

Basic Math

<code>exp(x)</code>	Exponential	<code>sum(x)</code>	Sum
<code>log(x)</code>	Natural log	<code>cumsum(x)</code>	Cumulative Sum
<code>max(x)</code>	Largest element	<code>ceil(x)</code>	Round up
<code>min(x)</code>	Smallest element	<code>floor(x)</code>	Round down
<code>x %% y</code>	Modulo		

Control Flow

<code>for (variable in sequence) {...}</code>	for-loop. If the loop body contains only a single line, the curly brackets can be omitted.
<code>while (condition) {...}</code>	while-loop
<code>if (i > 5) { ... else { ... }</code>	if-else-block
<code>foo = function(arg1, arg2, ...) { ... return (var) }</code>	function

Vectors

Creating Vectors

<code>c(2, 4, 6)</code>	Join elements into a vector
<code>2:6</code>	An integer sequence (end inclusive!)
<code>seq(2,3, by=0.5)</code>	Complex sequence (s. <code>np.linspace</code>)
<code>rep(1:2, 3)</code>	Repeat vector
<code>rep(1:2, 3:4)</code>	Repeat each element

Functions

<code>sort(x)</code>	Return x sorted.
<code>rev(x)</code>	Return x reversed.
<code>unique(x)</code>	See unique values.
<code>length(x)</code>	Length of x.

Selecting Vector Elements



By BarplotNorm

cheatography.com/barplotnorm/

Not published yet.

Last updated 31st January, 2025.

Page 1 of 9.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>

Vectors (cont)

By Position

<code>x[4]</code>	The fourth element
<code>x[-4]</code>	All but the fourth.
<code>x[2:4]</code>	Elements two to four
<code>x[-(2:4)]</code>	All elements except 2 to four
<code>x[c(1, 5)]</code>	Elements one and five.

By Value

<code>x[x == 10]</code>	All elements equal to 10
<code>x[x < 10]</code>	All elements less than 10.
<code>x[x %in% c(1, 2, 5)]</code>	Elements in the given set.

Named Vectors

<code>x['apple']</code>	Element with name 'apple'.
-------------------------	----------------------------

Tables

<code>table(data)</code>	get absolute frequencies of values
<code>as.num eri c(tab); as.vec tor (tab)</code>	Extract values and their absolute frequencies from table
<code>tab/le ngth(data)</code>	Compute relative frequencies

Matrices

<code>m = matrix(x, nrow = 3, ncol = 3)</code>	Create a matrix from vector x
<code>t(m)</code>	transpose
<code>m %*% n</code>	Matrix multiplication
<code>solve(m, n)</code>	Find x in $m * x = n$
<code>det(m)</code>	Determinant
<code>eigen(m)</code>	Find eigen vectors and values

Data sets

<code>Data=data.frame(price=c(11,20,14,15), number=c(40,50,60,20))</code>	Create a data set
---	-------------------

Interacting with data sets

<code>col_1 = data\$col_1_name</code>	Access column data
<code>Data[1,2]; Data[,2]; Data[[1]]</code>	Access data with index notation

I/O



By BarplotNorm

cheatography.com/barplotnorm/

Not published yet.

Last updated 31st January, 2025.

Page 2 of 9.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopadd.com>

Data sets (cont)

`data = read.csv("file.csv", header = FALSE, sep="")` Read csv (function arguments similar to that used in pandas)

`write.csv(data, "data.csv", row.names = FALSE, sep="")` Write data set as csv

Filter

`df[df$kids == "Jack",]` Filter data frame

`subset(df, kids == "John" & Grade == 1.3)` Filter multiple columns

`subset(df, kids %in% c("Jack", "John"))` Filter a column with multiple values

`unique(housing[, c("State", "region")])` Extract unique rows

Sort

`housing[order(housing$HomeValue),]` Order data frame in ascending order

`housing[order(housing$HomeValue, decreasing = TRUE),]` Order data frame in descending order

Meta

`dim(df)` Check the dimensions of a data frame

`colnames(d)` Return the column names

Manipulate data

`DataFrameNew <- DataFrame[-c(1), -c(1)]` Remove columns and/or rows from data frame

`rbind(df_1, df_2)` Combine data frames vertically

I/O

`write(data, "mydata.dat")` Write data as binary.

`scan("mydata.dat")` Read binary data.

`getcwd()` Current working directory

Random Numbers

`sample(1:3, prob = c(1/6, 1/3, 1/2), replace = TRUE, 20)` Draw 20 balls, labeled from 1 to 3, from box with replacement.

`r<distr.ID>(n, params)` Draw n numbers from distribution <distr.ID> with parameters params

(see *Distributions in R* for more details)



By BarplotNorm

cheatography.com/barplotnorm/

Not published yet.

Last updated 31st January, 2025.

Page 3 of 9.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish

Yours!

<https://apollopadd.com>

Characteristics of data sequences

<code>mean(x)</code>	Arithmetic mean of the data sequence
<code>var(x)</code>	Variance
<code>median(x)</code>	Median
<code>quantile(x, type=7)</code>	Quantile. <code>type=7</code> is the default computation algorithm, i.e. the function returns the value at position $k=1+p(n-1)$, if this is an integer. Otherwise, R computes a weighted mean of the two neighboring integers
<code>quantile(x, type=1)</code>	General inverse function of the ECDF (smallest p-quantile). Largest p-quantile can be obtained indirectly by slightly increasing p
<code>summary(x)</code>	Overview of important measures
<code>cov(x, y)</code>	Covariance
<code>cor(x, y)</code>	Correlation

Distributions in R

General usage

<code>d<distr.ID>(params)</code>	density function
<code>q<distr.ID>(params)</code>	quantile function. Always computes the smallest quantile
<code>p<distr.ID>(params)</code>	cumulative distribution function
<code>r<distr.ID>(params)</code>	random variate generation

Distributions

<code>dbinom(x, size=p, prob=p)</code>	Binomial
<code>dchisq(x, df, ncp=0)</code>	Chi-squared
<code>dexp(x, rate=1)</code>	Exponential
<code>dgamma(x, shape=r, rate=1)</code>	Gamma
<code>dgeom(x, prob=p)</code>	Geometric
<code>dnbinom(x, size, prob)</code>	Negative binomial
<code>dnorm(x, mean=0, sd=1)</code>	Normal
<code>dpois(x, lambda)</code>	Poisson
<code>dt(x, df, ncp)</code>	t-distribution
<code>dunif(x, min=0, max=1)</code>	Uniform



By BarplotNorm

cheatography.com/barplotnorm/

Not published yet.

Last updated 31st January, 2025.

Page 4 of 9.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>

Plotting

Basic plots

<code>plot(data)</code>	Plot quick overview.
<code>plot(x, y, xlab="m u", ylab="P owe r", type="l ", col="re d", ylim=c (0,1), lwd=1.5)</code>	Plot data with custom style options

Lines and curves

<code>abline (a, b,c ol= " red ")</code>	Add a red line with intercept a and slope b to the plot.
<code>abline (v= a,c ol= " red ")</code>	add vertical line at x=a
<code>abline (h= b,c ol= " red ")</code>	add horizontal line at y=b
<code>lines(x, y, col="gr een ", lwd=1.5)</code>	Add a generic line
<code>curve(sin ,-p i,p i,a dd= TRUE)</code>	Draw a curve of a function over the specified interval

Data visualization

<code>plot.e cdf (data)</code>	Plot ECDF.
<code>barplot(x, main="T itl e", xlab="x label")</code>	Annotated barplot of absolute frequencies
<code>hist(data, prob=TRUE, breaks=30)</code>	Histogram of relative frequencies (30 bins).
<code>rug(data)</code>	1D-plot
<code>boxplo t(d ata1, data2, ... ,range =1.5)</code>	Plot boxplots of one or more data sequences in one window. range determines the extend of the whiskers. Default range=1.5, i.e. 1.5 x IQR
<code>qqnorm(x)</code>	QQ-Plot against standard normal distribution
<code>qqPlot(x, dist="u nif ", ...)</code>	QQ-Plot against any R-standard distribution. Additional arguments such as df, ncp can also be specified.
<code>legend (x,y, legend =c(" n=1 0"), col =c(" red "), lty=1, cex=0.8)</code>	Add legend to plot as position (x, y)

Statistical hypothesis testing

One-Sample tests

<code>t.test (x, mu= mu0 ,al t="l ess ", conf.l eve l=1 -alph a)</code>	Performs one and two sample t-tests on vectors of data.
---	---



By BarplotNorm

Not published yet.

Last updated 31st January, 2025.

Page 5 of 9.

Sponsored by [ApolloPad.com](https://apollopad.com)

Everyone has a novel in them. Finish

Yours!

<https://apollopad.com>

Statistical hypothesis testing (cont)

<pre>power.t.test(n = 100, delta=0.1, sd=2, sig.level=0.1, type="one.sided", alt="one.sided")</pre>	Compute the power of the one- or two-sample t test, or determine parameters to obtain a target power.
---	--

<pre>binom.test(sum(x), n, p0, alt="greater", conf.level=1-alpha)</pre>	Performs an exact test of a simple null hypothesis about the probability of success in a Bernoulli experiment. It might happen that the decision based on the p-value differs from that of the confidence interval. Choose the decision based on the p-value in such cases.
---	--

Two-Sample tests

<pre>t.test(x1, x2, paired=FALSE, var.equal=TRUE)</pre>	Example for an unpaired sample t-test
---	--

<pre>var.test(x, y, conf.level=1-alpha)</pre>	Performs an F test to compare the variances of two samples from normal populations.
---	--

GOF tests

<pre>shapiro.test(x)</pre>	Performs the Shapiro-Wilk test of normality.
----------------------------	---

<pre>chisq.test(table(x), p=p0)</pre>	Test for distribution with probabilities p_0. If p is not specified, R tests for a uniform distribution
---------------------------------------	--

<pre>chisq.test(table(x), p=p0, simulate.p.value=TRUE)</pre>	Do not use χ^2-approximation to calculate the p-value
--	--

```
pwr.chisq.test(w=ncp, df=s-1, sig.level=alpha, power=0.9)
```

Determine the number of samples needed to reach the desired power at the given significance level

```
ks.test(x, "pnorm", 0, 1)
```

One-sample Kolmogorov-Smirnov test against hypothetical distribution

```
lillie.test(x)
```

Lilliefors (Kolmogorov-Smirnov) test for the composite hypothesis of normality

Tests of independence

```
chisq.test(M)
```

Chi²-test of independence. M has to be a matrix! (contingency table)



By BarplotNorm

cheatography.com/barplotnorm/

Not published yet.

Last updated 31st January, 2025.

Page 6 of 9.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>

Statistical hypothesis testing (cont)

<code>fisher.test(M)</code>	Exact test of Fisher. If the table entries are too large, use <code>simulate.p.value=TRUE</code>
<code>runs.test(x)</code>	Runs test of independence. <code>x</code> has to be a factor (use <code>as.factor()</code> if necessary)

Runs Test of Randomness

<code>rle(x)</code>	Compute the lengths and values of runs of equal values in a vector.
<code>rle(x)\$lengths</code>	Vector containing the length of each run.
<code>rle(x)\$values</code>	Vector of the same length as <code>lengths</code> with the corresponding values.

Optimization

<code>nlm(E2, 0.5)</code>	Carries out a minimization of the function <code>f</code> using a Newton-type algorithm. May not give all solutions. The function must be vectorized
<code>E2vec= Vectorize(E2, vectorize.args=c("n"))</code>	vectorize a function. <code>vectorize.args</code> : explicitly state arguments to be vectorized.

Distribution Fit

<code>fitdistr(x, "Poisson")</code>	Maximum-likelihood fitting of univariate distributions, allowing parameters to be held fixed if desired. (<code>library(MASS)</code>)
-------------------------------------	---

Regression

<code>reg=lm(x~t)</code>	Fit a linear function $x=a+bt$
<code>summary(reg)</code>	Obtain further information about regression result Important fields: - Residual standard error: sd of residuals (with normalization $n-2$) - t value: Test null hypothesis "estimate is 0" with assumption of a normally distributed random mechanism - multiple R-squared: squared corr. coef. Null hypothesis $r^2=0$ is tested with F-statistic



By BarplotNorm

cheatography.com/barplotnorm/

Not published yet.

Last updated 31st January, 2025.

Page 7 of 9.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>

Optimization (cont)

<code>reg=lm(x ~ t+ I(t^2))</code>	Fit a polynomial function. <code>I()</code> inhibits R from interpreting <code>t^2</code> as a formula
<code>form=x ~ a/(1+exp(-b*(t-c)))</code> <code>reg=nls(form, data=USPop, start= c(a =40 0,b =0.0 2, -</code> <code>c=2 000))</code>	perform non-linear least-squares regression
<code>plot(t, predict(reg))</code>	Plot regression result

Root finding

<code>res = uniroot(f, c(0,10))</code>	Searches interval for a root of the function <code>f</code> . <code>res\$root</code> and <code>res\$f.root</code> give the location of the root and the value of the function
--	---

Help

<code>?sqrt</code>	Display documentation of the command <code>sqrt</code>
<code>?'%%'</code>	use quotation marks for special characters

Miscellaneous

Printing

<code>print("Text")</code>	Default print
<code>sprintf("Formatted %s: %.3f", object, mean)</code>	Formatted print
<code>(x=3)</code>	enclose an R command with brackets to directly print the result
<code>edit(x)</code>	Invoke text editor on R object

Libraries

<code>library(MASS)</code>	Load package MASS
<code>uniroot(f, interval)</code>	find 1D root

Step functions

<code>stepfun(x, y)</code>	Given the vectors $(x_1, \dots, x_n)$ and $(y_0, \dots, y_n)$ (one value more!), returns an interpolating 'step' function
<code>knots(x)</code>	returns jump positions of stepfunction



By BarplotNorm

cheatography.com/barplotnorm/

Not published yet.

Last updated 31st January, 2025.

Page 8 of 9.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>