One sample confidence intervals/ tests

Mean (known variance) *No function – do it from 1st principles.*

Mean (unknown variance)

CI `t.test(x, alternative = "two.sided", conf.level = 0.95)`
Test `t.test(< data >, mu = 0, alternative = "two.sided", conf.level = 0.95)`
mu is value of mu in null hypothesis, default = 0

Variance *No function – do it from 1st principles,*

Binomial
CI `binom.test(< x >, < n >, alternative = "two.sided", conf.level = 0.95)`
Test `binom.test(x, n, p = 0.5, alternative = "two.sided", conf.level = 0.95)`
x is number of successes (or a vector of successes and trials)
n is number of trials (not needed if included in x)
p is the value of p in the null hypothesis, default = 0.5

Poisson
CI `poisson.test(x, T = 1, alternative = "two.sided", conf.level = 0.95)`
Test `poisson.test(x, T = 1, r = 1, alternative = "two.sided", conf.level = 0.95)`
x is number of events
T is time base for events that occurred, default = 1
r is the value of the rate (lambda) in the null hypothesis, default = 1

Two sample confidence intervals/tests

Mean (known variance) *No function – do it from 1st principles*

Means (unknown variance)

CI `t.test(< data1 >, < data2 >, alt = "two.sided", var.equal = FALSE,`
`conf = 0.95)`
Test `t.test(< data1 >, < data2 >, mu = 0, alt = "two.sided", var.equal = FALSE, conf = 0.95)`

mu is value of $\mu_1 - \mu_2$ *in the null hypothesis, default = 0*
var.equal is the option for whether we assume the variances are equal, default = FALSE

Means (paired data)
`t.test(< after >, < before >, mu = 0, alt = "two.sided", conf = 0.95, paired = TRUE)`

Variances ratio `var(< data1 >)/var(< data2 >)`
CI `var.test(< data1 >, < data2 >, alt = "two.sided", conf = 0.95)`
Test `var.test(< data1 >, < data2 >, ratio = 1, alt = "two.sided", conf = 0.95)`
ratio is value of ratio `var(< data1 >)/var(< data2 >)` *in null hypothesis, default = 1*

Binomial

CI prop.test(< successes >, < trials >, conf.level = 0.95, correct = TRUE)
x is number of successes (or a vector of successes and trials)
n is number of trials (not needed if included in x)
or matrix of successes and failures (not trials)
Test prop.test(x, n, p = NULL, alternative = "two.sided", conf.level = 0.95, correct = TRUE)
p is value of $p_1 - p_2$ in null hypothesis, default = 0

Poisson RATIO (not difference)

CI poisson.test(x, T = 1, conf.level = 0.95)
x is vector of events
T is vector of time base for events that occurred, default = 1
Test poisson.test(x, n, r = 1, alt = "two.sided", conf.level = 0.95)
r is value of lambda ratios in null hypothesis, default = 1

Contingency Table
Given matrix of observed frequencies:
chisq.test(< obs freq matrix >)
Note for a 2×2 it applies Yates continuity correction by default, to remove:
chisq.test(< obs freq matrix >, correct = FALSE)

Chi − squared goodness of fit tests
Goodness of fit
Given expected probabilities:
chisq.test(< obs freq >, p =< exptd probabilities >)
Default exptd probabilities are uniform
Note: p = is important or thinks doing contingency table with 2 vectors