Statistical tests with rstatix :: **CHEAT SHEET**

rstatix runs the everyday tests of base R with one pipe-friendly grammar and returns tidy data frames, ready to report and plot.

1 Choose the test

The **design** picks the row; the **assumptions** (section 3) pick the column. Grey: effect size or key option.

	ASSUMPTIONS MET	NOT MET / SMALL n
2 independent groups	`t_test()`	`wilcox_test()`
	`cohens_d()`	`wilcox_effsize()`
2 paired groups `+ paired = TRUE, id =`	`t_test()`	`wilcox_test()`
	`cohens_d()`	`wilcox_effsize()`
3+ independent groups	`anova_test()`	`kruskal_test()`
	ges column	`kruskal_effsize()`
post hoc pairs	`tukey_hsd()`	`dunn_test()`
3+ repeated `+ wid = , within =`	`anova_test()`	`friedman_test()`
	ges column	`friedman_effsize()`
2 numeric variables	`cor_test()`	`cor_test()`
	method = "pearson"	*method = "spearman"*
2 categorical variables	`chisq_test()`	`fisher_test()`
	`cramer_v()`	*expected counts < 5*
paired categorical	`mcnemar_test()`	

TIP `t_test()` is a **Welch** test by default (var.equal = FALSE). With 3+ groups and unequal variances use `welch_anova_test()` + `games_howell_test()`.

MULTIPLE TESTING `p.adjust.method =`

`"holm"` default for pairwise tests

`"bonferroni"` simplest, most conservative

`"BH"` false discovery rate

`"none"` only for one planned test

2 One grammar for all

```
library(rstatix)
df <- ToothGrowth   # len, supp, dose
df$dose <- factor(df$dose)
```

```
df %>% t_test(len ~ supp)
```

- `df %>%` — data comes first → pipes
- `t_test` — test function
- `len` — outcome
- `supp` — group

.y.	group1	group2	n1	n2	statistic	df	p
len	OJ	VC	30	30	1.92	55.3	**0.061**

A tibble, never a printout: filter, join, export or format it like any table.

BY SUBGROUP → ADJUST → STARS

```
df %>% group_by(dose) %>%
  t_test(len ~ supp) %>%
  adjust_pvalue(method = "bonferroni") %>%
  add_significance()
```

dose	p	p.adj	p.adj.signif
0.5	0.006	0.019	*
1	0.001	0.003	**
2	0.964	1.000	ns

Stars: **** < 0.0001 · *** < 0.001 · ** < 0.01 · * < 0.05 · ns

WHICH COMPARISONS?

`len ~ dose` all pairs (Holm-adjusted)

`ref.group = "0.5"` each level vs a reference

`comparisons = list(c("0.5", "2"))` only the pairs you planned

`effect.size = TRUE` adds Cohen's d: 0.49, small

`detailed = TRUE` adds estimate, 95% CI, method: OJ − VC = 3.7 (−0.17 to 7.57)

REPORT IN TEXT

```
res <- df %>% t_test(len ~ supp)
get_test_label(res, detailed = TRUE,
               type = "text")
```

→ **T test, t(55.31) = 1.92, p = 0.061, n = 60**

The default type returns a plot subtitle.

3 Check assumptions

NORMALITY (SHAPIRO–WILK)

```
df %>% group_by(supp, dose) %>%
  shapiro_test(len)
```

6 cells, smallest p = 0.17 → **no evidence against normality.**

Test the groups you compare. OJ pooled over doses gives p = 0.024: pooling mixes three doses.

```
fit <- lm(len ~ supp * dose, data = df)
shapiro_test(residuals(fit))  # p = 0.67
```

For ANOVA designs, check the model residuals.

```
ggpubr::ggqqplot(df, "len",
  facet.by = "dose")   # line = normal
```

EQUAL VARIANCES (LEVENE)

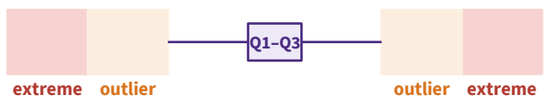
```
df %>% levene_test(len ~ supp * dose)
```

F(5, 54) = 1.71, p = 0.15 → **no evidence of unequal variances.**

OUTLIERS

```
df %>% group_by(dose) %>%
  identify_outliers(len)
```

dose	len	supp	is.outlier	is.extreme
0.5	21.5	OJ	TRUE	FALSE



Outlier: beyond 1.5 × IQR from Q1 or Q3. Extreme: beyond 3 × IQR. Investigate before removing.

TIP Shapiro–Wilk flags trivial departures in **large** samples and misses big ones in **small** samples: read it next to the QQ plot.

PAIRED DATA: TEST THE DIFFERENCES

```
d <- with(sleep, extra[group == 2] -
          extra[group == 1])
shapiro_test(d)   # p = 0.033
```

→ Not normal: use the paired Wilcoxon (p = 0.009).
Several outcomes at once: `mshapiro_test()`, `box_m()`

4 Describe and plot

SUMMARY STATISTICS

```
df %>% group_by(supp) %>%
  get_summary_stats(len, type = "mean_sd")
```

supp	variable	n	mean	sd
OJ	len	30	20.66	6.61
VC	len	30	16.96	8.27

`type =`

`"full"` `"common"` `"robust"`
`"five_number"` `"mean_sd"` `"mean_se"`
`"mean_ci"` `"median_iqr"` `"median_mad"`
`"quantile"`

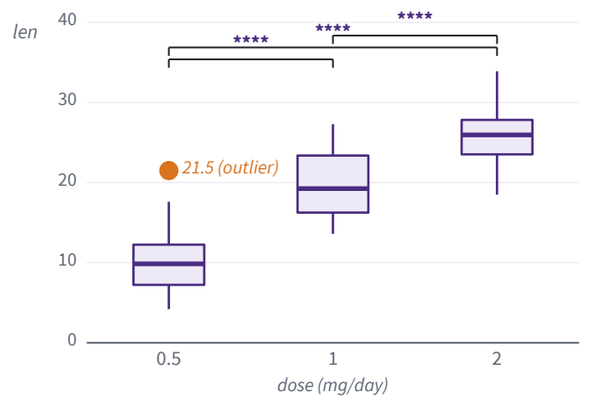
Or pick columns: `show = c("mean", "sd", "median", "iqr")`

COUNTS

```
df %>% freq_table(supp, dose)   # n, prop
```

FROM TEST TO PLOT

```
st <- df %>% t_test(len ~ dose) %>%
  add_xy_position(x = "dose")
ggpubr::ggboxplot(df, "dose", "len") +
  ggpubr::stat_pvalue_manual(st,
    label = "p.adj.signif")
```



WATCH OUT Paired tests match rows by **order**. Add `id = "ID"` to match by subject. With the rows of sleep shuffled once: p = 0.12 without it, p = 0.003 with it.

CC BY SA Anthony Romero-Cerdán • anthonyromero.netlify.app • Learn more at rpkgs.datanovia.com/rstatix • rstatix version 1.1.0 • Updated: 2026-10

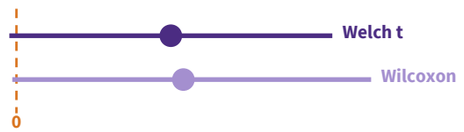
Statistical tests with rstatix :: **CHEAT SHEET**

Compare groups, find which ones differ, size the effect, then move to factorial designs, counts and correlations.

5 Two groups

```
df %>% t_test(len ~ supp, detailed = TRUE)
df %>% wilcox_test(len ~ supp,
  detailed = TRUE)
```

test	OJ − VC	95% CI	p
Welch t	3.7	−0.17 to 7.57	0.061
Wilcoxon	4.0	−0.10 to 8.50	0.064



Both intervals include 0: **no clear difference**, but the data are compatible with a sizeable one. Report the CI, not just p.

Wilcoxon estimate = Hodges–Lehmann shift, not a difference of medians.

EFFECT SIZE

```
df %>% cohens_d(len ~ supp)
df %>% wilcox_effsize(len ~ supp)
```

→ **d = 0.49 · r = 0.24, both small**



Cohen (1992) labels, added in the magnitude column. Or pass effect.size = TRUE to the test.

PAIRED (sleep: same 10 people, two drugs)

```
sleep %>% t_test(extra ~ group,
  paired = TRUE, id = "ID", detailed = TRUE)
sleep %>% cohens_d(extra ~ group,
  paired = TRUE, id = "ID")
```

Mean difference −1.58 (−2.46 to −0.70), t(9) = −4.06, p = 0.003; **d = −1.28, large.**

TIP `alternative = "greater"` gives a one-sided test (OJ > VC: p = 0.030). Choose the direction **before** seeing the data, never to rescue a p-value.

6 Three or more groups

OMNIBUS Does any group differ? → **POST HOC** Which pairs differ?

ONE-WAY ANOVA

```
df %>% anova_test(len ~ dose)
```

Effect	DFn	DFd	F	p	ges
dose	2	57	67.4	< 0.001	0.70

ges = generalized eta², the share of variance due to dose. `effect.size = "pes"` gives partial eta².

```
df %>% tukey_hsd(len ~ dose)
```

pair	difference	95% CI	p.adj
1 − 0.5	9.1	5.9 to 12.4	< 0.0001
2 − 0.5	15.5	12.3 to 18.7	< 0.0001
2 − 1	6.4	3.1 to 9.6	< 0.0001

Each extra 0.5 → 1 → 2 mg/day adds length.

UNEQUAL VARIANCES

```
df %>% welch_anova_test(len ~ dose)
df %>% games_howell_test(len ~ dose)
```

Welch F(2, 37.7) = 68.4, then Games–Howell pairs.

RANK-BASED

```
df %>% kruskal_test(len ~ dose)
df %>% kruskal_effsize(len ~ dose)
df %>% dunn_test(len ~ dose)   # Holm
```

H(2) = 40.7, p < 0.001; eta²[H] = 0.68, large.

pair	0.5 vs 1	0.5 vs 2	1 vs 2
p.adj	0.0008	< 0.0001	0.005
signif	***	****	**

TIP `pairwise_t_test(pool.sd = TRUE)` pools the SD of all groups, as the ANOVA does.

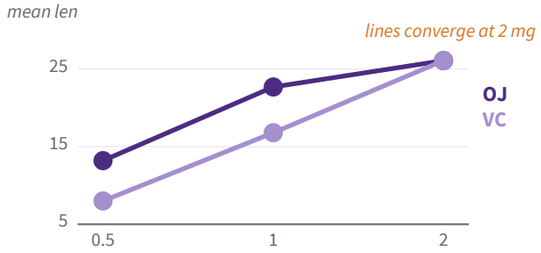
7 Factorial & repeated

TWO-WAY ANOVA

```
df %>% anova_test(len ~ supp * dose)
```

Effect	DFn	DFd	F	p	ges
supp	1	54	15.6	< 0.001	0.22
dose	2	54	92.0	< 0.001	0.77
supp:dose	2	54	4.1	**0.022**	0.13

Type II sums of squares by default; type = 3 for type III.



INTERACTION? TEST SIMPLE EFFECTS

```
df %>% group_by(dose) %>%
  emmeans_test(len ~ supp,
    p.adjust.method = "bonferroni")
```

OJ > VC at 0.5 mg (p = 0.002) and 1 mg (p < 0.001), **not at 2 mg** (p = 0.96).

REPEATED MEASURES

```
res <- sleep %>% anova_test(dv = extra,
  wid = ID, within = group)
get_anova_table(res)   # F(1, 9) = 16.5
```

Sphericity: Mauchly test and Greenhouse–Geisser correction applied **automatically** when needed (3+ levels). Mixed design: add `between =`. Rank-based:

```
sleep %>% friedman_test(extra ~ group | ID)
```

MANY OUTCOMES AT ONCE

```
iris %>% tidyr::pivot_longer(-Species,
    names_to = "measure") %>%
  group_by(measure) %>%
  kruskal_test(value ~ Species) %>%
  adjust_pvalue(method = "holm")
```

Don't name the column "variable": `get_summary_stats()` then fails.

8 Counts & correlation

```
bw <- MASS::birthwt
xt <- xtabs(~ smoke + low, data = bw)
```

low birth weight

	no	yes	% low
non-smoker	86	29	**25.2%**
smoker	44	30	**40.5%**

```
chisq_test(xt)
```

→ **$\chi^2(1) = 4.24$, p = 0.040 (Yates-corrected)**

```
expected_freq(chisq_test(xt))
```

→ **all ≥ 23: χ^2 is valid**

```
fisher_test(xt, detailed = TRUE)
```

→ **OR 2.01 (1.03–3.96), p = 0.036**

```
cramer_v(xt)
```

→ **0.16**

Which cells drive it? `chisq_descriptives()` lists standardized residuals (here ±2.22; beyond ±2 stands out).

Fisher when expected counts are small (< 5); mcnemar_test() for paired yes/no.

CORRELATION

```
mtcars %>% cor_test(mpg, wt)
```

→ **r = −0.87 (−0.93 to −0.74), p < 0.001** · Spearman ρ = −0.89

```
mtcars %>% select(mpg, wt, hp, qsec) %>%
  cor_mat() %>% cor_mark_significant()
```

	mpg	wt	hp
wt	−0.87 ****		
hp	−0.78 ****	0.66 ****	
qsec	0.42 *	−0.17	−0.71 ****

Violet: negative · orange: positive. P values: cor_get_pval().

CHECKLIST Report the test, statistic, df, exact p, **effect size with its CI** and n. A p-value alone says nothing about size.

CC BY SA Anthony Romero-Cerdán • anthonyromero.netlify.app • Learn more at rpkgs.datanovia.com/rstatix • rstatix version 1.1.0 • Updated: 2026-10