

rempsysc: Convenience functions for psychology

Rémi Thériault ¹

¹ Department of Psychology, Université du Québec à Montréal, Québec, Canada

DOI: [10.xxxxxx/draft](https://doi.org/10.xxxxxx/draft)

Software

- [Review](#) 
- [Repository](#) 
- [Archive](#) 

Editor: [Open Journals](#) 

Reviewers:

- [@openjournals](#)

Submitted: 01 January 1970

Published: unpublished

License

Authors of papers retain copyright
and release the work under a
Creative Commons Attribution 4.0
International License ([CC BY 4.0](#))

Summary

`{rempsysc}` is an R package of convenience functions that make the analysis-to-publication workflow faster, easier, and less error-prone. It enables the creation of publication-ready APA (American Psychological Association) tables exportable to Word (via `{flextable}`) and easily customizable APA-compliant plots (via `{ggplot2}`). It makes it easy to run statistical tests, check assumptions, and automate various tasks common in psychology research and social sciences more broadly.

Statement of need

There are many reasons to use R ([R Core Team, 2022](#)) for analyzing and reporting data from research studies, such as being compatible with the ideals of open science ([Quintana, 2020](#)). However, R has a major downside for novices: its steep learning curve due to its programmatic interface, in contrast to perhaps more user-friendly point-and-click software. Of course, this flexibility is also a strength, as the R community can and does come together to produce packages that make using R increasingly easier and more user-friendly (e.g., the *easystats* ecosystem [Lüdtke et al., 2019/2023](#)). The `{rempsysc}` package (Really Easy Methods for Psychology) contributes to this momentum by providing convenience functions that remove as much friction as possible between your script and your manuscript (in particular, if you are using Microsoft Word).

There are mainly three things that go into a manuscript: text, tables, and figures. `{rempsysc}` does not generate publication-ready text summarizing analyses; for this, see the `{report}` package ([Makowski et al., 2021/2023](#)). Instead, `{rempsysc}` focuses on the production of publication-ready tables and figures. Below, I go over a few quick examples of those.

Examples Features

Publication-Ready Tables

Many researchers using R still copy-paste the values from the R console to their manuscript, or retype them manually. Yet, this approach increases the risks of copy-paste and retyping errors so common in psychology. This problem is not trivial given that according to some estimates, up to 50% of articles in psychology have at least one statistical error ([Nuijten et al., 2016](#)). Ideally, one should be able to format the table directly in R, and to export it to Word directly.

Formatting a table properly in R is already a tedious and time-consuming task, but fortunately several packages take care of this step (e.g., the `{broom}` or `{report}` packages, [Makowski et al., 2021/2023](#); [Robinson et al., 2022](#), and there are several others). Exporting these formatted tables to Microsoft Word remains a challenge however. Some packages do export to Word (e.g., the `{apaTables}` package [Stanley & Spence, 2018](#)), but their formatting is often rigid especially when using analyzes or table formats that are not supported by default.

39 {rempsysc} solves this problem by allowing maximum flexibility: you manually create the data
40 frame exactly the way you want, and then only use the magical function, `nice_table()`, on
41 the resulting data frame. `nice_table()` works on any data frame, even non-statistical ones
42 like `mtcars`.

43 One of its main benefits however is the automatic formatting of statistical symbols and its
44 integration with other packages. We can for example create a {broom} table and then apply
45 `nice_table()` on it. It suits particularly well the pipe workflow.

```
46 library(rempsysc)
47
48 lm(mpg ~ cyl + wt * hp, mtcars) |>
49   broom::tidy(conf.int = TRUE) |>
50   nice_table(broom = "lm")
```

Term	<i>b</i>	<i>SE</i>	<i>t</i>	<i>p</i>	95% CI
(Intercept)	49.49	3.66	13.51	< .001	[41.97, 57.01]
cyl	-0.37	0.51	-0.72	.479	[-1.41, 0.68]
wt	-7.63	1.52	-5.01	< .001	[-10.75, -4.51]
hp	-0.11	0.03	-3.64	.001	[-0.17, -0.05]
wt × hp	0.03	0.01	3.23	.003	[0.01, 0.04]

51
52 We can do the same with a {report} table.

```
53 stats.table <- lm(mpg ~ cyl + wt * hp, mtcars) |>
54   report::report() |>
55   as.data.frame()
56
57 nice_table(stats.table)
```

Parameter	Fit	b	95% CI (b)	t	df	p	β	95% CI (β)
(Intercept)		49.49	[41.97, 57.01]	13.51	27	< .001	-0.18	[-0.36, -0.01]
cyl		-0.37	[-1.41, 0.68]	-0.72	27	.479	-0.11	[-0.42, 0.20]
wt		-7.63	[-10.75, -4.51]	-5.01	27	< .001	-0.62	[-0.85, -0.40]
hp		-0.11	[-0.17, -0.05]	-3.64	27	.001	-0.29	[-0.53, -0.04]
wt \times hp		0.03	[0.01, 0.04]	3.23	27	.003	0.29	[0.11, 0.47]
AIC	147.01							
AICc	150.37							
BIC	155.80							
R2	0.89							
R2 (adj.)	0.87							
Sigma	2.17							

58

59 The {report} package provides quite comprehensive tables, so one may request an abbreviated
60 table with the 'short' argument. For convenience, it is also possible to highlight significant
61 results for better visual discrimination, using the 'highlight' argument.[1] Once satisfied
62 with the table, we can add a title and note.

```
63 my_table <- nice_table(
64   stats.table, short = TRUE, highlight = 0.001,
65   title = c("Table 1", "A Pretty Regression Model"),
66   note = c("The data was extracted from the 1974 Motor Trend US magazine.",
67            "Greyed rows represent statistically significant differences, p < .001.")
68 my_table
```

Table 1

A Pretty Regression Model

Parameter	<i>b</i>	<i>t</i>	<i>df</i>	<i>p</i>	β	95% CI (β)
(Intercept)	49.49	13.51	27	< .001	-0.18	[-0.36, -0.01]
cyl	-0.37	-0.72	27	.479	-0.11	[-0.42, 0.20]
wt	-7.63	-5.01	27	< .001	-0.62	[-0.85, -0.40]
hp	-0.11	-3.64	27	.001	-0.29	[-0.53, -0.04]
wt \times hp	0.03	3.23	27	.003	0.29	[0.11, 0.47]

Note. The data was extracted from the 1974 Motor Trend US magazine.

Greyed rows represent statistically significant differences, $p < .001$.

69

One can then easily save the resulting table to Word with `flextable::save_as_docx()`, specifying the object name and desired path.

70

`flextable::save_as_docx(my_table, path = "nice_tablehere.docx")`

71

Additionally, tables created with `nice_table()` are `{flextable}` objects (Gohel & Skintzos, 2022), and can be modified as such.[2]

72

73 **Formattting Results of Analyses**

74

`{rempsyc}` also provides its own set of functions to prepare statistical tables before they can be fed to `nice_table()` and saved to Word.

75

76 ***t* tests**

77

```
78 nice_t_test(data = mtcars,
79             response = c("mpg", "disp", "drat"),
80             group = "am",
81             warning = FALSE) |>
82 nice_table()
83
```

84

Dependent Variable	<i>t</i>	<i>df</i>	<i>p</i>	<i>d</i>	95% CI
mpg	-3.77	18.33	.001	-1.48	[-2.27, -0.67]
disp	4.20	29.26	< .001	1.45	[0.64, 2.23]
drat	-5.65	27.20	< .001	-2.00	[-2.86, -1.12]

85

85 Contrasts

```
86 nice_contrasts(data = mtcars,  
87               response = c("mpg", "disp"),  
88               group = "cyl",  
89               covariates = "hp") |>  
90   nice_table(highlight = .001)
```

Dependent Variable	Comparison	df	t	p	d	95% CI
mpg	4 - 8	28	3.66	.001	3.59	[2.65, 4.52]
	6 - 8	28	1.29	.207	1.44	[0.82, 1.99]
	4 - 6	28	3.64	.001	2.15	[1.37, 3.12]
disp	4 - 8	28	-6.04	<.001	-4.80	[-5.92, -3.88]
	6 - 8	28	-4.86	<.001	-3.29	[-4.30, -2.26]
	4 - 6	28	-2.70	.012	-1.51	[-2.29, -0.91]

91

92 Regressions

```
93 data <- lapply(mtcars, scale)  
94 model1 <- lm(mpg ~ disp + wt * hp, data)  
95 model2 <- lm(qsec ~ drat + wt * hp, data)  
96 my.models <- list(model1, model2)  
97  
98 nice_lm(my.models) |>  
99   nice_table(highlight = TRUE)
```

Dependent Variable	Predictor	df	β	t	p	sr ²	95% CI
mpg	disp	27	-0.06	-0.35	.728	.00	[0.00, 0.01]
	wt	27	-0.63	-4.39	<.001	.08	[0.00, 0.16]
	hp	27	-0.32	-3.00	.006	.04	[0.00, 0.09]
	wt × hp	27	0.31	3.71	.001	.06	[0.00, 0.12]
qsec	drat	27	-0.03	-0.18	.862	.00	[0.00, 0.01]
	wt	27	0.49	2.37	.025	.07	[0.00, 0.19]
	hp	27	-1.05	-6.92	<.001	.62	[0.40, 0.83]
	wt × hp	27	0.02	0.10	.922	.00	[0.00, 0.00]

100

101 Simple Slopes

```
102 nice_lm_slopes(my.models, predictor = "wt", moderator = "hp", b.label = "B") |>
103   nice_table()
```

Dependent Variable	Predictor (+/-1 SD)	df	β	t	p	sr^2	95% CI
mpg	wt (LOW-hp)	27	-0.94	-5.60	< .001	.13	[0.02, 0.25]
	wt (MEAN-hp)	27	-0.63	-4.39	< .001	.08	[0.00, 0.16]
	wt (HIGH-hp)	27	-0.32	-1.94	.063	.02	[0.00, 0.05]
qsec	wt (LOW-hp)	27	0.47	1.56	.131	.03	[0.00, 0.10]
	wt (MEAN-hp)	27	0.49	2.37	.025	.07	[0.00, 0.19]
	wt (HIGH-hp)	27	0.51	2.41	.023	.07	[0.00, 0.19]

104

105 Correlation Matrices

106 It is also possible to export a colour-coded correlation matrix to Microsoft Excel. The
 107 `cormatrix_excel()` function has several benefits over conventional approaches. The base R
 108 `cor()` function for example does not use rounded values and the console is impractical for
 109 large matrices. One may manually round values and export it to a .csv file, which is an
 110 improvement but still unsatisfying.

111 The `{apaTables}` package (Stanley & Spence, 2018) allows exporting the correlation matrix
 112 to Word in an APA format, and in many cases this already meets the formal requirements of
 113 APA style. However, the Word format is not suitable for large matrices, as it will often spread
 114 beyond the document's margin limits.

115 Another approach is to export the matrix to an image, like the `{correlation}` package does
 116 (Makowski et al., 2020).[3] For very small matrices, this works extremely well, and the colour is
 117 an immense help to quickly identify which correlations are strong or weak, positive or negative,
 118 and significant or non-significant. Again, however, this does not work so well for large matrices
 119 because labels might overlap or navigating the large figure becomes difficult.

120 When the goal is more exploratory in nature, and one has large matrices, it can be beneficial
 121 to export them to Excel. `{rempsyc}` combines the idea of using a coloured correlation matrix
 122 from the `{correlation}` package with the idea of exporting to Excel using `{openxlsx2}` (Barbone
 123 & Garbuszus, 2023).

124 `{rempsyc}` also provides some usability improvements, like freezing the first row and column so
 125 as to be able to easily see which variables correlate with which other variables, regardless of
 126 how far or deep those variables are located within the matrix.

127 The colour represents the strength of the correlation, whereas the stars represent different
 128 significance thresholds for the p value is.[4] The exact p values are provided in a second tab
 129 for reference purposes, so all information is readily available in just one function call.

```
130 cormatrix_excel(data = infert,
131                 filename = "cormatrix1",
132                 select = c("age", "parity", "induced", "case", "spontaneous",
133                           "stratum", "pooled.stratum"))
```

	A	B	C	D	E	F	G	H	I
1	Parameter	age	parity	induced	case	spontaneous	stratum	pooled.stratum	
2	age	1.0	.08	-.10	.0	-.08	-.21 ***	-.17 *	
3	parity	.08	1.0	.45 ***	.01	.31 ***	-.31 ***	.12	
4	induced	-.10	.45 ***	1.0	.02	-.27 ***	-.10	.16 *	
5	case	.0	.01	.02	1.0	.36 ***	.0	.0	
6	spontaneous	-.08	.31 ***	-.27 ***	.36 ***	1.0	.06	.21 ***	
7	stratum	-.21 ***	-.31 ***	-.10	.0	.06	1.0	.75 ***	
8	pooled.stratum	-.17 *	.12	.16 *	.0	.21 ***	.75 ***	1.0	
9									
		r_values	p_values						
134									
	A	B	C	D	E	F	G	H	I
1	Parameter	age	parity	induced	case	spontaneous	stratum	pooled.stratum	
2	age	.0	.194	.113	.956	.186	.001	.006	
3	parity	.194	.0	.0	.889	.0	.0	.059	
4	induced	.113	.0	.0	.789	.0	.113	.010	
5	case	.956	.889	.789	.0	.0	.952	.939	
6	spontaneous	.186	.0	.0	.0	.0	.341	.001	
7	stratum	.001	.0	.113	.952	.341	.0	.0	
8	pooled.stratum	.006	.059	.010	.939	.001	.0	.0	
9									
		r_values	p_values						
135									

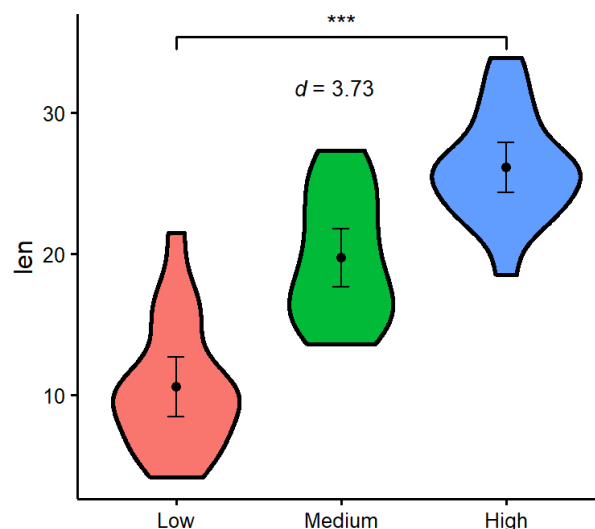
Publication-Ready Figures

Preparing figures according to APA style, having them look good, and being able to save them in high-resolution with the proper ratios is often challenging. Working with `{ggplot2}` (Wickham, 2016) provides tremendous flexibility, but an unintended consequence is that doing even trivial operations can at times be daunting.

This is why `{rempsyc}` setups a few default plot types, ready to be saved to your preferred format (.pdf, .tiff, or .png).

Violin Plots

```
nice_violin(data = ToothGrowth,
            group = "dose",
            response = "len",
            xlabels = c("Low", "Medium", "High"),
            comp1 = 1,
            comp2 = 3,
            has.d = TRUE,
            d.y = 30)
```



152

153 For an example of such use in publication, see Thériault et al. (2021).

154 One can easily save the resulting figure with `ggplot2::ggsave()`, specifying the desired file
155 name, extension, and resolution.

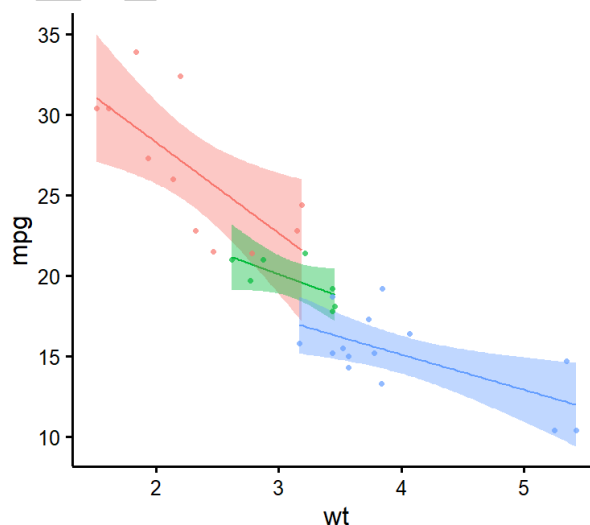
```
156 ggplot2::ggsave('nice_violinplotthere.pdf', width = 7, height = 7,  
157                unit = 'in', dpi = 300)
```

158 Recommended dimensions for saving {rempsyc} figures is 7 inches wide and 7 inches high
159 at 300 dpi, which makes sure that the resolution is high enough even if saving to non-vector
160 graphics formats like .png. That said, scalable vector graphics formats like .pdf or .eps are
161 still recommended for high-resolution submissions to scientific journals.

162 Scatter Plots

163 Figures are {ggplot2} objects (Wickham, 2016), and can be modified as such.

```
164 nice_scatter(data = mtcars,  
165             predictor = "wt",  
166             response = "mpg",  
167             group = "cyl",  
168             has.confband = TRUE)
```

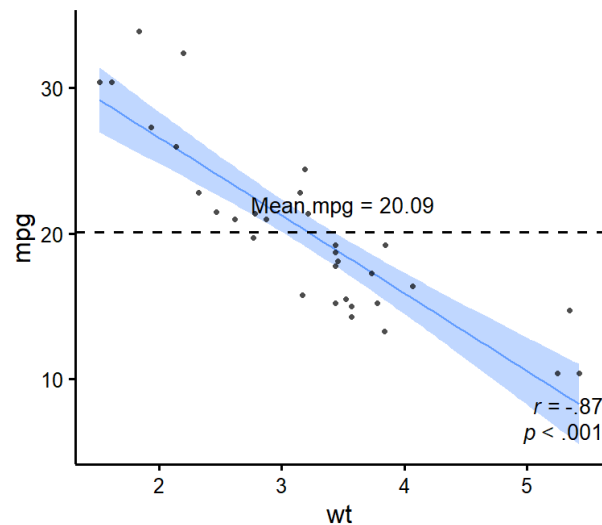


169


```

170 nice_scatter(data = mtcars,
171               predictor = "wt",
172               response = "mpg",
173               has.confband = TRUE,
174               has.r = TRUE,
175               has.p = TRUE) +
176   ggplot2::geom_hline(yintercept = mean(mtcars$mpg), colour = "black",
177                       linewidth = 1.4, linetype = "dashed") +
178   ggplot2::annotate("text", x = 3.5, y = 22, size = 7,
179                       label = paste("Mean mpg =", round(mean(mtcars$mpg), 2)))

```



```

180
181 For an example of such use in publication, see Krol et al. (2020).

```

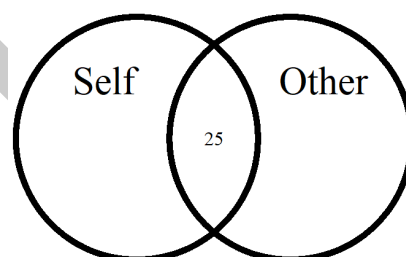
182 Overlapping Circles

183 For psychologists using the Inclusion of Other in the the Self Scale (Aron et al., 1992), it can
 184 be useful to interpolate the original discrete scores (1 to 7) into a group average representation
 185 of the conceptual self-other overlap. For example, assuming the group mean is 3.5 on the 1 to
 186 7 scale, `overlap_circle()` will draw a 25% overlap from interpolation:

```

187 overlap_circle(3.5)

```



```

188
189 For an example of such use in publication, see Thériault et al. (2021).

```

190 Testing assumptions

191 When comes time to test assumptions of a linear model, the best option is the `check_model()`
192 function from *easystats*' {performance} package, which allows direct visual evaluation of
193 assumptions (Lüdtke, Ben-Shachar, et al., 2021). Indeed, visual assessment of diagnostic
194 plots is recommended over statistical tests since they are overpowered in large samples and
195 underpowered in small samples (Kozak & Piepho, 2018).

196 That said, if for whatever reason one wants to check objective assumption tests for a linear
197 model, *rempsysc* makes this easy with the `nice_assumptions()` function, which provide p
198 values for normality (Shapiro–Wilk), homoscedasticity (Breusch–Pagan) and autocorrelation of
199 residuals (Durbin–Watson) in one call.

200 Categorical Predictors

201 `nice_normality()` makes it easy to visually check normality in the case of categorical predictors
202 (i.e., when using groups), through a combination of quantile-quantile plots, density plots, and
203 histograms.

```
204 nice_normality(data = iris,  
205               variable = "Sepal.Length",  
206               group = "Species",  
207               shapiro = TRUE,  
208               histogram = TRUE,  
209               title = "Density (Sepal Length)")
```



210

211 Similarly for univariate outliers using median absolute deviations from the median (MAD, Leys
212 et al., 2013).

```
213 plot_outliers(airquality,  
214              group = "Month",  
215              response = "Ozone")
```



216

217 Univariate outliers based on the median/MAD can also be simply requested with `find_mad()`.^[5]

218 `find_mad(airquality, names(airquality), criteria = 3)`

219

220 `## 8 outlier(s) based on 3 median absolute deviations for variable(s):`

221 `## Ozone, Solar.R, Wind, Temp, Month, Day`

222 `##`

223 `## Outliers per variable:`

224 `##`

225 `## $Ozone`

226 `## Row Ozone_mad`

227 `## 1 30 3.218284`

228 `## 2 62 3.989131`

229 `## 3 99 3.488081`

230 `## 4 101 3.025573`

231 `## 5 117 5.261028`

232 `## 6 121 3.333911`

233 `##`

234 `## $Wind`

235 `## Row Wind_mad`

236 `## 1 9 3.049871`

237 `## 2 48 3.225825`

238 Homoscedasticity can also be checked numerically with `nice_var()` or visually with

239 `nice_varplot()`.

240 `nice_var(data = iris,`

241 `variable = names(iris[1:4]),`

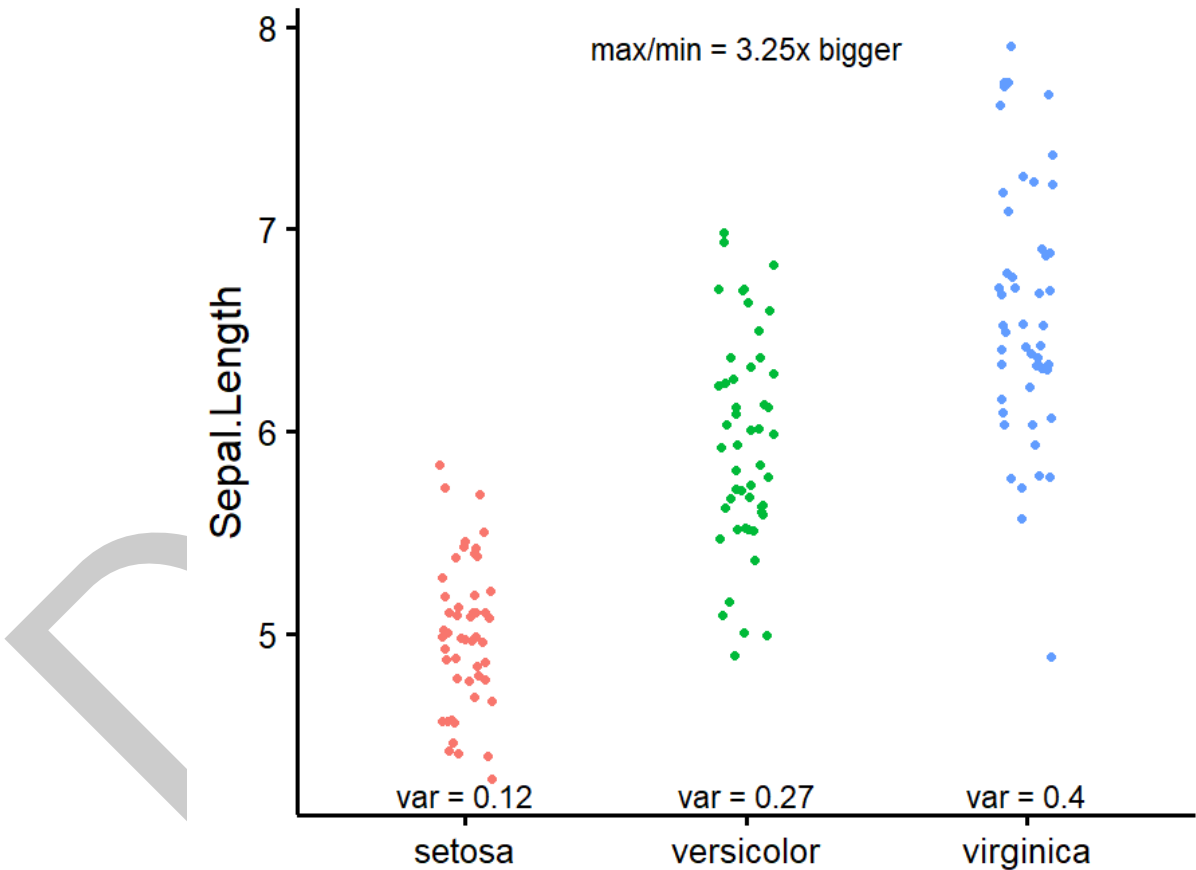
242 `group = "Species") |>`

243 `nice_table()`

Species	Setosa	Versicolor	Virginica	Variance.ratio	Criteria	Heteroscedastic
Sepal.Length	0.12	0.27	0.40	3.30	4.00	FALSE
Sepal.Width	0.14	0.10	0.10	1.50	4.00	FALSE
Petal.Length	0.03	0.22	0.30	10.20	4.00	TRUE
Petal.Width	0.01	0.04	0.07	6.80	4.00	TRUE

244

```
245 nice_varplot(data = iris,  
246               variable = "Sepal.Length",  
247               group = "Species")
```



248

249 Utility functions

250 Finally, with the idea of making the analysis workflow easier in mind, {rempsyc} also provides
251 a few other utility functions. `nice_na()` allows reporting item-level missing values per scale,
252 as well as participant's maximum number of missing items by scale, as per recommendations
253 (Parent, 2013).

254 `extract_duplicates()` creates a data frame of only observations with a duplicated ID or
255 participant number, so they can be investigated more thoroughly. `best_duplicate()` allows

256 to follow-up on this investigation and only keep the “best” duplicate, meaning those with the
257 fewer number of missing values, and in case of ties, the first one.

258 `nice_reverse()` permits the automatic reverse-coding of scores so common for psychology
259 questionnaires, provided the minimum and maximum score values are known.

260 There are other functions that the reader can explore at their leisure on the package official
261 website. However, hopefully, this overview has given the reader a gentle introduction to this
262 package.

263 Licensing and Availability

264 The `{rempsys}` package is licensed under the GNU General Public License (GPL v3.0). It is
265 available on CRAN, and can be installed using `install.packages("rempsys")`. The full tutorial
266 website can be accessed at: <https://rempsys.remi-theriault.com/>. All code is open-source and
267 hosted on GitHub, and bugs can be reported at <https://github.com/rempsys/rempsys/issues/>.

268 Acknowledgements

269 I would like to thank Jay Olson, Hugues Leduc, Charles-Étienne Lavoie, and Björn Büdenbender
270 for statistical or technical advice that helped inform some functions of this package and/or
271 useful feedback on this manuscript. I would also like to acknowledge funding from the Social
272 Sciences and Humanities Research Council of Canada.

273 References

- 274 Aron, A., Aron, E. N., & Smollan, D. (1992). Inclusion of Other in the Self Scale and the
275 structure of interpersonal closeness. *Journal of Personality and Social Psychology*, 63(4), 596.
276 <https://doi.org/10.1037/0022-3514.63.4.596>
- 277 Barbone, J. M., & Garbuszus, J. M. (2023). *openxlsx2: Read, write and edit 'xlsx' files*.
278 <https://github.com/JanMarvin/openxlsx2>
- 279 Gohel, D., & Skintzos, P. (2022). *Flextable: Functions for tabular reporting*. <https://CRAN.R-project.org/package=flextable>
- 281 Kozak, M., & Piepho, H.-P. (2018). What's normal anyway? Residual plots are more telling
282 than significance tests when checking ANOVA assumptions. *Journal of Agronomy and Crop*
283 *Science*, 204(1), 86–98. <https://doi.org/10.1111/jac.12220>
- 284 Krol, S. A., Thériault, R., Olson, J. A., Raz, A., & Bartz, J. A. (2020). Self-concept clarity
285 and the bodily self: Malleability across modalities. *Personality and Social Psychology Bulletin*,
286 46(5), 808–820. <https://doi.org/10.1177/0146167219879126>
- 287 Leys, C., Ley, C., Klein, O., Bernard, P., & Licata, L. (2013). Detecting outliers: Do not use
288 standard deviation around the mean, use absolute deviation around the median. *Journal of*
289 *Experimental Social Psychology*, 49(4), 764–766. <https://doi.org/10.1016/j.jesp.2013.03.013>
- 290 Lüdtke, D., Ben-Shachar, M. S., Patil, I., Waggoner, P., & Makowski, D. (2021). performance:
291 An R package for assessment, comparison and testing of statistical models. *Journal of Open*
292 *Source Software*, 6(60), 3139. <https://doi.org/10.21105/joss.03139>
- 293 Lüdtke, D., Makowski, D., Ben-Shachar, M. S., Patil, I., Wiernik, B. M., Bacher, E., &
294 Thériault, R. (2023). *easystats: Streamline model interpretation, visualization, and reporting*.
295 <https://easystats.github.io/easystats/> (Original work published 2019)

- 296 Lüdecke, D., Patil, I., Ben-Shachar, M. S., Wiernik, B. M., Waggoner, P., & Makowski, D.
297 (2021). see: An R package for visualizing statistical models. *Journal of Open Source Software*,
298 6(64), 3393. <https://doi.org/10.21105/joss.03393>
- 299 Makowski, D., Ben-Shachar, M. S., Patil, I., & Lüdecke, D. (2020). Methods and algorithms
300 for correlation analysis in R. *Journal of Open Source Software*, 5(51), 2306. <https://doi.org/10.21105/joss.02306>
301
- 302 Makowski, D., Lüdecke, D., Patil, I., Thériault, R., Ben-Shachar, M. S., & Wiernik, B. M.
303 (2023). *report: Automated reporting of results and statistical models*. <https://easystats.github.io/report/> (Original work published 2021)
304
- 305 Nuijten, M. B., Hartgerink, C. H., Van Assen, M. A., Epskamp, S., & Wicherts, J. M. (2016).
306 The prevalence of statistical reporting errors in psychology (1985–2013). *Behavior Research*
307 *Methods*, 48, 1205–1226. [https://doi.org/doi.org/10.3758/s13428-015-0664-2](https://doi.org/10.3758/s13428-015-0664-2)
- 308 Parent, M. C. (2013). Handling item-level missing data: Simpler is just as good. *The*
309 *Counseling Psychologist*, 41(4), 568–600. <https://doi.org/10.1177/0011000012445176>
- 310 Quintana, D. S. (2020). *Five things about open and reproducible science that every early*
311 *career researcher should know*. <https://osf.io/2jt9u>
- 312 R Core Team. (2022). *R: A language and environment for statistical computing*. R Foundation
313 for Statistical Computing. <https://www.R-project.org/>
- 314 Robinson, D., Hayes, A., & Couch, S. (2022). *broom: Convert statistical objects into tidy*
315 *tibbles*. <https://CRAN.R-project.org/package=broom>
- 316 Stanley, D. J., & Spence, J. R. (2018). Reproducible tables in psychology using the apaTables
317 package. *Advances in Methods and Practices in Psychological Science*, 1(3), 415–431. <https://doi.org/10.1177/2515245918773743>
318
- 319 Thériault, R., Olson, J. A., Krol, S. A., & Raz, A. (2021). Body swapping with a Black person
320 boosts empathy: Using virtual reality to embody another. *Quarterly Journal of Experimental*
321 *Psychology*, 74(12), 2057–2074. <https://doi.org/10.1177/17470218211024826>
- 322 Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York.
323 <https://ggplot2.tidyverse.org>
- 324 [1] This argument can be used logically, as 'TRUE' or 'FALSE', but can also be provided with
325 a numeric value representing the cut-off threshold for the *p* value
- 326 [2] A great resource for this is the {flextable} e-book: <https://ardata-fr.github.io/flextable-book/>
327
- 328 [3] Exporting the correlation matrix to an image through the {correlation} package also requires
329 the {see} package (Lüdecke, Patil, et al., 2021)
- 330 [4] For convenience, colours are only used when the corresponding *p* value is at least smaller
331 than .05
- 332 [5] Once one has identified outliers, it is also possible to winsorize them with the
333 `winsorize_mad()` function.