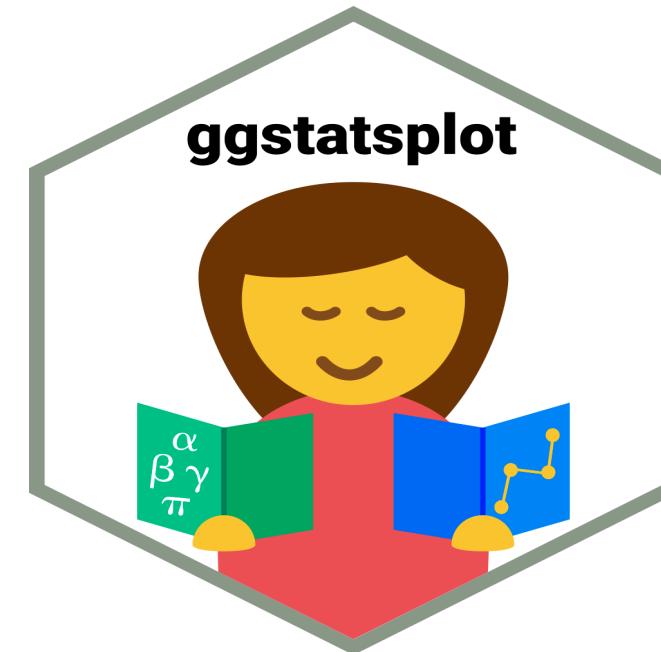


# {ggstatsplot}: Informative Statistical Visualizations

Indrajeet Patil



# Genesis

Why a new package?

# Life in the trenches (c. 2017)

(as an academic researcher)

## ⚠ Large-scale problems

- **Reporting errors:**

“half of all published psychology papers contained at least one  $p$ -value that was inconsistent”<sup>1</sup>

- **Interpretation errors:**

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

- **Replication crisis:**

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

and more...

## ⚠ Personal challenges

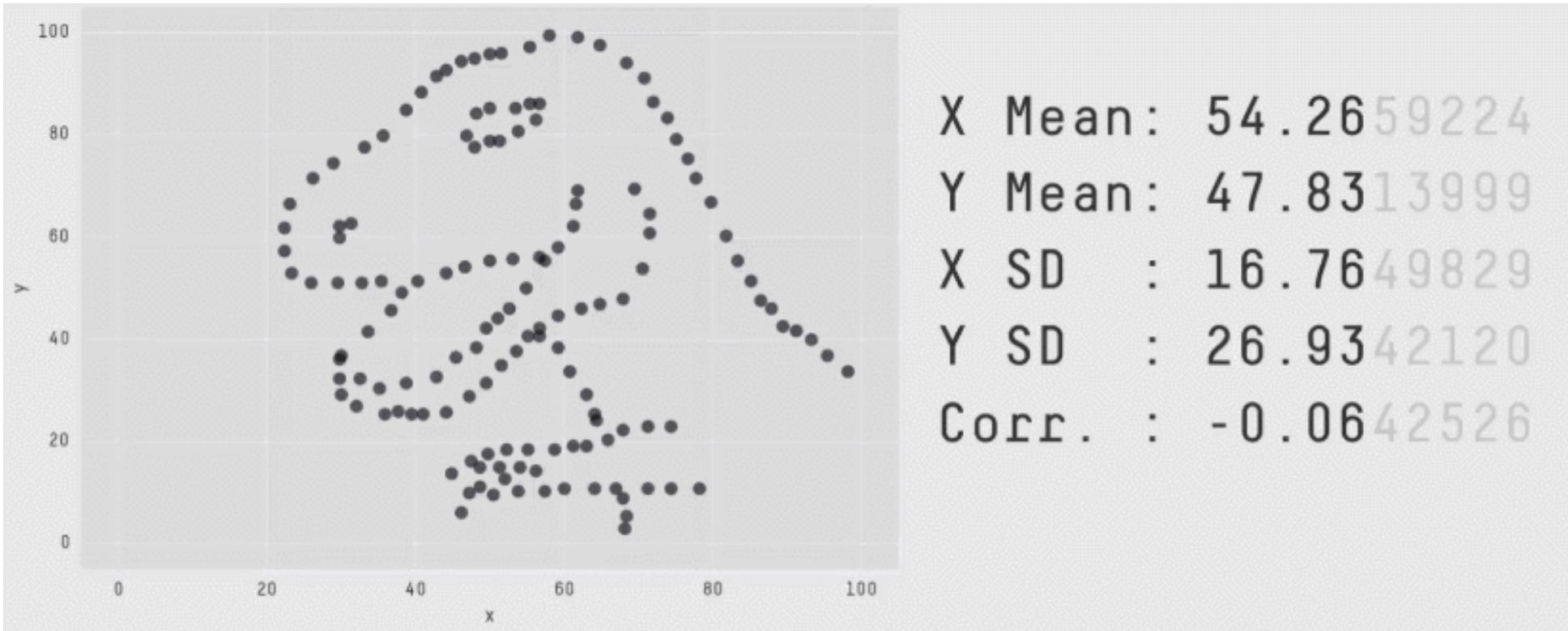


- How to avoid reporting errors?
- How to make visualizing data effortless?
- How to emphasize the importance of the effect?
- How to interpret null results?
- How to easily assess validity of model assumptions?
- How to increase reproducibility?

# Proposal

Information-rich, ready-made statistical visualizations

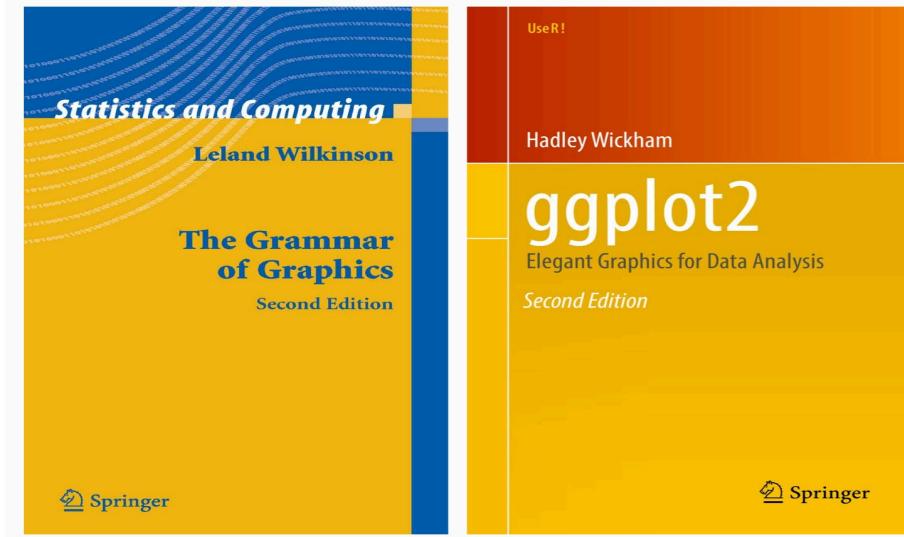
# Defaulting to a rich graphic



Visualizations reveal problems not discernible from descriptive statistics!

# Avoiding cooking a new plot

The grammar of graphics is a powerful framework and can help you make **any** data visualization!



💡 Using ready-made plots lowers the activation energy for visualizing data!

# Action Plan

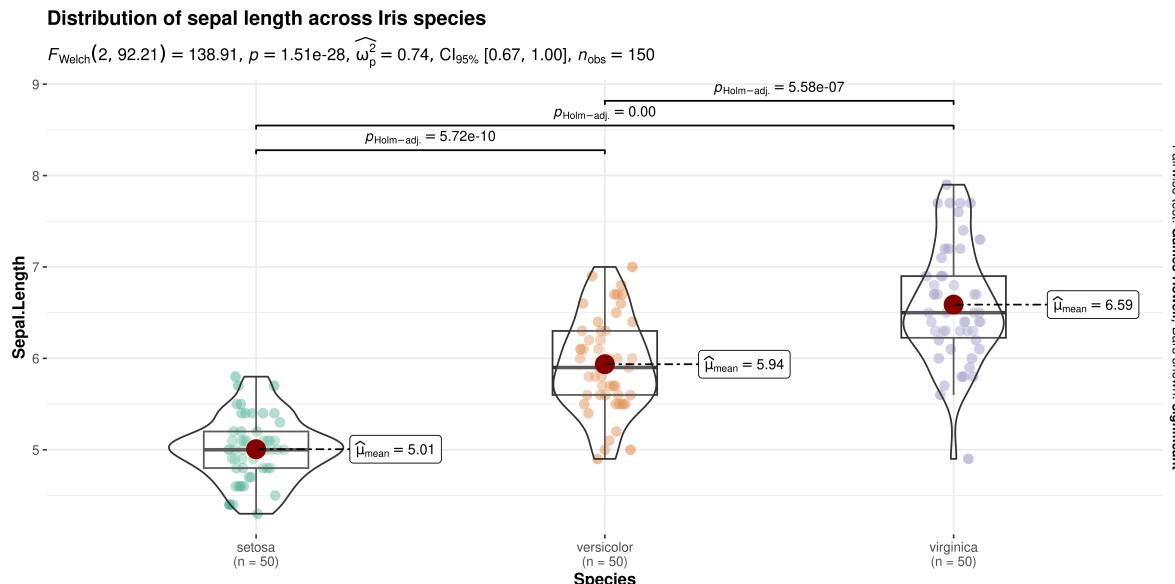
{ggstatsplot} was born!

(started out as loose scripts, which were later consolidated into a package)

# Example function

E.g., for hypothesis about group differences ([ggbetweenstats\(\)](#))

```
1 ggbetweenstats(  
2   data = iris,  
3   x     = Species,  
4   y     = Sepal.Length,  
5   title = "Distribution of sepal length across Iris species"  
6 )
```



## Important

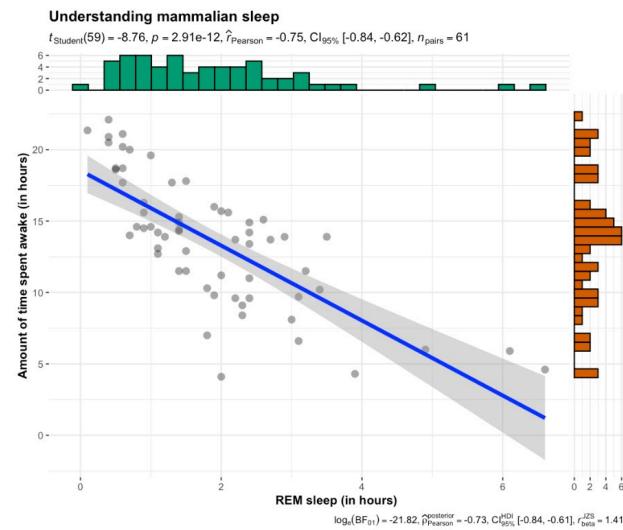
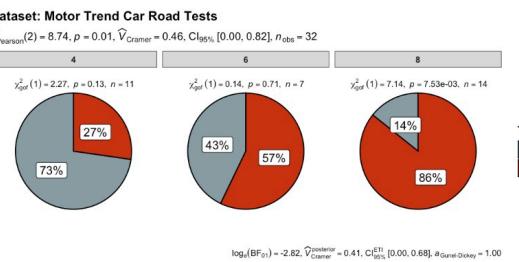
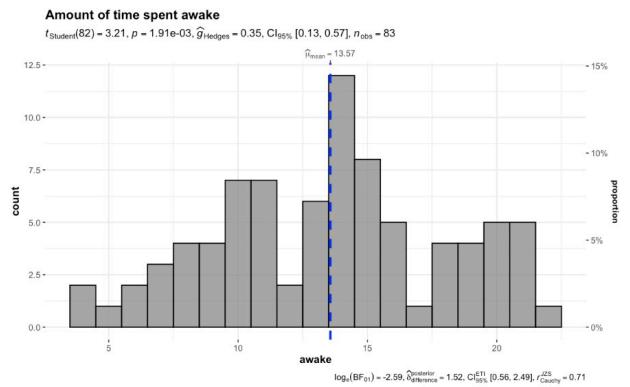
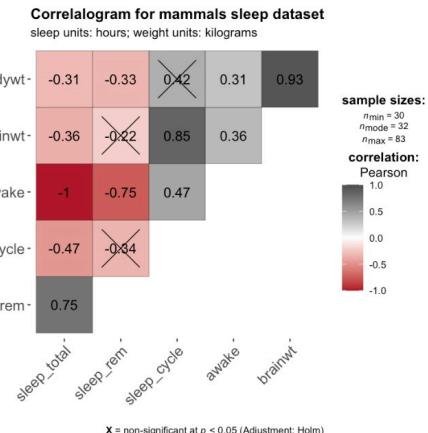
### Information-rich defaults

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

### Statistical approaches available

- parametric
- parametric
- robust
- Bayesian

# And there is more!



Appendix provides more details.

# Promised Land

Does it deliver?

# Show, don't tell

## Without `{ggstatsplot}`

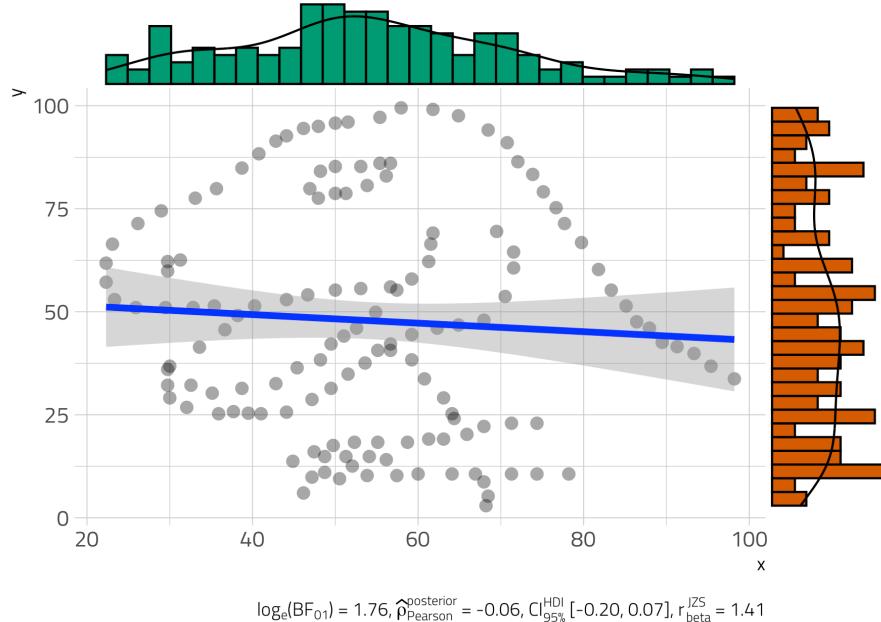
Pearson's correlation test revealed that, across 142 participants, variable `x` was negatively correlated with variable `y`:

$t(140) = -0.76, p = 0.446$ . 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}`

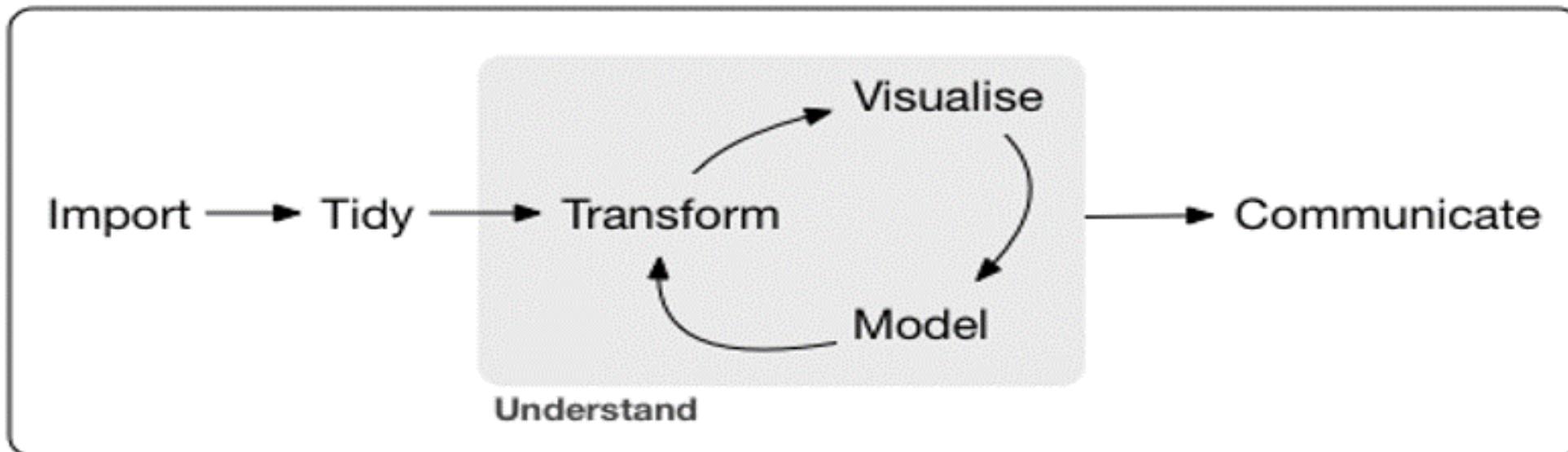
### Relationship between x and y

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



No need to worry about reporting or interpretation errors!

# Simplified data analysis workflow

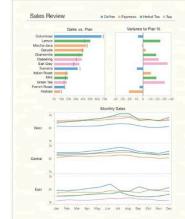


Quick insight into data by combining visualization and modeling!

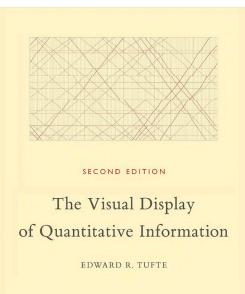
# Thoughtful Defaults

## >Data Visualization

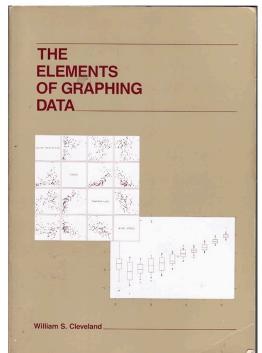
Second Edition  
Show Me the Numbers  
Designing Tables and Graphs to Enlighten



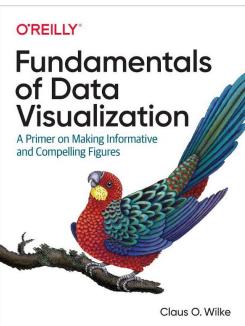
Stephen Few



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



William S. Cleveland



Claus O. Wilke

## Statistical Reporting



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

Template for Frequentist analysis

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

Template for Bayesian analysis

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

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

src: @patilindrajeets

(Doorn et al., 2020; APA Manual)

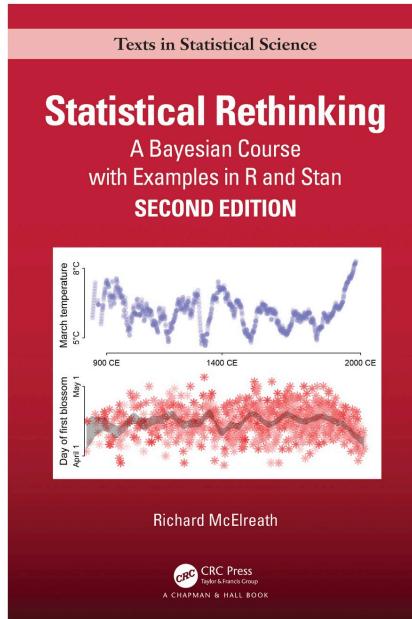


Encourages best practices in data visualization and statistical reporting!

# A grain of salt

The "Golem of Prague" problem

Promotes mindless application of statistical tests.



## Clunky API

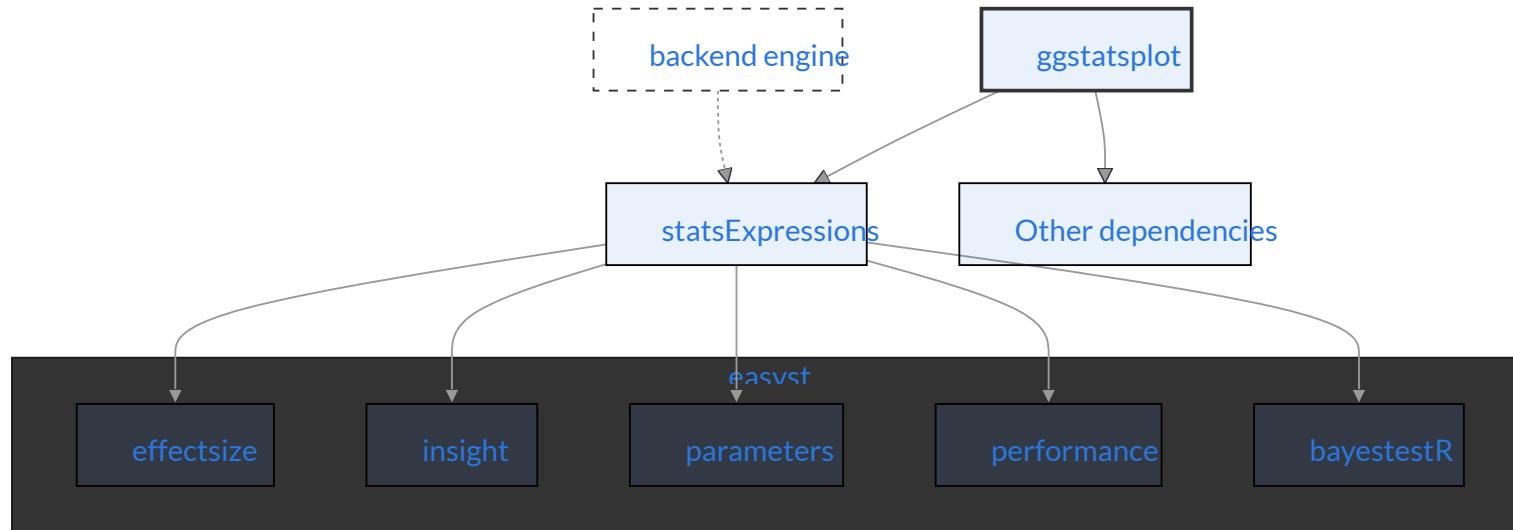
- Not a “real” `{ggplot2}` extension.
- Statistical proficiency needed.
- Not interactive.
- No stable (`1.0`) release yet.

# Pleasant Side Effects

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

# Software Architecture

Breaking down the monolith:  $20K \rightarrow 1K$  lines of code



## 💡 Team sport

While re-architecting `{ggstatsplot}`, I decided to contribute to upstream open-source dependencies:

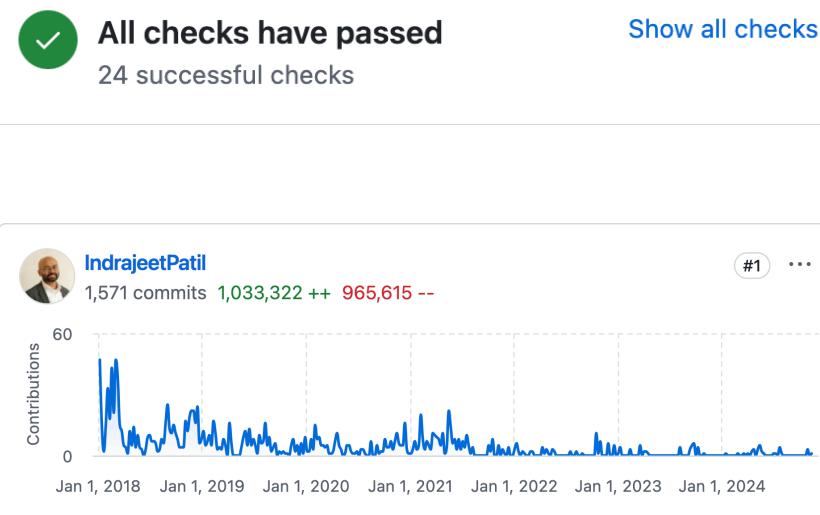
- joined the core `{easystats}` team and contributing to its ten component packages
- became a co-maintainer of `{ggsignif}`
- contributed to `{WRS2}` and `{ggcorrplot}`

# Quality Assurance

## CI Checks (GitHub Actions)

- Unit tests (random-order)
- Code coverage (100%)
- Linting (0 lints)
- Formatting (0 issues)
- Documentation (website, no link rot, many examples)
- CRAN checks (0 E, 0 W, 0 N)
- Pre-commit hooks (0 issues)
- Portability (Linux, macOS, Windows)
- Robustness (dependencies, R versions)

## Healthy and active code base



## Team sport

While improving QA tools for `{ggstatsplot}`, in the spirit of open-source, I decided to contribute to upstream dependencies:

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

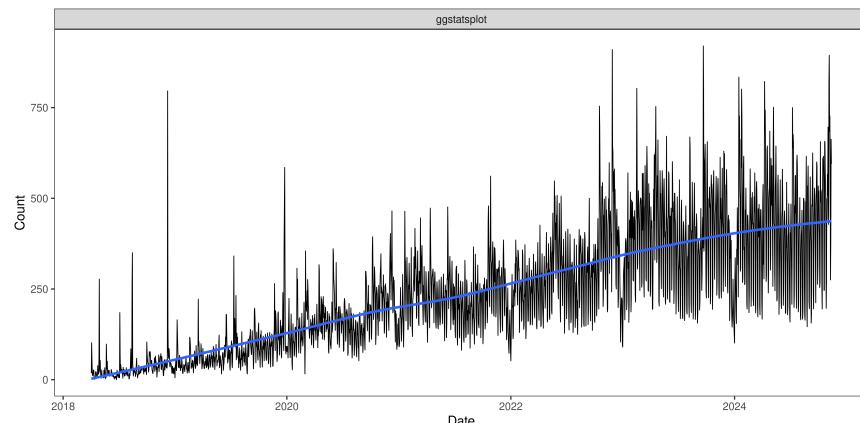
# Impact

*I can haz users?!*

# User Love



Total downloads > 500K (97 percentile)



Second most starred `{ggplot2}`-extension!



`ggstatsplot`

Public



Enhancing `{ggplot2}` plots with statistical analysis 📈



2k



190



Total citations > 1000



Visualizations with statistical details: The 'ggstatsplot' approach

Indrajeet Patil<sup>1</sup>

<sup>1</sup> Center for Humans and Machines, Max Planck Institute for Human Development, Berlin, Germany

## 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.

Visualizations with statistical details: The 'ggstatsplot' approach

1277  
\*

I Patil

Journal of Open Source Software 6 (61),  
3167, 2021

# Conclusion

The `{ggstatsplot}` package provides accessible interface to create rich statistical visualizations, and helps users adopt a rigorous, reliable, and robust data exploratory and reporting workflow.

# 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

Thank You 😊

# 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-az1052-290
ui        X11
language (EN)
collate  C.UTF-8
ctype    C.UTF-8
tz       UTC
date     2024-11-13
pandoc   3.5 @ /opt/hostedtoolcache/pandoc/3.5/x64/ (via rmarkdown)
quarto   1.6.33 @ /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
. . .
```

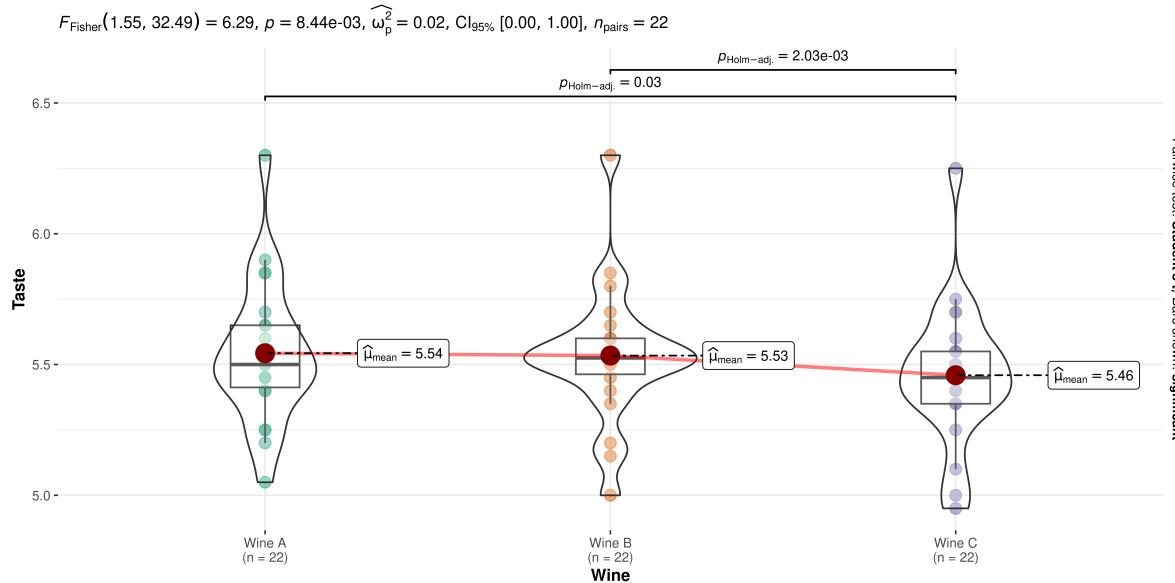
# Appendix

# 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

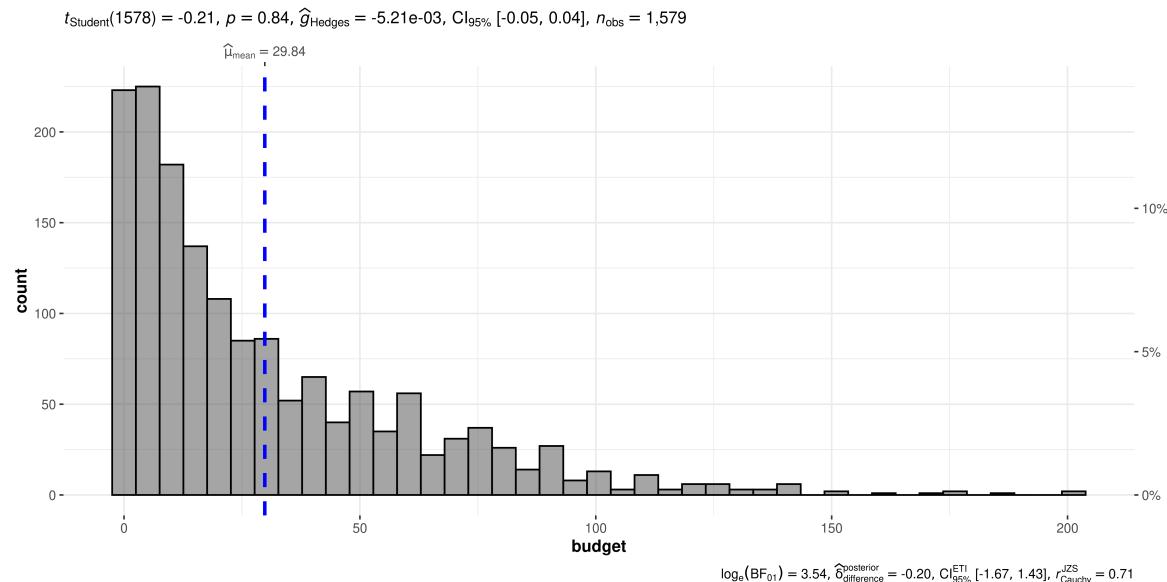
#### Statistical approaches available

- 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

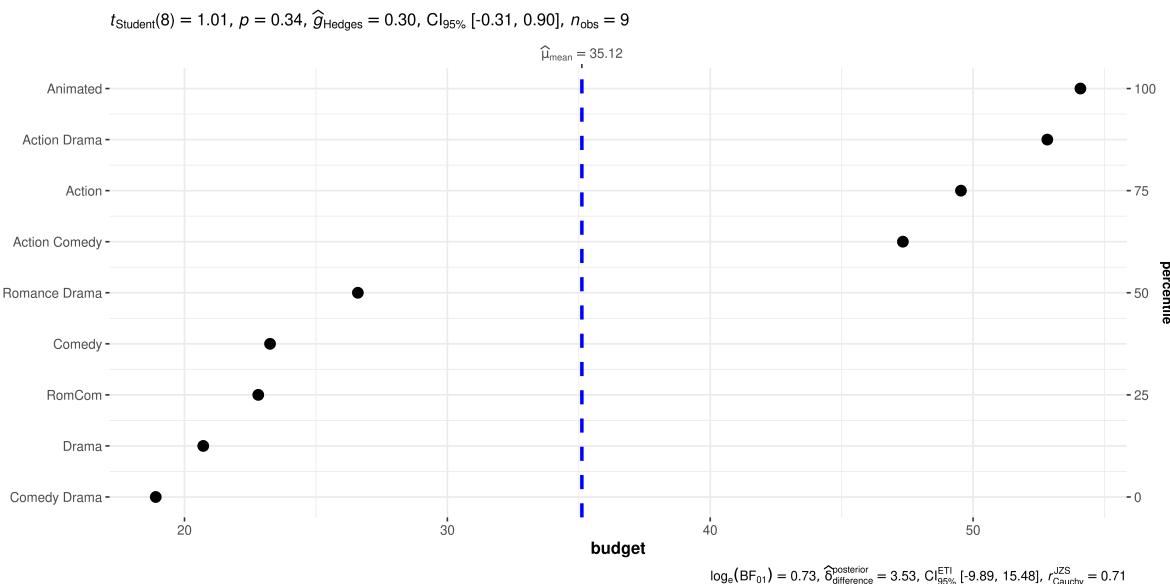
#### Statistical approaches available

- 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

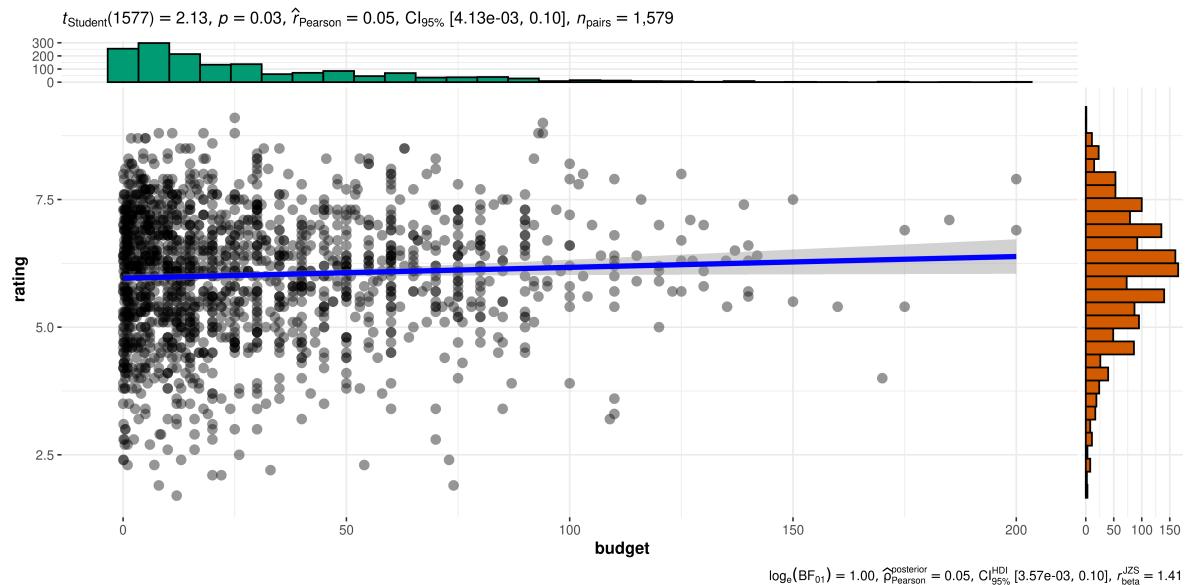
#### Statistical approaches available

- 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

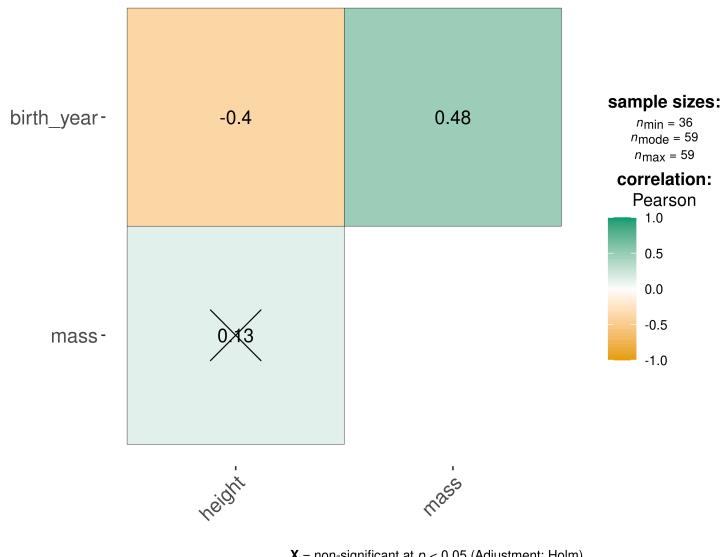
### Statistical approaches available

- 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

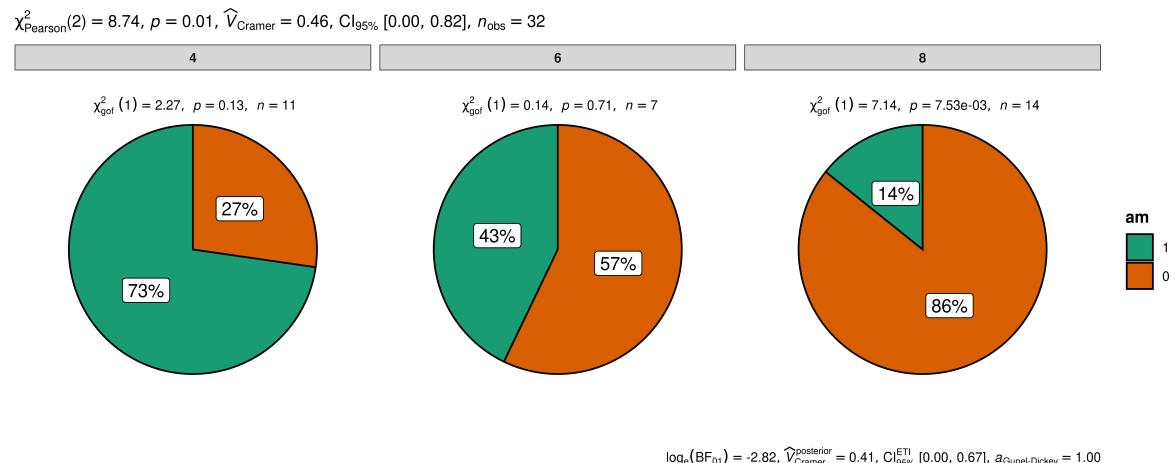
**Statistical approaches available**

- parametric
- parametric
- robust
- Bayesian

## ggpiestats()

### Hypothesis about composition of categorical variables

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



#### ! Important

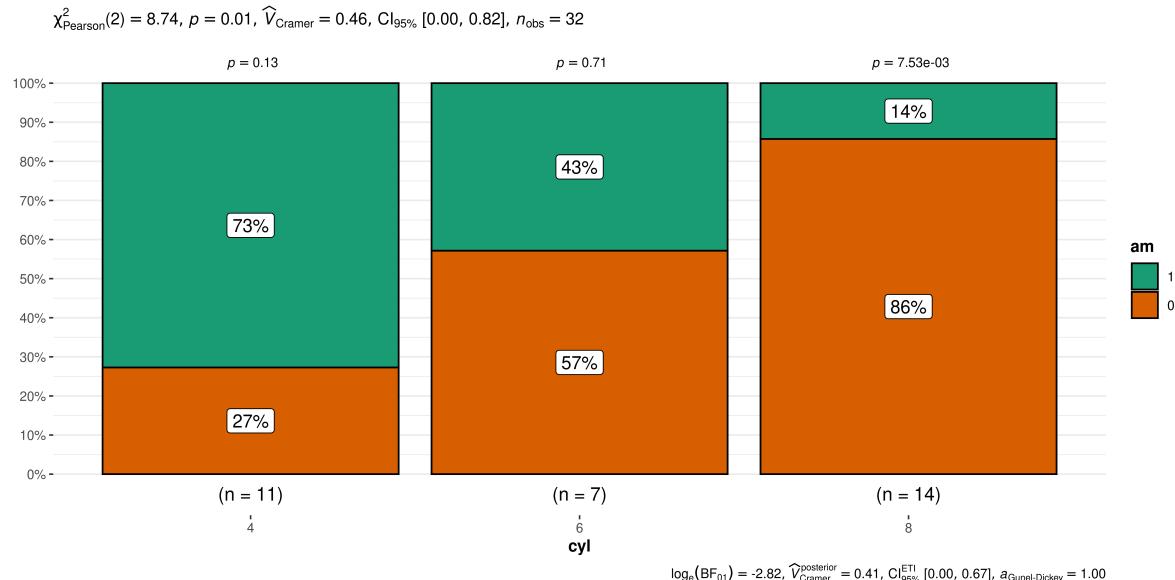
#### Defaults

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

## ggbarstats()

### Hypothesis about composition of categorical variables

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



#### ! Important

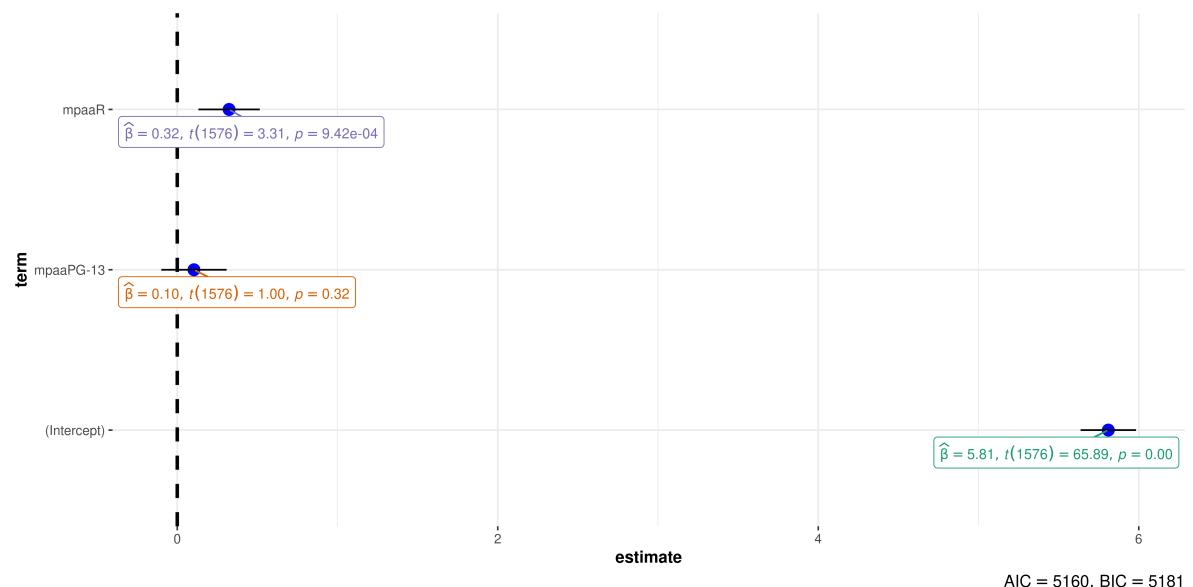
#### Defaults

- 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 ( $t, z, F, \chi^2$ )
- model fit indices (AIC + BIC)

Supports all regression models supported in `{easystats}` ecosystem.

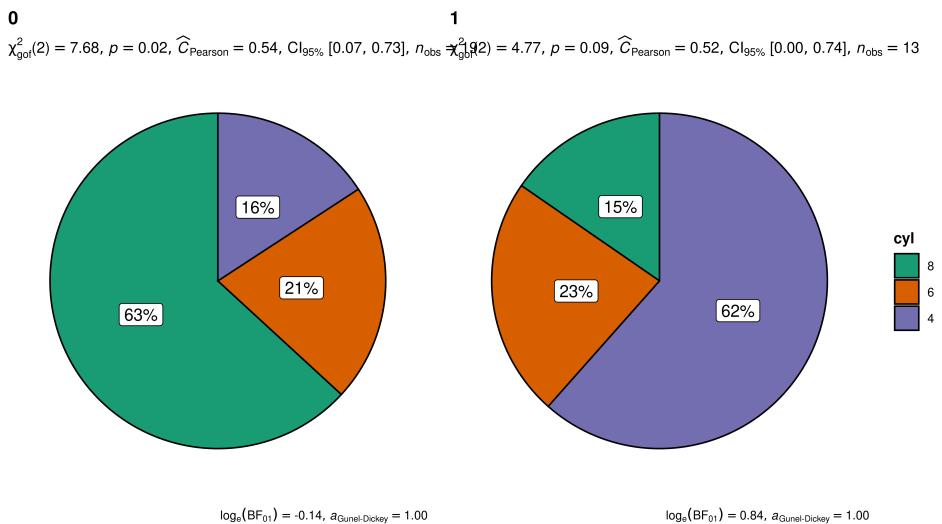
Meta-analysis is also supported!

# *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_ggcorrmat()`
- `grouped_ggpiestats()`
- `grouped_ggbarstats()`

# Customizability

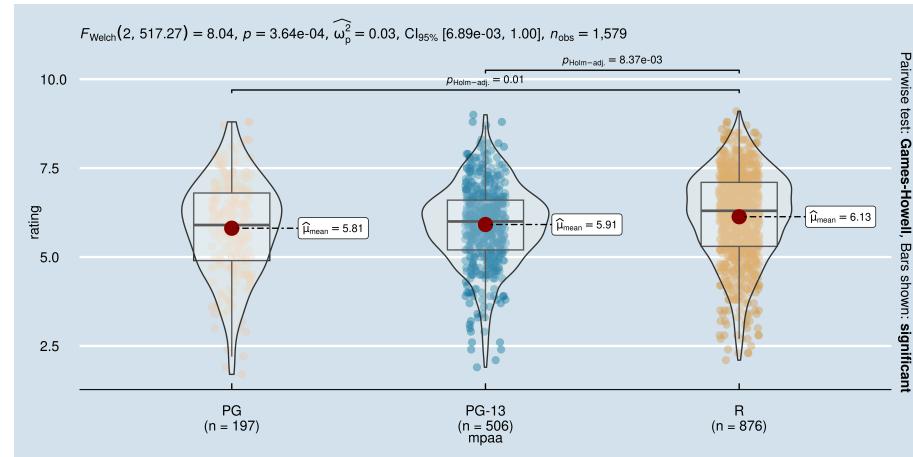
“What if I don’t like the default plots?” 🤔

# Modify the look



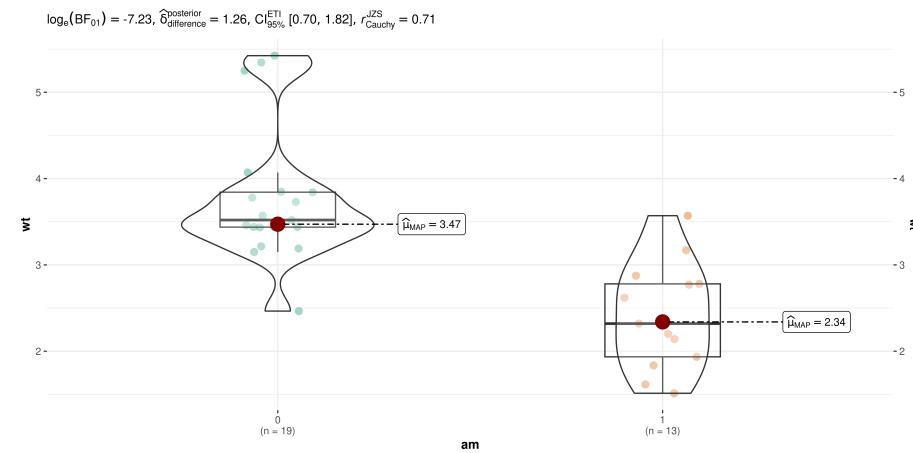
By changing **theme** and **palette**

```
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())
```

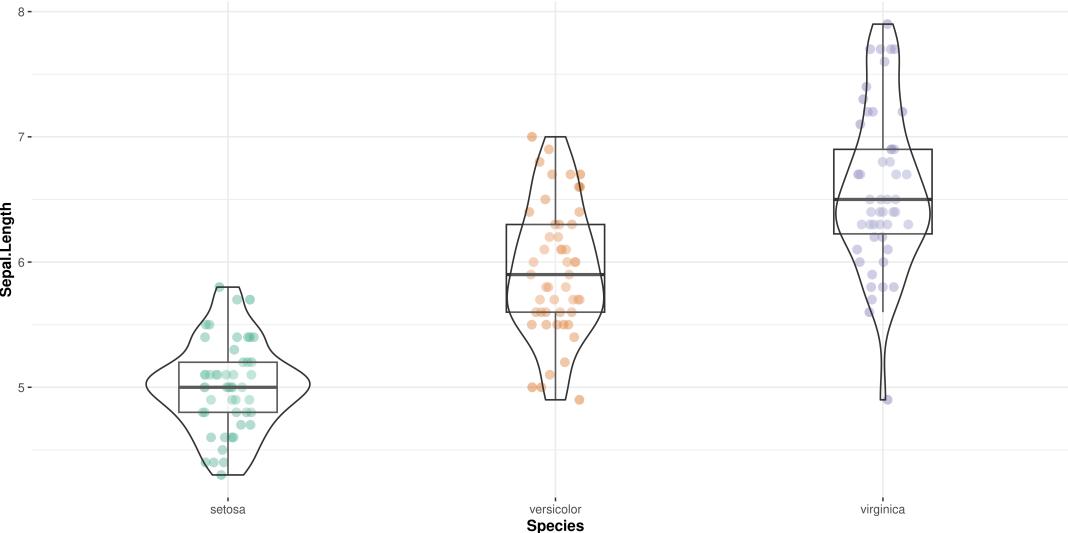


# Too much information



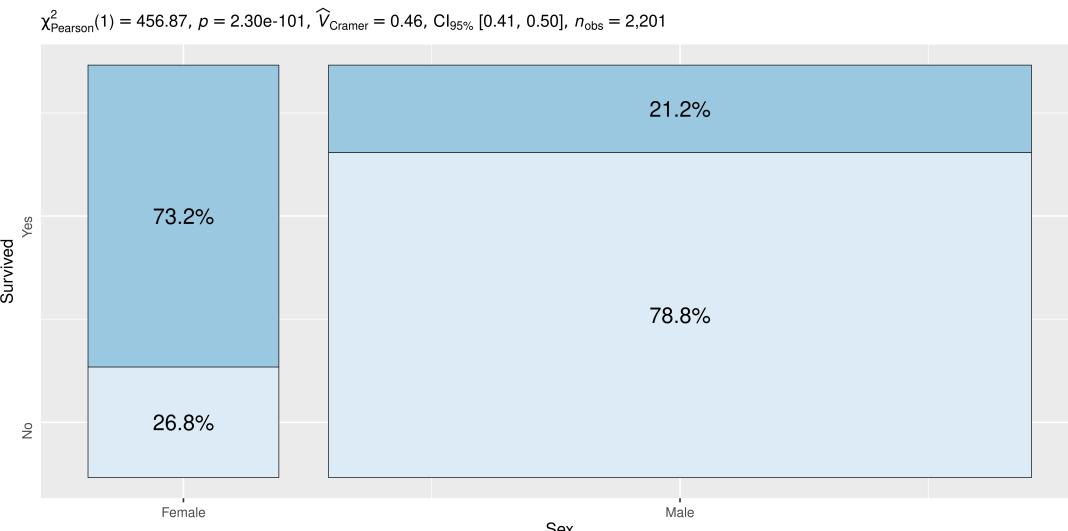
Get only plots:

```
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>ggcorrrmat()</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

# Footnotes

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

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

3. Open Science Collaboration, *Science*, 2015