List of distribution

1. Two Parameter Exponential
2. Two Parameter Weibul
3. Smallest Extreme Value
4. 3 Parameter Gamma
5. Logistic
6. Log Logistic

Reply



Joachim

February 18, 2021 8:32 am

Hey Vijeyachander,

Thank you very much for the kind words! 😊

I'm not an expert for this topic, but maybe the EWGoF package helps:

<https://rdrr.io/cran/EWGoF/man/EWGoF-package.html>

Regards,

Joachim

Reply



Juho Kopra

February 19, 2021 5:52 am

This appears as a useful listing. Could it be organised to subgroups? Like data manipulation functions in the same group, probability in one group etc.

Reply



Joachim

February 19, 2021 7:07 am

Hi Juho,

Thanks a lot for the feedback! This is something I have planned for the future, and I hope I can realize it soon.

Regards

Joachim

Reply



Lewis Mitchell

March 23, 2021 4:09 pm

Wow! This is amazing! I just started an R Class for a Business Degree, having no knowledge of computers, and was so confused. Thank you so much Joachim! May your life be blessed!

Reply



Joachim

March 23, 2021 4:12 pm

Thanks a lot for this awesome feedback Lewis! I'm very glad to hear that you like this list! 😊

Regards

Joachim

Reply

**Lawrence**

April 30, 2021 1:45 am

It's amazing

Reply

**Joachim**

April 30, 2021 5:44 am

Thanks Lawrence! 😊

Reply

**Tulsi**

May 2, 2021 5:29 pm

Well written. 👍

Reply

**Joachim**

May 3, 2021 8:18 am

Thanks a lot Tulsi, glad to hear that! 😊

Reply

**Angus McLean**

May 23, 2021 5:57 pm

The use of class() is useful to me when establishing the type of variable we e.g. data frame so is it eligible to be included? The list does look to be useful. Thank you for this accomplishment.

Reply

**Joachim**

May 25, 2021 6:46 am

Hey Angus,

Indeed, the class function should be on this list. Thanks for the hint, I will add it soon.

Regards

Joachim

Reply



Craig

June 11, 2021 4:27 am

Hey Joachim,

I am writing a package for r and possibly python and found this list in my research. I realize this is an older thread, but you seem to be active still. The package is pretty simple and I would love to get your input on it.

Reply



Joachim

June 11, 2021 7:23 am

Hey Craig,

I have never developed a package myself, so I'm not sure whether I'm the right person to ask. However, do you have any specific questions?

Regards

Joachim

Reply



wakkae Ali

June 12, 2021 10:25 am

Hello..... How is it possible to find the oblateness and skewness of the Weibull distribution and the Beta distribution

Reply



Joachim

June 15, 2021 3:18 pm

Hey Wakkae,

Please have a look at this tutorial: <https://statisticsglobe.com/calculate-skewness-kurtosis-in-r>

Regards

Joachim

Reply



Hans

August 5, 2021 10:15 am

Hey Joachim,

What an amazing website you have created, I find it very helpful! However I can't seem to find the method to sum different elements in a large data frame. For example, this particular data frame has 500 rows and 5 columns, then I would like to find the sum of row 1 column 6, row 6 column 6, and row 11 column 6 elements. How do you think should I this?

Thanks in advance for your reply!

Regards,

Hans

Reply



Joachim

August 5, 2021 11:31 am

Hey Hans,

Thanks a lot for this amazing feedback! I'm really glad you like my content! 😊

Would the following code do what you want?

```
sum(your_data[c(1, 6, 11), 6])
```

Regards

Joachim

Reply



Kellie

October 22, 2021 4:16 pm

Greetings. I am not sure if this is the proper venue for this question but hope you can help. I am taking a data analytics class where we are required to:

"Explore a command (or function) that describes sampling distributions in R that is NOT listed in the Handout". Can you help me find a list of R functions that would be used to describe sampling distributions? It is hard to explore commands if I don't know what commands are available.

I appreciate your advice.

Reply



Joachim

October 25, 2021 9:31 am

Hey Kellie,

This place is absolutely fine to ask such questions! 😊

I assume you are looking for this list? <https://statisticsglobe.com/probability-distributions-in-r>

Regards,
Joachim

Reply



Jason

October 24, 2021 7:36 am

I am a beginner, and will benefit from this ready reference of available functions. As I progress, I expect to benefit even more from this page. Thank you.

Reply



Joachim

October 25, 2021 10:08 am

Thanks a lot for this positive feedback Jason! Glad to hear that you find it useful!

Reply



Yordan

December 30, 2021 6:30 pm

Hey Joachim, happy holidays first of all

i have a given dataframe and i want to replace its rows by the string characters of state.abb which is given

in R. (you can type it)

`row.names(my.data.frame)=state.abb` it says invalid row.names length. Also my data frame rows are regular names and the ones in `state.abb` have "VT" string.

I'm really grateful for any help

Best regards,

Y

Reply



Joachim

January 4, 2022 7:48 am

Hey Yordan,

Thank you, happy new year and happy holidays to you as well! 😊

If you want to replace row names or column values by new values (i.e. `state.abb`), the number of values in your data and in your vector of new values have to be the same.

You may replace the `row.names` of your data by `state.abb` using the following code. I don't know if this makes sense in your case, though.

```
row.names(my_data) <- state.abb[1:nrow(my_data)]
```

Regards,

Joachim

Reply



MM

January 14, 2022 6:15 pm

Hi

Do you have a list where these functions are organised by use case? and is it possible to assemble a collection of functions for processing pipelines, say for a general category like time-series analyses, probability etc? I am struggling to recall specific functions for certain tasks (in a hurry!). I wonder if organising by general use case if this can improve my retention!

thanks

MM

Reply

**Joachim**

January 14, 2022 6:18 pm

Hey,

Thanks for the feedback and suggestions!

I'm planning to create certain overview articles that will categorize those tutorials/functions. But I haven't created these articles yet.

I'll keep you updated.

Joachim

Reply

**Jairon Leal**

March 22, 2022 2:43 am

Hey, guys

How is it possible to find the scale and shape parameters of the Weibull distribution based on Maximum Likelihood estimation in R?

Reply

**Joachim**

March 22, 2022 7:21 am

Hey Jairon,

Unfortunately, I'm not an expert on this topic. However, I have recently created a Facebook discussion group where people can ask questions about R programming and statistics. Could you post your question there? This way, others can contribute/read as well:

<https://www.facebook.com/groups/statisticsglobe>

Regards,

Joachim

Reply

Awais Anwar



April 22, 2022 4:05 pm

Hello Joachim

I am learning RStudio and dont know anything about R Programming Language. Do I need R to master RStudios?

Reply

**Joachim**

April 22, 2022 5:47 pm

Hey Awais,

RStudio is an IDE for the R programming language. In other words: You will write your R code in the RStudio interface.

I recommend [having a look here](#) to learn how to install R and RStudio.

Once you have installed all software, you may have a look at the [introduction to R course](#) on Statistics Globe.

Regards,

Joachim

Reply

**Katrin**

May 3, 2022 12:08 pm

Thanks for the effort and the useful explanations on this website!

I miss an explanation of the calculation of the risk ratio (e.g. "riskratio") and the odds ratio (e.g. "oddsratio") via for example the "epitools" library. In particular, I am interested in learning how to properly organize the data and how to calculate ratios of multiple independent groups within one calculation approach. Last but not least, I would like to plot this data in a forest plot. Is there already a tutorial on this and I'm just not finding it?

Greetings,

Katrin

Reply

**Joachim**

May 3, 2022 1:20 pm

Hey Katrin,

Thank you very much for the kind comments, glad you find my tutorials useful!

Also, thanks a lot for the topic suggestions! I don't have tutorials on these topics yet, but this might be interesting for future tutorials. I have noted it on my to-do list.

Regards,

Joachim

Reply

**gyanu**

May 27, 2022 4:31 pm

Hi Joachim

I have posted one problem that I'm facing while using carrplot in R Programming & Data Science (Moderated by Statistics Globe). I have personally messaged you in detail. I look forward to receiving your help.

Reply

**Joachim**

June 6, 2022 4:23 pm

Hi Gyanu,

Please see my comments in the Facebook Group.

Regards,

Joachim

Reply

**Ken Denike**

August 22, 2022 10:43 pm

having difficulty in R getting formula = $n*(n+1)*(2n+1)/6$ which is standard formula for sum x^2 .

Reply

**Joachim**

August 23, 2022 6:14 am

Hey Ken,

Could you please specify your question in some more detail?

Regards,

Joachim

Reply

**Ken Denike**

August 23, 2022 3:44 pm

> formula = $n*(n+1)/2$ works for s_n of x > formula = $n*(n+1)*(2n+1)/6$ doesn't for s_n of X^2 Error: unexpected symbol in "formula = $n*(n+1)*(2n$ "

Reply

**Joachim**

August 23, 2022 5:41 pm

I think you need to add a * symbol between $2n$. I.e.:

$n*(n+1)*(2*n+1)/6$

Regards,

Joachim

Reply

**SAHR EDWARD JAMES**

September 4, 2022 2:00 am

Joachim

you are the best teacher you lesson, and content are very useful to me since i start using r for my data Analytics skills

thanks so much

Reply

**Joachim**

September 5, 2022 11:09 am

Hey,

Thank you so much for the kind words, it's great to hear that you find my tutorials helpful!

Regards,

Joachim

Reply

**SAHR EDWARD JAMES**

September 4, 2022 2:02 am

i think i will recommend your channel to my friends here in Sierra Leone because for me you have the best lessons so far for R programming
once More thanks again.

Reply

**Joachim**

September 5, 2022 11:10 am

Thank you once again, and thanks a lot for sharing Statistics Globe with your friends! 😊

Greetings from Germany to Sierra Leone!

Joachim

Reply

**Mimansa**

October 9, 2022 7:34 am

Hi,

can you write a program for above ? determine how many functions are there in Base R by write a code?

Reply

**Joachim**

November 14, 2022 11:56 am

Hey Mimansa,

Please excuse the late response. I was on a long holiday so unfortunately I wasn't able to reply sooner. Still need help with your code?

Regards,
Joachim

Reply

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment *

Name *

Email *

Post Comment

AN ELITE CAFEMEDIA PUBLISHER