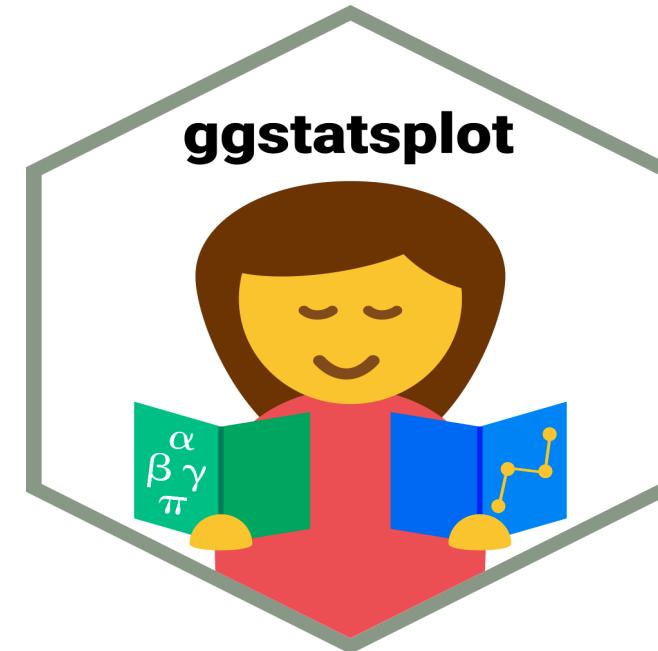


Statistical Visualizations with `{ggstatsplot}`: A Biography

Indrajeet Patil



Genesis

Why a new software?

(Or why did a social psychologist get into software development?)

Life in the trenches (c. 2017, Harvard)

⚠ External Stimulus

- **Reporting errors:**

“half of all published psychology papers contained at least one p -value that was inconsistent”¹

- **Interpretation errors:**

“in 72% of cases, nonsignificant results were misinterpreted [to mean] that effect was absent”²

- **Replication crisis:**

“39% of effects were subjectively rated to have replicated the original result”³

and more...

⚠ Internal Response



- How to avoid reporting errors?
- How to interpret null results?
- How to increase replicability?
- How to improve quality of statistical reporting?
- How to emphasize the importance of the effect?
- How to easily assess validity of model assumptions?

Proposal

Information-rich, ready-made statistical visualizations

(minimal effort and maximum transparency)

A visualization with statistical summary



Visualizations reveal problems not discernible from model summaries!

Ready-made plots with one-line syntax

The grammar of graphics framework can prepare any visualization! But building plots from scratch can be time-consuming.



💡 Using ready-made plots lowers the effort needed for visualizing data!

Action Plan

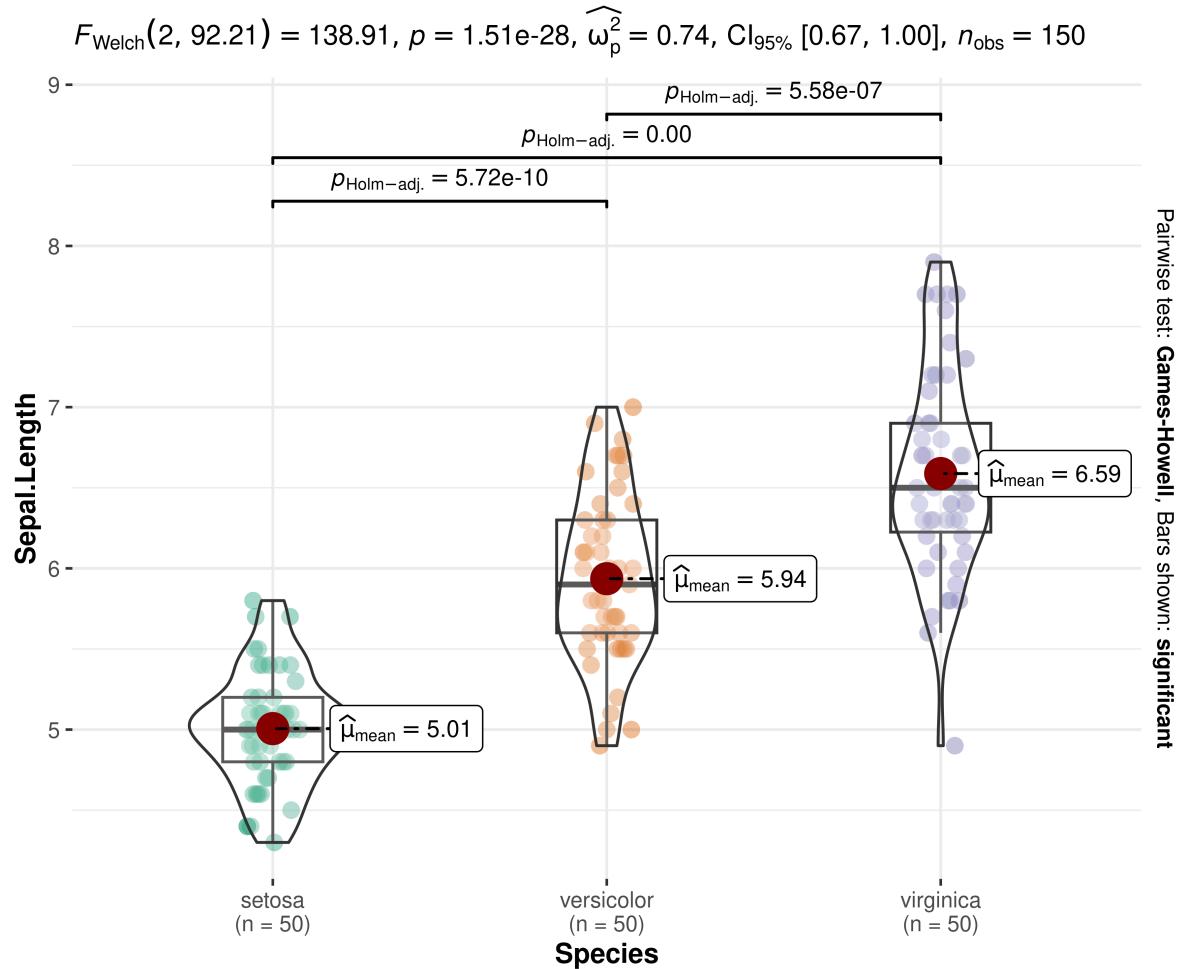
{ggstatsplot} was born!

(open-sourced on [GitHub](#) in 2017; still actively developed)

Example function

E.g., for hypothesis about differences *between* groups

```
1 ggbetweenstats(iris, Species, Sepal.Length)
```



! Important

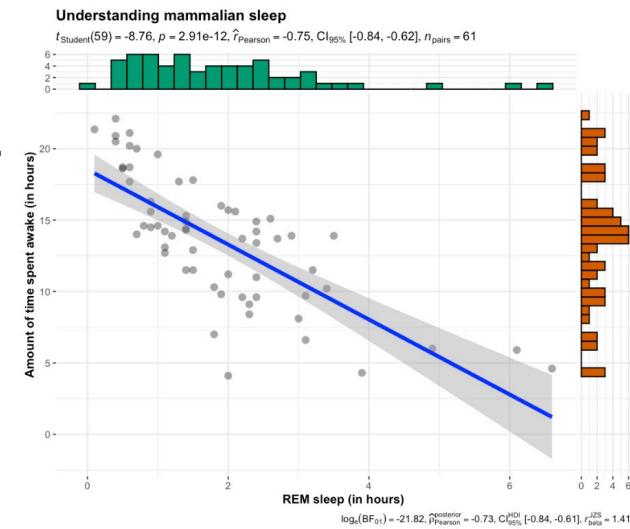
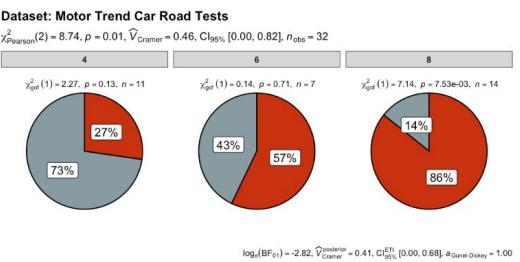
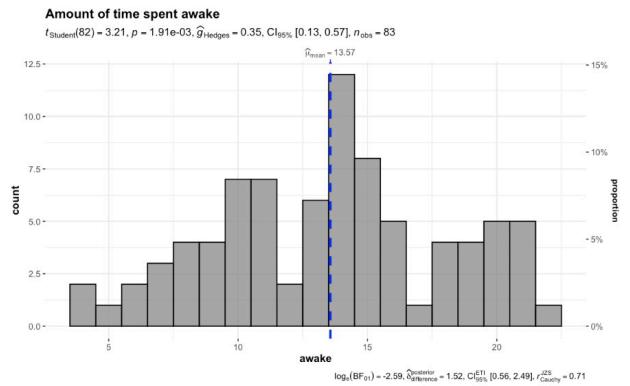
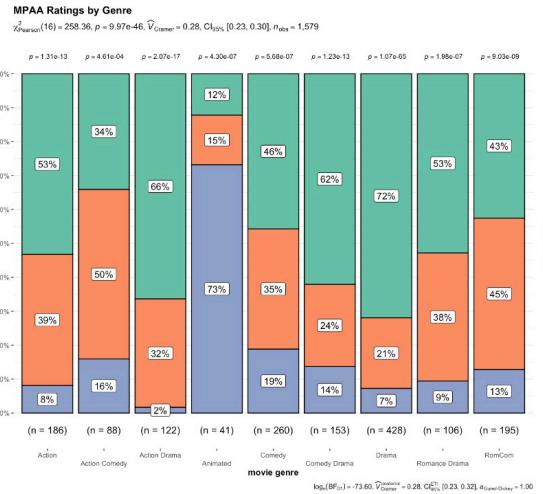
Information-rich defaults

- raw data + distributions
- descriptive statistics
- inferential statistics
- effect size + uncertainty
- pairwise comparisons
- Bayesian hypothesis-testing
- Bayesian estimation

Statistical approaches available

- parametric
- non-parametric
- robust
- Bayesian

And there is more!



Appendix provides more details.

Promised Land

Does it deliver?

Show, don't tell

Without `{ggstatsplot}`

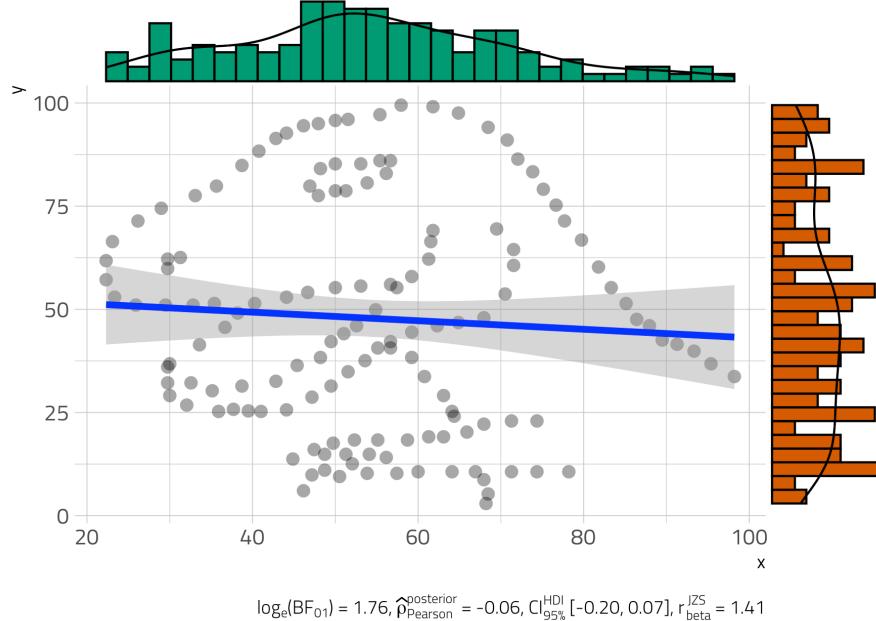
Pearson's correlation test revealed that, across 142 participants, variable `x` was negatively correlated with variable `y`:

$t(140) = -0.76, p = 0.446$. The effect size ($r = -0.06, 95\%CI[-0.23, 0.10]$) was small, as per Cohen's (1988) conventions. The Bayes Factor for the same analysis revealed that the data were 5.81 times more probable under the null hypothesis as compared to the alternative hypothesis. This can be considered moderate evidence (Jeffreys, 1961) in favor of the null hypothesis (absence of any correlation between `x` and `y`).

With `{ggstatsplot}`

Relationship between x and y

$t_{Student}(140) = -0.76, p = 0.446, \hat{r}_{Pearson} = -0.06, CI_{95\%} [-0.23, 0.10], n_{pairs} = 142$



$\log_e(BF_{01}) = 1.76, \hat{r}_{Pearson}^{posterior} = -0.06, CI_{95\%}^{HDI} [-0.20, 0.07], r_{beta}^{JZS} = 1.41$



No need to worry about reporting or interpretation errors!

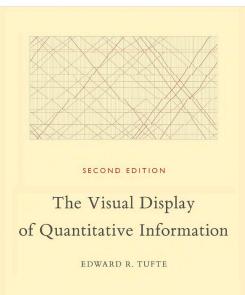
Thoughtful Defaults

>Data Visualization

Second Edition
Show Me the Numbers
Designing Tables and Graphs to Enlighten



Stephen Few



SECOND EDITION
The Visual Display
of Quantitative Information
EDWARD R. TUFTÉ



THE
ELEMENTS
OF GRAPHING
DATA



O'REILLY®
Fundamentals
of Data
Visualization
A Primer on Making Informative
and Compelling Figures

Statistical Reporting



Results from Welch's t-test with {statsExpressions}

Template for Frequentist analysis

test parameter statistic significance effect size type + estimate + confidence intervals number of observations
 $t_{\text{Welch}}(281.95) = -10.75, p = 8.31e-23, \hat{g}_{\text{Hedges}} = -1.27, \text{CI}_{99\%}[-1.61, -0.94], n_{\text{obs}} = 284$

Template for Bayesian analysis

evidence in favor of null over alternative hypothesis natural logarithm of Bayes Factor posterior type + estimate + credible intervals prior type and value
 $\log_e(\text{BF}_{01}) = -6.20, \delta_{\text{difference}}^{\text{posterior}} = -5.06, \text{CI}_{95\%}^{\text{HDI}} [-6.75, -3.53], r_{\text{cauchy}}^{\text{JZS}} = 0.71$

<<https://indrajeetpatil.github.io/statsExpressions/>>

src: @patilindrajeets

(Doorn et al., 2020; APA Manual)



Follows best practices in data visualization and statistical reporting!

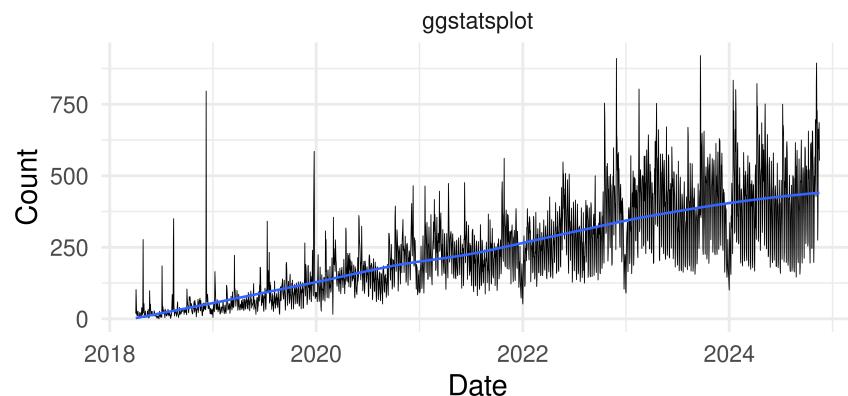
Impact

I can haz users?!

User Love



Total downloads > 500K (97 percentile)



Second most starred `{ggplot2}`-extension!



[ggstatsplot](#) Public



Enhancing `{ggplot2}` plots with statistical analysis 🎉

R ★ 2k 🔍 190



Total citations > 1000

From publications across a wide range of fields:
biology, medicine, psychology, economics, etc.



The Journal of Open Source Software

Visualizations with statistical details: The 'ggstatsplot' approach

Indrajeet Patil¹

¹ Center for Humans and Machines, Max Planck Institute for Human Development, Berlin, Germany

DOI: [10.21105/joss.03167](https://doi.org/10.21105/joss.03167)

Software

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

Editor: Charlotte Soneson

Reviewers:

- @njtierney
- @kevinrue

Submitted: 30 March 2021

Published: 25 May 2021

Summary

Graphical displays can reveal problems in a statistical model that might not be apparent from purely numerical summaries. Such visualizations can also be helpful for the reader to evaluate the validity of a model if it is reported in a scholarly publication or report. But, given the onerous costs involved, researchers often avoid preparing information-rich graphics and exploring several statistical approaches or tests available. The `ggstatsplot` package in the R programming language ([R Core Team, 2021](#)) provides a one-line syntax to enrich `ggplot2`-based visualizations with the results from statistical analysis embedded in the visualization itself. In doing so, the package helps researchers adopt a rigorous, reliable, and robust data exploratory and reporting workflow.

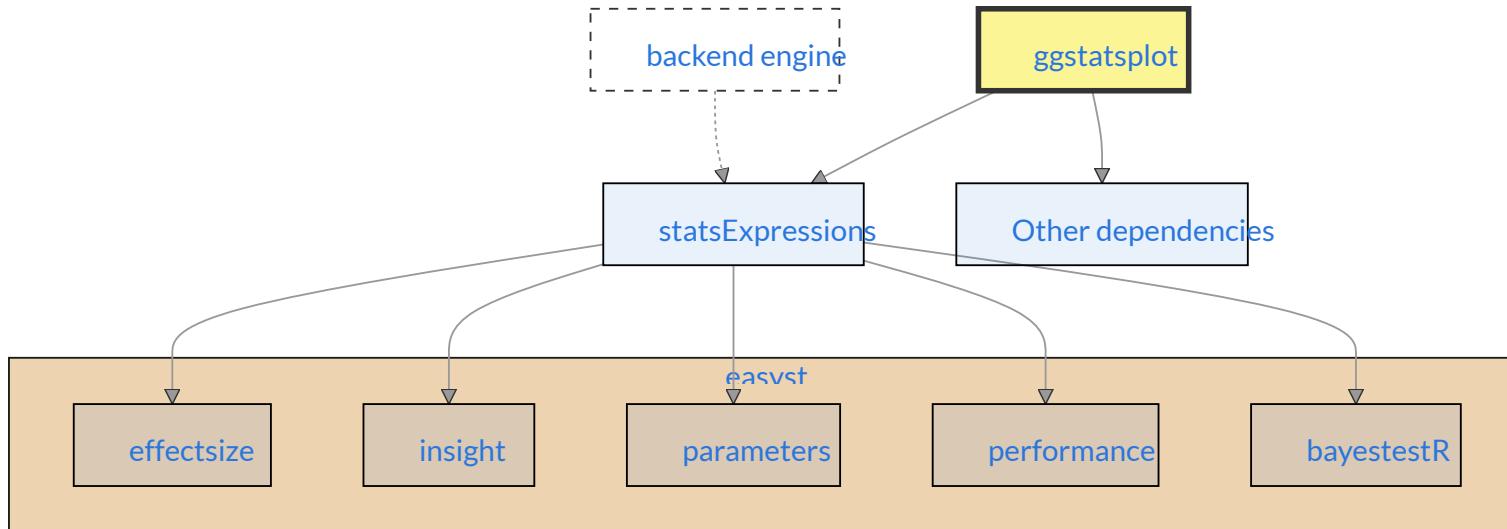
Improving Psychological Science Award (2020)

Pleasant Side Effects

Maybe the real treasure was the technical skills we picked up along the way!

Software Architecture

Breaking down the monolith: $20K_{(2017)} \rightarrow 1K_{(2024)}$ lines of code



Working with open-source teams

While re-architecting `{ggstatsplot}`, I started contributing upstream:

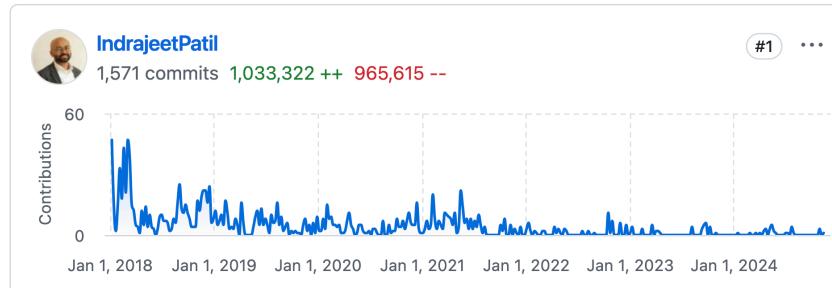
- joined the core `{easystats}` team (a collection of ten component packages)
- became a co-maintainer of `{ggsignif}`
- contributed to `{WRS2}` and `{ggcorrplot}`

Quality Assurance

ⓘ CI Checks (GitHub Actions)

- Unit tests (random-order)
- Code coverage (100%)
- Linting (0 lints)
- Formatting (0 issues)
- Documentation (website, no link rot, plenty examples)
- Pre-commit hooks (0 issues)
- Zero user-facing warnings
- Portability (Linux, macOS, Windows)
- Robustness (dependencies, language versions)
- CRAN checks (0 notes, 0 warnings, 0 errors)

❗ Healthy and active code base



All checks have passed
24 successful checks

Show all checks

💡 Working with open-source teams

While improving the QA tooling for `{ggstatsplot}`, I started contributing upstream:

- became co-author of `{lintr}` (linter) and `{styler}` (formatter)

Communication

Training material on best practices in software/package development to support community contributions keeping in mind the diverse backgrounds of contributors.

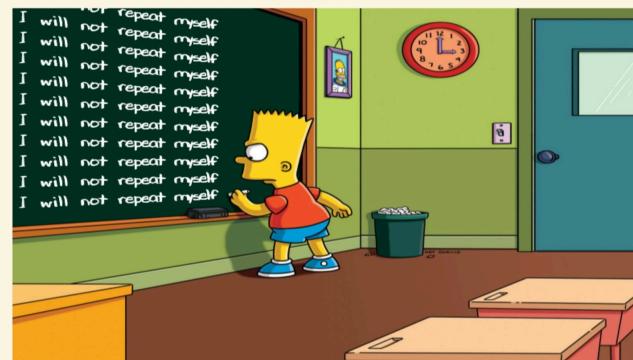
Preventive Care for R Packages

Indrajeet Patil



DRY PACKAGE DEVELOPMENT IN R

Indrajeet Patil



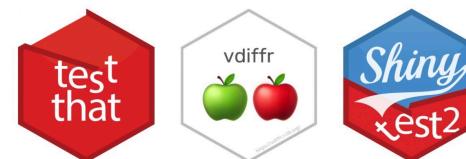
Dealing with the Second Hardest Thing in Computer Science

Indrajeet Patil



Introduction to Snapshot Testing in R

Indrajeet Patil



Conclusion

`{ggstatsplot}` provides an accessible interface to create rich statistical visualizations and helps users adopt a rigorous, reliable, and robust data exploratory and reporting workflows across diverse disciplines and domains in academia and industry alike. It's an actively developed tool with quality infrastructure and wide adoption.

Thank You 😊

For more

Source code for these slides can be found [on GitHub](#).

If you are interested in good programming and software development practices, check out my other [slide decks](#).

Find me at...

 Twitter

 LinkedIn

 GitHub

 Website

 E-mail

Session information

```
1 sessioninfo::session_info(include_base = TRUE)

-- Session info --
setting  value
version  R version 4.4.2 (2024-10-31)
os        Ubuntu 22.04.5 LTS
system   x86_64, linux-gnu
hostname fv-az1022-770
ui        X11
language (EN)
collate  C.UTF-8
ctype    C.UTF-8
tz       UTC
date     2024-11-17
pandoc   3.5 @ /opt/hostedtoolcache/pandoc/3.5/x64/ (via rmarkdown)
quarto   1.6.36 @ /usr/local/bin/quarto

-- Packages --
package      * version    date (UTC) lib source
base          * 4.4.2      2024-10-31 [3] local
BayesFactor    0.9.12-4.7 2024-01-24 [1] RSPM
bayestestR     0.15.0     2024-10-17 [1] RSPM
dplyr         1.1.0      2024-09-26 [1] RSPM
grid          4.4.2      2024-10-31 [3] local
gridExtra     2.3.3      2024-09-26 [1] RSPM
knitr         1.50.0     2024-09-26 [1] RSPM
lattice       0.22.0     2024-09-26 [1] RSPM
lubridate    1.9.0      2024-09-26 [1] RSPM
magrittr     2.0.3      2024-09-26 [1] RSPM
modelr        0.1.8      2024-09-26 [1] RSPM
munsell      0.5.1      2024-09-26 [1] RSPM
nloptr        1.2.9      2024-09-26 [1] RSPM
openxlsx     5.2.0      2024-09-26 [1] RSPM
parallel     4.4.2      2024-10-31 [3] local
purrr        1.0.1      2024-09-26 [1] RSPM
readr         2.2.0      2024-09-26 [1] RSPM
readxl        1.1.0      2024-09-26 [1] RSPM
rlang         1.1.0      2024-09-26 [1] RSPM
rmarkdown    2.20.0     2024-09-26 [1] RSPM
rstudioapi   0.20.0     2024-09-26 [1] RSPM
scales        1.2.1      2024-09-26 [1] RSPM
tibble       3.2.0      2024-09-26 [1] RSPM
tidyselect    1.2.0      2024-09-26 [1] RSPM
viridis      0.6.0      2024-09-26 [1] RSPM
viridisLite  0.4.0      2024-09-26 [1] RSPM
```

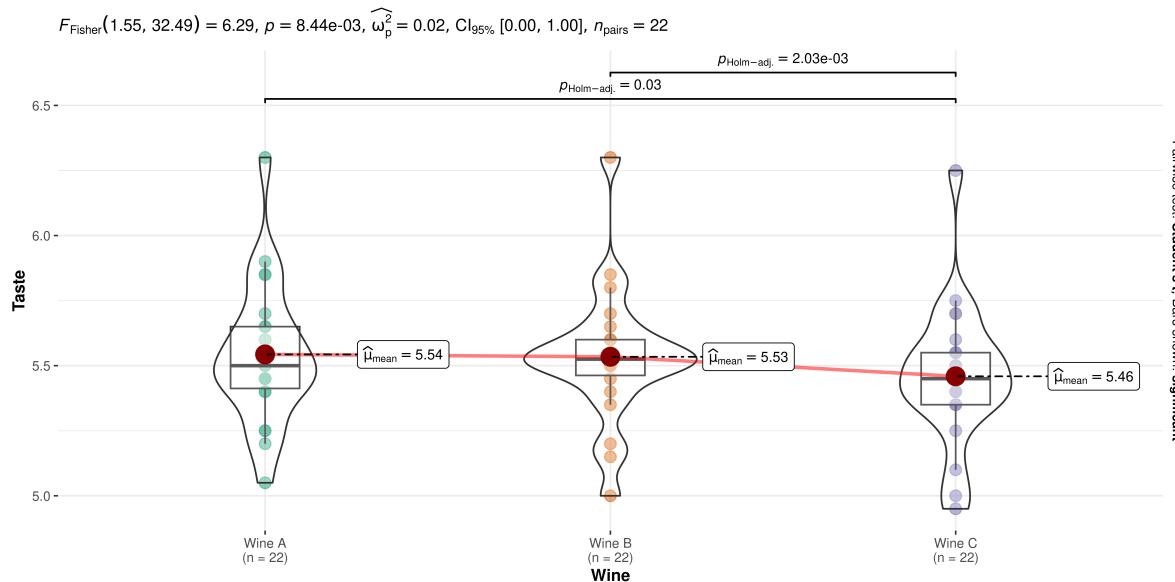
Appendix

Examples of other functions

ggwithinstats()

Hypothesis about group differences: repeated measures design

```
1 ggwithinstats(  
2   data = WRS2::WineTasting,  
3   x = Wine,  
4   y = Taste  
5 )
```



! Important

Defaults

- raw data + distributions
- descriptive statistics
- inferential statistics
- effect size + uncertainty
- pairwise comparisons
- Bayesian hypothesis-testing
- Bayesian estimation

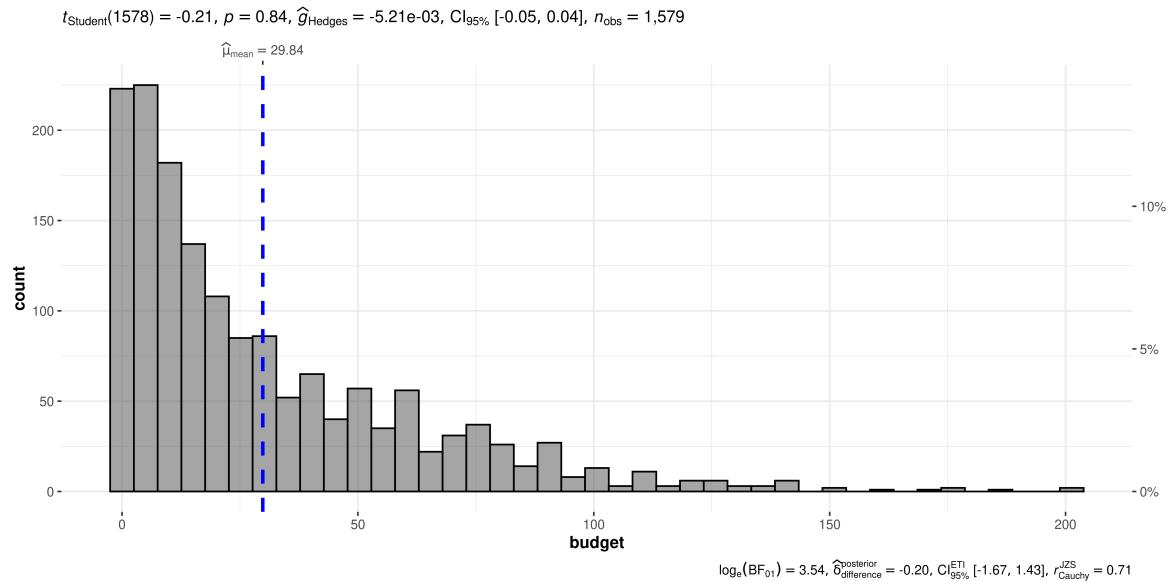
Statistical approaches available

- parametric
- parametric
- robust
- Bayesian

gghistostats()

Distribution of a numeric variable

```
1 gghistostats(  
2   data = movies_long,  
3   x = budget,  
4   test.value = 30  
5 )
```



! Important

Defaults

- counts + proportion for bins
- descriptive statistics
- inferential statistics
- effect size + uncertainty
- pairwise comparisons
- Bayesian hypothesis-testing
- Bayesian estimation

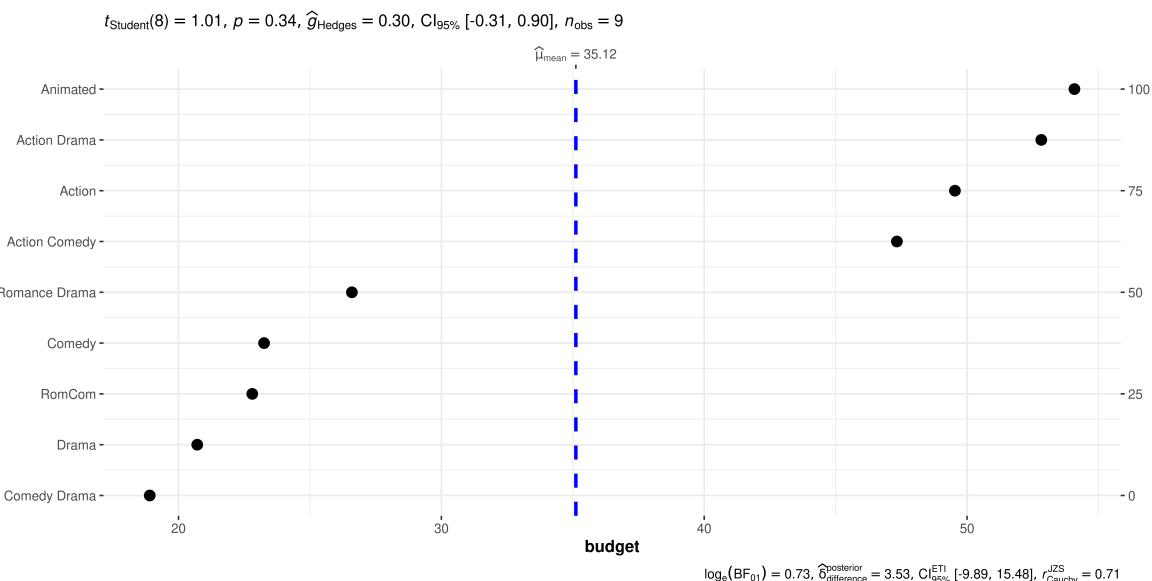
Statistical approaches available

- parametric
- parametric
- robust
- Bayesian

ggdotplotstats()

Labeled numeric variable

```
1 ggdotplotstats(  
2   data = movies_long,  
3   x = budget,  
4   y = genre,  
5   test.value = 30  
6 )
```



! Important

Defaults

- descriptive statistics
- inferential statistics
- effect size + uncertainty
- pairwise comparisons
- Bayesian hypothesis-testing
- Bayesian estimation

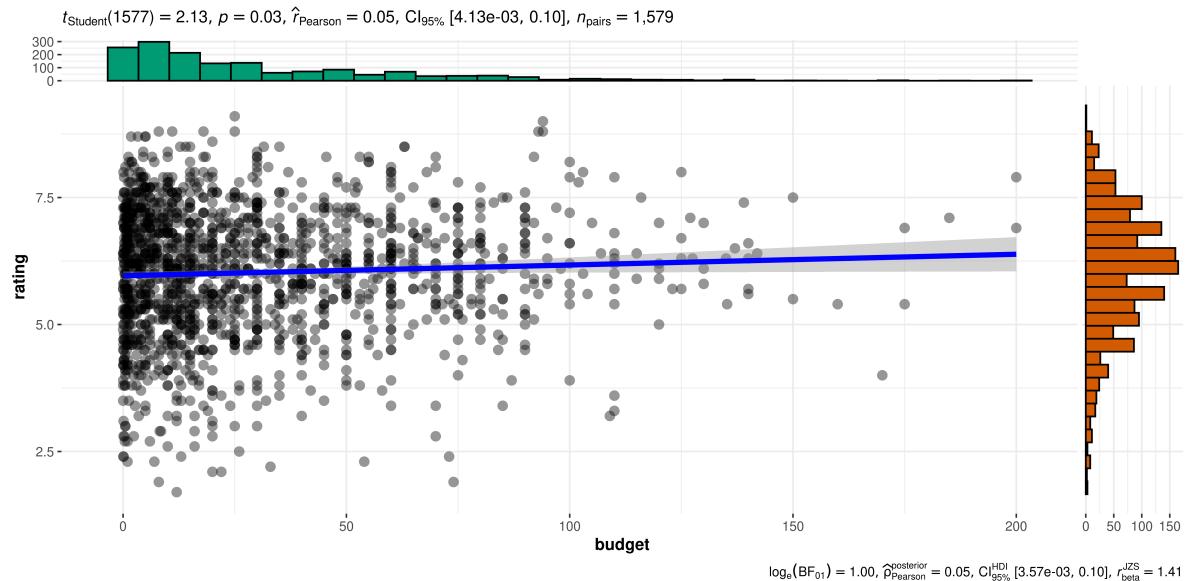
Statistical approaches available

- parametric
- parametric
- robust
- Bayesian

ggscatterstats()

Hypothesis about correlation: Two numeric variables

```
1 ggscatterstats(  
2   data = movies_long,  
3   x = budget,  
4   y = rating  
5 )
```



! Important

Defaults

- joint distribution
- marginal distribution
- effect size + uncertainty
- pairwise comparisons
- Bayesian hypothesis-testing
- Bayesian estimation

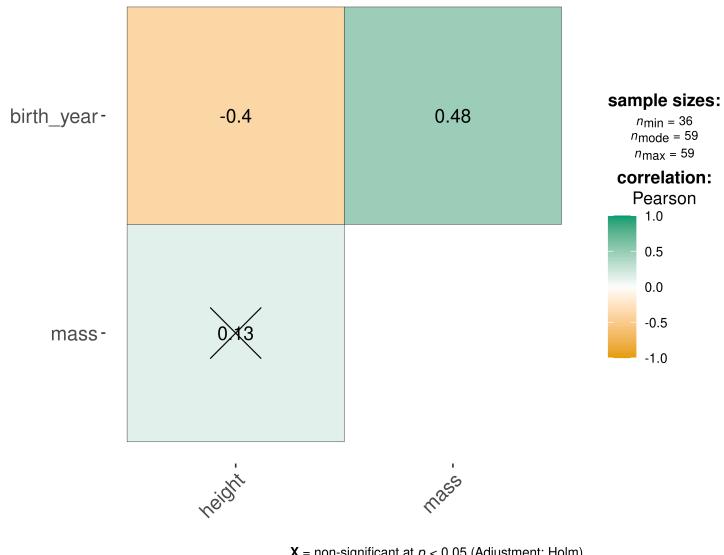
Statistical approaches available

- parametric
- parametric
- robust
- Bayesian

ggcorrmat ()

Hypothesis about correlation: Multiple numeric variables

```
1 ggcorrmat(dplyr::starwars)
```



! Important

Defaults

- inferential statistics
- effect size + uncertainty
- careful handling of [NAs](#)
- partial correlations

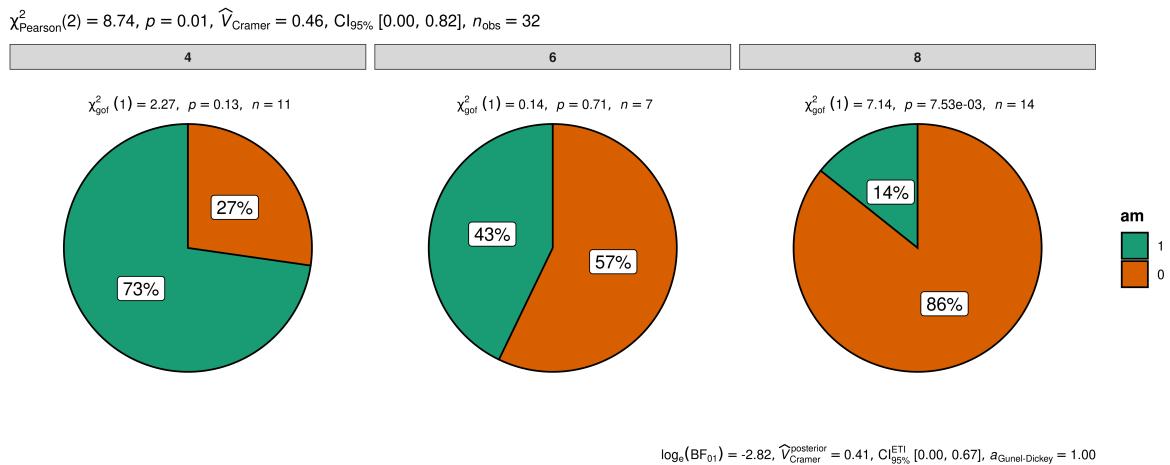
Statistical approaches available

- parametric
- parametric
- robust
- Bayesian

ggpiestats()

Hypothesis about composition of categorical variables

```
1 ggpiestats(  
2   data = mtcars,  
3   x = am,  
4   y = cyl  
5 )
```



! Important

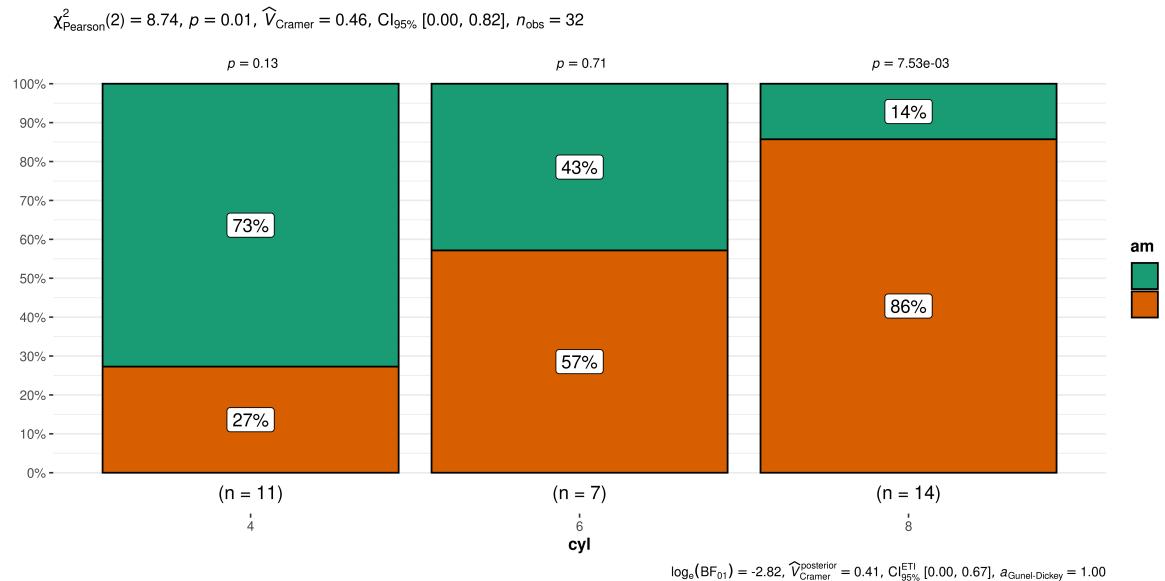
Defaults

- descriptive statistics
- inferential statistics
- effect size + uncertainty
- goodness-of-fit tests
- Bayesian hypothesis-testing
- Bayesian estimation

ggbarnstats()

Hypothesis about composition of categorical variables

```
1 ggbarnstats(  
2   data = mtcars,  
3   x = am,  
4   y = cyl  
5 )
```



! Important

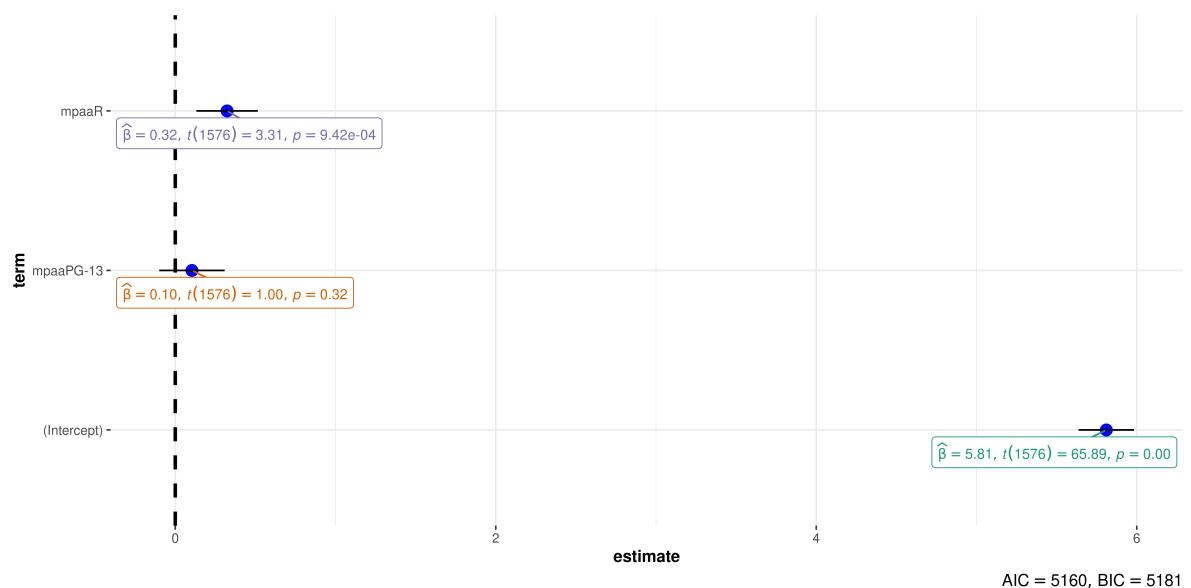
Defaults

- descriptive statistics
- inferential statistics
- effect size + uncertainty
- goodness-of-fit tests
- Bayesian hypothesis-testing
- Bayesian estimation

ggcoefstats()

Hypothesis about regression coefficients

```
1 mod <- lm(  
2   formula = rating ~ mpaa,  
3   data = movies_long  
4 )  
5  
6 ggcoefstats(mod)
```



! Important

Defaults

- estimate + uncertainty
- inferential statistics (t , z , F , χ^2)
- model fit indices (AIC + BIC)

Supports all regression models supported in `{easy stats}` ecosystem.

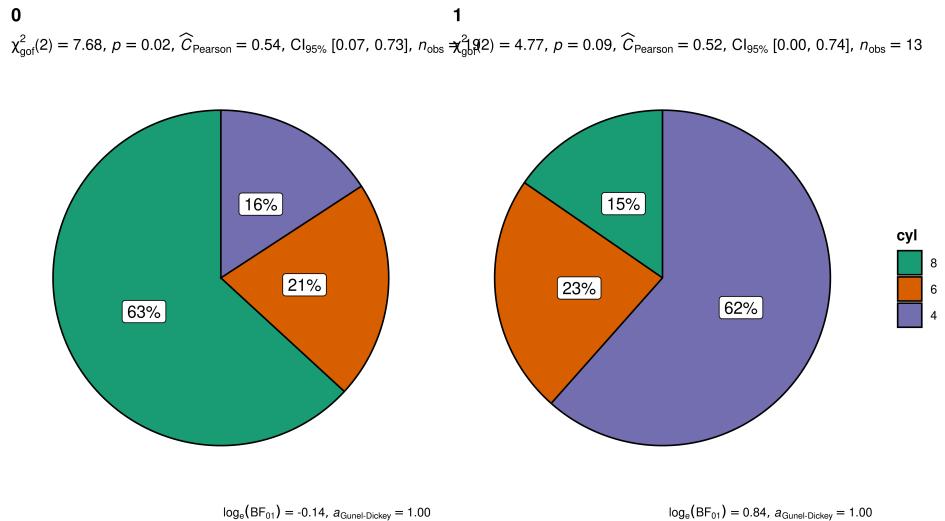
Meta-analysis is also supported!

grouped_ variants

Iterating over a grouping variable

grouped_ functions

```
1 grouped_ggpiestats(  
2   data = mtcars,  
3   x = cyl,  
4   grouping.var = am  
5 )
```



Available `grouped_` variants:

- `grouped_ggbetweenstats()`
- `grouped_ggwithinstats()`
- `grouped_gghistostats()`
- `grouped_ggdotplotstats()`
- `grouped_ggscatterstats()`
- `grouped_ggcormat()`
- `grouped_ggpiestats()`
- `grouped_ggbarstats()`

Customizability

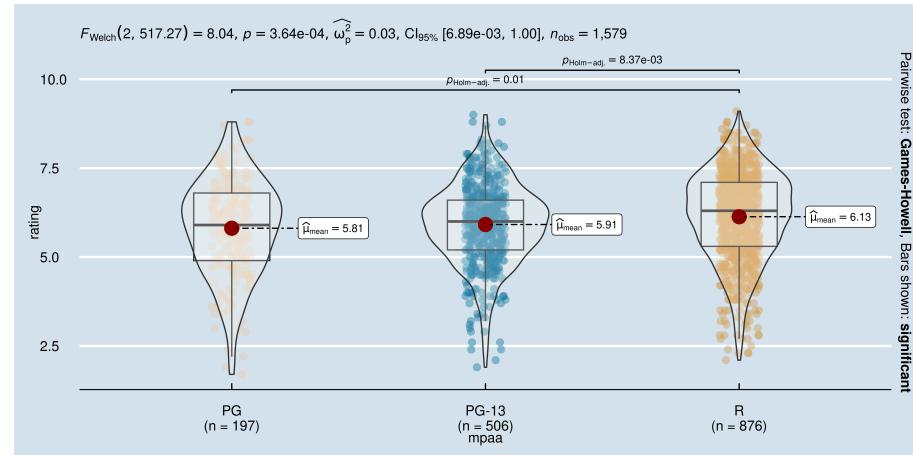
“What if I don’t like the default plots?” 🤔

Modify the look



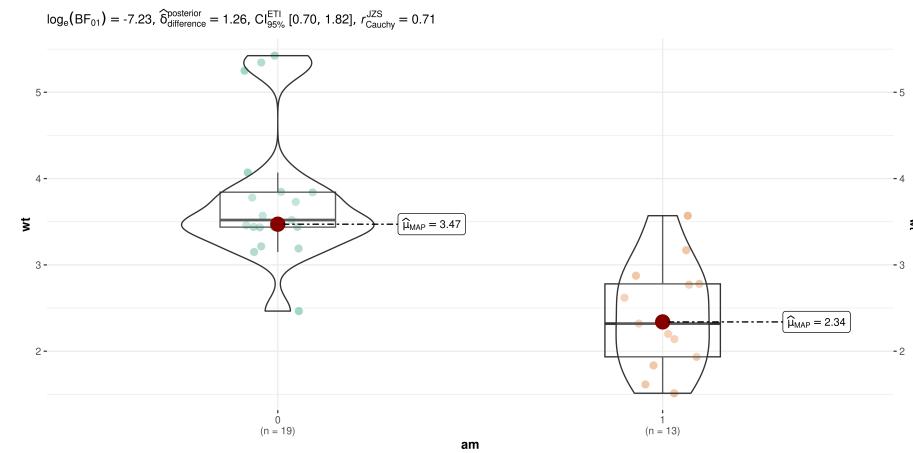
By changing **theme** and **palette**

```
1 ggbetweenstats(  
2   data = movies_long,  
3   x = mpaa,  
4   y = rating,  
5   ggtheme = ggthemes::theme_economist(),  
6   palette = "Darjeeling2",  
7   package = "wesanderson"  
8 )
```



By using `{ggplot2}` functions

```
1 ggbetweenstats(  
2   data = mtcars,  
3   x = am,  
4   y = wt,  
5   type = "bayes"  
6 ) +  
7   scale_y_continuous(sec.axis = dup_axis())
```

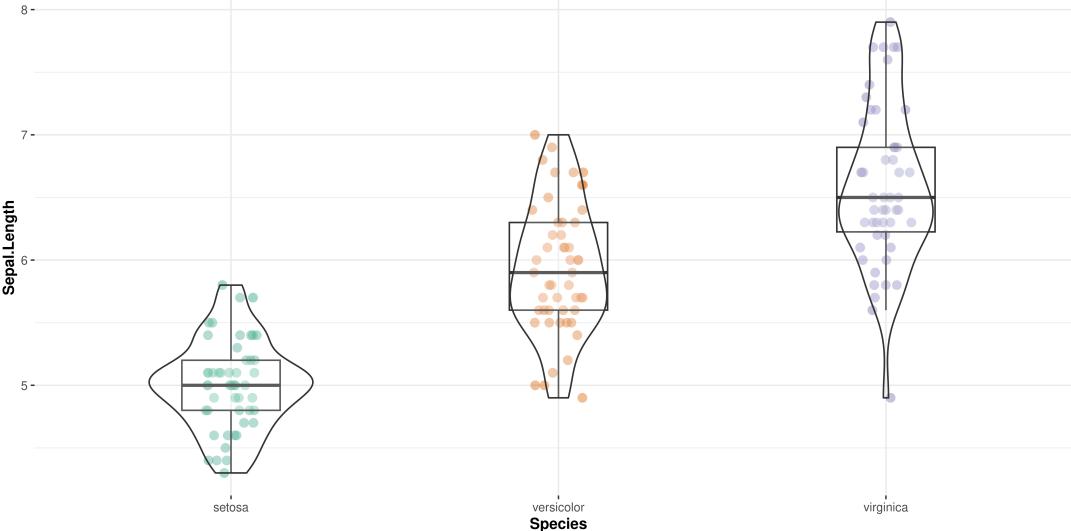


Too much information



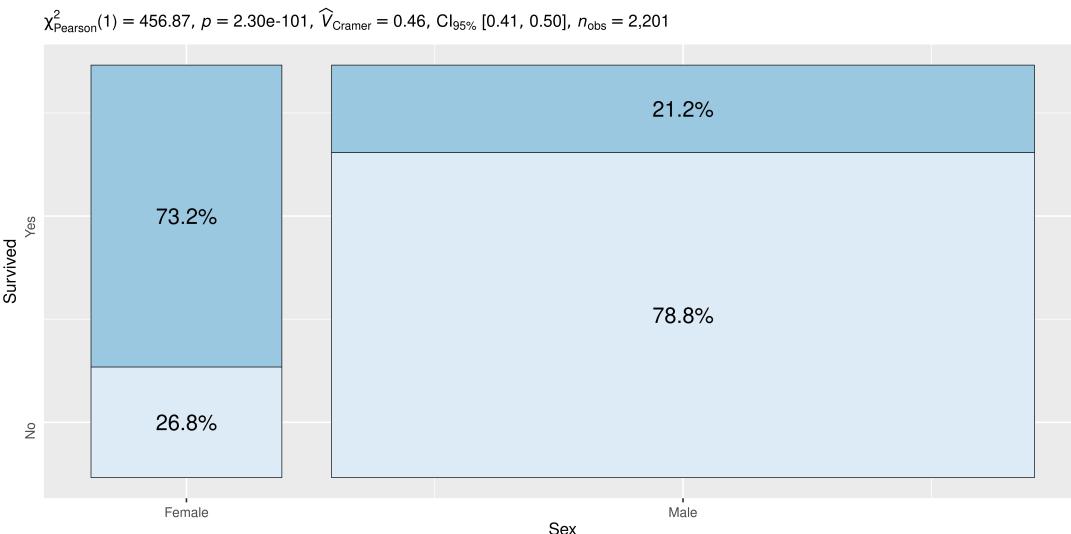
Get only plots:

```
1 ggbetweenstats(
2   data = iris,
3   x = Species,
4   y = Sepal.Length,
5   # turn off statistical analysis
6   centrality.plotting = FALSE,
7   results.subtitle = FALSE,
8   bf.message = FALSE,
9   # turn off pairwise comparisons
10 pairwise.display = "none"
11 )
```



Get only expressions:

```
1 stats_expr <- ggpiestats(
2   Titanic_full, Survived, Sex,
3 ) %>%
4   extract_subtitle()
5
6 ggiraphExtra:::ggSpine(
7   data = Titanic_full,
8   aes(x = Sex, fill = Survived)
9 ) +
10 labs(subtitle = stats_expr)
```



{ggstatsplot}: Details about statistical reporting

Supports different statistical approaches

Note

Functions	Description	Parametric	Non-parametric	Robust	Bayesian
<code>ggbetweenstats()</code>	Between group comparisons	✓	✓	✓	✓
<code>ggwithinstats()</code>	Within group comparisons	✓	✓	✓	✓
<code>gghistostats()</code> , <code>gddotplotstats()</code>	Distribution of a numeric variable	✓	✓	✓	✓
<code>ggcorrmat()</code>	Correlation matrix	✓	✓	✓	✓
<code>ggscatterstats()</code>	Correlation between two variables	✓	✓	✓	✓
<code>ggpiestats()</code> , <code>ggbarstats()</code>	Association between categorical variables	✓	NA	NA	✓
<code>ggpiestats()</code> , <code>ggbarstats()</code>	Equal proportions for categorical variable levels	✓	NA	NA	✓
<code>ggcoefstats()</code>	Regression modeling	✓	✓	✓	✓
<code>ggcoefstats()</code>	Random-effects meta-analysis	✓	NA	✓	✓

Toggling statistical approaches



Parametric

```
1 # anova
2 ggbetweenstats(
3   data = mtcars,
4   x = cyl,
5   y = wt,
6   type = "p"
7 )
8
9 # correlation analysis
10 ggscatterstats(
11   data = mtcars,
12   x = wt,
13   y = mpg,
14   type = "p"
15 )
16
17 # t-test
18 gghistostats(
19   data = mtcars,
20   x = wt,
21   test.value = 2,
22   type = "p"
23 )
```

Non-parametric

```
1 # anova
2 ggbetweenstats(
3   data = mtcars,
4   x = cyl,
5   y = wt,
6   type = "np"
7 )
8
9 # correlation analysis
10 ggscatterstats(
11   data = mtcars,
12   x = wt,
13   y = mpg,
14   type = "np"
15 )
16
17 # t-test
18 gghistostats(
19   data = mtcars,
20   x = wt,
21   test.value = 2,
22   type = "np"
23 )
```

Alternative: Pure Pain



Hunting for packages

- 📦 for inferential statistics (`{stats}`)
- 📦 computing effect size + CIs (`{effectsize}`)
- 📦 for descriptive statistics (`{skimr}`)
- 📦 pairwise comparisons (`{multcomp}`)
- 📦 Bayesian hypothesis testing (`{BayesFactor}`)
- 📦 Bayesian estimation (`{bayestestR}`)
- 📦 ...



Inconsistent APIs

- 🤔 accepts data frame, vector, matrix?
- 🤔 long/wide format data?
- 🤔 works with `NAs`?
- 🤔 returns data frame, vector, matrix?
- 🤔 works with tibbles?
- 🤔 has all necessary details?
- 🤔 ...



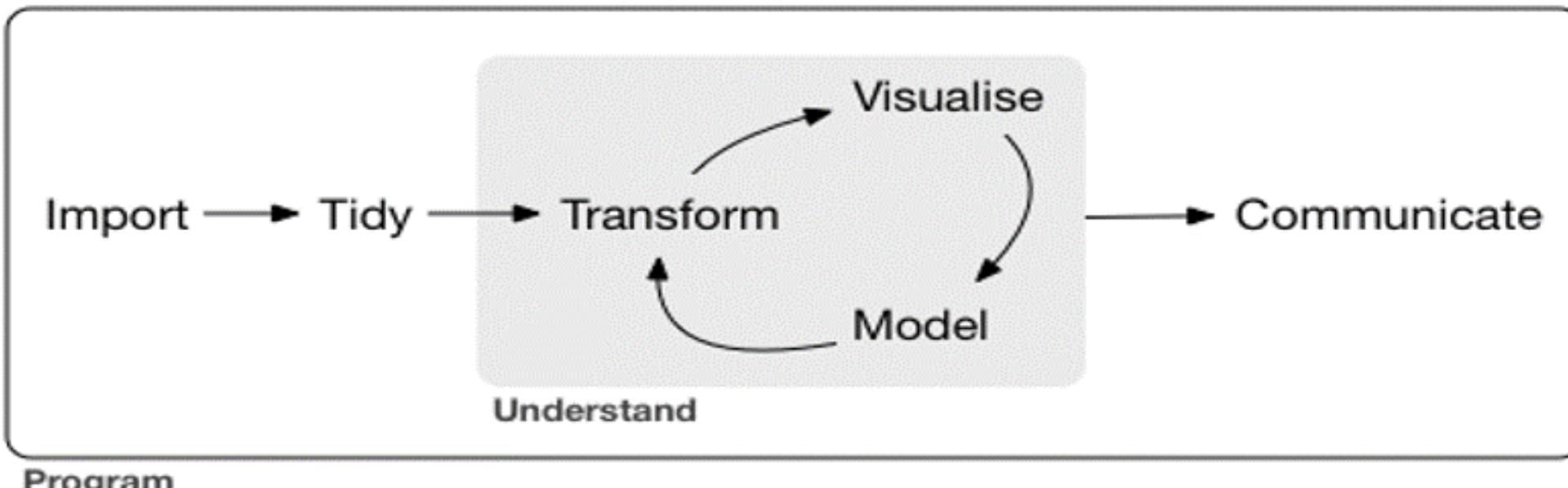
Benefits in details

`{ggstatsplot}` combines data visualization and statistical analysis in a single step.

It...

- provides ready-made plots with information-rich defaults
- minimizes the chances of making errors in statistical reporting
- follows best practices in data visualization and statistical reporting
- helps evaluate statistical analysis in the context of the underlying data
- highlights the importance of the effect by providing effect size measures
- provides an easy way to evaluate *absence* of an effect using Bayesian framework
- extremely beginner-friendly

Simplified data analysis workflow



Quick insight into data by combining visualization and modeling!

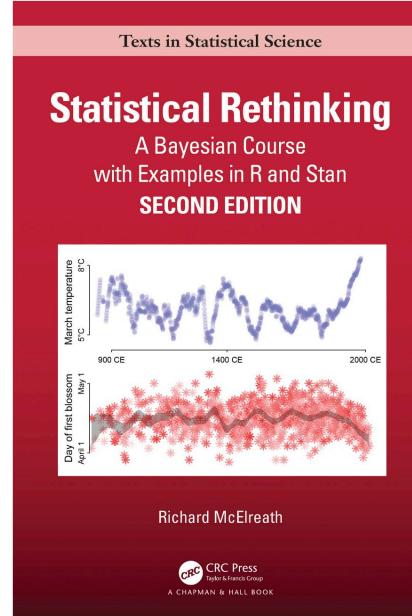
Community Involvement

- 11 contributors
- 3 reverse dependencies
- Widely covered in [YouTube videos](#) and social media posts
- Almost 100% resolution rate on [StackOverflow](#) (> 150 questions)
- Over 100 daily visitors on [GitHub repo](#)
- Usage in a wide [range of fields](#): psychology, biology, medicine, economics, etc.
- Usage in data science training programs

A grain of salt

The "Golem of Prague" problem

✗ Promotes mindless application of statistical tests.



No stable release yet.

Footnotes

1. (Nuijten et al., *Behavior Research Methods*, 2016)

2. (Aczel et al., *AMPPS*, 2018)

3. Open Science Collaboration, *Science*, 2015