Package: APCinteraction (via r-universe)

May 24, 2026

Type Package

Title Nonparametric Interaction Tests in Balanced Two-Way ANOVA Models

Version 0.1.0

Description Provides novel nonparametric tests, 'APCSSA' and 'APCSSM', for interaction in two-way ANOVA designs with balanced replications using all possible comparisons. These statistics extend previous methods, allow greater flexibility, and demonstrate higher power in detecting interactions for non-normal data. The package includes optimized functions for computing these test statistics, generating interaction plots, and simulating their null distributions. The companion package 'APCinteractionData' is available on 'GitHub' <https://github.com/tranbaokhue/APCinteractionData>. Methods are described and compared empirically in Tran, Wagaman, Nguyen, Jacobson, and Hartlaub (2024) <doi:10.48550/arXiv.2410.04700>.

License MIT + file LICENSE

Encoding UTF-8

URL https://github.com/tranbaokhue/APCinteraction

BugReports https://github.com/tranbaokhue/APCinteraction/issues

RoxygenNote 7.3.3

Roxygen list(markdown = TRUE)

Depends R (>= 4.0.0)

Imports magrittr, dplyr, parallel, pbapply, rlang

Suggests APCinteractionData, testthat (>= 3.0.0)

Config/testthat/edition 3

Repository https://baokhuetran.r-universe.dev

Date/Publication 2026-03-02 02:00:20 UTC

RemoteUrl https://github.com/baokhuetran/apcinteraction

RemoteRef HEAD

RemoteSha 6ae9c075e306a8470eb6735b3cf7ef6d9e6aa2ee

Contents

`APCinteraction` *APCinteraction: Nonparametric Interaction Tests in Balanced Two-Way ANOVA Models*

Description

Provides nonparametric tests for interaction in two-way ANOVA designs with balanced replications using All Possible Comparisons (APC). The `APCSSA` and `APCSSM` statistics extend previous methods, allow greater flexibility, and demonstrate higher power in detecting interactions for non-normal data. The package includes optimized functions for computing these test statistics, generating interaction plots, and simulating their null distributions.

Main functions

`APCSSA` Compute the APCSSA interaction test statistic.

`sim_nullAPCSSA` Simulate the null distribution for the APCSSA statistic.

`APCSSM` Compute the APCSSM interaction test statistic.

`sim_nullAPCSSM` Simulate the null distribution for the APCSSM statistic.

`save_null` Save the simulated null distributions locally for future usage.

Author(s)

Bao Khue Tran (maintainer) <baokhuetran@outlook.com>, Andrew Nguyen, Amy Wagaman, and Bradley Hartlaub

See Also

Useful links:

- https://github.com/tranbaokhue/APCinteraction
- Report bugs at https://github.com/tranbaokhue/APCinteraction/issues

APCSSA *APCSSA*

Description

This function tests for interaction in a two-way layout using all possible crossed comparisons (APC) based on aligned ranks, with alignment performed using the mean.

Usage

```
APCSSA(formula, data, numSim = 1e+05)
```

Arguments

`formula`	A formula specifying the model, with one response variable and two experimental factors.
`data`	A data frame in long format: the first column contains observed values of the response variable, the second column contains the levels of Factor A, and the third column contains the levels of Factor B.
`numSim`	An integer specifying the number of simulations used to estimate the null distribution. Defaults to 100,000 (1e+05). Even though this default might take longer to run for designs with factors of more than 5 levels, we recommend using at least 100,000 simulations to get reliable results.

Details

`APCSSA` performs a nonparametric test for interaction in a two-way layout with balanced replication. Two statistics, APCCRA and APCRCA, are computed by aligning the data with the averages and ranking within the rows or columns.

These statistics use all possible crossed comparisons to detect interaction effects. The final test statistic is the maximum of the two standardized statistics. A *p*-value is estimated by comparing this statistic to a pre-simulated null distribution specific to the design dimensions.

The companion package **APCinteractionData** provides 90 pre-computed APCSSA null distributions (100,000 simulations each) covering two-way designs with 2 to 6 factor levels and 1 to 10 replications. Install it with `devtools::install_github("tranbaokhue/APCinteractionData")` to enable fast p-value estimation for these designs.

If the design is not covered by the pre-computed distributions or if a higher precision is desired (`numSim` > 100,000), users must first generate the null distribution using `sim_nullAPCSSA` before using this function to evaluate the interaction effect.

For more information, see the referenced article.

Value

An object of class `"APC"`, which is a data frame with columns:

- **Interaction**: The interaction term.

- **Statistic**: The APCSSA test statistic (maximum of APCCRA and APCRCA).
- **p.value**: The estimated *p*-value.

Use `summary.APC` to print a formatted summary table and generate an interaction plot.

References

Tran, B. K., Wagaman, A. S., Nguyen, A., Jacobson, D., & Hartlaub, B. (2024). Nonparametric tests for interaction in two-way ANOVA with balanced replications. *arXiv preprint* arXiv:2410.04700. https://arxiv.org/abs/2410.04700

See Also

`summary.APC`, `sim_nullAPCSSA`, `APCSSM`

Examples

```
# Set the seed for reproducibility
set.seed(206)

# Parameters - number of levels for factors A and B, and replications
nA <- 3; nB <- 3; nrep <- 3

# Generate levels and create the full design
A_vals <- 1:nA
B_vals <- 1:nB
design <- expand.grid(A = A_vals, B = B_vals)
design <- design[rep(seq_len(nrow(design)), each = nrep), ]

# Compute response as interaction (product of numeric factor levels)
mu <- A_vals[design$A] * B_vals[design$B]
value <- mu + rnorm(length(mu))  # Add normal noise

# Assemble the data
data <- data.frame(
  value = value,
  A = factor(design$A),
  B = factor(design$B)
)

# Run the APCSSA test and view summary
result <- APCSSA(value ~ A + B, data = data)
summary(result)
```

APCSSM *APCSSM*

Description

This function tests for interaction in a two-way layout using all possible crossed comparisons (APC) based on aligned ranks, with alignment performed using the median.

Usage

```
APCSSM(formula, data, numSim = 1e+05)
```

Arguments

`formula`	A formula specifying the model, with one response variable and two experimental factors.
`data`	A data frame in long format: the first column contains observed values of the response variable, the second column contains the levels of Factor A, and the third column contains the levels of Factor B.
`numSim`	An integer specifying the number of simulations used to estimate the null distribution. Defaults to 100,000 (1e+05). Even though this default might take longer to run for designs with factors of more than 5 levels, we recommend using at least 100,000 simulations to get reliable results.

Details

`APCSSM` performs a nonparametric test for interaction in a two-way layout with balanced replication. Two statistics, APCCRM and APCRCM, are computed by aligning the data with the medians and ranking within the rows or columns.

These statistics use all possible crossed comparisons to detect interaction effects. The final test statistic is the maximum of the two standardized statistics. A *p*-value is estimated by comparing this statistic to a pre-simulated null distribution specific to the design dimensions.

The companion package **APCinteractionData** provides 90 pre-computed APCSSM null distributions (100,000 simulations each) covering two-way designs with 2 to 6 factor levels and 1 to 10 replications. Install it with `devtools::install_github("tranbaokhue/APCinteractionData")` to enable fast p-value estimation for these designs.

If the design is not covered by the pre-computed distributions or if a higher precision is desired (`numSim` > 100,000), users must first generate the null distribution using `sim_nullAPCSSM` before using this function to evaluate the interaction effect.

For more information, see the referenced article.

Value

An object of class `"APC"`, which is a data frame with columns:

- **Interaction**: The interaction term.

- **Statistic**: The APCSSM test statistic (maximum of APCCRM and APCRCM).
- **p.value**: The estimated p-value.

Use `summary.APC` to print a formatted summary table and generate an interaction plot.

References

Tran, B. K., Wagaman, A. S., Nguyen, A., Jacobson, D., & Hartlaub, B. (2024). Nonparametric tests for interaction in two-way ANOVA with balanced replications. *arXiv preprint* arXiv:2410.04700. https://arxiv.org/abs/2410.04700

See Also

`summary.APC`, `sim_nullAPCSSM`, `APCSSA`

Examples

```
# Set seed for reproducibility
set.seed(36)

# Parameters - number of levels for factos A and B, replications, and interaction effect
nA <- 3; nB <- 4; nrep <- 2; c <- 1.25

# Generate levels and create the full design
A_vals <- seq(-2, 2, length.out = nA)
B_vals <- seq(-1.5, 1.5, length.out = nB)
design <- expand.grid(A = 1:nA, B = 1:nB)
design <- design[rep(seq_len(nrow(design)), each = nrep), ]

# Create the specific interaction matrix: alternate ±c in top rows
specInt <- matrix(0, nA, nB)
specInt[1:2, 1:nB] <- matrix(c(c, -c, -c, c), nrow = 2, byrow = TRUE)

# Compute response
mu <- A_vals[design$A] + B_vals[design$B] + specInt[cbind(design$A, design$B)]
value <- mu + rt(length(mu), df = 1)

# Assemble the data
data <- data.frame(
  value = value,
  A = factor(design$A, labels = round(A_vals, 2)),
  B = factor(design$B, labels = round(B_vals, 2))
)

# With Cauchy errors, we opt for APCSSM to check for interaction
result <- APCSSM(value ~ A + B, data = data)
summary(result)
```

save_null *Save null distribution to a directory*

Description

Saves a null distribution from the package's cache to RData files in the specified directory.

Usage

```
save_null(type, i, j, k, path)
```

Arguments

type	Character string: either "APCSSA" or "APCSSM."
i	The number of levels for factor A.
j	The number of levels for factor B.
k	The number of replications at each combination of Factor A and Factor B.
path	Character string specifying the directory to save the files to.

Value

Invisibly returns the file paths to where the data were saved.

See Also

sim_nullAPCSSA, sim_nullAPCSSM

Examples

```
## Not run:
# First generate the desired null distribution
sim_nullAPCSSA(2, 2, 2, 50000)

# Save to a temporary directory
save_null("APCSSA", 2, 2, 2, path = tempdir())

# Or save to a custom location
save_null("APCSSA", 2, 2, 2, path = "path/to/directory")

## End(Not run)
```

`sim_nullAPCSSA` *Simulate the null distribution for APCSSA*

Description

A function for simulating the null distribution of APCSSA.

Usage

```
sim_nullAPCSSA(i, j, k, numSim = 1e+05, parallel = TRUE, verbose = TRUE)
```

Arguments

`i`	The number of levels for factor A.
`j`	The number of levels for factor B.
`k`	The number of replications at each combination of Factor A and Factor B.
`numSim`	The number of simulations (data sets) to generate. Defaults to 100,000 (1e+05). Even though this default might take longer to run for designs with factors of more than 5 levels, we recommend using at least 100,000 simulations to get reliable results.
`parallel`	Boolean; if TRUE (the default), uses one fewer than the total number of cores for parallel computation.
`verbose`	Logical; if TRUE (the default), prints a startup notice and progress bar.

Details

While we have provided null distributions for various settings in this package, there are still two-way settings that we don't readily have the null distribution to evaluate the significance of the test statistics (see **References** for the complete list of settings available). For example, one might wish to estimate the *p*-value with a higher `numSim`, e.g., 250,000.

Accompanying this function is our customized `save_null` to help researchers save the null distribution generated for future reference and calculations of the statistics in similar experiment design.

Value

A data frame with the null mean and standard deviation for the two test statistics (APCCRA and APCRCA) that will get standardized into APCSSA and a numeric vector with length equal to the `numSim` of all APCSSA statistics on the null data sets.

Note

Benchmarks for 100,000 simulations (no bias toward any chip type):

- **3×3×3** (243 summands): ~ 5 hours on an Intel i5
- **2×5×5** (625 summands): < 5 hours on Apple M1 (7 cores, parallel)

- **6×6×5** (9375 summands): ~ 108 hours on Apple M1 (7 cores, parallel) Factor levels (i,j) have some impact, but total summands scale as *k^4* (see **References** for more information), so replication depth (k) dominates run time.

Tip: leave `parallel = TRUE` to shorten run times dramatically.

Results are stored in the package's cache. The customized function `save_null` is highly recommended for saving the null distribution to your local storage for future reference and testing in the same design.

References

Tran, B. K., Wagaman, A. S., Nguyen, A., Jacobson, D., & Hartlaub, B. (2024). Nonparametric tests for interaction in two-way ANOVA with balanced replications. *arXiv preprint* arXiv:2410.04700. https://arxiv.org/abs/2410.04700

See Also

`save_null`, `APCSSA`, `sim_nullAPCSSM`

Examples

```
## Not run:
sim_nullAPCSSA(2, 2, 2, 50000)

## End(Not run)
```

`sim_nullAPCSSM` *Simulate the null distribution for APCSSM*

Description

A function for simulating the null distribution of APCSSM

Usage

```
sim_nullAPCSSM(i, j, k, numSim = 1e+05, parallel = TRUE, verbose = TRUE)
```

Arguments

`i`	The number of levels for factor A.
`j`	The number of levels for factor B.
`k`	The number of replications at each combination of Factor A and Factor B.
`numSim`	The number of simulations (data sets) to generate. Defaults to 100,000 (1e+05). Even though this default might take longer to run for designs with factors of more than 5 levels, we recommend using at least 100,000 simulations to get reliable results.

`parallel`	Boolean; if TRUE (the default), uses one fewer than the total number of cores for parallel computation.
`verbose`	Logical; if TRUE (the default), prints a startup notice and progress bar.

Details

While we have provided null distributions for various settings in this package, there are still two-way settings that we don't readily have the null distribution to evaluate the significance of the test statistics (see **References** for the complete list of settings available). For example, one might wish to estimate the *p*-value with a higher `numSim`, e.g., 250,000.

#' Accompanying this function is our customized `save_null` to help researchers save the null distribution generated for future reference and calculations of the statistics in similar experiment design.

Value

A data frame with the null mean and standard deviation for the two test statistics (APCCRM and APCRCM) that will get standardized into APCSSM and a numeric vector with length equal to the `numSim` of all APCSSM statistics on the null data sets.

Note

Benchmarks for 100,000 simulations (no bias toward any chip type):

- **3×3×3** (243 summands): ~5 hours on an Intel i5
- **2×5×5** (625 summands): < 5 hours on Apple M1 (7 cores, parallel)
- **6×6×5** (9375 summands): ~108 hours on Apple M1 (7 cores, parallel) Factor levels (i,j) have some impact, but total summands scale as *k^4* (see **References** for more information), so replication depth (k) dominates run time.

Tip: leave `parallel = TRUE` to shorten run times dramatically.

Results are stored in the package's cache. The customized function `save_null` is highly recommended for saving the null distribution to your local storage for future reference and testing in the same design.

References

Tran, B. K., Wagaman, A. S., Nguyen, A., Jacobson, D., & Hartlaub, B. (2024). Nonparametric tests for interaction in two-way ANOVA with balanced replications. *arXiv preprint* arXiv:2410.04700. https://arxiv.org/abs/2410.04700

See Also

`save_null`, `APCSSM`, `sim_nullAPCSSA`

Examples

```
## Not run:
sim_nullAPCSSM(2, 2, 2, 50000)

## End(Not run)
```

`summary.APC` *Summary method for APC objects*

Description

Prints a formatted summary table and generates an interaction plot for objects returned by APCSSA or APCSSM.

Usage

```
## S3 method for class 'APC'
summary(object, ...)
```

Arguments

`object`	An object of class "APC", as returned by APCSSA or APCSSM.
`...`	Additional arguments to be passed to or from other methods.

Value

Invisibly returns the input object.

See Also

APCSSA, APCSSM

Index