

Package: solomonR (via r-universe)

October 9, 2026

Title Analyze Solomon Four-Group Designs

Version 0.8.2

Date 2026-10-09

Description Tools for planning, analyzing, teaching, and reporting Solomon four-group experiments, introduced by Solomon (1949) <[doi:10.1037/h0062958](https://doi.org/10.1037/h0062958)>. The historical analyses are reproduced and labeled as such, from Solomon's improvement scores to the Tests A-I sequence of Walton Braver and Braver (1988) <[doi:10.1037/0033-2909.104.1.150](https://doi.org/10.1037/0033-2909.104.1.150)> and its published revisions. Contemporary analyses include one generalized linear model for all the groups, of four-group designs and of designs with several treatments, with robust or cluster-robust standard errors, for continuous, binary, and count outcomes; randomization inference; full-information maximum likelihood, following van Engelenburg (1999) <<https://eric.ed.gov/?id=ED435692>>; observed- and latent-variable structural equation models with 'lavaan'; and equivalence tests of pretest sensitization. Also includes design checks, power and sample-size planning, simulated teaching data, published data sets, and reports in the style of the American Psychological Association, with method-specific references.

License GPL-3

Copyright See the file inst/COPYRIGHTS for the copyright holders and licenses of the bundled published data.

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

Depends R (>= 4.1)

Imports stats, broom, dplyr, sandwich, clubSandwich, generics, MASS, ggplot2, tibble, withr, lifecycle

Suggests testthat (>= 3.0.0), knitr, rmarkdown, lavaan, mmmrm

VignetteBuilder knitr

URL <https://github.com/JUhalt/solomonR>,
<https://juhalt.github.io/solomonR/>,
<https://juhalt.r-universe.dev/solomonR>

BugReports <https://github.com/JUhalt/solomonR/issues>

LazyData true

Config/roxygen2/version 8.1.0

Config/pak/sysreqs libicu-dev

Repository <https://juhalt.r-universe.dev>

Date/Publication 2026-10-09 13:36:25 UTC

RemoteUrl <https://github.com/JUhalt/solomonR>

RemoteRef v0.8.2

RemoteSha 17b1632a9e607dc6e2f3cf053bf599b4780c5759

Contents

analysis_plan_solomon	3
baseline_solomon	6
check_solomon_assumptions	8
check_solomon_missing	10
compare_solomon_methods	12
elkarkri2025a	14
equivalence_solomon	16
fisher_solomon	18
fit_solomon_1949	20
fit_solomon_classic	22
fit_solomon_glm	26
fit_solomon_mi	32
fit_solomon_ml	35
fit_solomon_mmrn	37
fit_solomon_sem	40
fit_solomon_sem_latent	42
fit_solomon_steyn	45
invariance_solomon	50
jordaan2014	53
kvalem1996	55
lana1959	56
mai2020	57
marginal_solomon	59
p_to_z	63
perm_solomon	63
plan_solomon	66
plot_classic_flow	69
plot_perm	70
plot_power_solomon	71

plot_sensitization	72
plot_solomon	74
plot_solomon_change	75
plot_solomon_design	76
plot_solomon_effects	78
plot_solomon_gg	79
plot_solomon_means	80
plot_tipping_point	81
power_solomon	82
report_solomon	84
simulate_solomon	86
solomon_demo	89
solomon_effect_sizes	90
solomon_example	92
solomon_from_summary	93
solomon1949	95
solomonR	97
steyn2005	98
stouffer_solomon	100
tipping_point_solomon	101
validate_solomon	103

Index**107**

analysis_plan_solomon *Write an analysis plan for a Solomon four-group study*

Description

[Experimental] Writes an editable Markdown analysis plan, for preregistration or a protocol, from a few choices and, optionally, a `plan_solomon()` result. The plan follows the three sections of van 't Veer and Giner-Sorolla's (2016, Appendix A) preregistration template (hypotheses, methods, and analysis plan) and names the SPIRIT 2013 item each part answers (Chan et al., 2013). The Solomon-specific content comes from the package: the four contrasts, the pretest adjustment in the pretested groups, how to claim that sensitization is negligible, and how missing posttests are handled. Text in square brackets is for the researcher to complete.

Usage

```
analysis_plan_solomon(
  plan = NULL,
  outcome = "[the primary outcome, measured at posttest]",
  treatment = "[the treatment, and the control condition]",
  direction = c("increase", "decrease", "two-sided"),
  sensitization = c("test", "equivalence", "larger_pretested", "smaller_pretested",
    "exploratory"),
  equivalence_bound = NULL,
  alpha = 0.05,
```

```

occasions = 1,
primary_occasion = NULL,
tipping_groups = "treatment",
title = "Analysis plan for a Solomon four-group study",
file = NULL
)

```

Arguments

plan	Optional result of <code>plan_solomon()</code> . Its sample sizes and planning values fill the planned-sample section.
outcome	Description of the primary outcome, measured at posttest.
treatment	Description of the treatment and of the control condition.
direction	Expected direction of the treatment effect: "increase" (default), "decrease", or "two-sided" for no predicted direction.
sensitization	How the Pretest x Treatment contrast is treated: "test" (default), a confirmatory two-sided test of sensitization; "equivalence", a confirmatory claim that sensitization is negligible, tested with two one-sided tests against <code>equivalence_bound</code> ; "larger_pretested" or "smaller_pretested", a confirmatory prediction of the direction of sensitization; or "exploratory", estimated and reported but not confirmatory.
equivalence_bound	The smallest sensitization of interest, in posttest units, for <code>sensitization = "equivalence"</code> .
alpha	Significance level of the confirmatory tests. Default 0.05.
occasions	Number of posttest occasions, or their labels. With more than one, the primary analysis is <code>fit_solomon_mmrn()</code> , a mixed model for repeated measures (Mallinckrodt et al., 2008), and the confirmatory contrasts are those at <code>primary_occasion</code> .
primary_occasion	The occasion whose contrasts are confirmatory, when there are several. Default: the last.
tipping_groups	Groups whose imputed posttests are shifted in the tipping-point sensitivity analysis, as in <code>tipping_point_solomon()</code> . Default "treatment". When sensitization is confirmatory, a second analysis in the pretested treatment group alone is added.
title	Title of the plan.
file	Optional path. When given, the plan is also written there as UTF-8 Markdown.

Details

What the plan covers.

- **Hypotheses** (SPIRIT item 12). Numbered predictions for the treatment effect and for sensitization. For interactions, the template asks for "the expected shape" (van 't Veer & Giner-Sorolla, 2016, p. 10), so the sensitization hypothesis states which treatment effect is expected to be larger. A claim that sensitization is negligible needs an equivalence test against a smallest effect of interest fixed in advance (Lakens, 2017).

- **Methods** (items 14 and 16a). Randomization to the four groups; the pretest-posttest interval, since pretest effects can depend on it (Entwisle, 1961); identical measurement in all groups (French et al., 2021b, Recommendation 11); and the planned sample, from plan when given, with its planning values.
- **Analysis plan** (items 20a-c). The model, the confirmatory contrasts, the test of sensitization, the handling of missing posttests, with a tipping-point sensitivity analysis (White et al., 2011), and the checks run before the model.
- **Deviations** (item 25). A section for recording departures from the plan, which are to be reported with the results (Nosek et al., 2018, p. 2602).

What it does not decide. The plan states the package’s defaults; it does not choose the hypotheses, the sample size, or the smallest effect of interest, and every sentence can be edited. Prediction and postdiction must stay distinguishable (Nosek et al., 2018, p. 2602), so the plan should be registered before the outcomes are seen.

The design. The plan is for the four-group design: one treatment and a control, each with and without a pretest. Plans for designs with several treatments are not yet supported; see `fit_solomon_glm()` for their analysis.

The round trip. The returned object records the confirmatory contrasts. Passing it to `report_solomon()` as `design = list(plan = )` makes the report state whether the sensitization analysis was pre-specified, from the plan rather than by guessing.

Value

An object of class `solomon_analysis_plan`: text (the Markdown lines), settings (the choices, including confirmatory), and date. Printing it shows the Markdown.

References

- Chan, A.-W., Tetzlaff, J. M., Altman, D. G., Laupacis, A., Gøtzsche, P. C., Krleža-Jerić, K., Hróbjartsson, A., Mann, H., Dickersin, K., Berlin, J. A., Doré, C. J., Parulekar, W. R., Summerskill, W. S. M., Groves, T., Schulz, K. F., Sox, H. C., Rockhold, F. W., Rennie, D., & Moher, D. (2013). SPIRIT 2013 statement: Defining standard protocol items for clinical trials. *Annals of Internal Medicine*, 158(3), 200–207. <https://doi.org/10.7326/0003-4819-158-3-201302050-00583>
- Entwisle, D. R. (1961). Interactive effects of pretesting. *Educational and Psychological Measurement*, 21(3), 607–620. <https://doi.org/10.1177/001316446102100307>
- Fitzmaurice, G. M., Laird, N. M., & Ware, J. H. (2011). *Applied longitudinal analysis* (2nd ed.). Wiley. <https://doi.org/10.1002/9781119513469>
- French, D. P., Miles, L. M., Elbourne, D., Farmer, A., Gulliford, M., Locock, L., Sutton, S., McCambridge, J., & MERIT Collaborative Group. (2021b). Reducing bias in trials from reactions to measurement: The MERIT study including developmental work and expert workshop. *Health Technology Assessment*, 25(55), 1–72. <https://doi.org/10.3310/hta25550>
- Lakens, D. (2017). Equivalence tests: A practical primer for t tests, correlations, and meta-analyses. *Social Psychological and Personality Science*, 8(4), 355–362. <https://doi.org/10.1177/1948550617697177>
- Mallinckrodt, C. H., Lane, P. W., Schnell, D., Peng, Y., & Mancuso, J. P. (2008). Recommendations for the primary analysis of continuous endpoints in longitudinal clinical trials. *Drug Information Journal*, 42(4), 303–319. <https://doi.org/10.1177/009286150804200402>

- Nosek, B. A., Ebersole, C. R., DeHaven, A. C., & Mellor, D. T. (2018). The preregistration revolution. *Proceedings of the National Academy of Sciences*, 115(11), 2600–2606. <https://doi.org/10.1073/pnas.1708274114>
- van 't Veer, A. E., & Giner-Sorolla, R. (2016). Pre-registration in social psychology—A discussion and suggested template. *Journal of Experimental Social Psychology*, 67, 2–12. <https://doi.org/10.1016/j.jesp.2016.03.004>
- White, I. R., Horton, N. J., Carpenter, J., & Pocock, S. J. (2011). Strategy for intention to treat analysis in randomised trials with missing outcome data. *BMJ*, 342, Article d40. <https://doi.org/10.1136/bmj.d40>

See Also

`plan_solomon()` for the sample size, `report_solomon()` for the results, and the R Markdown template "Solomon four-group study" (`rmarkdown::draft("study.Rmd", "solomon-study", package = "solomonR")`), which runs the planned analysis in order.

Examples

```
p <- plan_solomon(delta = 0.4, sens = 0, estimand = "ate")
analysis_plan_solomon(
  plan = p,
  outcome = "reading comprehension (0-100)",
  treatment = "a six-week tutoring program, against the usual lessons",
  sensitization = "equivalence",
  equivalence_bound = 3
)
```

baseline_solomon

Baseline comparison of the pretested arms

Description

[Stable] Compares the pretest scores of the treated and control pretested groups: the check of baseline equivalence that a Solomon design allows. It matters most when groups were not formed by random assignment. In a design with several treatments, each treatment's pretested group is compared with the pretested control group.

Usage

```
baseline_solomon(
  y_pre = NULL,
  treat = NULL,
  pretested = NULL,
  n = NULL,
  mean = NULL,
  sd = NULL,
  conf_level = 0.95,
  control = NULL,
  data = NULL
)
```

Arguments

y_pre, treat, pretested	Individual data: pretest scores, treatment indicator, and pretest indicator. Only pretested participants with a pretest score are used. <code>treat</code> is a 0/1 (or logical) indicator, or a factor or character vector of conditions with the control named by <code>control</code> .
n, mean, sd	Alternatively, the pretest sample size, mean, and standard deviation of the two pretested groups, treated first. With <code>control</code> , vectors named by condition, one element for each pretested group.
conf_level	Confidence level. Default 0.95.
control	The control condition when <code>treat</code> is a factor or character vector with more than two conditions, a Solomon N-group design; see <code>fit_solomon_glm()</code> . With two conditions the result is the same as with a 0/1 <code>treat</code> . With summary statistics, the name of the control group in <code>n</code> , <code>mean</code> , and <code>sd</code> .
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (<code>y_pre = pre</code>) or as strings (<code>y_pre = "pre"</code>).

Details

The comparison reports the pretest difference with a *t* interval (pooled variance) and the standardized difference, Hedges's *g*, with a confidence interval from the noncentral *t* distribution (Cumming & Finch, 2001; Kelley, 2007). No equivalence threshold is applied; the estimate and its interval are reported for the reader to judge.

Designs with several treatments: give `treat` as a factor or character vector of conditions and name the control with `control`, or give `n`, `mean`, and `sd` as vectors named by condition together with `control`. Each treatment is then compared with the control, and each comparison uses only the two groups it compares: their pooled SD, *t* test, and Hedges's *g* are the same as in a four-group analysis of that treatment and the control. The *p*-values are not adjusted for the number of comparisons.

What the design cannot check. The unpretested arms form the posttest-only control group design, which relies on randomization rather than a pretest for the equivalence of its groups (Campbell & Stanley, 1963/1966, p. 25). Without random assignment they form a static-group comparison, for which there are "no formal means of certifying that the groups would have been equivalent" (p. 12). Selection bias in the unpretested comparison, which is the comparison that isolates pretest sensitization, therefore cannot be checked or adjusted for with the study's own data. Edmonds and Kennedy (2017) identify selection bias as the largest threat to internal validity in quasi-experimental research (p. 7) and, with instrumentation, as the threat most common in quasi-experimental Solomon designs (p. 94).

Value

An object of class `solomon_baseline` with the group statistics, the difference with its interval and *t* test, and Hedges's *g* with its interval. For a design with several treatments, `groups` holds the statistics of every pretested group, `comparisons` has one row for each treatment against the control (`comparison`, `difference`, `std.error`, `conf.low`, `conf.high`, `statistic`, `df`, `p.value`, `g`, `g.low`, and `g.high`), and `conditions` names the control and the treatments.

References

- Campbell, D. T., & Stanley, J. C. (1966). *Experimental and quasi-experimental designs for research*. Rand McNally. (Original work published 1963)
- Cumming, G., & Finch, S. (2001). A primer on the understanding, use, and calculation of confidence intervals that are based on central and noncentral distributions. *Educational and Psychological Measurement*, 61(4), 532–574. <https://doi.org/10.1177/00131640121971374>
- Edmonds, W. A., & Kennedy, T. D. (2017). *An applied guide to research designs: Quantitative, qualitative, and mixed methods* (2nd ed.). SAGE Publications. <https://doi.org/10.4135/9781071802779>
- Kelley, K. (2007). Confidence intervals for standardized effect sizes: Theory, application, and implementation. *Journal of Statistical Software*, 20(8), 1–24. <https://doi.org/10.18637/jss.v020.i08>
- Mai, N. N., Takahashi, Y., & Oo, M. M. (2020). Testing the effectiveness of transfer interventions using Solomon four-group designs. *Education Sciences*, 10(4), Article 92. <https://doi.org/10.3390/educsci10040092>

See Also

`report_solomon()` with `design = list(assignment = "nonrandom")`.

Examples

```
# El Karkri et al. (2025a), Table 7: intact classes, one per condition.
pre <- elkarkri2025a[elkarkri2025a$pretested == 1, ]
baseline_solomon(n = pre$n, mean = pre$pre_mean, sd = pre$pre_sd)

# Mai et al. (2020): two treatments, each compared with the control.
baseline_solomon(pre_behavior, condition, pretested, control = "Control",
                 data = mai2020)
```

check_solomon_assumptions

Descriptive assumption diagnostics for Solomon analyses

Description

[Stable] Reports Brown-Forsythe tests of equal posttest variance (Brown & Forsythe, 1974), Shapiro-Wilk normality tests within cells, and a test of homogeneous pretest-posttest slopes among pretested participants with complete scores.

Usage

```
check_solomon_assumptions(
  y_post,
  treat,
  pretested,
  y_pre,
  control = NULL,
  data = NULL
)
```

Arguments

y_post	numeric posttest
treat	0/1 (or logical) treatment indicator; or a factor or character vector of conditions, with the control named by control
pretested	0/1 (or logical) pretest indicator
y_pre	numeric pretest (NA for unpretested)
control	The control condition when treat is a factor or character vector with more than two conditions, a Solomon N-group design; see fit_solomon_glm() . With two conditions the result is the same as with a 0/1 treat.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Details

These results are descriptive. Choosing an analysis according to whether a preliminary assumption test is significant can distort Type I error rates (Zimmerman, 2004), so solomonR does not use them as gates: heteroskedasticity-consistent standard errors are a reasonable default for the unified GLM regardless of these results (Long & Ervin, 2000). A small slope-homogeneity p-value is substantively informative: it suggests that the treatment effect among pretested participants depends on the pretest score.

Designs with several treatments: give treat as a factor or character vector of conditions and name the control with control. The tests then cover all $2(k + 1)$ cells of a design with k treatments (Steyn, 2009): the Brown-Forsythe tests compare all the posttest cells and the $k + 1$ unpretested groups, and the slope-homogeneity test is a test on k degrees of freedom that the pretest-posttest slope is the same in the $k + 1$ pretested groups.

Value

An object of class solomon_checks with the p-values brown_forsythe_4cell_p (all four posttest cells), brown_forsythe_unpre_p (the unpretested cells), shapiro_p_by_cell (one p-value for each cell, named by the pretest indicator and the treatment, such as 1.0), and ancova_slope_homogeneity_p. For a design with several treatments, the test across all the posttest cells is brown_forsythe_cells_p, the cells of shapiro_p_by_cell are named by the pretest indicator and the condition, such as 1.RP, and conditions names the control and the treatments.

References

- Brown, M. B., & Forsythe, A. B. (1974). Robust tests for the equality of variances. *Journal of the American Statistical Association*, 69(346), 364–367. <https://doi.org/10.1080/01621459.1974.10482955>
- Long, J. S., & Ervin, L. H. (2000). Using heteroscedasticity consistent standard errors in the linear regression model. *The American Statistician*, 54(3), 217–224. <https://doi.org/10.1080/00031305.2000.10474549>
- Mai, N. N., Takahashi, Y., & Oo, M. M. (2020). Testing the effectiveness of transfer interventions using Solomon four-group designs. *Education Sciences*, 10(4), Article 92. <https://doi.org/10.3390/educsci10040092>
- Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

Zimmerman, D. W. (2004). A note on preliminary tests of equality of variances. *British Journal of Mathematical and Statistical Psychology*, 57(1), 173–181. <https://doi.org/10.1348/000711004849222>

Examples

```
with(solomon_example, check_solomon_assumptions(y_post, treat, pretested, y_pre))

# A six-group design: two treatments and a control (Mai et al., 2020).
check_solomon_assumptions(post_behavior, condition, pretested, pre_behavior,
                           control = "Control", data = mai2020)
```

check_solomon_missing *Distinguish structural and incidental missingness in a Solomon design*

Description

[Stable] Classifies missing values in Solomon design data and explains the supported response to each kind. Pretest scores are *structurally absent* for participants assigned to the unpretested groups: withholding the pretest is the experimental manipulation (Solomon, 1949), so those values must never be imputed. Other missing values are *incidental* and are handled according to the missing-data literature (Rubin, 1976; Little & Rubin, 2019).

Usage

```
check_solomon_missing(
  y_post,
  treat,
  pretested,
  y_pre = NULL,
  control = NULL,
  data = NULL
)
```

Arguments

y_post	Numeric posttest scores.
treat	Treatment indicator coded 0/1 (or logical); or a factor or character vector of conditions, with the control named by control.
pretested	Pretest indicator coded 0/1 (or logical).
y_pre	Optional numeric pretest scores, missing by design for unpretested participants.
control	The control condition when treat is a factor or character vector with more than two conditions, a Solomon N-group design; see fit_solomon_glm() . With two conditions the result is the same as with a 0/1 treat.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Details

Categories:

- **Structural pretest absence:** unpretested participants have no pretest. Unlike planned missing-data designs, in which unmeasured values exist and can be imputed (Graham et al., 2006), an imputed pretest here would describe a measurement that never occurred.
- **Incidental pretest missingness:** pretested participants without a pretest score. `solomonR` analyses currently use complete cases and warn. Because treatment is randomized, deterministic mean imputation of the pretest (without using treatment or outcome), with a missingness indicator when pretests may not be missing completely at random, retains these participants without biasing the treatment effect (White & Thompson, 2005; Groenwold et al., 2012).
- **Incidental posttest missingness:** missing outcomes. Complete-case analysis is unbiased when missingness is unrelated to the outcome given the variables in the model (Little & Rubin, 2019); attrition that differs across the groups should be reported.
- **Unexpected pretest scores:** pretest values recorded for unpretested participants, which usually indicate a coding or assignment error. They are ignored by `solomonR` analyses.
- **Unassigned participants:** missing treatment or pretest assignment.

Pretest categories are NA when `y_pre` is not supplied.

Designs with several treatments: give `treat` as a factor or character vector of conditions and name the control with `control`. The counts are then given for each of the $2(k + 1)$ groups of a design with k treatments (Steyn, 2009): each treatment and the control, with and without a pretest.

Not supported: this function does not test the missingness mechanism, perform imputation, or provide sensitivity analyses for outcome missingness that depends on unobserved values.

Value

An object of class `solomon_missing` with `by_cell` (counts by Solomon cell), `counts` (totals by category), `pattern` ("none", "structural", "incidental", or "mixed"), and `guidance` (the interpretation, supported response, and sources for each category present). For a design with several treatments, `by_cell` has one row for each of the $2(k + 1)$ groups, its `treat` column holds the condition, and `conditions` names the control and the treatments.

References

- Graham, J. W., Taylor, B. J., Olchowski, A. E., & Cumsille, P. E. (2006). Planned missing data designs in psychological research. *Psychological Methods*, 11(4), 323–343. <https://doi.org/10.1037/1082-989X.11.4.323>
- Groenwold, R. H. H., White, I. R., Donders, A. R. T., Carpenter, J. R., Altman, D. G., & Moons, K. G. M. (2012). Missing covariate data in clinical research: When and when not to use the missing-indicator method for analysis. *Canadian Medical Association Journal*, 184(11), 1265–1269. <https://doi.org/10.1503/cmaj.110>
- Little, R. J. A., & Rubin, D. B. (2019). *Statistical analysis with missing data* (3rd ed.). Wiley. <https://doi.org/10.1002/9781119482260>
- Mai, N. N., Takahashi, Y., & Oo, M. M. (2020). Testing the effectiveness of transfer interventions using Solomon four-group designs. *Education Sciences*, 10(4), Article 92. <https://doi.org/10.3390/educsci10040092>
- Rubin, D. B. (1976). Inference and missing data. *Biometrika*, 63(3), 581–592. <https://doi.org/10.1093/biomet/63.3.581>

Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>

Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

White, I. R., & Thompson, S. G. (2005). Adjusting for partially missing baseline measurements in randomized trials. *Statistics in Medicine*, 24(7), 993–1007. <https://doi.org/10.1002/sim.1981>

See Also

`validate_solomon()`

Examples

```
data(solomon_example)

# Structural absence only: pretests are absent by design in Groups 3 and 4.
with(solomon_example, check_solomon_missing(y_post, treat, pretested, y_pre))

# Mixed: add a lost pretest and a missing posttest.
d <- solomon_example
d$y_pre[which(d$pretested == 1)[1]] <- NA
d$y_post[which(d$pretested == 0)[1]] <- NA
with(d, check_solomon_missing(y_post, treat, pretested, y_pre))

# A six-group design: two treatments and a control (Mai et al., 2020).
check_solomon_missing(post_behavior, condition, pretested, pre_behavior,
  control = "Control", data = mai2020)
```

compare_solomon_methods

Compare Solomon analyses and their estimands

Description

[Stable] Fits several Solomon analyses to the same data and lines up their estimates of the four Solomon contrasts, so that differences in pretest adjustment, variance assumptions, and reference distributions are visible side by side. Stating the target quantity explicitly is what makes such comparisons meaningful (Lundberg et al., 2021).

Usage

```
compare_solomon_methods(
  y_post,
  treat,
  pretested,
  y_pre = NULL,
  methods = c("glm", "ml", "classic", "sem"),
```

```

    conf_level = 0.95,
    data = NULL
  )

```

Arguments

y_post	Numeric posttest scores (continuous).
treat	Treatment indicator coded 0/1 (or logical). Designs with several treatments are not supported; see <code>fit_solomon_glm()</code> .
pretested	Pretest indicator coded 0/1 (or logical).
y_pre	Optional numeric pretest scores. Maximum likelihood, the classic analyses, and the SEM ANCOVA require them.
methods	Analyses to include: any of "glm" (unified GLM with HC3), "ml" (maximum likelihood, reported with both its default Wald inference and the small-sample option), "classic", and "sem" (requires the lavaan package).
conf_level	Confidence level for intervals. Default is 0.95.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Value

An object of class `solomon_comparison` with results (one row per method and contrast, with the adjustment, variance assumption, reference distribution, estimate, standard error, interval, and p-value), estimands (definitions of the four contrasts), `not_compared` (analyses excluded and why), and `skipped` (requested methods that could not be fitted and why).

Estimands

Every row estimates one of four population contrasts in posttest means (treatment minus control): the treatment effect among pretested participants, the treatment effect among unpretested participants, their difference (Pretest x Treatment, or sensitization), and their equal-weighted average. In a randomized Solomon experiment with a continuous outcome, all included estimators target these same contrasts. Pretest adjustment changes precision rather than the target (Lin, 2013), and the variance assumptions and reference distributions change the standard errors and intervals.

Some algebraic identities make this concrete. The unified GLM, maximum likelihood, classic Tests C and H, and the SEM mean structure give the same estimate of the treatment effect among unpretested participants. The unified GLM (with pretest scores), maximum likelihood, and classic Test E give the same pretest-adjusted estimate among pretested participants. Their uncertainty differs.

Not compared

Analyses that do not estimate a raw-scale Solomon contrast are listed in `not_compared` rather than aligned with the others:

- `perm_solomon()` is a randomization test of no treatment effect, not an interval estimator of the contrasts.

- Test I (Walton Braver & Braver, 1988) combines one-tailed p-values.
- `fit_solomon_sem_latent()` estimates contrasts on a latent-variable scale.
- Hedges' g from `fit_solomon_classic()` is a standardized mean difference.

Unsupported: non-continuous outcomes, for which adjusted and unadjusted estimators can target different noncollapsible quantities (Daniel et al., 2021), and clustered designs.

References

- Daniel, R., Zhang, J., & Farewell, D. (2021). Making apples from oranges: Comparing noncollapsible effect estimators and their standard errors after adjustment for different covariate sets. *Biometrical Journal*, 63(3), 528–557. <https://doi.org/10.1002/bimj.201900297>
- Lin, W. (2013). Agnostic notes on regression adjustments to experimental data: Reexamining Freedman's critique. *The Annals of Applied Statistics*, 7(1), 295–318. <https://doi.org/10.1214/12-AOAS583>
- Lundberg, I., Johnson, R., & Stewart, B. M. (2021). What is your estimand? Defining the target quantity connects statistical evidence to theory. *American Sociological Review*, 86(3), 532–565. <https://doi.org/10.1177/00031224211004187>
- Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

See Also

[fit_solomon_glm\(\)](#), [fit_solomon_ml\(\)](#), [fit_solomon_classic\(\)](#), [fit_solomon_sem\(\)](#)

Examples

```
data(solomon_example)
with(
  solomon_example,
  compare_solomon_methods(
    y_post, treat, pretested, y_pre,
    methods = c("glm", "ml", "classic")
  )
)
```

Description

The posttest sample size, mean, and standard deviation of the four Solomon groups in a classroom study of the Cognitive Acceleration through Science Education programme (El Karkri et al., 2025a, Table 8, p. 11), and the pretest mean and standard deviation of the two pretested groups (Table 7, p. 10). Numbers reported in the publication are reused with citation.

Usage

elkarkri2025a

Format

A data frame with 4 rows, one per Solomon group, and 8 variables:

group The Solomon group.

pretested, treat Indicators (1 = yes).

n, mean, sd Posttest sample size, mean, and standard deviation.

pre_mean, pre_sd Pretest mean and standard deviation (pretested groups only; the pretest sample sizes equal the posttest ones).

Details

Design and caveats. The authors describe the study as quasi-experimental: each Solomon group was one intact class, so class and condition are confounded, and differences between the groups can reflect the classes as well as the treatment and the pretest. `validate_solomon()` flags this design when class membership is supplied. The pretested classes already differed at pretest (9.61 against 7.86), which `baseline_solomon()` reports; the unpretested classes have no pretest. The authors reported a significant Pretest x Treatment interaction.

Known result. `solomon_from_summary()` reproduces the published two-way ANOVA from these numbers within rounding: interaction $F(1, 84) = 11.46$ against the published 11.482, treatment 6.78 against 6.794, and pretest 0.18 against 0.186 (pp. 11-12).

Source

El Karkri, M., Quesada, A., & Romero-Ariza, M. (2025a). The dual impact of pretest sensitisation and the cognitive acceleration through science education programme in the Solomon four-group design. *Brain Sciences*, 16(1), Article 64. <https://doi.org/10.3390/brainsci16010064>

See Also

`solomon_from_summary()`, `baseline_solomon()`, `kvalet1996`

Examples

```
with(elkarkri2025a, solomon_from_summary(n, mean, sd))
pre <- elkarkri2025a[elkarkri2025a$pretested == 1, ]
baseline_solomon(n = pre$n, mean = pre$pre_mean, sd = pre$pre_sd)
```

equivalence_solomon *Equivalence test for a Solomon contrast*

Description

[Stable] Tests whether a Solomon contrast, by default the Pretest x Treatment (sensitization) contrast, is small enough to be considered negligible, using the two one-sided tests (TOST) procedure (Schuirmann, 1987; Lakens, 2017). A nonsignificant sensitization test is not evidence that sensitization is absent; an equivalence test against a prespecified smallest effect size of interest (SESOI) can provide that evidence.

Usage

```
equivalence_solomon(
  fit,
  bounds,
  contrast = "Pretest x Treatment",
  alpha = 0.05,
  comparison = NULL,
  object = deprecated()
)
```

Arguments

fit	A fit from <code>fit_solomon_glm()</code> or <code>fit_solomon_ml()</code> .
bounds	Equivalence bounds on the raw posttest scale: one positive number delta, giving <code>c(-delta, delta)</code> , or <code>c(lower, upper)</code> with $\text{lower} < 0 < \text{upper}$. There is no default; bounds must be chosen in advance.
contrast	Solomon contrast to test. Default is "Pretest x Treatment".
alpha	Significance level for each one-sided test. Default is 0.05.
comparison	For a design with several treatments, the comparison to test, one of the comparison values of <code>fit\$effects</code> (see "Designs with several treatments"). Leave it NULL for a four-group design.
object	[Deprecated] Use <code>fit</code> .

Value

An object of class `solomon_equivalence` containing the estimate, standard error, degrees of freedom, both one-sided tests (`t_lower`, `p_lower`, `t_upper`, `p_upper`), the equivalence p-value (`p_equivalence`), the test against zero (`statistic`, `p_zero`), both confidence intervals, the logical results `equivalent`, `different`, and `exceeds_bounds`, the outcome, and a plain-language interpretation. For a design with several treatments, it also holds the comparison, its weights over the conditions, and the conditions of the fit.

Choosing equivalence bounds

The bounds define the smallest effect size of interest and should be set before the data are examined, for example in a preregistration (Lakens, 2017; Lakens et al., 2018). Justify them substantively, such as the smallest change in posttest scores that would alter a conclusion, or from prior research.

Bounds are on the raw posttest scale. To use a standardized SESOI (for example, $d = 0.2$), multiply it by a standard deviation fixed in advance, such as one reported in prior studies. Do not use the standard deviation of the current data: the same standardized bound then implies different raw bounds in different samples (Lakens, 2017).

Inference

Each one-sided test uses the estimate, standard error, and reference distribution of the fitted model: t with the model's degrees of freedom for `fit_solomon_glm()` (Satterthwaite degrees of freedom with CR2); for `fit_solomon_ml()`, the normal distribution with its default Wald inference or t with Welch-Satterthwaite degrees of freedom with `inference = "satterthwaite"`. The equivalence p -value is the larger of the two one-sided p -values, and the matching interval has confidence level $1 - 2\alpha$ (90% The conventional two-sided test against zero is reported alongside, with its $1 - \alpha$ interval.

Outcomes

Combining the equivalence test with the test against zero gives four outcomes (Lakens, 2017):

- "equivalent": statistically equivalent and not different from zero;
- "trivial": different from zero but statistically equivalent, so smaller than the smallest effect size of interest;
- "different": different from zero and not statistically equivalent;
- "inconclusive": neither different from zero nor statistically equivalent.

`exceeds_bounds` is TRUE when the $1 - 2\alpha$ interval lies entirely beyond one bound, which rejects effects no larger than the smallest effect size of interest in that direction (a minimum-effect test; Murphy & Myors, 1999).

Designs with several treatments

For a `fit_solomon_glm()` fit of a design with several treatments (class `solomon_ngroup`), name the comparison to test with `comparison`, such as "RP vs Control"; it may be left out only when the fit has a single comparison. The test uses that comparison's estimate of the chosen contrast and its standard error. It is not adjusted for the other comparisons of the design.

References

- Lakens, D. (2017). Equivalence tests: A practical primer for t tests, correlations, and meta-analyses. *Social Psychological and Personality Science*, 8(4), 355–362. <https://doi.org/10.1177/1948550617697177>
- Lakens, D., Scheel, A. M., & Isager, P. M. (2018). Equivalence testing for psychological research: A tutorial. *Advances in Methods and Practices in Psychological Science*, 1(2), 259–269. <https://doi.org/10.1177/2515245918770963>

Murphy, K. R., & Myors, B. (1999). Testing the hypothesis that treatments have negligible effects: Minimum-effect tests in the general linear model. *Journal of Applied Psychology*, 84(2), 234–248. <https://doi.org/10.1037/0021-9010.84.2.234>

Schuurmann, D. J. (1987). A comparison of the two one-sided tests procedure and the power approach for assessing the equivalence of average bioavailability. *Journal of Pharmacokinetics and Biopharmaceutics*, 15(6), 657–680. <https://doi.org/10.1007/BF01068419>

Examples

```
data(solomon_example)
fit <- with(solomon_example, fit_solomon_glm(y_post, treat, pretested, y_pre))

# Illustrative bounds only: half a standard deviation (5 points) on this
# test. In a real study, justify the smallest effect size of interest and
# fix the bounds before examining the data.
equivalence_solomon(fit, bounds = 5)

# A six-group design: test one comparison at a time. The bounds here are
# again illustrative only.
fit6 <- fit_solomon_glm(post_behavior, condition, pretested, pre_behavior,
                        control = "Control", data = mai2020)
equivalence_solomon(fit6, bounds = 0.3, comparison = "RP vs Control")
```

fisher_solomon

Historical categorical analysis of a binary Solomon outcome

Description

[Stable] Reproduces the categorical path described by El Karkri et al. (2025b) for qualitative outcomes: the treatment comparison is tested separately among pretested and unpretested participants with Fisher's exact test (Pearson's chi-square test is also reported), and pretest sensitization is "judged to exist if a significant effect is observed for pre-tested groups but not for non-pre-tested groups" (p. 7).

Usage

```
fisher_solomon(
  y_post,
  treat,
  pretested,
  alpha = 0.05,
  data = NULL,
  y = deprecated()
)
```

Arguments

y_post	Binary posttest outcome coded 0/1 (or logical).
treat	Treatment indicator coded 0/1 (or logical). Designs with several treatments are not supported; see fit_solomon_glm() .
pretested	Pretest indicator coded 0/1 (or logical).
alpha	Significance level for the historical rule. Default 0.05.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").
y	[Deprecated] Use y_post.

Details

This is provided for teaching and replication. The rule compares significance, not effects: when the treatment effect is the same in both pretest conditions, it still declares sensitization whenever the pretested comparison reaches significance and the unpretested one does not. As Gelman and Stern (2006) put it, "even large changes in significance levels can correspond to small, nonsignificant changes in the underlying quantities" (p. 328). A test of sensitization compares the effects themselves; see [marginal_solomon\(\)](#).

Value

An object of class `solomon_fisher` with tests (one row per pretest condition and for both combined: counts, proportions, the uncorrected Pearson chi-square, and Fisher's exact p-value) and sensitization (the historical rule's verdict).

References

- El Karkri, M., Quesada, A., & Romero-Ariza, M. (2025b). Methodological aspects of the Solomon four-group design: Detecting pre-test sensitisation and analysing qualitative and quantitative variables in education research. *Review of Education*, 13(1), Article e70050. <https://doi.org/10.1002/rev3.70050>
- Gelman, A., & Stern, H. (2006). The difference between "significant" and "not significant" is not itself statistically significant. *The American Statistician*, 60(4), 328–331. <https://doi.org/10.1198/000313006X152649>

See Also

[marginal_solomon\(\)](#)

Examples

```
# Kvalem et al. (1996): condom use at most recent intercourse, 6 months.
kvalem <- data.frame(
  treat = c(1, 1, 0, 0), pretested = c(1, 0, 1, 0),
  events = c(51, 21, 76, 69), n = c(73, 49, 148, 133)
)
d <- kvalem[rep(1:4, kvalem$n), c("treat", "pretested")]
d$y <- unlist(lapply(1:4, function(i) {
  rep(c(1, 0), c(kvalem$events[i], kvalem$n[i] - kvalem$events[i]))
}))
```

```
fisher_solomon(y, treat, pretested, data = d)
```

fit_solomon_1949

Solomon's (1949) improvement-score analysis

Description

[Stable] Reproduces the analysis Solomon (1949) proposed with the design: an inferred pretest mean for the unpretested groups, an improvement score for each group, and an interaction term I. It is a historical procedure, kept to document how the analysis of the design began; the recommended analysis is [fit_solomon_glm\(\)](#).

Usage

```
fit_solomon_1949(
  y_post = NULL,
  treat = NULL,
  pretested = NULL,
  y_pre = NULL,
  post_mean = NULL,
  pre_mean = NULL,
  n = NULL,
  inferred_pretest = c("average", "pooled"),
  data = NULL
)
```

Arguments

y_post, treat, pretested, y_pre

Individual data: posttest scores, training (treatment) and pretest indicators coded 0/1, and pretest scores, missing by design for unpretested participants. A design with no participant who is neither pretested nor trained is analyzed as the three-group design. Designs with several treatments are not supported; see [fit_solomon_glm\(\)](#).

post_mean, pre_mean

Alternatively, group means: post_mean has three values (experimental, Control I, Control II) for the three-group design or four (adding Control III) for the four-group design, and pre_mean the two pretest means (experimental, Control I).

n

Optional group sizes, in the order of post_mean, for the printout and for inferred_pretest = "pooled".

inferred_pretest

"average" (default) or "pooled"; see Details.

data

Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Details

The three-group design (Solomon, 1949, pp. 141–143, Table I). An experimental group and Control Group I are pretested; the experimental group and Control Group II are trained; all three are posttested. The pretest mean Control Group II would have had is inferred from the pretested groups, $i = (a1 + a2) / 2$, where $a1$ and $a2$ are their pretest means (Tables I–III). Each group's improvement is its posttest mean minus its observed or inferred pretest mean: $d1 = b1 - a1$, $d2 = b2 - a2$, and $d3 = b3 - i$. The interaction is $I = d1 - (d2 + d3)$ (p. 143): the part of the experimental group's improvement not explained by the effect of the pretest alone ($d2$) and of training alone ($d3$).

The four-group design (p. 147, Table V). Control Group III, with neither pretest nor training, receives the same inferred pretest, its improvement $d4 = b4 - i$ is attributed to outside events between the two occasions, and the interaction becomes $I = d1 - (d2 + d3 - d4)$.

The inferred pretest. Solomon's text describes it as the grand mean of the pooled pretested groups (p. 141), and his tables compute it as $(a1 + a2) / 2$ (Tables I–III, V). The two agree when the pretested groups are the same size. `inferred_pretest = "average"` (the default) follows the tables; `"pooled"` weights the two means by their sample sizes, which requires `n` or individual data.

No test. Solomon gave standard errors for the observed means but printed the error of $d3$ as "?" (Tables II–III), and judged the interaction from "an examination of the variabilities" (p. 144). This function therefore reports his point estimates only. With individual data it also reports the standard errors of the observed means, as his tables do.

Relation to the later analysis (a `solomonR` note, not Solomon's). In the four-group design the inferred pretest cancels, so $I = (b1 - b2 - b3 + b4) - (a1 - a2)$: the posttest interaction contrast of the 2×2 analysis, less the pretest difference between the two pretested groups. With random assignment the pretest difference has expectation zero, so I estimates the same Pretest $\times$ Treatment interaction as the posttest contrast.

The later verdict. Campbell (1957, p. 303) first rejected the analysis with an inferred pretest: it restricts the degrees of freedom, violates independence, and leaves no legitimate test. He recommended the 2×2 analysis of variance of the four posttests instead (`fit_solomon_classic()`, Test A). Campbell and Stanley (1963/1966, p. 25) repeated the recommendation and judged Solomon's gain-score suggestions unacceptable. Solomon and Lessac (1968, p. 147) still used the combined pretest mean of the pretested groups as the best estimate for the unpretested groups, to judge whether those groups improved or deteriorated in absolute terms.

Value

An object of class `solomon_1949` with design ("three-group" or "four-group"), groups (a data frame with each group's pretest and training, size, pretest mean and whether it was observed or inferred, posttest mean, and improvement, plus standard errors of the observed means when individual data are given), `inferred_pretest`, `I`, and, for the four-group design, `posttest_contrast` and `pretest_difference`.

References

- Campbell, D. T. (1957). Factors relevant to the validity of experiments in social settings. *Psychological Bulletin*, 54(4), 297–312. <https://doi.org/10.1037/h0040950>
- Campbell, D. T., & Stanley, J. C. (1966). *Experimental and quasi-experimental designs for research*. Rand McNally. (Original work published 1963)

Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>

Solomon, R. L., & Lessac, M. S. (1968). A control group design for experimental studies of developmental processes. *Psychological Bulletin*, 70(3, Pt. 1), 145–150. <https://doi.org/10.1037/h0026147>

See Also

[solomon1949](#) for Solomon's data, [fit_solomon_classic\(\)](#) for the later historical tests, and [fit_solomon_glm\(\)](#) for the recommended analysis.

Examples

```
# Solomon's fifth-grade spelling experiment (Table II, p. 144).
g5 <- solomon1949[solomon1949$grade == 5, ]
fit_solomon_1949(post_mean = g5$mean, pre_mean = g5$pre_mean[1:2], n = g5$n)

# The four-group version, from individual data.
with(solomon_example, fit_solomon_1949(y_post, treat, pretested, y_pre))
```

fit_solomon_classic	<i>Historical Solomon Four-Group Analysis</i>
---------------------	---

Description

[Stable] Implements the historical Test A-I framework associated with analysis of the Solomon four-group design. All tests are calculated when possible, while the historical decision pathway is stored separately.

Usage

```
fit_solomon_classic(
  y_post,
  treat,
  pretested,
  y_pre,
  alpha = 0.05,
  pretested_test = c("ancova", "gain", "repeated"),
  stouffer = TRUE,
  stouffer_direction = c("greater", "less"),
  conf_level = 0.95,
  flow = c("1988", "1990", "1995"),
  alpha_allocation = c("none", "method1_conservative", "method1_liberal",
    "method2_conservative", "method2_liberal"),
  data = NULL,
  combine_with_stouffer = deprecated()
)
```

Arguments

y_post	Numeric posttest scores.
treat	Treatment indicator coded 0 = control and 1 = treatment. Designs with several treatments are not supported; see <code>fit_solomon_glm()</code> .
pretested	Pretest indicator coded 0 = unpretested and 1 = pretested.
y_pre	Numeric pretest scores. Values should be missing for participants assigned to the unpretested groups.
alpha	Significance level used to reconstruct the historical decision pathway. Default is 0.05.
pretested_test	Which historical pretested-group analysis should be followed in the decision pathway: "ancova", "gain", or "repeated". All three are still calculated and returned.
stouffer	Logical. If TRUE, include Test I, the Walton Braver & Braver (1988) Stouffer combination, in the decision pathway when earlier treatment tests are nonsignificant.
stouffer_direction	Direction of the historical one-tailed treatment hypothesis used for Test I: "greater" or "less".
conf_level	Confidence level for intervals. Default is 0.95.
flow	Which published version of the decision sequence to follow. "1988" (the default) is the original sequence of Walton Braver and Braver (1988, pp. 151–153): testing stops at the first significant test among D, the selected pretested-groups test (E, F, or G), and H, and Test I is reached only when all are nonsignificant. "1990" is the authors' amendment (Braver & Walton Braver, 1990, p. 322): once Tests A and D are nonsignificant, every test through Test I is run and Test I is regarded as the most definitive. "1995" is the authors' later revision, which removed Test D (Walton Braver & Braver, 1995, as cited in Sawilowsky, 1996, p. 2); the revision itself is unpublished. The remaining tests follow the 1988 rule, each reached only if the ones before it are nonsignificant, as in Sawilowsky's (1996) simulation of the revised sequence. Only the path and conclusion differ; every test is always calculated.
alpha_allocation	Test-wise significance levels for the historical sequence. "none" (the default) uses alpha for every test. The other options are the two Bonferroni-type allocations of Sawilowsky (1996, Table 4), which he calibrated by Monte Carlo so that the experiment-wise Type I error of the 1995 sequence stays within Bradley's (1968, as cited in Sawilowsky, 1996) conservative ("_conservative") or liberal ("_liberal") robustness limit for a nominal alpha of .05: • "method1_conservative" and "method1_liberal": Tests A, E, H, and I each at .020 or .0275; • "method2_conservative" and "method2_liberal": Test A at .05, as a preliminary test, and Tests E, H, and I each at .005 or .020. An allocation requires flow = "1995", alpha = 0.05, pretested_test = "ancova", and stouffer = TRUE, the conditions for which the levels were obtained. Tests B and C, reached only after a significant Test A, are outside the allocation and keep alpha; that choice is solomonR's.

`data` Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (`y_post = post`) or as strings (`y_post = "post"`).

`combine_with_stouffer` **[Deprecated]** Use `stouffer`.

Details

The historical sequence includes the four-group posttest factorial model, simple treatment effects, ANCOVA, gain-score analysis, the equivalent two-wave repeated-measures interaction, the posttest-only comparison, and the optional Stouffer meta-analytic combination.

Test I, the Walton Braver & Braver (1988) Stouffer meta-analytic combination, is included for historical replication and teaching. It converts the one-tailed p-values of the pretested-groups test and Test H, in the direction of `stouffer_direction`, to z and combines them (p. 152). The combined z is judged by its two-tailed p-value, as in their worked example (p. 153: $z = 2.05$, $p = .040$); the one-tailed value is kept as `p_one_tailed`. Later work (see Sawilowsky et al., 1994) raised concerns about experiment-wise Type I error when the procedure is used conditionally. Its presence in this function should not be interpreted as a general contemporary recommendation.

Value

An object of class `solomon_classic`. The `tests` component contains Tests A-I, while `path` records the historical decision sequence for the observed data under the chosen flow. The `history` component holds the history/maturation comparisons.

Error rates and the published replication

The package's replication of the published Type I error rates (issue #51; normal data, 30 per group, 20,000 datasets per condition) found:

- **Tests A to H** reached and rejected at the rates Sawilowsky et al. (1994, Table 2) and Sawilowsky (1996, Table 1) report.
- **Test I** rejected far more often than published. With its two-tailed criterion, the experiment-wise error rates were .137 for the 1988 sequence and .135 for the 1995 sequence, against the published .137 and .125. A post hoc investigation reproduced the published Test I rates only when the two-sided p-values of Tests E and H were converted to z as if one-tailed, a reading that differs from Walton Braver and Braver's (1988, p. 152) definition.
- **The alpha allocations**, with Test I as defined here, gave experiment-wise error rates of .059, .079, .056, and .081 for "method1_conservative", "method1_liberal", "method2_conservative", and "method2_liberal", against Sawilowsky's .054, .072, .057, and .076. All four exceed their limits of .055 and .075.

Until this version, Test I was judged by the one-tailed p of the combined z. It now follows Walton Braver and Braver's worked example (p. 153), which reports the two-tailed p.

See the article "Historical Tests: Replicating the Published Error Rates".

History and maturation

The history component compares the unpretested control posttest (O6) with the pretests of the pretested groups (O1 and O3) by independent-samples *t* tests, as Mai et al. (2020, p. 8) report. Neither group had received the treatment when these scores were measured, so a difference estimates the combined effect of history and maturation between the two occasions (Campbell & Stanley, 1963/1966, p. 25), provided assignment was random and the measure is comparable at both occasions. Solomon (1949, pp. 146–148) introduced the fourth group for this purpose, attributing its change from the pretest to outside events and the passage of time, and Campbell (1957, p. 303) first proposed the *t* test, together with the 2 x 2 analysis of variance of the posttests that Tests A–D carry out. It is reported as a historical check, not a test of the treatment.

Confidence intervals

Tests A–H are *t* tests (equivalently, *F* tests with one numerator degree of freedom) with residual degrees of freedom, and each result carries the matching confidence interval (conf.low, conf.high). Test I combines *p*-values and has no interval. The Groups 3–4 standardized mean difference (g_post) reports Hedges' *g* with a noncentral *t* interval for the population standardized mean difference (Cumming & Finch, 2001; Kelley, 2007).

References

- Braver, S. L., & Walton Braver, M. C. (1990). Meta-analysis for Solomon four-group designs reconsidered: A reply to Sawilowsky and Markman. *Perceptual and Motor Skills*, 71(1), 321–322. <https://doi.org/10.2466/pms.1990.71.1.321>
- Campbell, D. T. (1957). Factors relevant to the validity of experiments in social settings. *Psychological Bulletin*, 54(4), 297–312. <https://doi.org/10.1037/h0040950>
- Campbell, D. T., & Stanley, J. C. (1966). *Experimental and quasi-experimental designs for research*. Rand McNally. (Original work published 1963)
- Cumming, G., & Finch, S. (2001). A primer on the understanding, use, and calculation of confidence intervals that are based on central and noncentral distributions. *Educational and Psychological Measurement*, 61(4), 532–574. <https://doi.org/10.1177/00131640121971374>
- Huck, S. W., & Sandler, H. M. (1973). A note on the Solomon 4-group design: Appropriate statistical analyses. *The Journal of Experimental Education*, 42(2), 54–55. <https://doi.org/10.1080/00220973.1973.11011460>
- Kelley, K. (2007). Confidence intervals for standardized effect sizes: Theory, application, and implementation. *Journal of Statistical Software*, 20(8), 1–24. <https://doi.org/10.18637/jss.v020.i08>
- Mai, N. N., Takahashi, Y., & Oo, M. M. (2020). Testing the effectiveness of transfer interventions using Solomon four-group designs. *Education Sciences*, 10(4), Article 92. <https://doi.org/10.3390/educsci10040092>
- Sawilowsky, S. S. (1996, June 23). *Controlling experiment-wise Type I error of meta-analysis in the Solomon four-group design* [Paper presentation]. First International Conference on Multiple Comparisons, Tel Aviv, Israel. https://digitalcommons.wayne.edu/coe_tbf/29/
- Sawilowsky, S. S., Kelley, D. L., Blair, R. C., & Markman, B. S. (1994). Meta-analysis and the Solomon four-group design. *The Journal of Experimental Education*, 62(4), 361–376. <https://doi.org/10.1080/00220973.1994.10555555>
- Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>

Van Breukelen, G. J. P. (2006). ANCOVA versus change from baseline had more power in randomized studies and more bias in nonrandomized studies. *Journal of Clinical Epidemiology*, 59(9), 920–925. <https://doi.org/10.1016/j.jclinepi.2006.02.007>

Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

Examples

```
classic <- with(solomon_example, fit_solomon_classic(y_post, treat, pretested, y_pre))
classic

# The 1995 revision of the sequence, without Test D (Sawilowsky, 1996).
with(solomon_example, fit_solomon_classic(y_post, treat, pretested, y_pre, flow = "1995"))
```

fit_solomon_glm

Fit the unified GLM for a Solomon Four-Group design

Description

[Stable] Fits one generalized linear model to all four Solomon groups and reports four Solomon contrasts: the equal-weighted average treatment effect, the Pretest x Treatment (sensitization) contrast, and the treatment effect within each pretesting condition.

Usage

```
fit_solomon_glm(
  y_post,
  treat,
  pretested,
  y_pre = NULL,
  covariates = NULL,
  robust = c("HC3", "none", "CR2"),
  cluster = NULL,
  family = stats::gaussian(),
  conf_level = 0.95,
  exposure = NULL,
  control = NULL,
  contrasts = "control",
  adjust = c("holm", "bonferroni", "none"),
  data = NULL,
  y = deprecated(),
  pretest_score = deprecated()
)
```

Arguments

y_post	numeric posttest vector
treat	0/1 (or logical) treatment indicator (1 = treatment); or, for a design with several treatments, a factor or character vector of conditions, with the control named by control
pretested	0/1 (or logical) pretest indicator (1 = group received pretest)
y_pre	numeric pretest vector: the pretest score for pretested participants and NA for the others
covariates	optional data frame of additional covariates, or, with data, the names of its columns to use
robust	character: "HC3" (default), "none", or "CR2" (cluster-robust; requires cluster)
cluster	optional clustering id (e.g., class/site), one value per participant. CR2 fits refuse designs in which a Solomon cell contains a single cluster, because cluster and condition are then confounded; see validate_solomon() . A classed warning (solomonR_small_df_warning) flags Solomon contrasts whose Satterthwaite degrees of freedom are below 4, where Tipton (2015) advises that p-values not be trusted.
family	model family (default gaussian()), given as a family object, a family function, or its name; or "negative_binomial" for the NB2 model for counts.
conf_level	confidence level for intervals (default 0.95)
exposure	optional positive exposure (for example, observation time) for each participant, entered as a log offset; requires a log-link family such as poisson() or "negative_binomial". Contrasts are then log rate ratios per unit of exposure.
control	the control condition, when treat is a factor or character vector. With two conditions the design is a four-group design and the result is the same as with a 0/1 treat; with three or more it is an N-group design (see "Designs with several treatments").
contrasts	for designs with several treatments: "control" (each treatment against the control), "pairwise" (every pair of conditions), or a named list of weight vectors named by condition, one per comparison, each summing to zero.
adjust	for designs with several treatments: the adjustment of the p-values of each contrast across the comparisons, "holm" (Holm, 1979; the default), "bonferroni", or "none".
data	optional data frame. When supplied, the other data arguments are looked up in it first: give them as bare column names (y_post = post) or as strings (y_post = "post").
y, pretest_score	[Deprecated] Use y_post and y_pre.

Details

When y_pre is supplied, it enters the model as pre_obs, equal to the pretest score in the pretested groups and 0 in the unpretested groups, so the structurally absent pretests do not remove Groups 3 and 4. Regression adjustment for baseline covariates in randomized experiments, and the case

for pairing it with heteroskedasticity-robust standard errors, is discussed by Lin (2013). Pretested participants with a missing pretest score are excluded with a warning; incidental missingness is never imputed.

Value

An object of class `solomon_glm`: a list with the fitted model, coefficient and contrast tables (including degrees of freedom and confidence limits `conf.low` and `conf.high`), the covariance matrix, the Pearson dispersion statistic for binomial, Poisson, and negative-binomial fits, `theta` (its estimate, standard error, and $\alpha = 1/\theta$) for negative-binomial fits, and the settings used.

For a design with several treatments, an object of class `solomon_ngroup`, with the same elements and these changes: `effects` has a comparison column and the adjusted p-values `p.adjusted`; `omnibus` holds the omnibus tests (`statistic`, `df1`, `df2`, `p.value`, and `reference`, "F" or "chisq"); `conditions` names the control and the treatments and their model terms; `weights` holds the weights of each comparison; and `adjust` names the adjustment.

Inference

- `robust = "HC3"` (default): heteroskedasticity-consistent covariance (MacKinnon & White, 1985), recommended for routine use and particularly below about 250 observations (Long & Ervin, 2000; Hayes & Cai, 2007). Heteroskedasticity is expected in the unified Solomon model: when the pretest predicts the posttest, adjusting for it reduces residual variance only in the pretested groups.
- `robust = "none"`: conventional model-based covariance.
- `robust = "CR2"`: bias-reduced cluster-robust covariance (Bell & McCaffrey, 2002) with Satterthwaite degrees of freedom (Pustejovsky & Tipton, 2018), computed with the `clubSandwich` package. Use this when participants are nested in clusters such as classrooms or sites. In the package's simulation of cluster-randomized Solomon designs (issue #19), CR2 tests exceeded the nominal level with four clusters per arm (Type I error up to 0.068 with no effect in any cluster); with few clusters, the cluster-level randomization test of `perm_solomon()` is preferred.

Tests and confidence intervals use the same reference distribution. For "HC3" and "none" it is the t distribution with residual degrees of freedom when the dispersion is estimated, as in Gaussian models, which is the conventional choice when t approximations are used with robust standard errors (Imbens & Kolesár, 2016; Rajh-Weber et al., 2025). Families with a fixed dispersion (binomial, Poisson) use the normal distribution. With a noncollapsible link (such as the logit) and `y_pre`, the link-scale contrasts compare a treatment effect conditional on the pretest among pretested participants with a marginal effect among unpretested participants, who have no pretest. When the pretest predicts the outcome, the Pretest x Treatment contrast is then nonzero even without sensitization (Daniel et al., 2021): in the package's simulation study (issue #43) it averaged 0.08 to 0.21 on the log-odds scale with no sensitization present. Such fits give the classed warning `solomonR_noncollapsible_warning`; for binary outcomes, estimate the Solomon contrasts on a common scale with `marginal_solomon()`. It takes the fit of a four-group design: for a design with several treatments, subset the data to one treatment and the control and fit the subset. Identity and log links are collapsible and are not affected.

Count outcomes: with `family = poisson()`, the default HC3 covariance gives the robust (quasi-likelihood) inference that Cameron and Trivedi (2013) describe. The Poisson estimator stays consistent when the counts are not Poisson, provided the mean is correctly specified (p. 72), whereas

model-based standard errors should not be used under overdispersion. The Pearson dispersion statistic is returned as `dispersion` for description; values well above 1 indicate overdispersion. Log-link rate ratios are collapsible, so the noncollapsibility caution above does not apply. Use `exposure` for counts observed over different times or exposures, and `marginal_solomon()` for rate differences. No published Solomon study with a count outcome has been identified, so applying these count-data methods to the Solomon contrasts is a `solomonR` extension; its simulation validation is on issue #44.

`family = "negative_binomial"` fits the NB2 model, with variance $\mu + \alpha\mu^2$, by maximum likelihood using `MASS::glm.nb()` (Venables & Ripley, 2002). Its coefficient estimates stay consistent when the counts are not negative binomial, provided the mean is correctly specified, but its model-based standard errors do not, so robust standard errors are advised (Cameron & Trivedi, 2013, pp. 84–85); the default HC3 covariance provides them. The estimated $\theta = 1/\alpha$ is returned as `theta`. When the counts show little overdispersion, θ does not converge and a classed warning (`solomonR_theta_boundary_warning`) is given. Clustered (CR2) negative-binomial fits are not supported.

In the package's pre-registered simulation (issue #62), which reused the datasets of issue #44, the NB2 fit with HC3 met the coverage and Type I tolerances in 85% of the overdispersed contrasts with 50 or 100 participants per cell, against 82% for robust Poisson. That fell short of the 90% set in advance for recommending it, although its estimates were on average 3.5% more precise with strong overdispersion. Robust Poisson therefore remains the recommendation, and Poisson is preferred when there is no evidence of overdispersion. Model-based NB2 standard errors fell outside the coverage tolerance in 162 of 384 contrasts and should not be used.

The degrees of freedom are returned in the `df` columns (Inf for normal reference distributions). Imbens and Kolesár (2016) further recommend Bell-McCaffrey degrees of freedom for heteroskedasticity-robust intervals; that refinement is under evaluation. For non-identity links, the contrasts are on the link scale.

In the package's simulation validation (issues #10 and #22), HC3 intervals for the Solomon contrasts were conservative with 10 or fewer participants per cell (mean coverage of nominal 95% intervals was 0.961 with 6 per cell and 0.958 with 10, and the Pretest x Treatment test had a Type I error of 0.034 with 6 per cell) and close to nominal with 20 or more (mean coverage 0.951 to 0.954).

Confidence intervals for the Wald partial R-squared use the noncentral F method (Steiger, 2004) and are reported only for conventional Gaussian fits; no corresponding interval is available with robust covariance.

Designs with several treatments

A Solomon N-group design crosses k treatments and a control with pretesting, giving $2(k + 1)$ groups: six for two treatments and eight for three (Steyn, 2009). Give `treat` as a factor or character vector of conditions and name the control with `control`. One model is then fitted to all the groups, with an indicator for each treatment, and the four Solomon contrasts are estimated for each comparison:

- each treatment against the control (`contrasts = "control"`, the default);
- every pair of conditions (`contrasts = "pairwise"`);
- or comparisons given as weights over the conditions, such as the main effects of two treatments crossed factorially: `contrasts = list(Lecture = c(Lecture = 0.5, Both = 0.5, Service = -0.5, None = -0.5))`. The weights of each comparison must sum to zero.

Omnibus Wald tests ask whether the conditions differ on each contrast. The Pretest x Condition test asks whether pretesting changes the effect of any treatment.

The p-values of each contrast are adjusted across the comparisons, by Holm's (1979) procedure by default. It controls the familywise error rate "for any combination of true hypotheses" (p. 65). The confidence intervals are not adjusted. The result has class `solomon_ngroup`.

Published studies with several treatments analyzed them as overlapping four-group designs: one for each treatment against the control (McCarthy & Tucker, 2002), or one for each pair of conditions (Mai et al., 2020). Those analyses reuse the same groups, so their tests are dependent, and each extra analysis adds to the chance of a false finding. The joint model asks each question once. In Mai et al.'s (2020) six-group study, the Pretest x Condition test of the posttests alone (without the pretest as a covariate), with conventional covariance, gives $F(2, 127) = 1.86$, $p = .161$. `fit_solomon_steyn()` carries out the sequence of tests Steyn (2009) proposed for these designs.

[Experimental] The analysis of designs with several treatments is experimental. In the package's simulation study (issue #45; 112 scenarios with two or three treatments and 10 to 50 participants per group, 5,000 replications each):

- the contrasts were unbiased, and coverage of their 95% intervals was 0.939 to 0.967;
- the familywise error rates of the Holm-adjusted comparisons were at most 0.059;
- the Pretest x Condition test and the test of the conditions averaged over pretest rejected a true null hypothesis in 0.036 to 0.055 of replications;
- with three treatments and 10 participants per group, the omnibus tests of Condition | pretested and Condition | unpretested rejected in 0.055 to 0.069 of replications at the .05 level. The rule for error control set before the study was therefore not met, which is why the analysis is experimental. With groups that small, judge those two questions by the adjusted comparisons.

The study did not cover binary or count outcomes, clustered designs, or comparisons given as weights. It is reported in the article "Designs With Several Treatments: Validating the Joint Model".

References

- Bell, R. M., & McCaffrey, D. F. (2002). Bias reduction in standard errors for linear regression with multi-stage samples. *Survey Methodology*, 28(2), 169–181.
- Cameron, A. C., & Trivedi, P. K. (2013). *Regression analysis of count data* (2nd ed.). Cambridge University Press. <https://doi.org/10.1017/CBO9781139013567>
- Daniel, R., Zhang, J., & Farewell, D. (2021). Making apples from oranges: Comparing noncollapsible effect estimators and their standard errors after adjustment for different covariate sets. *Biometrical Journal*, 63(3), 528–557. <https://doi.org/10.1002/bimj.201900297>
- Hayes, A. F., & Cai, L. (2007). Using heteroskedasticity-consistent standard error estimators in OLS regression: An introduction and software implementation. *Behavior Research Methods*, 39(4), 709–722. <https://doi.org/10.3758/BF03192961>
- Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2), 65–70. <https://www.jstor.org/stable/4615733>
- Imbens, G. W., & Kolesár, M. (2016). Robust standard errors in small samples: Some practical advice. *The Review of Economics and Statistics*, 98(4), 701–712. https://doi.org/10.1162/REST_a_00552

- Lin, W. (2013). Agnostic notes on regression adjustments to experimental data: Reexamining Freedman's critique. *The Annals of Applied Statistics*, 7(1), 295–318. <https://doi.org/10.1214/12-AOAS583>
- Long, J. S., & Ervin, L. H. (2000). Using heteroscedasticity consistent standard errors in the linear regression model. *The American Statistician*, 54(3), 217–224. <https://doi.org/10.1080/00031305.2000.10474549>
- MacKinnon, J. G., & White, H. (1985). Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties. *Journal of Econometrics*, 29(3), 305–325. [https://doi.org/10.1016/0304-4076\(85\)90158-7](https://doi.org/10.1016/0304-4076(85)90158-7)
- Mai, N. N., Takahashi, Y., & Oo, M. M. (2020). Testing the effectiveness of transfer interventions using Solomon four-group designs. *Education Sciences*, 10(4), Article 92. <https://doi.org/10.3390/educsci10040092>
- McCarthy, A. M., & Tucker, M. L. (2002). Encouraging community service through service learning. *Journal of Management Education*, 26(6), 629–647. <https://doi.org/10.1177/1052562902238322>
- Pustejovsky, J. E., & Tipton, E. (2018). Small-sample methods for cluster-robust variance estimation and hypothesis testing in fixed effects models. *Journal of Business & Economic Statistics*, 36(4), 672–683. <https://doi.org/10.1080/07350015.2016.1247004>
- Rajh-Weber, H., Huber, S. E., & Arendasy, M. (2025). A practice-oriented guide to statistical inference in linear modeling for non-normal or heteroskedastic error distributions. *Behavior Research Methods*, 57(12), Article 338. <https://doi.org/10.3758/s13428-025-02801-4>
- Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>
- Steiger, J. H. (2004). Beyond the F test: Effect size confidence intervals and tests of close fit in the analysis of variance and contrast analysis. *Psychological Methods*, 9(2), 164–182. <https://doi.org/10.1037/1082-989X.9.2.164>
- Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>
- Tipton, E. (2015). Small sample adjustments for robust variance estimation with meta-regression. *Psychological Methods*, 20(3), 375–393. <https://doi.org/10.1037/met0000011>
- Venables, W. N., & Ripley, B. D. (2002). *Modern applied statistics with S* (4th ed.). Springer. <https://doi.org/10.1007/978-0-387-21706-2>

Examples

```
fit <- fit_solomon_glm(y_post, treat, pretested, y_pre, data = solomon_example)
fit

# The four Solomon contrasts as a data frame.
fit$effects

# A six-group design: two treatments and a control (Mai et al., 2020).
fit6 <- fit_solomon_glm(post_behavior, condition, pretested, pre_behavior,
                        control = "Control", data = mai2020)
fit6
```

fit_solomon_mi

*Solomon analysis with multiply imputed posttests***Description**

[Experimental] Multiply imputes missing posttests, analyzes each completed data set with `fit_solomon_glm()`, and combines the four Solomon contrasts with Rubin's rules. With `delta = 0` the imputations assume the posttests are missing at random (MAR). A nonzero delta shifts the imputed posttests of each group by a fixed amount, the delta-adjusted pattern-mixture sensitivity analysis of Carpenter et al. (2023, section 10.3). `tipping_point_solomon()` repeats it over a range of offsets.

Usage

```
fit_solomon_mi(
  y_post,
  treat,
  pretested,
  y_pre = NULL,
  delta = 0,
  m = 100,
  robust = c("HC3", "none"),
  conf_level = 0.95,
  seed = NULL,
  data = NULL
)
```

Arguments

<code>y_post</code>	Numeric posttest scores, with NA for missing posttests.
<code>treat</code>	Treatment indicator coded 0/1 (or logical). Designs with several treatments are not supported; see <code>fit_solomon_glm()</code> .
<code>pretested</code>	Pretest indicator coded 0/1 (or logical).
<code>y_pre</code>	Optional numeric pretest scores, missing by design for unpretested participants.
<code>delta</code>	Offsets added to the imputed posttests, on the posttest scale: one number for all four groups, or four numbers for Groups 1-4 (pretested treatment, pretested control, unpretested treatment, unpretested control). The default, 0, is the MAR analysis.
<code>m</code>	Number of imputations. Default 100.
<code>robust</code>	Covariance for each completed-data analysis: "HC3" (default) or "none"; see <code>fit_solomon_glm()</code> .
<code>conf_level</code>	Confidence level. Default 0.95.
<code>seed</code>	Optional random-number seed. The global random number state is restored afterwards.
<code>data</code>	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (<code>y_post = post</code>) or as strings (<code>y_post = "post"</code>).

Value

An object of class `solomon_mi` with effects (the pooled Solomon contrasts, with their degrees of freedom, fraction of missing information `fmi`, and Monte Carlo standard error `mc_se`), `delta`, `m`, `missing` (missing posttests by group), `excluded`, the per-imputation estimates and `std_errors`, and the settings used.

Method

1. **Imputation model.** A normal linear regression is fitted separately in each Solomon group to the participants with an observed posttest: on the pretest in the pretested groups, and on a constant in the unpretested groups, which have no pretest by design. The analysis contains the Treatment x Pretest interaction of two fully observed indicators, and imputing separately in the groups they define is the simplest approach to such interactions (Carpenter et al., 2023, section 6.3.5, p. 149).
2. **Proper imputation.** For each imputation the residual variance and the coefficients are drawn from their posterior distribution, and the missing posttests are then drawn from the model (Carpenter et al., 2023, pp. 81-83).
3. **Offsets.** Each imputed posttest in group j is shifted by `delta[j]`: those with a missing posttest are assumed to differ from those observed by an offset in each group, as in the pattern-mixture analysis of Little et al. (2012, p. 1358), and "a clinically plausible amount" is added to the imputed outcomes (White et al., 2011, "Perform Sensitivity Analyses" section, para. 1).
4. **Analysis and pooling.** Each completed data set is analyzed with the model of `fit_solomon_glm()` (computed directly, since the design is the same in every completed data set), and the contrasts are combined with Rubin's rules: the mean estimate, with variance $W + (1 + 1/m)B$ (Carpenter et al., 2023, Eq. 2.16). Tests and intervals use t with the small-sample degrees of freedom of Barnard and Rubin (1999, as cited in van Buuren, 2018, Eqs. 2.30-2.32), taking the complete-data residual degrees of freedom as their starting point.

With fixed offsets, multiple imputation with Rubin's variance is information-anchored: the sensitivity analysis neither adds nor removes information relative to the MAR analysis (Cro et al., 2019; Carpenter et al., 2023, p. 278).

Structural and incidental missingness. Pretests absent by design are never imputed. Pretested participants with a missing pretest are excluded with a warning, as in `fit_solomon_glm()`; `check_solomon_missing()` describes the options for them.

Number of imputations. The default $m = 100$ follows Carpenter et al. (2023, p. 56), who note that p -values accurate to about .005 need at least 100 imputations and advise erring "towards too many imputations rather than too few". With much missing information, more are needed: in the worked example on `mai2020`, where the fraction of missing information is about 0.4, p -values varied by about .01 from one seed to another with 100 imputations. The `mc_se` column gives the Monte Carlo standard error of each estimate due to the finite number of imputations.

Validation

A simulation study under a protocol posted on issue #82 before any run (24 scenarios, 2,000 replications each; see the article "Missing Posttests: Validating the Sensitivity Analysis") found:

- **With 60 or more participants per group**, coverage of 95% intervals from 0.9415 to 0.9595 and Type I error within 0.040 to 0.060, under missing at random and under the pattern-mixture

departures studied when the offsets were right. Bias exceeded 2 Monte Carlo standard errors in 2 of 64 contrasts, by at most 0.011 SD.

- **With 30 per group**, conservative intervals: coverage up to 0.967 and model standard errors 3.9% above the empirical ones on average.
- **Missing at random**: agreement with the complete-case analysis, as theory predicts (Carpenter et al., 2023, p. 256).
- **Departures that differ between the pretested groups**: the analyses that assume missing at random biased the sensitization contrast by 0.10 to 0.16 SD.

Lifecycle

Experimental. The study's pre-specified rule for validation, every tolerance met in 90% of cells and in every cell with 60 or more per group, was not met: 84 of 96 cells and 62 of 64.

References

- Carpenter, J. R., Bartlett, J. W., Morris, T. P., Wood, A. M., Quartagno, M., & Kenward, M. G. (2023). *Multiple imputation and its application* (2nd ed.). Wiley. <https://doi.org/10.1002/9781119756118>
- Cro, S., Carpenter, J. R., & Kenward, M. G. (2019). Information-anchored sensitivity analysis: Theory and application. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 182(2), 623–645. <https://doi.org/10.1111/rssa.12423>
- Little, R. J., D'Agostino, R., Cohen, M. L., Dickersin, K., Emerson, S. S., Farrar, J. T., Frangakis, C., Hogan, J. W., Molenberghs, G., Murphy, S. A., Neaton, J. D., Rotnitzky, A., Scharfstein, D., Shih, W. J., Siegel, J. P., & Stern, H. (2012). The prevention and treatment of missing data in clinical trials. *The New England Journal of Medicine*, 367(14), 1355–1360. <https://doi.org/10.1056/NEJMSr1203730>
- van Buuren, S. (2018). *Flexible imputation of missing data* (2nd ed.). CRC Press. <https://doi.org/10.1201/9780429492259>
- White, I. R., Horton, N. J., Carpenter, J., & Pocock, S. J. (2011). Strategy for intention to treat analysis in randomised trials with missing outcome data. *BMJ*, 342, Article d40. <https://doi.org/10.1136/bmj.d40>

See Also

[tipping_point_solomon\(\)](#), [check_solomon_missing\(\)](#)

Examples

```
d <- solomon_example
set.seed(82)
d$y_post[sample(nrow(d), 20)] <- NA

# Missing at random.
fit_solomon_mi(y_post, treat, pretested, y_pre, m = 20, seed = 1, data = d)

# Missing posttests in the treatment groups 3 points lower than MAR predicts.
fit_solomon_mi(y_post, treat, pretested, y_pre, delta = c(-3, 0, -3, 0),
               m = 20, seed = 1, data = d)
```

fit_solomon_ml

*Full-information ML analysis for a Solomon Four-Group Design***Description**

[Stable] Fits the maximum-likelihood regression model described by van Engelenburg (1999). Pretest information is incorporated for the pretested groups while structurally missing pretests in the unpretested groups are handled through a separate residual variance.

Usage

```
fit_solomon_ml(
  y_post,
  treat,
  pretested,
  y_pre,
  weights = c("equal"),
  control = list(),
  conf_level = 0.95,
  inference = c("wald", "satterthwaite"),
  data = NULL
)
```

Arguments

y_post	Numeric posttest scores.
treat	Treatment indicator coded 0 = control and 1 = treatment. Designs with several treatments are not supported; see fit_solomon_glm() .
pretested	Pretest indicator coded 0 = unpretested and 1 = pretested.
y_pre	Numeric pretest scores. These should be missing by design for participants assigned to the unpretested groups.
weights	Character. How to define the average treatment effect across pretest conditions. Currently "equal" gives equal weight to the pretested and unpretested treatment effects.
control	Optional list passed to <code>stats::optim()</code> .
conf_level	Confidence level for intervals. Default is 0.95.
inference	How standard errors, tests, and intervals are computed: "wald" (default) for van Engelenburg's (1999) large-sample Wald inference, or "satterthwaite" for the small-sample option. The point estimates are the same. See the Inference options section.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (<code>y_post = post</code>) or as strings (<code>y_post = "post"</code>).

Details

The model estimates the treatment effect, pretest effect, Treatment x Pretest interaction, pretest-posttest slope, and separate residual standard deviations for pretested and unpretested participants.

Value

An object of class `solomon_ml`.

Inference options

The point estimates are maximum-likelihood estimates, which coincide with separate regressions in the pretested and unpretested groups.

- `inference = "wald"` (default) follows van Engelenburg (1999): standard errors come from the observed information matrix, and tests and intervals use a normal reference distribution, the usual large-sample basis for maximum-likelihood inference.
- `inference = "satterthwaite"` is a small-sample option. Standard errors use unbiased residual variances within each pretest condition. Contrasts within one condition use t tests with that condition's residual degrees of freedom, and contrasts that combine the conditions (the ATE and Pretest x Treatment) use Welch-Satterthwaite degrees of freedom (Satterthwaite, 1946; Welch, 1947).

In the package's simulation validation (issues #10 and #22; 84 scenarios with 2,000 replications each, reported in the article "Validating `fit_solomon_ml()`" on the package website), both options recovered the Solomon contrasts without bias. Wald intervals were too narrow in small samples: mean coverage of nominal 95% intervals was 0.893 with 6 participants per cell, 0.920 with 10, 0.936 with 20, 0.941 with 30, and 0.948 with 100, and the Pretest x Treatment test rejected a true null hypothesis in 9.9% of samples with 6 per cell and 5.9% with 30. The small-sample option had mean coverage of 0.949 to 0.950 and Type I error of 0.050 to 0.053 at every cell size studied, from 6 to 100 per cell.

When the smallest cell has fewer than 40 participants and `inference` is not supplied, `fit_solomon_ml()` issues a warning of class `solomonR_small_sample_warning` that suggests the small-sample option. Supplying `inference = "wald"` explicitly keeps the default without the warning. The threshold follows a rule set before the validation results were examined: 40 is the smallest cell size at which Wald inference had mean coverage of at least 0.940 and Type I error of at most 0.060, with equal and unequal residual variances, at that size and every larger size studied.

References

- Satterthwaite, F. E. (1946). An approximate distribution of estimates of variance components. *Biometrics Bulletin*, 2(6), 110–114. <https://doi.org/10.2307/3002019>
- van Engelenburg, G. (1999). *Statistical analysis for the Solomon four-group design* (Research Report 99-06). University of Twente. ERIC. <https://eric.ed.gov/?id=ED435692>
- Welch, B. L. (1947). The generalization of "Student's" problem when several different population variances are involved. *Biometrika*, 34(1–2), 28–35. <https://doi.org/10.1093/biomet/34.1-2.28>

Examples

```
# With fewer than 40 participants per cell, use the small-sample
# (Satterthwaite) inference; see "Inference options".
with(solomon_example, fit_solomon_ml(y_post, treat, pretested, y_pre,
                                     inference = "satterthwaite"))
```

fit_solomon_mmrmm	<i>Mixed model for repeated measures in a longitudinal Solomon design</i>
-------------------	---

Description

[Experimental] Analyzes a Solomon four-group design with several posttest occasions by a mixed model for repeated measures (MMRM), and estimates the four Solomon contrasts at each occasion and the change in pretest sensitization from the first occasion to the last. The model is likelihood-based, so it is valid when posttests are missing at random, for example when participants drop out depending on their earlier scores (Fitzmaurice et al., 2011, pp. 497, 505). Per-occasion analyses of the participants still observed, and unweighted generalized estimating equations, require the stronger assumption that the posttests are missing completely at random (Liang & Zeger, 1986; Fitzmaurice et al., 2011, pp. 358, 498).

Usage

```
fit_solomon_mmrmm(
  y_post,
  treat,
  pretested,
  id,
  occasion,
  y_pre = NULL,
  df = c("kenward-roger", "satterthwaite"),
  conf_level = 0.95,
  data = NULL
)
```

Arguments

y_post	Numeric posttest scores, one per participant and occasion (long format), with NA for missing posttests.
treat	Treatment indicator coded 0/1 (or logical), the same in every row of a participant. Designs with several treatments are not supported; see fit_solomon_glm() .
pretested	Pretest indicator coded 0/1 (or logical), the same in every row of a participant.
id	Participant identifier.
occasion	Posttest occasion. A factor keeps its level order; otherwise the sorted unique values are used.
y_pre	Optional pretest score, the same in every row of a participant and missing by design for unpretested participants.

df	Degrees of freedom for tests and intervals: "kenward-roger" (default) or "satterthwaite".
conf_level	Confidence level. Default 0.95.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Value

An object of class `solomon_mmrn` with effects (the Solomon contrasts at each occasion and the change in sensitization), model (the mmrm fit), covariance (the structure used), observed (observed posttests by group and occasion), and the settings used.

Model

The specification follows the example of Mallinckrodt et al. (2008, p. 312), adapted to the Solomon design:

- **Fixed effects.** Occasion x Treatment x Pretested, all categorical.
- **Pretest adjustment.** The pretest enters as in `fit_solomon_glm()`: the score in the pretested groups and 0 in the unpretested groups, with a separate slope at each occasion ("a full interaction of the covariate with time", p. 312). Pretests absent by design are never imputed.
- **Covariance.** Unstructured within participant, estimated by restricted maximum likelihood (Laird & Ware, 1982). With a pretest, it is estimated separately for pretested and unpretested participants: the pretest adjustment makes the residual covariance conditional on the pretest in the pretested groups only, so a single shared matrix would be misspecified.
- **Fallback.** If the unstructured model does not converge, the heterogeneous Toeplitz, heterogeneous AR(1), and heterogeneous compound symmetry structures are fitted, and the converged one with the smallest AIC is used (Mallinckrodt et al., 2008, p. 312). The output names the structure used.
- **Inference.** t tests and intervals with Kenward-Roger degrees of freedom and adjusted covariance (Kenward & Roger, 1997, as cited in Fitzmaurice et al., 2011, p. 101), as Mallinckrodt et al. (2008, p. 312) specify, or with Satterthwaite (1946) degrees of freedom. The Kenward-Roger adjustment depends on how the covariance is parameterized; it is computed with the matrix's own elements as the parameters, the form that reproduces SAS PROC MIXED (mmrm's "Kenward-Roger-Linear"; Sabanes Bove et al., 2026). Fitzmaurice et al. (2011, p. 101) note that the small-sample properties of both approximations in longitudinal models "have not been extensively studied"; the simulation study posted on issue #57 compares them in Solomon designs.

The model is fitted with the mmrm package (Sabanes Bove et al., 2026), which must be installed.

Validation

A simulation study under a protocol posted on issue #57 before any run (12 scenarios, 2,000 replications each; see the article "Longitudinal Designs: Validating the Repeated-Measures Analysis") generated monotone dropout that depended on the last posttest. With 30 to 120 participants per group:

- **Coverage and Type I error.** With Kenward-Roger degrees of freedom, 95% intervals covered from 0.937 to 0.962 of the time, and Type I error where the true value was zero ranged from 0.041 to 0.0615. Satterthwaite degrees of freedom performed alike.
- **Bias.** The contrasts were unbiased; the largest bias was 0.029 SD.
- **Normal reference distribution.** It gave coverage as low as 0.935 at 30 per group.
- **Shared covariance.** A covariance shared by all four groups misstated the standard errors by up to 12% for the unpretested contrasts and 17% for the pretested ones.
- **Complete-case analyses.** Per-occasion analyses of the participants still observed were biased by up to 0.09 SD at the last occasion.

Lifecycle

Experimental. The study's pre-specified rule for choosing the default degrees of freedom, every tolerance met in 90% of cells and in every cell with 60 or more per group, was not met by either approximation (145 of 156 cells each; 97 and 96 of 104), although Monte Carlo error alone makes the second part unlikely to be met even by a correctly calibrated method. Kenward-Roger remains the default, as Mallinckrodt et al. (2008) specify.

References

- Fitzmaurice, G. M., Laird, N. M., & Ware, J. H. (2011). *Applied longitudinal analysis* (2nd ed.). Wiley. <https://doi.org/10.1002/9781119513469>
- Laird, N. M., & Ware, J. H. (1982). Random-effects models for longitudinal data. *Biometrics*, 38(4), 963–974. <https://doi.org/10.2307/2529876>
- Liang, K.-Y., & Zeger, S. L. (1986). Longitudinal data analysis using generalized linear models. *Biometrika*, 73(1), 13–22. <https://doi.org/10.1093/biomet/73.1.13>
- Mallinckrodt, C. H., Lane, P. W., Schnell, D., Peng, Y., & Mancuso, J. P. (2008). Recommendations for the primary analysis of continuous endpoints in longitudinal clinical trials. *Drug Information Journal*, 42(4), 303–319. <https://doi.org/10.1177/009286150804200402>
- Sabanes Bove, D., Li, L., Dedic, J., Kelkhoff, D., Kunzmann, K., Lang, B. M., Stock, C., Wang, Y., James, D., Sidi, J., Leibovitz, D., Sjoberg, D. D., Krieger, N. I., Panagos, A., & Jones, J. (2026). *mmrm: Mixed models for repeated measures* (Version 0.3.18) [R package]. <https://doi.org/10.32614/CRAN.package.mmrmm>
- Satterthwaite, F. E. (1946). An approximate distribution of estimates of variance components. *Biometrics Bulletin*, 2(6), 110–114. <https://doi.org/10.2307/3002019>

See Also

[fit_solomon_glm\(\)](#) for a single posttest occasion; [jordaana2014](#) and the article "Worked Example: Repeated Posttests" for a published study with three posttest occasions.

Examples

```
if (requireNamespace("mmrm", quietly = TRUE)) {
  # Three posttest occasions for the example participants, with dropout.
  set.seed(57)
  d <- solomon_example
  d$id <- seq_len(nrow(d))
}
```

```

long <- do.call(rbind, lapply(1:3, function(t) {
  x <- d
  x$occasion <- t
  x$y_post <- d$y_post - 2 * (t - 1) + rnorm(nrow(d), 0, 4)
  x
}))
long$y_post[long$occasion == 3 & runif(nrow(long)) < 0.3] <- NA
fit_solomon_mmrn(y_post, treat, pretested, id, occasion, y_pre, data = long)
}

```

fit_solomon_sem	<i>SEM analysis for Solomon Four-Group designs (mean-structure; optional ANCOVA)</i>
-----------------	--

Description

[Experimental] Two modes:

1. mean-structure (default): 4-group SEM estimating posttest means for P1, P0, U1, U0, and reporting ATE, Sens (Pretest×Treatment), Pre_Eff, Unpre_Eff.
2. ancova = TRUE: restricts to pretested groups (P1, P0) and fits $y_{\text{post}} \sim \beta y_{\text{pre}}$ with group means; reports the pretested simple effect (Pre_Eff). This avoids structural missingness of y_{pre} in U1/U0 and matches Huck & Sandler.

Usage

```

fit_solomon_sem(
  y_post,
  treat,
  pretested,
  y_pre = NULL,
  equal_var = FALSE,
  ancova = FALSE,
  estimator = "MLR",
  conf_level = 0.95,
  data = NULL
)

```

Arguments

y_post	numeric posttest
treat	0/1 (or logical) treatment indicator. Designs with several treatments are not supported; see fit_solomon_glm() .
pretested	0/1 (or logical) pretest indicator
y_pre	optional pretest score (required if ancova = TRUE)

equal_var	logical; if TRUE, constrain posttest variances equal across groups
ancova	logical; if TRUE, fit ANCOVA in pretested groups only (P1 vs P0)
estimator	lavaan estimator (default "MLR")
conf_level	confidence level for intervals (default 0.95)
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Details

The four-group mean-structure model is saturated, so its global fit indices are not diagnostic. Its contrasts are unadjusted posttest mean differences, whereas the ANCOVA mode adjusts for the pretest within the pretested groups.

Tests and confidence intervals for the contrasts are lavaan's Wald results, which use a large-sample normal reference distribution.

Value

An object of class `solomon_sem` with mode ("mean" or "ancova_pretested"), the lavaan fit, the contrasts in effects (estimate, standard error, z, p value, and confidence interval), the fit measures, and `conf_level`.

Lifecycle

Experimental. Its tests rest on large-sample (MLR) theory, and no simulation study in the package has yet checked their error rates at Solomon sample sizes. `fit_solomon_glm()` is the validated analysis of an observed outcome. The defaults may change after such a study.

References

- Huck, S. W., & Sandler, H. M. (1973). A note on the Solomon 4-group design: Appropriate statistical analyses. *The Journal of Experimental Education*, 42(2), 54–55. <https://doi.org/10.1080/00220973.1973.11011460>
- Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, 48(2), 1–36. <https://doi.org/10.18637/jss.v048.i02>

Examples

```
if (requireNamespace("lavaan", quietly = TRUE)) {
  with(solomon_example, fit_solomon_sem(y_post, treat, pretested, y_pre))
}
```

fit_solomon_sem_latent

Latent SEM for Solomon Four-Group designs (POST means; optional latent ANCOVA)

Description

[Experimental] This function provides a fully latent analysis path: (A) A 4-group SEM that defines a latent POST factor from multiple indicators and estimates group-specific latent means for P1, P0, U1, and U0. From these we compute ATE, Sens (Pretest x Treat), and simple effects on the latent outcome. (B) Optionally, a 2-group SEM in **pretested** groups only (P1 vs P0) with a latent PRE factor and latent POST factor, fitting a latent ANCOVA (POST ~ PRE), and reporting the pretested simple effect.

Usage

```
fit_solomon_sem_latent(
  data,
  post_items,
  treat,
  pretested,
  pre_items = NULL,
  invariance_post = c("scalar", "metric", "configural"),
  ancova = FALSE,
  invariance_pre = c("scalar", "metric", "configural"),
  estimator = "MLR",
  std_lv = TRUE,
  conf_level = 0.95,
  partial_post = NULL,
  partial_pre = NULL,
  check_invariance = TRUE
)
```

Arguments

data	data.frame containing all variables
post_items	character vector of posttest item names (required)
treat	0/1 (or logical) treatment indicator (length nrow(data)). Designs with several treatments are not supported; see fit_solomon_glm() .
pretested	0/1 (or logical) pretest indicator (length nrow(data))
pre_items	character vector of pretest item names (for ANCOVA branch)
invariance_post	measurement invariance for POST; must be "scalar"
ancova	logical; if TRUE, also fit latent ANCOVA in pretested groups
invariance_pre	measurement invariance for the pretested branch; must be "scalar"

estimator	lavaan estimator, default "MLR" (robust)
std_lv	logical; if TRUE (default), lavaan's <code>std.lv = TRUE</code> : the latent variance is fixed at 1 in the first group, the pretested treatment group (P1), so latent contrasts are in that group's latent SD units. In the latent ANCOVA (<code>ancova = TRUE</code>) the PRE variance and the POST residual variance are fixed at 1 in P1.
conf_level	confidence level for intervals (default 0.95)
partial_post, partial_pre	Optional freed parameters of the POST or PRE indicators for a partial-invariance model, in lavaan syntax (for example, "post3 ~ 1"); see invariance_solomon() .
check_invariance	If TRUE (the default), run invariance_solomon() on the POST indicators first, and warn when a criterion does not support scalar (or, with <code>partial_post</code> , partial scalar) invariance. Set it to FALSE when invariance was established elsewhere. See Invariance check.

Details

Latent mean contrasts require scalar measurement invariance (equal loadings and intercepts) across groups (Meredith, 1993; Vandenberg & Lance, 2000). `invariance_post` and `invariance_pre` therefore accept only "scalar"; configural and metric models are rejected with an explanation. [invariance_solomon\(\)](#) tests invariance across the four groups. If only some indicators are non-invariant, `partial_post` and `partial_pre` free their parameters for a partial-invariance model, which permits latent mean comparisons when enough indicators remain invariant (Byrne et al., 1989). Freed parameters must be chosen on substantive grounds before the analysis and involve only a minority of the indicators (Vandenberg & Lance, 2000, p. 38).

Identification: with scalar invariance, the latent POST mean of the unpretested control group (U0) is fixed at 0 and the other latent means are estimated relative to it. In the pretested ANCOVA model, the pretested control group (P0) is the reference. The Solomon contrasts are differences between latent means, so they do not depend on the reference choice.

Tests and confidence intervals for the contrasts are lavaan's Wald results, which use a large-sample normal reference distribution.

Value

An object of class `solomon_sem_latent` with:

- `fit_post`: lavaan object for the 4-group POST model
- `effects_post`: data.frame of ATE, Sens, Pre_Eff, Unpre_Eff on latent POST
- `fitmeasures_post`: named vector (CFI, RMSEA, SRMR, df)
- `fit_pre` (optional): lavaan object for pretested latent ANCOVA
- `effects_pre` (optional): data.frame with Pre_Eff on latent POST (pretested)
- `fitmeasures_pre` (optional)
- `invariance`: the [invariance_solomon\(\)](#) result, or NULL when the check was not run, and `invariance_status`, a one-line summary

Invariance check

By default the function first runs `invariance_solomon()` on the POST indicators, stores the result in `invariance`, and prints a one-line summary. When the chi-square difference test or the change-in-fit criterion does not support scalar (or, with `partial_post`, partial scalar) invariance, it warns with class `solomonR_invariance_warning`. It does not refuse the contrasts.

This follows a simulation study whose decision rules were posted on issue #55 before any run (see the article "Latent Contrasts: Validating the Invariance Check"). A criterion would have governed an automatic refusal only if it falsely rejected invariance at a rate of .060 or less in Solomon-sized groups, and none did. With six indicators and 60 participants per group, the false-rejection rates were:

- .100 for the scaled chi-square difference test (Satorra & Bentler, 2001);
- .178 for Chen's (2007) change-in-fit cutoffs;
- .090 when both had to agree.

The warning is therefore a prompt to examine the invariance results, not a verdict. The study also found which kind of noninvariance matters for the Solomon contrasts:

- **A shift common to both pretested groups.** When a pretest shifted an indicator's intercept equally in both pretested groups, the sensitization contrast stayed unbiased, because the shift cancels within the pretested condition.
- **A shift in one group only.** A shift in the unpretested control group alone biased it (by 0.07 to 0.16 latent standard deviations across the scenarios studied), and freeing the shifted intercept with `partial_post` removed the bias.

Lifecycle

Experimental. The issue #55 study found no invariance criterion that holds its false-rejection rate in Solomon-sized groups (see the Invariance check section), so how the check decides may change as better small-sample criteria are published.

References

- Byrne, B. M., Shavelson, R. J., & Muthén, B. (1989). Testing for the equivalence of factor covariance and mean structures: The issue of partial measurement invariance. *Psychological Bulletin*, 105(3), 456–466. <https://doi.org/10.1037/0033-2909.105.3.456>
- Chen, F. F. (2007). Sensitivity of goodness of fit indexes to lack of measurement invariance. *Structural Equation Modeling: A Multidisciplinary Journal*, 14(3), 464–504. <https://doi.org/10.1080/10705510701301834>
- Meredith, W. (1993). Measurement invariance, factor analysis and factorial invariance. *Psychometrika*, 58(4), 525–543. <https://doi.org/10.1007/BF02294825>
- Rosseel, Y. (2012). lavaan: An R package for structural equation modeling. *Journal of Statistical Software*, 48(2), 1–36. <https://doi.org/10.18637/jss.v048.i02>
- Satorra, A., & Bentler, P. M. (2001). A scaled difference chi-square test statistic for moment structure analysis. *Psychometrika*, 66(4), 507–514. <https://doi.org/10.1007/BF02296192>
- Vandenberg, R. J., & Lance, C. E. (2000). A review and synthesis of the measurement invariance literature: Suggestions, practices, and recommendations for organizational research. *Organizational Research Methods*, 3(1), 4–70. <https://doi.org/10.1177/109442810031002>

Examples

```
if (requireNamespace("lavaan", quietly = TRUE)) {
  set.seed(55)
  g <- rep(1:4, each = 120)
  treat <- c(1, 0, 1, 0)[g]
  pretested <- c(1, 1, 0, 0)[g]
  f <- stats::rnorm(480, 0.4 * treat)
  items <- data.frame(y1 = f + stats::rnorm(480, 0, 0.6),
                     y2 = 0.9 * f + stats::rnorm(480, 0, 0.6),
                     y3 = 0.8 * f + stats::rnorm(480, 0, 0.6),
                     y4 = 0.7 * f + stats::rnorm(480, 0, 0.6))
  fit_solomon_sem_latent(items, c("y1", "y2", "y3", "y4"), treat, pretested)
}
```

fit_solomon_steyn

Steyn's (2009) analysis of the extended Solomon design

Description

[Stable] Carries out the sequence of tests Steyn (2009) proposed for the Solomon four-group design and for its extension to several interventions, the extended Solomon design: k interventions and a control, each with and without a pretest, giving $2(k + 1)$ groups. The sequence checks the threats to internal validity the design can detect (nonequivalent groups, history and maturation, testing, the pretest-intervention interaction, instrumentation, regression to the mean, and attrition) before it compares the interventions. It is a published proposal, kept for replication and teaching. The package's recommended analysis is `fit_solomon_glm()` with `control =`, which estimates the Solomon contrasts of every intervention in one model.

Usage

```
fit_solomon_steyn(
  y_post,
  treat,
  pretested,
  y_pre = NULL,
  control = NULL,
  alpha = 0.05,
  include_unpretested_control = FALSE,
  posthoc = c("scheffe", "holm"),
  data = NULL
)
```

Arguments

<code>y_post</code>	Numeric posttest scores, with NA for participants who dropped out.
<code>treat</code>	The condition: a 0/1 (or logical) indicator for one intervention (1 = intervention), or a factor or character vector of conditions, with the control named by <code>control</code> .

pretested	Pretest indicator coded 0/1 (or logical).
y_pre	Optional numeric pretest scores, NA for unpretested participants. Without them the steps that use the pretest are skipped.
control	The control condition, when treat is a factor or character vector.
alpha	Significance level of every test in the sequence. Default 0.05.
include_unpretested_control	Logical. If TRUE, the equivalence step adds a one-way ANOVA with the unpretested control's posttests (Of) as a further group. Default FALSE.
posthoc	The post hoc tests of pairs of groups: "scheffe" (the default), Scheffé's (1953) tests, as in Steyn (2005); or "holm", pairwise t tests with Holm's (1979) adjustment. Both use a pooled SD. See the Operationalization section.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Value

An object of class `solomon_steyn`, a list with:

- equivalence, testing, reliability, regression: tables of tests for steps 1, 3, 5, and 6. testing adds comparison, term ("Pretest", "Intervention", "Interaction"), and sum_sq; regression adds var_pre, var_post, and direction.
- history: tests, and pattern, the classification of step 2.
- classic: summary (one row per intervention: Test A, the path, and the conclusion) and fits, the `fit_solomon_classic()` results.
- attrition: counts (randomized, observed, and missing posttests in each group), dropouts (the two-by-two table of dropouts), and tests.
- effects: tests (E1 to E5), groups (for several interventions, whether each intervention group differs from 0d and from 0f), and highest (the intervention with the highest combined posttest mean in E5, or NA; see the Operationalization section).
- path: the steps of step 8 that Steyn's sequence acts on.
- conclusions: one plain-language summary for each step.
- notes: skipped steps and data notes.
- n: counts of participants, posttests, and pretests in each group.
- conditions: the control and the interventions.
- settings: alpha, include_unpretested_control, posthoc, k, and pretest (whether pretest scores were supplied).

The tables of tests share the columns step, test, groups (Steyn's labels with the condition names), n, estimate (the first group's mean minus the second's for a t test; the mean change for a paired t test; r for a correlation; the variance ratio; or the difference in dropout proportions), statistic, reference ("t", "F", "chisq", or "z"), df1, df2, p.value, p.adjusted (the post hoc p-value of a pair of groups: Scheffé's, or Holm-adjusted, as posthoc sets; NA for the other tests), and interpretation (Steyn's reading, where he gives one). For a post hoc test, statistic and p.value are the pairwise t with a pooled SD and its unadjusted p-value; the decisions of the sequence read p.adjusted.

Steyn's sequence

Steyn labels the scores by group: 0a1, 0b1 are the pretest and posttest of the pretested group of intervention 1 (EG1); 0c, 0d the pretest and posttest of the pretested control (CG1); 0e1 the posttest of the unpretested group of intervention 1 (CG2.1); and 0f the posttest of the unpretested control (CG3). With one intervention the number is dropped (0a, 0b, 0e; EG, CG2). The steps below follow a pre-publication draft of the article, dated March 2, 2009 (see Source). In Steyn's order:

1. **Equivalence after randomization.** The pretests of the pretested groups are compared: a t test for one intervention, a one-way ANOVA otherwise. If they differ, Steyn advises post hoc tests to locate the difference, an explanation, and reconsidering whether to continue. With `include_unpretested_control = TRUE`, a further one-way ANOVA adds 0f, which Steyn suggests when the study is short or history and maturation are believed negligible.
2. **History and maturation.** A paired t test of 0c and 0d; a t test of all the pretests (the Time 1 value for the unpretested control) against 0f; and a one-way ANOVA of 0c, 0d, and 0f. If 0c differs from 0d and 0f, Steyn reads history or maturation; if 0d differs from 0c and 0f, the pretest (a testing effect). Steyn (2009) presents this reading of the three sets of scores (0c, 0d, and 0f) as new.
3. **Testing effect.** The pretest main effect in the two-way between-groups ANOVA of the four posttest groups of each intervention and the control.
4. **Pretest-intervention interaction.** The decision sequence of Walton Braver and Braver (1988), `fit_solomon_classic()`, for each intervention against the control.
5. **Test-retest reliability and instrumentation.** The correlation of 0c and 0d; the paired t test of 0c and 0d; and a t test of 0c and 0f.
6. **Regression to the mean.** A chi-square test for the variance of 0d against that of 0c. Steyn reads an increase in variance as a sign of regression to the mean, a threat when groups were selected for extreme scores.
7. **Attrition.** A z test of the dropout proportions of the intervention groups against the non-intervention groups (CG1 and CG3), and Steyn's chi-square on the two-by-two table of dropouts, intervention or not by pretested or not ($df = 1$). That chi-square asks whether intervention and pretesting are associated among the dropouts: whether the dropouts of the intervention groups were pretested more, or less, often than the dropouts of the non-intervention groups. It counts dropouts only, so it does not compare dropout rates, and it depends on how many participants each group started with. To compare dropout rates by group, see `attrition$counts` and `check_solomon_missing()`, which counts the missing posttests in each group.
8. **Effects of the interventions.** For one intervention: E1, a one-way ANOVA of the four posttest groups, and E2, a t test of 0b plus 0e against 0d plus 0f; Steyn (2009) credits these two comparisons to his doctoral thesis (Steyn, 2001, as cited in Steyn, 2009). For several:
 - E1, a one-way ANOVA of all $2(k + 1)$ posttest groups;
 - E2, post hoc tests of every pair of groups, summarized by whether each intervention group differs from 0d and from 0f;
 - E3, a one-way ANOVA of the $2k$ intervention groups;
 - E4, for each intervention, a t test of 0b against 0e;
 - E5, when no E4 test is significant, a one-way ANOVA of the k interventions with their two groups combined, with post hoc tests, which Steyn uses to find which intervention had the greatest effect. When an E4 test is significant, Steyn cautions that internal validity is in question and the groups are not combined.

Every test is computed when the data allow it; path records the steps Steyn's sequence acts on at `alpha`. Steps 1, 2, 4, 5, and 6 use the pretest and are skipped, with a note, when `y_pre` is `NULL`.

Operationalization

Steyn's description leaves these choices open. `solomonR` made them as follows.

- **t tests.** Independent-samples t tests are Student's pooled-variance tests, matching the equal-variance one-way ANOVAs.
- **Post hoc tests.** Steyn (2009) names no post hoc test. In the study the article describes, Steyn (2005, Table 5.61, p. 153) used Scheffé tests after the one-way ANOVA of the eight posttest groups, so they are the default (`posthoc = "scheffe"`). They are the method of Scheffé (1953). For a pair of groups among G groups with N scores in all, the Scheffé statistic is the squared pairwise t divided by $G - 1$, on $G - 1$ and $N - G$ degrees of freedom, with the pooled error variance of the groups in the ANOVA. The pair differs at α exactly when Scheffé's criterion holds: $|t|$ exceeds S , where S^2 is $G - 1$ times the upper α point of $F(G - 1, N - G)$ (Scheffé, 1953, pp. 87–89, 97). Scheffé's procedure protects every contrast among the groups, not only the pairs, so it is conservative when only pairs are compared; for pairs of equally precise means (equal group sizes), Scheffé recommended Tukey's method instead (pp. 89, 92–93, 96–97). Because the F test rejects exactly when some contrast is significant (pp. 87, 95–96), no pair can differ unless the ANOVA of the same groups is significant, and a significant ANOVA need not yield a pair that differs; step 2 then reports a pattern Steyn's rule does not cover. `posthoc = "holm"` gives pairwise t tests with the same pooled SD and Holm's (1979) adjustment, the default of `stats::pairwise.t.test()`, which covers the pairwise comparisons only. In step 1 the post hoc tests follow a significant ANOVA of more than two groups; in step 2, and in step 8 with several interventions (E2 and E5), they are always computed.
- **Two-way ANOVA.** Type III sums of squares, with effect coding, one analysis for each intervention against the control.
- **History classification.** Two sets "differ" when their post hoc p-value (Scheffé's, or Holm-adjusted) is below α . A nonsignificant ANOVA gives "no evidence of history, maturation, or a testing effect"; 0c differing from 0d and from 0f, which do not differ, gives "history or maturation"; 0d differing from 0c and from 0f, which do not differ, gives "the pretest (a testing effect)"; any other pattern is reported as one Steyn's rule does not cover.
- **Scores used.** The one-way ANOVA of 0c, 0d, and 0f and the t test of 0c and 0f use every available score in each set. The paired t test, the correlation, and the variance test use the pretested controls with both scores.
- **Variance test.** The statistic is $(n - 1)$ times the variance of 0d divided by the variance of 0c, on $n - 1$ degrees of freedom, with a two-sided p-value (twice the smaller tail).
- **Attrition.** A participant with a missing posttest is a dropout. "Intervention" means any intervention. The z test pools the two proportions; neither test has a continuity correction. The chi-square is computed on the dropout counts, as Steyn describes; a note is recorded when an expected count is below 5.
- **"Differs from both control groups" (E2).** An intervention group differs from both when its post hoc p-values against 0d and against 0f are both below α . The sequence continues to E3 when at least one intervention group does.

- **"Greatest effect"** (E5). Steyn does not say how the intervention with the greatest effect is identified. `solomonR` reports the intervention with the highest combined posttest mean (highest) and the post hoc tests of every pair. The highest mean is the greatest effect only when the interventions raise the scores; when they lower them, as in Steyn (2005), the greatest effect is the lowest mean. Read `highest` with the direction of the outcome and the post hoc tests. In the English report of that study, Steyn and Mynhardt (2008, pp. 569–570) ranked the treatments by the difference between treated and untreated means and by d , from separate 2×2 analyses, without a test of the differences between treatments.
- **Missing scores.** Participants with a missing posttest stay in the data for the attrition step and are left out of the posttest analyses. The pretest analyses use the pretested participants with a pretest.
- **Tests that cannot be computed.** A test of groups that are empty, too small, or without variation is left out, with a note in notes.
- **Step 4** uses `fit_solomon_classic(flow = "1988")` with its defaults (ANCOVA as the pretested-groups test, and Test I).

Cautions

- Steyn notes that multiple ANOVAs capitalize on chance. The sequence runs many tests, each at α , with no control of the error rate across them.
- A nonsignificant test is not evidence that groups are equivalent, or that a threat is absent.
- The one-way ANOVA of O_c , O_d , and O_f , and its post hoc tests, treat the paired scores of O_c and O_d , which come from the same participants, as independent. Scheffé's (1953, p. 87) tests take the covariances of the group means as known; here that of O_c and O_d is taken to be zero.
- The chi-square test for the variance treats the variance of O_c as a known value and ignores the pairing of O_c and O_d .
- A difference between O_b and O_e (E4) mixes a pretest main effect with pretest sensitization. Step 4 tests the interaction for each intervention separately; the joint model (`fit_solomon_glm()`) tests it for all the interventions in one model, with adjusted p-values for the comparisons.
- The package's recommended analysis is `fit_solomon_glm(control = )`.

Source

This function follows a pre-publication draft of Steyn's (2009) article, dated March 2, 2009, which the author provided (R. Steyn, personal communication, September 30, 2026). The published article was not available for comparison. The draft has no page numbers, so none are cited.

Acknowledgment

We thank Renier Steyn for providing the draft of his article, and for noting that his model is intended for situations with a large amount of data and ample time (R. Steyn, personal communication, September 30, 2026).

References

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2), 65–70. <https://www.jstor.org/stable/4615733>

Scheffé, H. (1953). A method for judging all contrasts in the analysis of variance. *Biometrika*, 40(1–2), 87–104. <https://doi.org/10.1093/biomet/40.1-2.87>

Steyn, R. (2005). *Self-evaluasie en die vorming van selfdoeltreffendheidspersepsies* [Self-evaluation and the forming of self-efficacy perceptions] [Doctoral thesis, University of South Africa]. Unisa Institutional Repository. <https://hdl.handle.net/10500/1745>

Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

Steyn, R., & Mynhardt, J. (2008). Factors that influence the forming of self-evaluation and self-efficacy perceptions. *South African Journal of Psychology*, 38(3), 563–573. <https://doi.org/10.1177/008124630803800310>

Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

See Also

[fit_solomon_glm\(\)](#) for the recommended analysis, [fit_solomon_classic\(\)](#), [check_solomon_missing\(\)](#), [report_solomon\(\)](#) for APA 7 text of the results, and [steyn2005](#) for the summary statistics of Steyn's (2005) eight-group study.

Examples

```
# Six groups: two interventions and a control.
fit <- fit_solomon_steyn(post_behavior, condition, pretested, pre_behavior,
                        control = "Control", data = mai2020)
fit

# Pairwise t tests with Holm's adjustment as the post hoc tests.
fit_solomon_steyn(post_behavior, condition, pretested, pre_behavior,
                  control = "Control", posthoc = "holm", data = mai2020)

# The four-group design: one intervention.
fit_solomon_steyn(y_post, treat, pretested, y_pre, data = solomon_example)
```

invariance_solomon	<i>Measurement invariance across the four Solomon groups</i>
--------------------	--

Description

[Experimental] Tests whether a set of indicators measures the same construct in the same way in the four Solomon groups, the condition latent mean contrasts require (Meredith, 1993). It fits the configural, metric (equal loadings), and scalar (equal loadings and intercepts) models in the sequence Vandenberg and Lance (2000, pp. 56–57) recommend and compares each with the one before it.

Usage

```
invariance_solomon(
  data,
  items,
  treat,
  pretested,
  estimator = "MLR",
  partial = NULL,
  alpha = 0.05
)
```

Arguments

<code>data</code>	A data frame with the indicators.
<code>items</code>	Names of at least three indicators.
<code>treat, pretested</code>	Treatment and pretest indicators (0/1). Designs with several treatments are not supported; see fit_solomon_glm() .
<code>estimator</code>	lavaan estimator, default "MLR".
<code>partial</code>	Optional character vector of freed parameters (see Partial invariance).
<code>alpha</code>	Significance level of the chi-square difference test. Default 0.05.

Value

An object of class `solomon_invariance`: the three fits, the fit indexes in models, the step tests with the decision under each criterion, the most constrained level supported under each ("configural", "metric", "scalar", or "partial scalar"), and the cutoffs applied.

Two published criteria

Each step compares the more constrained model with the one before it in two ways, and the output gives the decision under each.

- **Chi-square difference test** (`noninvariant_chisq`): noninvariance when the difference test rejects at α . For the robust estimators MLR, MLM, and MLMV the difference is scaled (Satorra & Bentler, 2001). Vandenberg and Lance (2000, p. 46) recommend this test as the primary criterion, with the change in CFI as a supplement.
- **Change in fit** (`noninvariant_chen`): noninvariance when CFI drops by at least .005 and, in addition, RMSEA rises by at least .010 or SRMR by at least .025 (loadings) or .005 (intercepts), the cutoffs Chen (2007, pp. 501–502) gives for a total N of 300 or less with unequal group sizes. For a total N above 300 with equal group sizes they are .010, .015, and .030 or .010. Chen does not cover the two mixed cases, for which `solomonR` uses the small-sample values. Reading "a change in CFI, supplemented by a change in RMSEA or SRMR" as requiring both is `solomonR`'s; Chen chose CFI as the main criterion (p. 502). Cheung and Rensvold (2002, pp. 234–235) favor such changes over the chi-square difference, which depends on sample size.

The criteria can disagree, and both rest on simulations with two groups and maximum likelihood estimation of multivariate normal data (Cheung & Rensvold, 2002, p. 251; Chen, 2007). Chen (2007, p. 502) notes that RMSEA and SRMR tend to over-reject invariant models when samples are small, as Solomon groups often are. A simulation study under a protocol posted on issue #55 found that neither criterion, nor the two together, kept false rejections of invariance at or below .060 in Solomon-sized groups (see the article "Latent Contrasts: Validating the Invariance Check"). This function therefore reports both and decides nothing for the user, and `fit_solomon_sem_latent()` runs it and warns, rather than refuses, when a criterion flags noninvariance. The fit indexes are those of the maximum likelihood fit, whose estimates MLR shares.

Partial invariance

When scalar invariance fails, latent means can still be compared if the noninvariant parameters are freed and enough indicators stay invariant (Byrne et al., 1989, p. 458). `partial` names the freed parameters in lavaan syntax (for example, "item3 ~ 1" for an intercept). They must be chosen on substantive grounds, not by searching the data (Byrne et al., 1989, p. 465), and must involve only a minority of the indicators (Vandenberg & Lance, 2000, p. 38); at least two indicators must stay fully invariant. The function never chooses them.

Lifecycle

Experimental. The issue #55 study found no criterion that holds its false-rejection rate in Solomon-sized groups (see Two published criteria), so the criteria reported may change as better small-sample criteria are published.

References

- Byrne, B. M., Shavelson, R. J., & Muthén, B. (1989). Testing for the equivalence of factor covariance and mean structures: The issue of partial measurement invariance. *Psychological Bulletin*, 105(3), 456–466. <https://doi.org/10.1037/0033-2909.105.3.456>
- Chen, F. F. (2007). Sensitivity of goodness of fit indexes to lack of measurement invariance. *Structural Equation Modeling: A Multidisciplinary Journal*, 14(3), 464–504. <https://doi.org/10.1080/10705510701301834>
- Cheung, G. W., & Rensvold, R. B. (2002). Evaluating goodness-of-fit indexes for testing measurement invariance. *Structural Equation Modeling: A Multidisciplinary Journal*, 9(2), 233–255. https://doi.org/10.1207/S15328007SEM0902_5
- Meredith, W. (1993). Measurement invariance, factor analysis and factorial invariance. *Psychometrika*, 58(4), 525–543. <https://doi.org/10.1007/BF02294825>
- Satorra, A., & Bentler, P. M. (2001). A scaled difference chi-square test statistic for moment structure analysis. *Psychometrika*, 66(4), 507–514. <https://doi.org/10.1007/BF02296192>
- Vandenberg, R. J., & Lance, C. E. (2000). A review and synthesis of the measurement invariance literature: Suggestions, practices, and recommendations for organizational research. *Organizational Research Methods*, 3(1), 4–70. <https://doi.org/10.1177/109442810031002>

See Also

`fit_solomon_sem_latent()`

Examples

```
if (requireNamespace("lavaan", quietly = TRUE)) {
  set.seed(55)
  g <- rep(1:4, each = 80)
  treat <- c(1, 0, 1, 0)[g]
  pretested <- c(1, 1, 0, 0)[g]
  f <- stats::rnorm(320, 0.4 * treat)
  items <- data.frame(y1 = f + stats::rnorm(320, 0, 0.6),
                     y2 = 0.9 * f + stats::rnorm(320, 0, 0.6),
                     y3 = 0.8 * f + stats::rnorm(320, 0, 0.6),
                     y4 = 0.7 * f + stats::rnorm(320, 0, 0.6))
  invariance_solomon(items, c("y1", "y2", "y3", "y4"), treat, pretested)
}
```

jordan2014

Group statistics from Jordaan's (2014) study with three posttest occasions

Description

The sample sizes, means, and standard deviations of a Solomon four-group design measured on three posttest occasions: an evaluation of a life skills program for young adult male offenders (Jordaan, 2014, Table 7.3, p. 112). The outcomes are the three subscales of the Coping Strategy Indicator. Numbers reported in the publication are reused with citation.

Usage

jordan2014

Format

A data frame with 42 rows, one per subscale, group, and occasion, and 8 variables:

subscale Social support, Problem solving, or Avoidance.

group The group's number in the thesis: 1 = program with pretest, 2 = program without pretest, 3 = control with pretest, 4 = control without pretest.

treat 1 for the program groups, 0 for the control groups.

pretested 1 if the group was pretested, 0 if not.

occasion Pretest (groups 1 and 3 only), Posttest (after the program), Follow-up 1 (3 months later), or Follow-up 2 (6 months later).

n Sample size.

mean, sd Mean and standard deviation.

Details

Design. 120 offenders aged 21 to 25 with long sentences were selected by systematic random sampling in a maximum-security correctional center and assigned at random to the program or the control condition. Half of each condition was assigned at random to be pretested (pp. 86–87). The program ran for six months; the control group followed the center's normal daily activities (p. 87). All groups were tested after the program and again 3 and 6 months later (p. 98).

Attrition. Transfers removed 17 offenders from the program groups and 7 from the control groups, leaving 96 (p. 109). Those 96 completed every posttest: every analysis in the thesis has 92 error degrees of freedom. Because more were lost from the program groups, the groups analyzed are not guaranteed to be comparable, even though they were randomized.

Measures. Each subscale of the Coping Strategy Indicator has 11 items scored from 1 (not at all) to 3 (a lot), so scores range from 11 to 33. High scores on problem solving and on seeking social support, and low scores on avoidance, indicate better coping (p. 92).

Known results. These statistics reproduce, within rounding, the 2 x 2 analyses of variance of the posttests on each occasion (Tables 7.4–7.19, pp. 113–127). The Pretest x Treatment interaction F values are:

- **Social support:** 9.678 ($p = .002$) after the program, 0.266 at 3 months, and 2.306 at 6 months.
- **Problem solving:** 0.819, 0.563, and 5.556 ($p = .021$).
- **Avoidance:** 0.373, 0.688, and 0.327.

The thesis followed the decision sequence of Walton Braver and Braver (1988) on each occasion separately (`fit_solomon_classic()`). The correlations between occasions are not reported, so analyses of change across occasions cannot be reproduced; see the article "Worked Example: Repeated Posttests".

Source

Jordaan, J. (2014). *The development and evaluation of a life skills programme for young adult prisoners* [Doctoral thesis, University of the Free State]. KopsieScholar. <https://hdl.handle.net/11660/832>

References

Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

See Also

[solomon_from_summary\(\)](#), [fit_solomon_mmrn\(\)](#), [steyn2005](#)

Examples

```
# Problem solving, 6 months after the program (Table 7.13, p. 121).
ps6 <- subset(jordaan2014, subscale == "Problem solving" & occasion == "Follow-up 2")
with(ps6, solomon_from_summary(n, mean, sd, treat = treat, pretested = pretested))
```

kvalem1996

*Condom use in the Solomon study of Kvalem et al. (1996)***Description**

The number of students who used condoms at their most recent intercourse, at the 6-month posttest, among students who had had intercourse before the intervention, in each Solomon group of a school-based sex-education trial (Kvalem et al., 1996, p. 42). Numbers reported in the publication are reused with citation.

Usage

kvalem1996

Format

A data frame with 4 rows, one per Solomon group, and 5 variables:

group The Solomon group.

pretested, treat Indicators (1 = yes).

events Students who used condoms at their most recent intercourse.

n Students in the group with intercourse before the intervention who answered the 6-month posttest.

Details

Design and caveats. Whole classes were randomized: 30 classes to the intervention, 15 of them pretested, and 94 to control, 47 of them pretested (p. 38). The authors analyzed individuals rather than classes because class sizes varied too much for a class-level analysis (p. 39). These counts are therefore individual-level, and analyses of them ignore the clustering; see [perm_solomon\(\)](#) for cluster-level inference when class membership is available.

Known result. Fisher's exact and chi-square comparisons by [fisher_solomon\(\)](#) reproduce the reported pattern: 70% (51/73) against 51% (76/148) among pretested students, chi-square = 6.85, and 43% (21/49) against 52% (69/133) among unpretested students, chi-square = 1.17 (p. 42).

Source

Kvalem, I. L., Sundet, J. M., Rivø, K. I., Eilertsen, D. E., & Bakketeig, L. S. (1996). The effect of sex education on adolescents' use of condoms: Applying the Solomon four-group design. *Health Education Quarterly*, 23(1), 34–47. <https://doi.org/10.1177/109019819602300103>

See Also

[fisher_solomon\(\)](#), [marginal_solomon\(\)](#), [elkarkri2025a](#)

Examples

```
# One row per student.
students <- kva1em1996[rep(seq_len(4), kva1em1996$n), c("pretested", "treat")]
students$used <- unlist(Map(function(e, n) rep(1:0, c(e, n - e)),
                           kva1em1996$events, kva1em1996$n))
with(students, fisher_solomon(used, treat, pretested))
```

lana1959

Posttest statistics from Lana's (1959) attitude experiment

Description

The sample size, posttest mean, and standard deviation of the four Solomon groups in one of the first experiments built on the Solomon design: a recorded pro-vivisection talk and a 10-item vivisection questionnaire given to introductory psychology classes (Lana, 1959, Table 2, p. 297). Numbers reported in the publication are reused with citation.

Usage

lana1959

Format

A data frame with 4 rows, one per Solomon group, and 7 variables:

group The Solomon group.

lana_group Lana's group label (I to IV).

pretested, treat Indicators (1 = yes); treat is the recorded talk.

n, mean, sd Posttest sample size, mean, and standard deviation.

Details

Design. Five classes (156 students) were randomly assigned to five conditions (p. 295). Four form a Solomon design:

- **Group I.** Pretest, then the talk 12 days later, then the posttest.
- **Group IV.** Pretest, then the posttest 12 days later.
- **Group II.** The talk, then the posttest.
- **Group III.** The posttest only.

A fifth group, with a delayed posttest, is not part of the four-group design and is not included. Because whole classes were assigned, class and condition are confounded, as in [elkarkri2025a](#).

Known result. Lana analyzed the posttest means as a two-by-two analysis of variance (Table 3, p. 297). The treatment effect was significant, $F = 5.35$, and the pretest and the interaction were not. [solomon_from_summary\(\)](#) reproduces it within rounding (treatment $F = 5.36$ on 1 and 152 df). Lana's sums of squares are on the scale of the cell means (an unweighted-means analysis).

Dividing the package's sums of squares by the harmonic mean of the cell sizes recovers them, and the F tests are the same.

Historical note. Lana concluded that the pretest did not sensitize participants to the talk (p. 298), the opposite of Solomon's (1949) spelling result. Together they begin the record that Willson and Putnam (1982) later pooled.

Source

Lana, R. E. (1959). Pretest-treatment interaction effects in attitudinal studies. *Psychological Bulletin*, 56(4), 293–300. <https://doi.org/10.1037/h0044646>

See Also

[solomon_from_summary\(\)](#), [solomon1949](#), [elkarkri2025a](#)

Examples

```
lana1959
with(lana1959, solomon_from_summary(n, mean, sd))
```

mai2020

Transfer-intervention data from Mai et al. (2020)

Description

Individual-level data from a Solomon six-group study of interventions to support the transfer of time-management training: pretesting crossed with relapse prevention, proximal plus distal goal setting, and a control condition (Mai et al., 2020). The data were published as the article's supplementary material and are redistributed under its Creative Commons Attribution 4.0 license (see License).

Usage

```
mai2020
```

Format

A data frame with 211 rows, one per participant, and 6 variables:

id Row number in the published file.

gender Female or Male.

pretested 1 if assigned to be pretested, 0 if not.

condition RP (relapse prevention), GS (proximal plus distal goal setting), or Control.

pre_behavior Self-reported time-management behavior before the workshop (pretested groups only).

post_behavior Self-reported time-management behavior after the workshop (see Details for the timing); missing for participants who did not answer.

Details

Design. Final-year students at a management college took a three-hour time-management workshop.

- **Pretesting.** Before the workshop, they were "randomly assigned into either the pretested or unpretested groups based on their roll numbers" (p. 5). The article does not say how roll numbers were used.
- **Interventions.** After it, they were randomly assigned to relapse prevention (RP), goal setting (GS), or control (p. 6).
- **Measures.** Self-reported time-management behavior was measured before the workshop in the pretested groups and 10 weeks after it in all groups (p. 7; the note to Table 3 on p. 8 says eight weeks).

Every pair of conditions forms a Solomon four-group design. The authors analyzed three such pairs, reusing groups across them. `fit_solomon_glm()` with `control = "Control"` fits one model to all six groups (see the examples and the article "Designs With Several Treatments").

Sample and attrition. The file has 211 participants, and the article reports 210 (p. 5). 133 answered the posttest, with attrition that differed by group (see the example). The unpretested groups have no pretest by design.

Scale. `pre_behavior` and `post_behavior` are the authors' scale scores for the 30 self-report items. The article describes the response scale as running from 1 (always) to 5 (never) (p. 7). The value labels in the data file run from 1 (never) to 5 (always). The scores are not the simple means of the 30 items in the file, so the items are not included.

Known results. The two-by-two ANOVAs of the posttest scores reproduce the authors' Table 4 (p. 8). For relapse prevention against control, the Pretest x Treatment sum of squares is 0.508, the error sum of squares 13.347, and $F = 3.461$. The history checks of their Table 5 (p. 8) and the ANCOVAs of the first two comparisons in their Table 7 (p. 9) are reproduced as well. The third ANCOVA, relapse prevention against goal setting, gives the published difference of -0.219 with $p = .044$ against the published .046. The published cell statistics of their Table 6 do not match the data (see issue #53). The article "Worked Example: A Published Solomon Study" carries the relapse-prevention comparison through the package's analyses.

Acknowledgment

We thank Nu Nu Mai, Yoshi Takahashi, and Mon Mon Oo for making their data publicly available with their article. For the study's design, measures, and findings, read the published article (Mai et al., 2020, cited under Source).

License

The data are © 2020 by the authors (Nu Nu Mai, Yoshi Takahashi, and Mon Mon Oo) and are published with the article under the Creative Commons Attribution 4.0 International license (<https://creativecommons.org/licenses/by/4.0/>). MDPI's open access policy (<https://www.mdpi.com/openaccess>) states that its articles, "including their data, graphics, and supplementary material", may be reused with attribution. Changes made in `solomonR`: 6 of the 117 variables in the published file are kept, renamed, and recoded to factors or 0/1 indicators, and an `id` column is added. The original file and the script that builds this data set are in the package's `data-raw` folder on GitHub. See also `inst/COPYRIGHTS`.

Source

Mai, N. N., Takahashi, Y., & Oo, M. M. (2020). Testing the effectiveness of transfer interventions using Solomon four-group designs. *Education Sciences*, 10(4), Article 92. <https://doi.org/10.3390/educsci10040092> (supplementary material, Table S1: dataset.sav)

See Also

[baseline_solomon\(\)](#), [fit_solomon_glm\(\)](#), [elkarkri2025a](#), [kvalem1996](#)

Examples

```
# Posttest completion by group.
with(mai2020, table(pretested, condition, posttest = !is.na(post_behavior)))

# Relapse prevention against control, as a Solomon four-group design.
rp <- subset(mai2020, condition %in% c("RP", "Control"))
with(rp, fit_solomon_glm(post_behavior, treat = as.integer(condition == "RP"),
  pretested = pretested, pretest_score = pre_behavior))

# One model for all six groups.
fit_solomon_glm(post_behavior, condition, pretested, pre_behavior,
  control = "Control", data = mai2020)
```

marginal_solomon

Marginal Solomon contrasts for binary and count outcomes

Description

[Stable] Estimates the Solomon contrasts for a binary outcome as risk differences, risk ratios, or odds ratios, and for a count outcome as rate differences or rate ratios, comparing marginal risks or rates in every cell.

Usage

```
marginal_solomon(
  fit,
  scale = c("difference", "ratio", "odds_ratio"),
  method = c("bootstrap", "delta", "cluster_summary"),
  R = 999,
  seed = NULL,
  conf_level = fit$conf_level
)
```

Arguments

<code>fit</code>	A fit from <code>fit_solomon_glm()</code> with <code>family = binomial()</code> (logit link), <code>family = poisson()</code> (log link), or <code>family = "negative_binomial"</code> , with HC3 or model-based covariance, or, for binary outcomes, CR2 covariance. Designs with several treatments are not supported; see <code>fit_solomon_glm()</code> .
<code>scale</code>	One or more of "difference", "ratio", and, for binary outcomes, "odds_ratio". For count outcomes the default is <code>c("difference", "ratio")</code> .
<code>method</code>	"bootstrap" (default for binary outcomes), "delta" (the only method for count outcomes, and the default for clustered fits), or "cluster_summary" (clustered binary fits, risk differences only; see Clustered fits).
<code>R</code>	Number of bootstrap resamples (at least 99). Default 999.
<code>seed</code>	Optional integer seed for the bootstrap. The global random number state is restored afterwards.
<code>conf_level</code>	Confidence level. Defaults to the fit's.

Details

A logistic model that adjusts for the pretest estimates, among pretested participants, a treatment effect conditional on the pretest, but among unpretested participants a marginal effect, because they have no pretest. Odds ratios are noncollapsible: when the pretest predicts the outcome, a conditional and a marginal odds ratio differ even without confounding (Daniel et al., 2021). The logit-scale Pretest x Treatment contrast of `fit_solomon_glm()` can therefore be nonzero when there is no sensitization. This function compares like with like.

Marginal risks are estimated by standardization (Daniel et al., 2021; Localio et al., 2007). In the pretested cells, each pretested participant's risk is predicted under treatment and under control from the fitted logistic model and averaged over all pretested participants. In the unpretested cells the same is done over the unpretested participants; without covariates, these are the observed cell proportions. The contrasts are then:

- Treatment | pretested and Treatment | unpretested: the treatment effect within each pretest condition;
- Pretest x Treatment: their difference on the chosen scale (a difference of risk differences, or a ratio of risk ratios or of odds ratios);
- ATE (avg over pretest): the effect in a population with equal numbers of pretested and unpretested participants, computed from the averaged risks.

Sensitization depends on the scale: an effect can be modified on the risk-difference scale and not on the ratio scale, or the reverse. Report the scale with every result.

Intervals use a nonparametric bootstrap by default, resampling participants within each of the four Solomon cells and reporting percentile intervals; Daniel et al. (2021) and Localio et al. (2007) found the bootstrap performed better than the delta method, which is available as `method = "delta"` (using the fit's covariance matrix, with covariates held fixed). p-values use the bootstrap or delta-method standard error on the analysis scale: risk differences, log risk ratios, or log odds ratios. Bootstrap resamples in which the logistic fit fails (no convergence, a Solomon cell with only events or only non-events, or a fitted probability within $1e-8$ of 0 or 1) are excluded and counted; when more than 10% fail, intervals are not reported.

Count outcomes: for a fit with `family = poisson()` or `family = "negative_binomial"`, rates per unit of exposure are standardized in the same way and compared as rate differences or rate ratios. Log-link rate ratios are collapsible (Daniel et al., 2021), so they agree with the fitted model's coefficients when there are no other covariates; rate differences depend on the covariate distribution. For counts, intervals use the delta method with the fit's (by default robust HC3) covariance, which Cameron and Trivedi (2013) recommend under overdispersion; a bootstrap for counts has not been evaluated and is not offered. The fit's Pearson dispersion statistic, and for negative-binomial fits the estimated theta, are printed for description. No published Solomon study with a count outcome has been identified, so this use of count-data methods is a `solomonR` extension.

Clustered fits: for binary outcomes fitted with `robust = "CR2"`, the delta-method standard errors use the CR2 cluster-robust covariance (Bell & McCaffrey, 2002), and intervals and tests use the `t` distribution with Satterthwaite degrees of freedom for the delta method's linear approximation (Pustejovsky & Tipton, 2018). Applying those degrees of freedom to the linearized contrast is a `solomonR` extension. Only the delta method is offered, because the bootstrap resamples participants rather than clusters, and a classed warning flags degrees of freedom below 4 (Tipton, 2015).

In the package's simulation of cluster-randomized Solomon designs (issue #64; 4 to 47 clusters per cell or arm, intracluster correlations of 0.02 and 0.10), these intervals met the coverage and Type I error tolerances for risk differences in 140 of 144 scenario-contrasts and for odds ratios in 137 of 144. The four risk-difference shortfalls, with coverage from 0.935 to 0.961, all had an intracluster correlation of 0.10: 4 or 10 clusters per arm with pretesting assigned within clusters, 15 and 47 clusters per arm with pretesting within clusters, and 15, 15, 47, and 47 clusters per cell. Risk-ratio intervals met the tolerances in 111 of 144 and were conservative with 4 clusters per cell or arm (coverage up to 0.973). A normal reference distribution undercovered (coverage down to 0.880), so it is not used for clustered fits.

In the three shortfalls with pretesting assigned within clusters, the unweighted comparison of cluster-level summaries (Hayes & Moulton, 2017, pp. 211–215) met the tolerances, so a cluster-level analysis is the recommended alternative for risk differences in such designs when clustering is strong. `method = "cluster_summary"` computes it, as the study did: the unweighted mean of the cluster proportions in each arm, compared with a `t` interval that uses separate variances and Satterthwaite degrees of freedom. With pretesting assigned within clusters, each cluster contributes its pretested and unpretested proportions, and treated and control clusters are compared. It warns when an arm has fewer than four clusters, the minimum Hayes and Moulton (2017, p. 128) recommend. The differences were small, and the cluster-level comparison met the tolerances less often across all scenarios (119 of 144), mostly by covering more than 96% of the time with 4 clusters per cell or arm. `perm_solomon()` gives a cluster-level randomization test. Clustered count fits have not been validated and are refused; for them, the log rate-ratio contrasts of `fit_solomon_glm()` with CR2 covariance and `perm_solomon()` account for clustering.

The package's simulation validation of this function is described on issue #43 (binary outcomes), issue #44 (count outcomes), issue #62 (negative-binomial fits), and issue #64 (clustered fits).

Value

An object of class `solomon_marginal` with effects (one row per scale and contrast: estimate and interval on the reporting scale, standard error on the analysis scale, p-value), risks (binary) or rates (counts, per unit of exposure) for the four cells, and the settings, including the number of failed bootstrap resamples.

References

- Bell, R. M., & McCaffrey, D. F. (2002). Bias reduction in standard errors for linear regression with multi-stage samples. *Survey Methodology*, 28(2), 169–181.
- Cameron, A. C., & Trivedi, P. K. (2013). *Regression analysis of count data* (2nd ed.). Cambridge University Press. <https://doi.org/10.1017/CBO9781139013567>
- Daniel, R., Zhang, J., & Farewell, D. (2021). Making apples from oranges: Comparing noncollapsible effect estimators and their standard errors after adjustment for different covariate sets. *Biometrical Journal*, 63(3), 528–557. <https://doi.org/10.1002/bimj.201900297>
- Hayes, R. J., & Moulton, L. H. (2017). *Cluster randomised trials* (2nd ed.). Chapman and Hall/CRC. <https://doi.org/10.4324/9781315370286>
- Localio, A. R., Margolis, D. J., & Berlin, J. A. (2007). Relative risks and confidence intervals were easily computed indirectly from multivariable logistic regression. *Journal of Clinical Epidemiology*, 60(9), 874–882. <https://doi.org/10.1016/j.jclinepi.2006.12.001>
- Pustejovsky, J. E., & Tipton, E. (2018). Small-sample methods for cluster-robust variance estimation and hypothesis testing in fixed effects models. *Journal of Business & Economic Statistics*, 36(4), 672–683. <https://doi.org/10.1080/07350015.2016.1247004>
- Tipton, E. (2015). Small sample adjustments for robust variance estimation with meta-regression. *Psychological Methods*, 20(3), 375–393. <https://doi.org/10.1037/met0000011>

See Also

`fit_solomon_glm()`, `fisher_solomon()`

Examples

```
d <- solomon_example
d$passed <- as.integer(d$y_post > 55)
fit <- with(d, fit_solomon_glm(passed, treat, pretested, y_pre,
                             family = binomial()))
marginal_solomon(fit, method = "delta")

marginal_solomon(fit, R = 499, seed = 1)

# A count outcome observed over different exposure times.
set.seed(2)
d$days <- runif(nrow(d), 5, 15)
d$visits <- rpois(nrow(d), d$days * exp(-1.5 + 0.3 * d$treat))
counts <- with(d, fit_solomon_glm(visits, treat, pretested, y_pre,
                                family = poisson(), exposure = days))
marginal_solomon(counts)
```

p_to_z

*Convert p-value to Z (one-tailed) for Stouffer's method***Description****[Stable]****Usage**

p_to_z(p)

Arguments

p numeric vector of p-values assumed one-tailed and aligned in the same direction

Value

numeric Z-scores

References

Stouffer, S. A., Suchman, E. A., DeVinney, L. C., Star, S. A., & Williams, R. M., Jr. (1949). *The American soldier: Adjustment during army life* (Vol. 1). Princeton University Press.

Examples

p_to_z(c(0.05, 0.025, 0.5))

perm_solomon

*Permutation test for a Solomon contrast***Description**

[Stable] Performs a randomization-based test by permuting treatment assignment within pretest strata. This preserves the Solomon four-group design while generating the null distribution for a selected treatment contrast. Participants are permuted in unclustered designs and whole clusters in clustered ones.

Usage

```
perm_solomon(
  fit,
  contrast = "ATE (avg over pretest)",
  reps = 5000L,
  seed = NULL,
  return_dist = FALSE,
  statistic = c("studentized", "difference"),
  object = deprecated()
)
```

Arguments

fit	An object returned by <code>fit_solomon_glm()</code> . Designs with several treatments are not supported; see <code>fit_solomon_glm()</code> .
contrast	Character string identifying the contrast to test. One of "ATE (avg over pretest)", "Pretest x Treatment", "Treatment pretested", or "Treatment unpretested".
reps	Number of permutations. Default is 5000. In clustered designs with at most reps possible allocations, every allocation is used.
seed	Optional random-number seed for reproducibility. The global random-number state is restored when the function exits.
return_dist	Logical. If TRUE, return the permutation distribution in addition to the observed statistic and p-value.
statistic	"studentized" (the default) divides the contrast by its standard error; "difference" uses the contrast itself.
object	[Deprecated] Use fit.

Details

Unclassified designs. Treatment labels of participants are permuted within pretest strata and the model is refitted for each permutation. The default statistic is the HC3-studentized contrast. The permutation p-value is a valid test of the sharp null hypothesis that treatment has no effect for any participant; the +1 correction keeps the Monte Carlo p-value from being zero (Phipson & Smyth, 2010). That result is for the count of permuted statistics at least as extreme as the observed one, so a permutation whose statistic equals the observed one is counted. Such ties occur when a permutation repeats the observed assignment or swaps arms of equal size, and often when scores are tied, as with binary outcomes, counts, and ratings. So that rounding error does not decide, statistics that differ by less than a small relative tolerance are treated as equal: 1e-10 for a linear model and 1e-6 for models fitted by iteration. The tolerance is relative to the observed statistic or, when that is smaller, to 1 for the studentized statistic and to the largest permuted difference for the difference statistic. In a model fitted by iteration, equal fits usually agree to about 1e-7; with a covariate they can differ by more, most with a link other than the canonical one, and such a tie can still be missed (issue #134). Studentizing the statistic makes permutation tests asymptotically robust when only an average effect is hypothesized to be zero (DiCiccio & Romano, 2017; Wu & Ding, 2021); for the Pretest x Treatment contrast that robustness should be regarded as approximate.

Clustered designs. Randomization inference must permute the unit that was randomized, so for fits with a cluster variable the treatment labels of whole clusters are permuted. Two assignment mechanisms are supported:

- whole clusters assigned to the four Solomon conditions, as in Kvalem et al. (1996), where treatment labels are permuted among clusters within each pretest condition; and
- treatment assigned to clusters and pretesting to participants within clusters, where treatment labels are permuted among all clusters and every cluster must contain pretested and unpretested participants.

Designs that assign treatment to participants within clusters are refused, and stratified or restricted randomization of clusters is not supported.

The statistic is built from cluster-level summaries (Gail et al., 1996; Hayes & Moulton, 2017, ch. 10). A Stage 1 model with every term of the fit except treatment (the pretest indicator, the pretest score, covariates and any exposure offset) gives each cluster a covariate-adjusted difference residual: observed minus expected, divided by the number of participants or, for counts, by the total exposure (Hayes & Moulton, 2017, pp. 221–224, following Bennett et al., 2002). Because Stage 1 ignores treatment, the test remains exact under the sharp null hypothesis. When pretesting is assigned within clusters, each cluster has a pretested and an unpretested residual, and each contrast compares treated and control clusters on one combination of the two.

The contrast is estimated from unweighted means of the cluster residuals, so each cluster counts once (Hayes & Moulton, 2017, pp. 202–205). It therefore estimates the average effect across clusters, which can differ from the participant-weighted contrast of `fit_solomon_glm()` when cluster sizes vary and effects depend on cluster size. The studentized statistic divides the contrast by its separate-variances standard error (Hayes & Moulton, 2017, p. 212), which is the studentization Wu and Ding (2021) use for weak null hypotheses, applied here with clusters as the units. Gail et al. (1996, p. 1079) showed that the unstudentized difference can exceed the nominal level under the weak null hypothesis when the arms have unequal numbers of clusters and unequal variances.

When the number of possible allocations is at most reps, all of them are enumerated and the p-value is exact; the smallest attainable p-value is then reported. Hayes and Moulton (2017, p. 239) note that at least four clusters per arm are needed for a two-sided p below .05.

Simulation evidence. In the package’s pre-registered simulation study (issue #19; 96 scenarios, 2,000 replications each):

- Under the sharp null hypothesis, both statistics had Type I errors of at most 0.063, consistent with the exactness of randomization tests. With four clusters per arm, few allocations exist and the test is conservative: the attainable level at .05 is 2/70, about 0.03.
- When treatment made treated clusters four times as variable as control clusters, so that only the average effect was zero, the studentized statistic’s Type I error reached 0.065 with equal numbers of treated and control clusters, 0.0685 with 15 treated and 47 control clusters per pretest condition, and 0.0805 with 4 and 8. The difference statistic reached 0.158 with 15 and 47, as Gail et al. (1996) found for unbalanced designs. A classed warning (`solomonR_unbalanced_clusters_warning`) is therefore given whenever treated and control clusters differ in number.
- The CR2 tests of `fit_solomon_glm()` reached 0.068 with four clusters per arm even under the sharp null, where this test is exact, so the permutation test is preferred for designs with few clusters.

Value

A list of class `solomon_perm` containing the contrast, the statistic type, the level permuted ("participant" or "cluster"), the estimated contrast (`estimate`), the observed statistic (`z_obs`; the contrast itself when `statistic = "difference"`), the permutation p-value (`p_perm`), the number of permutations, and whether the p-value is exact. Clustered fits also return the design, the number of possible allocations, the smallest attainable p-value when exact, and the numbers of treated and control clusters. If `return_dist = TRUE`, the permutation distribution (`z_perm`) is also returned.

References

- Bennett, S., Parpia, T., Hayes, R., & Cousens, S. (2002). Methods for the analysis of incidence rates in cluster randomized trials. *International Journal of Epidemiology*, 31(4), 839–846. <https://doi.org/10.1093/ije/31.4.839>
- DiCiccio, C. J., & Romano, J. P. (2017). Robust permutation tests for correlation and regression coefficients. *Journal of the American Statistical Association*, 112(519), 1211–1220. <https://doi.org/10.1080/01621459.2016.120>
- Gail, M. H., Mark, S. D., Carroll, R. J., Green, S. B., & Pee, D. (1996). On design considerations and randomization-based inference for community intervention trials. *Statistics in Medicine*, 15(11), 1069–1092. [https://doi.org/10.1002/\(SICI\)1097-0258\(19960615\)15:11%3C1069::AID-SIM220%3E3.0.CO;2-Q](https://doi.org/10.1002/(SICI)1097-0258(19960615)15:11%3C1069::AID-SIM220%3E3.0.CO;2-Q)
- Hayes, R. J., & Moulton, L. H. (2017). *Cluster randomised trials* (2nd ed.). Chapman and Hall/CRC. <https://doi.org/10.4324/9781315370286>
- Kvalem, I. L., Sundet, J. M., Rivø, K. I., Eilertsen, D. E., & Bakketeig, L. S. (1996). The effect of sex education on adolescents' use of condoms: Applying the Solomon four-group design. *Health Education Quarterly*, 23(1), 34–47. <https://doi.org/10.1177/109019819602300103>
- Phipson, B., & Smyth, G. K. (2010). Permutation p-values should never be zero: Calculating exact p-values when permutations are randomly drawn. *Statistical Applications in Genetics and Molecular Biology*, 9(1), Article 39. <https://doi.org/10.2202/1544-6115.1585>
- Wu, J., & Ding, P. (2021). Randomization tests for weak null hypotheses in randomized experiments. *Journal of the American Statistical Association*, 116(536), 1898–1913. <https://doi.org/10.1080/01621459.2020.17504>

Examples

```
fit <- with(solomon_example, fit_solomon_glm(y_post, treat, pretested, y_pre))
# Few permutations keep the example fast; use the default for analyses.
perm_solomon(fit, reps = 199, seed = 1)
```

plan_solomon

Sample-size planning for Solomon designs

Description

[Stable] Finds the smallest Solomon four-group design, at a fixed allocation across the four cells, whose power for each Solomon estimand reaches a target. `power_solomon()` answers the forward question, the power of a given design; `plan_solomon()` answers the inverse.

Usage

```

plan_solomon(
  power = 0.8,
  delta,
  sens = 0,
  rho = 0.5,
  sigma = 1,
  alpha = 0.05,
  estimand = c("ate", "sensitization", "pretested", "unpretested"),
  allocation = c(1, 1, 1, 1),
  method = c("analytic", "simulation"),
  sims = 1000,
  seed = NULL,
  max_n = 10000
)

```

Arguments

power	Target power, between alpha and 1. Default is 0.80.
delta	Treatment effect among unpretested participants, on the posttest scale.
sens	Sensitization: the additional treatment effect among pretested participants. Default is 0.
rho	Pretest-posttest correlation among pretested participants.
sigma	Posttest residual standard deviation in all cells.
alpha	Two-sided significance level. Default is 0.05.
estimand	Which estimands to plan for: any of "ate", "sensitization", "pretested", and "unpretested". Default is all four.
allocation	Relative cell sizes for n1 (pretested treatment), n2 (pretested control), n3 (unpretested treatment), and n4 (unpretested control). Default is equal allocation.
method	"analytic" (default) or "simulation"; see the Methods section.
sims	Monte Carlo replications per evaluation when method = "simulation".
seed	Optional integer seed for method = "simulation". Every evaluation reuses it, so designs are compared on common random numbers.
max_n	Largest size considered for the smallest cell.

Details

The data-generating model is the one `power_solomon()` uses and validates: normally distributed posttests with residual standard deviation `sigma` in every cell, a pretest-posttest correlation of `rho` among pretested participants, a treatment effect of `delta` among unpretested participants, and `delta + sens` among pretested participants.

Value

A data frame with one row per estimand: the estimand, its true effect, the four cell sizes, the total sample size, the achieved power, the basis ("analytic" or "simulation"), the Monte Carlo standard error (NA for analytic rows), the target power, alpha, and a note. Estimands whose true effect is zero, or whose target is not reached by max_n, return NA sizes with an explanatory note. The planning values are kept in the attribute settings, which `analysis_plan_solomon()` reads.

Planning for sensitization

The sensitization contrast (Pretest x Treatment) is the difference between the two simple treatment effects, and the equal-weighted average treatment effect is their average. The sensitization contrast therefore has exactly four times the sampling variance of the average treatment effect, for any pretest correlation and any allocation. Detecting sensitization as large as the average treatment effect needs about four times as many participants, and detecting sensitization half as large needs about sixteen times as many. A Solomon study powered only for the average treatment effect is usually underpowered for the question the design exists to answer.

Methods

- `method = "analytic"` (default) uses normal-theory power: a two-sample t test for the unpretested effect, an ANCOVA comparison whose residual variance is reduced by the squared pretest-posttest correlation for the pretested effect, and Welch-Satterthwaite degrees of freedom for the contrasts that combine pretest conditions. The search is exact: at the returned design power reaches the target, and with one fewer participant in the smallest cell it does not.
- `method = "simulation"` starts from the analytic design and increases it until the rejection rate of the package's own GLM test, estimated with `power_solomon()`, reaches the target. It reports the Monte Carlo standard error of the achieved power. The validation of `power_solomon()` found the default HC3 standard errors conservative with 20 or fewer participants per cell, so for small designs the simulated answer can be larger than the analytic one; from about 30 per cell the two agree closely.

Designs not covered

Binary and count outcomes, clustered assignment, and longitudinal follow-ups are not supported; the calculations assume independent, normally distributed posttests and no missing data. Designs with several treatments are not supported; plan each treatment-control comparison as a four-group design.

References

- Morris, T. P., White, I. R., & Crowther, M. J. (2019). Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, 38(11), 2074–2102. <https://doi.org/10.1002/sim.8086>
- Satterthwaite, F. E. (1946). An approximate distribution of estimates of variance components. *Biometrics Bulletin*, 2(6), 110–114. <https://doi.org/10.2307/3002019>
- Welch, B. L. (1947). The generalization of "Student's" problem when several different population variances are involved. *Biometrika*, 34(1–2), 28–35. <https://doi.org/10.1093/biomet/34.1-2.28>

See Also

[power_solomon\(\)](#) for the power of a given design, and [analysis_plan_solomon\(\)](#) for an analysis plan built on the result.

Examples

```
# Equal allocation, a treatment effect of 0.4 SD, and sensitization of 0.2 SD
plan_solomon(power = 0.80, delta = 0.4, sens = 0.2, rho = 0.5)

# Pretesting is expensive: half as many participants in the pretested cells
plan_solomon(power = 0.80, delta = 0.4, sens = 0.2, rho = 0.5,
              estimand = "ate", allocation = c(1, 1, 2, 2))
```

plot_classic_flow	<i>Historical Solomon decision path</i>
-------------------	---

Description

[Stable] Draws the conditional Tests A-I sequence implemented in [fit_solomon_classic\(\)](#) as a decision tree. Test A (the Pretest x Treatment interaction) decides the branch. If it is significant, Tests B and C examine the treatment effect within each pretest condition. If not, Test D examines the treatment main effect, followed if necessary by the selected pretested-groups test (E, F, or G), Test H, and finally Test I, the Walton Braver and Braver (1988) Stouffer combination. In the original 1988 sequence each of these is reached only if the one before it is nonsignificant; in the 1990 amendment (Braver & Walton Braver, 1990), all of them are run once Test D is nonsignificant. The 1995 revision removed Test D (Walton Braver & Braver, 1995, as cited in Sawilowsky, 1996, p. 2).

Usage

```
plot_classic_flow(fit = NULL, flow = c("1988", "1990", "1995"))
```

Arguments

fit	Optional fit from fit_solomon_classic() . Without one, the generic diagram is drawn.
flow	Version of the sequence to draw when fit is not given: "1988", "1990", or "1995". With a fit, the fit's own version is used.

Details

Given a fitted classic analysis, the tests it visited are highlighted with their p-values, following the fit's recorded path exactly, and the caption gives its conclusion.

The figure is a teaching and replication aid, not a recommended workflow. Sawilowsky et al. (1994) found that the conditional sequence ending in Test I has an experiment-wise Type I error rate well above its nominal level; see `vignette("solomon-methods")`.

Value

A ggplot object.

References

Braver, S. L., & Walton Braver, M. C. (1990). Meta-analysis for Solomon four-group designs reconsidered: A reply to Sawilowsky and Markman. *Perceptual and Motor Skills*, 71(1), 321–322. <https://doi.org/10.2466/pms.1990.71.1.321>

Sawilowsky, S. S. (1996, June 23). *Controlling experiment-wise Type I error of meta-analysis in the Solomon four-group design* [Paper presentation]. First International Conference on Multiple Comparisons, Tel Aviv, Israel. https://digitalcommons.wayne.edu/coe_tbf/29/

Sawilowsky, S. S., Kelley, D. L., Blair, R. C., & Markman, B. S. (1994). Meta-analysis and the Solomon four-group design. *The Journal of Experimental Education*, 62(4), 361–376. <https://doi.org/10.1080/00220973.1994.10800022>

Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

Examples

```
plot_classic_flow()
classic <- with(solomon_example, fit_solomon_classic(y_post, treat, pretested, y_pre))
plot_classic_flow(classic)
plot_classic_flow(flow = "1990")
plot_classic_flow(flow = "1995")
```

plot_perm

Plot permutation distribution for a Solomon contrast

Description

[Stable] Draws the permutation distribution of the test statistic with the observed value marked. The subtitle reports the same p-value as `perm_solomon()`, and says whether it came from every possible allocation or from sampled permutations.

Usage

```
plot_perm(perm)
```

Arguments

perm An object returned by `perm_solomon(..., return_dist = TRUE)`.

Value

A ggplot object

Examples

```
fit <- with(solomon_example, fit_solomon_glm(y_post, treat, pretested, y_pre))
perm <- perm_solomon(fit, reps = 199, seed = 1, return_dist = TRUE)
plot_perm(perm)
```

plot_power_solomon *Power curves for a Solomon design*

Description

[Stable] Draws power against sample size for each Solomon estimand, so the effect of cell size, effect size, pretest-posttest correlation, and sensitization on a planned study can be seen rather than read from a table. A horizontal line marks the target power, and each curve is annotated with the design `plan_solomon()` returns for that target.

Usage

```
plot_power_solomon(
  n = seq(10, 150, by = 10),
  delta = 0.3,
  sens = 0,
  rho = 0.5,
  sigma = 1,
  alpha = 0.05,
  estimand = c("ate", "sensitization", "pretested", "unpretested"),
  allocation = c(1, 1, 1, 1),
  target = 0.8,
  method = c("analytic", "simulation"),
  sims = 500,
  seed = NULL
)
```

Arguments

n	Sizes of the smallest cell to plot. Other cells follow allocation.
delta, sens, rho	Treatment effect among unpretested participants, sensitization, and pretest-posttest correlation, as in <code>power_solomon()</code> . At most one of them may have more than one value; that one is shown by color.
sigma	Posttest residual standard deviation in all cells.
alpha	Two-sided significance level. Default is 0.05.
estimand	Estimands to plot: any of "ate", "sensitization", "pretested", and "unpretested".
allocation	Relative cell sizes for n1 to n4, as in <code>plan_solomon()</code> .
target	Target power marked on the figure. Default is 0.80. Use NULL to omit the target line and the planned designs.
method	"analytic" (default) or "simulation".
sims	Replications per point when method = "simulation".
seed	Optional seed for method = "simulation".

Details

Curves use the normal-theory power calculations validated for `power_solomon()` by default. With `method = "simulation"`, each point is the rejection rate of the package's own GLM test from `power_solomon()`, shown with a band of two Monte Carlo standard errors.

Power differs sharply between estimands: the sensitization contrast has four times the sampling variance of the average treatment effect, so its curve rises far more slowly. The figure is faceted by estimand to keep that difference visible.

Value

A ggplot object.

See Also

`power_solomon()`, `plan_solomon()`

Examples

```
plot_power_solomon(n = seq(10, 200, by = 10), delta = 0.4, sens = 0.2)
plot_power_solomon(n = seq(10, 200, by = 10), delta = 0.4, rho = c(0, 0.5, 0.8),
  estimand = c("ate", "pretested"))
```

plot_sensitization	<i>Pretest sensitization figure</i>
--------------------	-------------------------------------

Description

[Stable] Draws the Pretest x Treatment interaction that the Solomon design exists to test: model-adjusted posttest means for the four groups, with confidence intervals, joined within each pretest condition so that sensitization appears as lines that are not parallel.

Usage

```
plot_sensitization(fit, bounds = NULL, alpha = 0.05, show_observed = TRUE)
```

Arguments

<code>fit</code>	A fit from <code>fit_solomon_glm()</code> or <code>fit_solomon_ml()</code> .
<code>bounds</code>	Optional equivalence bounds for the sensitization contrast, as one positive number or <code>c(lower, upper)</code> . When supplied, the caption reports the outcome of <code>equivalence_solomon()</code> with these bounds, which should be fixed before the data are examined. Not available for a design with several treatments.
<code>alpha</code>	Significance level for the equivalence test when bounds is supplied. Default is 0.05.
<code>show_observed</code>	Logical; if TRUE (default), overlay observed cell means as hollow points.

Details

The treatment effect among pretested participants is adjusted for the pretest, so the sensitization contrast reported by `fit_solomon_glm()` or `fit_solomon_ml()` is not the difference of differences among raw cell means. The figure therefore draws model-adjusted means: pretested groups are evaluated at the mean pretest score among pretested participants (the usual ANCOVA adjusted mean), unpretested groups without a pretest, and any covariates at their sample means. Because neither model has a treatment-by-pretest-score term, the difference of differences among these adjusted means equals the fitted Pretest x Treatment estimate exactly. Observed cell means are shown as hollow points for comparison.

Intervals for the adjusted means use the fit's own covariance matrix and reference distribution:

- for `fit_solomon_glm()`, t with residual degrees of freedom, Satterthwaite t for CR2, or the normal distribution for binomial and Poisson models, whose means are shown on the link scale;
- for `fit_solomon_ml()`, the normal distribution under the default Wald inference (van Engelenburg, 1999), or Welch-Satterthwaite t under `inference = "satterthwaite"`.

The sensitization estimate and interval in the subtitle are taken unchanged from the fit.

Value

A ggplot object.

Designs with several treatments

For a fit from `fit_solomon_glm()` with several treatments and a control, the figure shows the model-adjusted mean of every condition with and without a pretest, adjusted as above, with one line for each condition, the control first. Pretest sensitization appears as a treatment line that is not parallel to the control line: for each treatment, the difference of differences between its means and the control's equals its fitted Pretest x Treatment estimate. The subtitle reports the omnibus Pretest x Condition test of the fit, which asks whether pretesting changes the effect of any treatment. `bounds` is not available for these fits; test equivalence one comparison at a time with `equivalence_solomon()`.

References

van Engelenburg, G. (1999). *Statistical analysis for the Solomon four-group design* (Research Report 99-06). University of Twente. ERIC. <https://eric.ed.gov/?id=ED435692>

See Also

`equivalence_solomon()`, `plot_solomon_effects()`

Examples

```
fit <- with(solomon_example, fit_solomon_glm(y_post, treat, pretested, y_pre))
plot_sensitization(fit)
plot_sensitization(fit, bounds = 5)
```

```

ml <- with(solomon_example, fit_solomon_ml(y_post, treat, pretested, y_pre,
                                           inference = "satterthwaite"))
plot_sensitization(ml)

# A six-group design: two treatments and a control.
fit6 <- fit_solomon_glm(post_behavior, condition, pretested, pre_behavior,
                       control = "Control", data = mai2020)
plot_sensitization(fit6)

```

plot_solomon

Plot Solomon posttest cell means in base graphics (deprecated)

Description

[Deprecated] `plot_solomon()` is replaced by `plot_solomon_means()`, which draws the same means and intervals with `ggplot2`, as every other plot in the package does. `plot_solomon()` still works, with a deprecation warning, and still draws in base graphics and returns the cell summaries invisibly.

Usage

```
plot_solomon(y, treat, pretested)
```

Arguments

<code>y</code>	Numeric vector of posttest scores.
<code>treat</code>	Treatment indicator coded 0 = control and 1 = treatment.
<code>pretested</code>	Pretest indicator coded 0 = not pretested and 1 = pretested.

Details

`plot_solomon()` draws the four-group design only. For a design with several treatments, use `plot_solomon_means()` with control.

Value

Invisibly returns a data frame containing cell summaries.

Examples

```

# Use plot_solomon_means() instead:
plot_solomon_means(y_post, treat, pretested, data = solomon_example)

```

plot_solomon_change *Pretest-to-posttest change in a Solomon design*

Description

[Stable] Draws mean pretest and posttest scores for the two pretested groups, joined to show change, alongside posttest means for the two unpretested groups. The unpretested groups appear at posttest only: their missing pretest is the experimental manipulation (Solomon, 1949), not missing data, and the figure labels it that way.

Usage

```
plot_solomon_change(
  y_post,
  treat,
  pretested,
  y_pre,
  show_individuals = FALSE,
  conf_level = 0.95,
  control = NULL,
  data = NULL
)
```

Arguments

y_post	Numeric posttest scores.
treat	Treatment indicator coded 0 = control and 1 = treatment; or, for a design with several treatments, a factor or character vector of conditions, with the control named by control.
pretested	Pretest indicator coded 0 = unpretested and 1 = pretested.
y_pre	Numeric pretest scores, missing by design for unpretested participants.
show_individuals	Logical; if TRUE, draw each pretested participant's change as a faint line behind the means. Default FALSE.
conf_level	Confidence level for the intervals, which use the t distribution with n - 1 degrees of freedom. Default is 0.95.
control	The control condition, when treat is a factor or character vector. With two conditions the figure is the same as with a 0/1 treat.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Details

Trajectories use pretested participants with both scores observed. Pretested participants whose pretest is incidentally missing are left out of the trajectories and counted in the caption, and are never imputed; see `check_solomon_missing()`.

A change reading of this figure applies only to the pretested groups. The Solomon contrasts the package reports compare posttests, adjusting for the pretest where one exists; see `compare_solomon_methods()` for the estimand behind each analysis.

Value

A ggplot object.

Designs with several treatments

For a Solomon N-group design, with k treatments and a control each with and without a pretest (Steyn, 2009), give `treat` as a factor or character vector of conditions and name the control with `control`. The figure then shows all $2(k + 1)$ groups: color marks the condition, the control first as in `plot_sensitization()`, and the point shape marks whether the group was pretested.

References

Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>

Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

Examples

```
with(solomon_example, plot_solomon_change(y_post, treat, pretested, y_pre))

# A six-group design: two treatments and a control.
plot_solomon_change(post_behavior, condition, pretested, pre_behavior,
  control = "Control", data = mai2020)
```

plot_solomon_design *Schematic of a Solomon design*

Description

[Stable] Draws the Solomon (1949) four-group design in the notation of Campbell and Stanley (1963/1966, p. 6): each row is a randomized group (R), O marks an observation (pretest or posttest), and X marks the treatment. Groups 1 and 2 are pretested; Groups 3 and 4 are not, by design, so their missing pretest is part of the experiment rather than missing data.

Usage

```
plot_solomon_design(
  y_post = NULL,
  treat = NULL,
  pretested = NULL,
  fit = NULL,
  control = NULL,
  treatments = NULL,
  data = NULL,
  x = deprecated()
)
```

Arguments

y_post	Optional numeric posttest scores, with treat and pretested. A fit from <code>fit_solomon_glm()</code> given here is used as fit.
treat	Treatment indicator coded 0 = control and 1 = treatment, when y_post is given; or, for a design with several treatments, a factor or character vector of conditions, with the control named by control.
pretested	Pretest indicator coded 0 = unpretested and 1 = pretested, when y_post is given.
fit	Optional fit from <code>fit_solomon_glm()</code> , used instead of y_post, treat, and pretested.
control	The control condition, when treat is a factor or character vector. With treatments, an optional label for the control (default "Control" with treatment labels, "control" with a number).
treatments	For the teaching schematic without data: the number of treatments, or a character vector of treatment labels. NULL (the default) or 1 draws the four-group design.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").
x	[Deprecated] Use y_post or fit.

Details

Called with no data, the function draws the generic schematic used for teaching. Given data or a fitted model, it labels each group with its size and posttest mean. Groups with no participants, or with fewer than two observed posttest scores, are flagged, using the same rule as `validate_solomon()`.

Value

A ggplot object.

Designs with several treatments

A Solomon N-group design crosses k treatments and a control with pretesting, giving $2(k + 1)$ groups: six for two treatments and eight for three (Steyn, 2009). The rows run in the order `solomonR`

uses for these designs: the pretested treatment groups, the pretested control group, then the unpretested groups in the same order. With two or more treatments, X1, X2, and so on mark the treatments, and the subtitle gives the key.

To draw such a design, give a fit from `fit_solomon_glm()` with several treatments, or data with `treat` as a factor or character vector of conditions and the control named by `control`. For the teaching schematic without data, give treatments: the number of treatments (`treatments = 2`) or their labels (`treatments = c("RP", "GS")`).

References

Campbell, D. T., & Stanley, J. C. (1966). *Experimental and quasi-experimental designs for research*. Rand McNally. (Original work published 1963)

Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>

Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

Examples

```
plot_solomon_design()
plot_solomon_design(y_post, treat, pretested, data = solomon_example)

# A six-group design: two treatments and a control.
plot_solomon_design(treatments = 2)
plot_solomon_design(post_behavior, condition, pretested,
                     control = "Control", data = mai2020)
```

plot_solomon_effects *Forest plot of the Solomon contrasts*

Description

[Stable] Plots the four Solomon contrasts from a fitted model: the average treatment effect across pretest conditions, the Pretest x Treatment sensitization contrast, and the treatment effects among pretested and unpretested participants, each with its confidence interval and a reference line at zero.

Usage

```
plot_solomon_effects(fit, bounds = NULL)
```

Arguments

<code>fit</code>	A fit from <code>fit_solomon_glm()</code> , <code>fit_solomon_ml()</code> , <code>fit_solomon_sem()</code> , or <code>fit_solomon_sem_latent()</code> .
<code>bounds</code>	Optional equivalence bounds for the sensitization contrast, as one positive number or <code>c(lower, upper)</code> , drawn as a shaded band on that row. Use the same bounds as <code>equivalence_solomon()</code> , fixed before the data were examined.

Details

Estimates and intervals are taken unchanged from the fitted object, so the figure agrees with its printed output, and the caption states the confidence level and the reference distribution the model used. Contrasts a model does not estimate are omitted and named in the caption rather than drawn as zero.

Value

A ggplot object.

Designs with several treatments

For a fit from `fit_solomon_glm()` with several treatments and a control, the figure has one panel for each of the four contrasts and one row in each panel for each comparison, such as "RP vs Control", in the order of the fit. The intervals are the unadjusted ones the fit reports. The caption names the adjustment the fit applied to its p-values, by default Holm's (1979) procedure, and says that the intervals are not adjusted. With bounds, the band is drawn on every Pretest x Treatment row.

References

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2), 65–70. <https://www.jstor.org/stable/4615733>

Examples

```
fit <- with(solomon_example, fit_solomon_glm(y_post, treat, pretested, y_pre))
plot_solomon_effects(fit)
plot_solomon_effects(fit, bounds = 5)

# A six-group design: two treatments and a control.
fit6 <- fit_solomon_glm(post_behavior, condition, pretested, pre_behavior,
  control = "Control", data = mai2020)
plot_solomon_effects(fit6)
```

plot_solomon_gg	<i>Plot Solomon posttest cell means with ggplot2 (deprecated)</i>
-----------------	---

Description

[Deprecated] `plot_solomon_gg()` was renamed `plot_solomon_means()` in `solomonR` 0.8.0, so that each plot in the package is named for what it shows. It still works, with a deprecation warning, and returns the plot that `plot_solomon_means()` draws.

Usage

```
plot_solomon_gg(y, treat, pretested)
```

Arguments

y	Numeric vector of posttest scores.
treat	Treatment indicator coded 0 = control and 1 = treatment.
pretested	Pretest indicator coded 0 = not pretested and 1 = pretested.

Value

A ggplot object.

Examples

```
# Use plot_solomon_means() instead:
plot_solomon_means(y_post, treat, pretested, data = solomon_example)
```

plot_solomon_means	<i>Plot Solomon posttest cell means</i>
--------------------	---

Description

[Stable] Displays the posttest mean of each of the four Solomon groups with its confidence interval, with the pretested and unpretested groups side by side. Intervals use the t distribution with $n - 1$ degrees of freedom within each group. For the model-adjusted means behind the sensitization contrast, see [plot_sensitization\(\)](#); for pretest-to-posttest change, see [plot_solomon_change\(\)](#).

Usage

```
plot_solomon_means(
  y_post,
  treat,
  pretested,
  conf_level = 0.95,
  control = NULL,
  data = NULL
)
```

Arguments

y_post	Numeric posttest scores.
treat	Treatment indicator coded 0 = control and 1 = treatment; or, for a design with several treatments, a factor or character vector of conditions, with the control named by control.
pretested	Pretest indicator coded 0 = not pretested and 1 = pretested.
conf_level	Confidence level for the intervals. Default is 0.95.
control	The control condition, when treat is a factor or character vector. With two conditions the figure is the same as with a 0/1 treat.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Value

A ggplot object. Its data element holds the group summaries: n, mean, sd, se, and the interval limits lo and hi. Its treat column is 0/1 for a four-group design and holds the condition for a design with several treatments.

Designs with several treatments

For a Solomon N-group design, with k treatments and a control each with and without a pretest (Steyn, 2009), give treat as a factor or character vector of conditions and name the control with control. Each panel then shows the k + 1 conditions, the control first, as with the four-group design.

References

Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

See Also

[plot_solomon_change\(\)](#), [plot_sensitization\(\)](#)

Examples

```
plot_solomon_means(y_post, treat, pretested, data = solomon_example)

# A six-group design: two treatments and a control.
plot_solomon_means(post_behavior, condition, pretested,
                    control = "Control", data = mai2020)
```

plot_tipping_point	<i>Plot a tipping-point analysis</i>
--------------------	--------------------------------------

Description

[Experimental] Plots the pooled estimate of a Solomon contrast, with its confidence interval, against the offset added to the imputed posttests, from [tipping_point_solomon\(\)](#). Dashed lines mark the offsets at which the conclusion changes.

Usage

```
plot_tipping_point(tipping)
```

Arguments

tipping A result from [tipping_point_solomon\(\)](#).

Value

A ggplot object.

See Also

`tipping_point_solomon()`

Examples

```
d <- solomon_example
set.seed(82)
d$y_post[sample(nrow(d), 20)] <- NA
tp <- tipping_point_solomon(y_post, treat, pretested, y_pre, groups = "treatment",
                           deltas = seq(-10, 10, by = 2.5), m = 10, seed = 1, data = d)
plot_tipping_point(tp)
```

power_solomon

Power simulation for Solomon designs

Description

[Stable] Simulates normally distributed Solomon four-group data and reports the rejection rate of each Solomon test at the chosen alpha, with its Monte Carlo standard error.

Usage

```
power_solomon(
  n = 50,
  delta = 0.3,
  sens = 0,
  rho = 0.5,
  sigma = 1,
  alpha = 0.05,
  sims = 2000,
  stouffer = TRUE,
  seed = NULL
)
```

Arguments

n	Cell sizes: a single number used for all four cells, or four sizes given as a list or vector with elements n1 (pretested treatment), n2 (pretested control), n3 (unpretested treatment), and n4 (unpretested control).
delta	Treatment effect among unpretested participants, on the posttest scale.
sens	Sensitization: the additional treatment effect among pretested participants (0 = none).
rho	Pretest-posttest correlation among pretested participants.

sigma	Posttest residual standard deviation in all cells.
alpha	Significance level. Default is 0.05.
sims	Number of Monte Carlo replications.
stouffer	Logical; if TRUE, also report the historical Test I rejection rate: one-tailed p-values in the direction treatment > control are combined, and the combined z is judged by its two-tailed p-value, as in <code>fit_solomon_classic()</code> and Walton Braver and Braver's (1988, p. 153) worked example.
seed	Optional integer seed. The global random number state is restored afterwards.

Details

Every participant has a latent baseline drawn from a standard normal distribution, which only pretested participants observe, so structural pretest absence is part of the design rather than missing data. The posttest residual standard deviation is sigma in every cell, and the pretest-posttest correlation among pretested participants is rho. The treatment effect is delta among unpretested participants and delta + sens among pretested participants, so the equal-weighted average treatment effect is $\text{delta} + \text{sens} / 2$.

The reported power is a rejection rate: when true_effect is zero it estimates the Type I error rather than power.

Tests are computed from the same functions users would call, so reported power reflects the package's default inference: `fit_solomon_glm()` with HC3 standard errors and t reference distributions, the 2x2 ANOVA interaction, and the historical Test I (Walton Braver & Braver, 1988).

Value

A data frame with one row per test, giving the estimand, the test used, the true effect, the rejection rate and its Monte Carlo standard error, the replications that produced a usable fit, the number of failures, and alpha. Test I rows report NA for true_effect, because the combination targets a directional hypothesis rather than a single contrast, and all Test I values are NA when stouffer = FALSE.

Validation

A pre-specified simulation study of 126 scenarios and 315,000 replications checked this function against normal-theory benchmarks. The protocol and its amendment were posted to issue #18 before any results were examined, and the article "Validating power_solomon()" on the package website reports the study in full.

- The 2x2 ANOVA interaction agreed with its analytic benchmark in every scenario, with mean differences of 0.002 or less.
- Test I held its nominal size under the complete null, rejecting between 4.4% and 5.6% of the time at alpha = .05.
- Rejection rates were invariant to the residual scale, and no fit failed.
- Rejection rates from the unified GLM are conservative in small samples, because the HC3 standard errors the package uses by default are conservative there. Type I error averaged 0.041 with 10 participants per cell and 0.049 with 100, and simulated power fell below the normal-theory benchmark by 0.030 on average with 10 per cell (up to 0.083 in one scenario), 0.017 with 20, and 0.003 with 100.

With 20 or fewer participants per cell, treat GLM-based power as a conservative figure rather than an exact one.

Only continuous outcomes are simulated. Binary and count outcomes are not supported; for binary outcomes see `marginal_solomon()` and its simulation validation on issue #43. Designs with several treatments are not supported; compute power for each treatment-control comparison as a four-group design.

The arguments follow the order of `plan_solomon()` and `plot_power_solomon()`: `n`, `delta`, `sens`, `rho`, `sigma`, `alpha`. Before `solomonR` 0.8.0, `rho` came before `sens` and `alpha` came last, so a call that passes these by position gets a warning that it was read in the new order. Naming the arguments avoids the ambiguity.

References

Morris, T. P., White, I. R., & Crowther, M. J. (2019). Using simulation studies to evaluate statistical methods. *Statistics in Medicine*, 38(11), 2074–2102. <https://doi.org/10.1002/sim.8086>

Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

Examples

```
power_solomon(n = 30, delta = 0.5, sens = 0.2, sims = 50, seed = 1)
```

report_solomon	APA 7 results text for a Solomon analysis
----------------	---

Description

[Stable] Turns a fitted Solomon analysis into APA 7 results sentences, a short design statement, and the references for exactly the methods that analysis used.

Usage

```
report_solomon(fit, design = NULL, digits = 2, format = c("text", "markdown"))
```

Arguments

fit	A fitted Solomon analysis (see Details).
design	Optional list describing the design: randomized, the numbers assigned to the four groups (pretested treatment, pretested control, unpretested treatment, unpretested control); prespecified, TRUE or FALSE for whether the sensitization analysis was pre-specified; plan, the <code>analysis_plan_solomon()</code> result the study registered, which sets prespecified from the plan’s confirmatory contrasts; measurement, one description of the measurement procedure or one per group; and assignment, "random" or "nonrandom". For a design with k

	treatments, randomized (and measurement, when given per group) has one entry for each of the $2(k + 1)$ groups, in this order: the pretested treatments (in the order of the levels of <code>treat</code>), the pretested control, the unpretested treatments, and the unpretested control. For <code>mai2020</code> , that is pretested RP, pretested GS, pretested Control, unpretested RP, unpretested GS, and unpretested Control.
<code>digits</code>	Decimal places for estimates and statistics. Default 2.
<code>format</code>	"text" (default) or "markdown", which italicizes statistical symbols.

Details

Supported objects come from `fit_solomon_glm()`, `fit_solomon_ml()`, `fit_solomon_classic()`, `perm_solomon()`, `marginal_solomon()`, `equivalence_solomon()`, `fisher_solomon()`, `solomon_from_summary()`, `fit_solomon_sem()`, `fit_solomon_sem_latent()`, `baseline_solomon()`, `fit_solomon_mi()`, `tipping_point_solomon()`, `fit_solomon_mmrn()`, and `fit_solomon_steyn()`. The references depend on the options the fit used: for example, a CR2 fit cites Bell and McCaffrey (2002) and Pustejovsky and Tipton (2018), and the 1990 flow of the classic analysis adds Braver and Walton Braver (1990). Every reference matches the package's canonical APA 7 bibliography.

The design statement follows the MERIT recommendations on reporting measurement in trials (French et al., 2021b): the numbers analyzed in each group, attrition by group when `design$randomized` is given, whether the sensitization analysis was pre-specified, and the measurement procedure in each group (Recommendation 11 is to use identical measurement protocols in all arms, p. 34).

Designs with several treatments. For a `fit_solomon_glm()` fit with control and three or more conditions (class `solomon_ngroup`), the design statement names the treatments, the control, and the $2(k + 1)$ groups of the design (Steyn, 2009), with the numbers analyzed in each. The results give the omnibus tests of the Pretest $\times$ Condition interaction and of the conditions averaged over pretest conditions, then the Solomon contrasts of each comparison. Their p-values are adjusted within each contrast across the comparisons, by Holm's (1979) procedure unless the fit chose another adjustment; the confidence intervals are not adjusted. Comparisons defined by weights are reported with their weights. An `equivalence_solomon()` test of one comparison and the `baseline_solomon()` comparisons of such a design are reported with the same design statement. `solomonR` follows a pre-publication draft of Steyn (2009), which the author provided; see `fit_solomon_steyn()`.

Nonrandomized designs. With `design$assignment = "nonrandom"`, the results describe differences between groups rather than treatment effects, and the design statement names the threats that random assignment would otherwise control: selection bias, the largest threat to internal validity in quasi-experimental research (Edmonds & Kennedy, 2017, p. 7), and instrumentation, which with selection bias Edmonds and Kennedy name as the threats most common in quasi-experimental Solomon designs (p. 94). It adds that baseline differences can be examined only in the pretested arms (`baseline_solomon()`): without random assignment the unpretested arms form a static-group comparison, whose groups cannot be shown to have been equivalent (Campbell & Stanley, 1963/1966, pp. 12, 25).

What the report does not decide. It states results; it does not interpret them. Whether the analysis was pre-specified must be supplied, never inferred, and the choice of analysis, the reading of the results, and the conclusions remain the researcher's responsibility.

Value

An object of class `solomon_report` with method, results, and design (character vectors of sentences), table (the estimates), and references (APA 7 reference entries, in APA order).

References

- Campbell, D. T., & Stanley, J. C. (1966). *Experimental and quasi-experimental designs for research*. Rand McNally. (Original work published 1963)
- Edmonds, W. A., & Kennedy, T. D. (2017). *An applied guide to research designs: Quantitative, qualitative, and mixed methods* (2nd ed.). SAGE Publications. <https://doi.org/10.4135/9781071802779>
- French, D. P., Miles, L. M., Elbourne, D., Farmer, A., Gulliford, M., Locock, L., Sutton, S., McCambridge, J., & MERIT Collaborative Group. (2021b). Reducing bias in trials from reactions to measurement: The MERIT study including developmental work and expert workshop. *Health Technology Assessment*, 25(55), 1–72. <https://doi.org/10.3310/hta25550>
- Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2), 65–70. <https://www.jstor.org/stable/4615733>
- Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

Examples

```
fit <- with(solomon_example, fit_solomon_glm(y_post, treat, pretested, y_pre))
report_solomon(fit, design = list(prespecified = TRUE))

# A six-group design: two treatments and a control (Mai et al., 2020).
fit6 <- fit_solomon_glm(post_behavior, condition, pretested, pre_behavior,
                        control = "Control", data = mai2020)
report_solomon(fit6)
```

simulate_solomon

Simulate a Solomon four-group study with known effects

Description

[Stable] Generates individual data for a randomized Solomon four-group study with chosen treatment, pretest, and sensitization effects, and attaches the true value of every Solomon estimand. It is a solomonR teaching tool: instructors can show that each analysis recovers the effects that were built in, and researchers can check an analysis plan on data whose answer is known. With several treatment effects it simulates a Solomon N-group design (see "Designs with several treatments").

Usage

```
simulate_solomon(
  n = 30,
  delta = 0,
  sens = 0,
  pretest_effect = 0,
  rho = 0.5,
  sigma = 1,
```

```

    mean = 0,
    digits = NULL,
    limits = NULL,
    seed = NULL
  )

```

Arguments

n	Participants per group: one number for all groups, or one number per group. For the four-group design, four numbers in the order pretested treatment, pretested control, unpretested treatment, unpretested control. For k treatments, 2(k + 1) numbers in the order given in "Designs with several treatments". Default 30 per group.
delta	Treatment effect among unpretested participants. For a design with several treatments, one effect per treatment, named by treatment. Default 0.
sens	Pretest x Treatment interaction (sensitization): the extra treatment effect among pretested participants. For a design with several treatments, one number for every treatment or one per treatment. Default 0.
pretest_effect	Effect of taking the pretest on the posttest, among controls. Default 0.
rho	Pretest-posttest correlation within the pretested groups. Default 0.5.
sigma	Standard deviation of the scores within each group. Default 1.
mean	Mean pretest and control-group posttest score. Default 0.
digits	Optional number of decimal places to round scores to.
limits	Optional lower and upper limits for scores, applied after rounding.
seed	Optional integer seed. The global random number state is left unchanged.

Details

Data-generating model. It is the model of [power_solomon\(\)](#), whose rebuild was validated under a protocol posted on issue #18, with a pretesting main effect and a location added. Every participant has a latent baseline A from a standard normal distribution. Participants in the pretested groups have a pretest score of $\text{mean} + \text{sigma} * A$. Every participant's posttest score is

$$\text{mean} + \text{delta} * \text{treat} + \text{pretest_effect} * \text{pretested} + \text{sens} * \text{treat} * \text{pretested} + \text{sigma} * (\text{rho} * A + \sqrt{1 - \text{rho}^2} * e),$$

where e is independent standard normal error. So rho is the pretest-posttest correlation within each pretested group, and sigma is the standard deviation of both scores within each group. The four groups are generated in the package's order (pretested treatment, pretested control, unpretested treatment, unpretested control), and the baselines are drawn before the errors.

Relation to the package's other simulations. With $\text{mean} = 0$, $\text{sigma} = 1$, and $\text{pretest_effect} = 0$, the data are the first data set that [power_solomon\(\)](#) simulates with the same seed. With $n = 30$, $\text{delta} = 5$, $\text{pretest_effect} = 2$, $\text{rho} = 0.6$, $\text{sigma} = 10$, $\text{mean} = 50$, $\text{seed} = 20260915$, $\text{digits} = 0$, and $\text{limits} = c(0, 100)$, they are the bundled [solomon_example](#) data.

True values. The "truth" attribute gives each estimand:

- the treatment effect among unpretested participants, delta;

- among pretested participants, $\delta + \text{sens}$;
- the Pretest $\times$ Treatment interaction, sens ;
- the equal-weighted average treatment effect, $\delta + \text{sens} / 2$;
- the pretest effect among controls, `pretest_effect`.

Rounding and limits (digits, limits) make the scores look like test scores, but they shift the true values slightly, and more so when many scores reach a limit.

Choosing values. Willson and Putnam (1982) found an average pretest effect of about a fifth of a standard deviation in randomized studies; the article "Planning a Solomon Study" discusses planning values.

Value

A data frame with `y_post`, `treat`, `pretested`, and `y_pre` (missing by design in the unpretested groups), in the form the analysis functions take. `treat` is 0/1 for the four-group design and a factor of conditions for a design with several treatments. Its "truth" attribute is a data frame of the true estimands, and its "settings" attribute holds the arguments, with the group sizes named by group.

Designs with several treatments

A Solomon N-group design crosses k treatments and a control with pretesting, giving $2(k + 1)$ groups: six for two treatments and eight for three (Edmonds & Kennedy, 2017; Steyn, 2009). Give δ one effect per treatment, named by treatment, for example $\delta = c(A = 0.5, B = 0.2)$. Without names the treatments are called "T1", "T2", and so on; the control is "Control". sens is one number for every treatment or one per treatment, in the order of δ ; a named sens is matched to the treatments by name.

The model is the one above, with the effects $\delta[j]$ and $\text{sens}[j]$ of treatment j : each treatment differs from the control only by its own effects. n is one size for every group, or $2(k + 1)$ sizes in this order: the pretested treatments (in the order of δ), the pretested control, the unpretested treatments, and the unpretested control. Sizes named by group, as in the "settings" attribute (such as "Pretested, A"), or named n_1 , n_2 , and so on for the groups in that order, may be given in any order. Other names are refused. The groups are generated in the order above, and the baselines are drawn before the errors.

The `treat` column is then a factor whose levels are the treatments and "Control", so the data go directly to `fit_solomon_glm(y_post, treat, pretested, y_pre, control = "Control")`. The "truth" attribute has one row for each treatment-control comparison and contrast, with the columns comparison (such as "A vs Control"), contrast, and `true_value`, in the order of the effects table of that fit; a last row gives the pretest effect among controls. With one treatment effect the data and the "truth" attribute are those described above.

References

- Edmonds, W. A., & Kennedy, T. D. (2017). *An applied guide to research designs: Quantitative, qualitative, and mixed methods* (2nd ed.). SAGE Publications. <https://doi.org/10.4135/9781071802779>
- Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

Willson, V. L., & Putnam, R. R. (1982). A meta-analysis of pretest sensitization effects in experimental design. *American Educational Research Journal*, 19(2), 249–258. <https://doi.org/10.3102/00028312019002249>

See Also

`power_solomon()` and `plan_solomon()` for planning, `fit_solomon_glm()` for the recommended analysis, and the article "Teaching with solomonR".

Examples

```
d <- simulate_solomon(n = 40, delta = 0.5, sens = 0.4, rho = 0.6, seed = 1)
attr(d, "truth")
with(d, fit_solomon_glm(y_post, treat, pretested, y_pre))

# The bundled teaching data.
ex <- simulate_solomon(n = 30, delta = 5, pretest_effect = 2, rho = 0.6,
                      sigma = 10, mean = 50, digits = 0, limits = c(0, 100),
                      seed = 20260915)
attr(ex, "truth") <- attr(ex, "settings") <- NULL
identical(ex, solomon_example)

# A six-group design: two treatments and a control.
d6 <- simulate_solomon(n = 50, delta = c(A = 0.5, B = 0.2), sens = c(0.3, 0),
                      rho = 0.6, seed = 1)
attr(d6, "truth")
fit_solomon_glm(y_post, treat, pretested, y_pre, control = "Control", data = d6)
```

solomon_demo

Demo Solomon four-group data set (second example)

Description

A synthetic data set with 25 observations in each of the four combinations of treatment and pretest assignment. It is kept as a second example because two of its features make different analyses of the same effect disagree:

Format

A data frame with 100 rows and 4 variables:

y_post Numeric posttest score.

treat Treatment indicator: 0 = control, 1 = treatment.

pretested Pretest indicator: 0 = not pretested, 1 = pretested.

y_pre Numeric pretest score. Structurally missing for participants assigned to the unpretested groups.

Details

- among pretested participants, the pretest and posttest are negatively correlated ($r = -0.14$); and
- the pretested treatment group's mean pretest score is about 4.5 points higher than the pretested control group's.

As a result, unadjusted, ANCOVA, and gain-score estimates of the treatment effect among pretested participants differ markedly (see [compare_solomon_methods\(\)](#) and `vignette("getting-started")`). The parameters used to generate these data were not recorded, so their true effects are unknown. For the primary example with documented true values, see [solomon_example](#).

Source

Synthetic data generated for examples and package documentation.

See Also

[solomon_example](#)

solomon_effect_sizes *Solomon effect sizes for meta-analysis*

Description

[Stable] Computes standardized treatment effects for the pretested and unpretested pairs of a Solomon four-group study, with sampling variances in the y_i/v_i form that meta-analysis software reads.

Usage

```
solomon_effect_sizes(
  n,
  mean_post,
  sd_post,
  mean_pre = NULL,
  sd_pre = NULL,
  r = NULL
)
```

Arguments

<code>n</code>	Sample sizes of the four groups, in the order pretested treated, pretested control, unpretested treated, unpretested control.
<code>mean_post, sd_post</code>	Posttest means and standard deviations of the four groups.
<code>mean_pre, sd_pre</code>	Pretest means and standard deviations of the two pretested groups (treated, control). Optional; without them only the unpretested pair is returned.
<code>r</code>	Pre-post correlation in the pretested groups, needed with <code>mean_pre</code> and <code>sd_pre</code> .

Details

Pretested pair. The effect size is Morris's (2008) d_{ppc2} : the difference between the treated and control groups' mean pre-post change, divided by the pooled pretest standard deviation and multiplied by a small-sample bias correction (Eqs. 8-10). The correction here is the exact form (Eq. 22). Its sampling variance is Eq. 25, which needs the pre-post correlation r , assumed equal in the two groups. Morris (2008, p. 374) found that Eq. 25 was within 3% of the simulated variance in most conditions. When the treatment inflates posttest variance, however, it underestimated the true variance by 21% to 48% (p. 380), and a correlation that differs between the groups can make it less accurate still.

Unpretested pair. The effect size is Hedges's g for the two posttest groups, with the pooled posttest standard deviation. Its variance comes from the same noncentral t argument that Morris (2008, pp. 371-373) uses for Eq. 25, following Hedges (1981), without the pretest adjustment.

Variances are evaluated at the estimated effect size. The two rows use different standardizers (the pretest SD and the posttest SD), so their difference is not a clean measure of pretest sensitization.

Value

A data frame with one row per pair: the estimator, y_i (effect size), v_i (sampling variance), se_i (standard error), and the group sizes.

References

- Hedges, L. V. (1981). Distribution theory for Glass's estimator of effect size and related estimators. *Journal of Educational Statistics*, 6(2), 107–128. <https://doi.org/10.3102/10769986006002107>
- Morris, S. B. (2008). Estimating effect sizes from pretest-posttest-control group designs. *Organizational Research Methods*, 11(2), 364–386. <https://doi.org/10.1177/1094428106291059>

See Also

[solomon_from_summary\(\)](#)

Examples

```
# Pretested pair: the first study in Morris (2008, Table 1);
# unpretested pair: hypothetical.
solomon_effect_sizes(
  n = c(20, 20, 20, 20),
  mean_post = c(38.5, 19.7, 36.0, 25.0),
  sd_post = c(11.6, 14.8, 13.0, 14.0),
  mean_pre = c(30.6, 23.1),
  sd_pre = c(15.0, 13.8),
  r = 0.47
)
```

solomon_example

*Simulated Solomon four-group study (primary example)***Description**

A simulated randomized Solomon four-group study with 30 participants per group and scores on a 0-100 scale. Because the data were simulated, the true effects are known, so estimates can be compared with the truth. This is the main example in the package documentation; [solomon_demo](#) is a second example in which different analyses of the same effect disagree.

Format

A data frame with 120 rows and 4 variables:

y_post Posttest score (0-100).

treat Treatment indicator: 0 = control, 1 = treatment.

pretested Pretest indicator: 0 = not pretested, 1 = pretested.

y_pre Pretest score (0-100). Structurally missing for participants assigned to the unpretested groups.

Data-generating mechanism

Every participant has a latent baseline ability A drawn from a standard normal distribution. Participants in the pretested groups have a pretest score of $50 + 10A$. Every participant's posttest score is $50 + 5 \times \text{treat} + 2 \times \text{pretested} + 10(0.6A + 0.8e)$, where e is independent standard normal error. Scores are rounded to whole points and kept within 0-100.

The seed (20260915) and all parameters were fixed and posted in issue #21 before the data were generated, and the data were not regenerated to obtain particular results. The script is `data-raw/solomon_example.R` in the package repository.

True values and this sample

The treatment effect is 5 points among both pretested and unpretested participants, so there is no sensitization. The pretest effect is 2 points, the equal-weighted average treatment effect (ATE) is 5 points, and the pretest-posttest correlation is 0.6.

A single study can miss a real effect. With 30 participants per group, this design has approximately 85% power at $\alpha = .05$ for the ATE, 67% for the treatment effect among pretested participants, and 48% among unpretested participants. In this sample, `fit_solomon_glm()` estimates the ATE at about 2.7 points, with a 95% confidence interval that includes both 0 and the true value of 5. The sample pretest-posttest correlation is 0.61, and the two pretested groups have nearly identical mean pretest scores.

Source

Simulated for `solomonR`; see `data-raw/solomon_example.R` in the package repository.

See Also

[solomon_demo](#)

Examples

```
data(solomon_example)
with(solomon_example, validate_solomon(y_post, treat, pretested, y_pre))
with(solomon_example, fit_solomon_glm(y_post, treat, pretested, y_pre))
```

<i>solomon_from_summary</i>	<i>Solomon analysis from summary statistics</i>
-----------------------------	---

Description

[Stable] Reanalyzes a published Solomon four-group study from the sample size, mean, and standard deviation of the posttest in each of the four groups. With `treat` and `pretested` it also reanalyzes designs with several treatments (see "Designs with several treatments").

Usage

```
solomon_from_summary(
  n,
  mean,
  sd,
  conf_level = 0.95,
  treat = NULL,
  pretested = NULL,
  control = NULL
)
```

Arguments

<code>n</code> , <code>mean</code> , <code>sd</code>	Posttest sample size, mean, and standard deviation of the four groups, in the order above. With <code>treat</code> and <code>pretested</code> , one value for every group of the design, in any order.
<code>conf_level</code>	Confidence level for intervals. Default 0.95.
<code>treat</code>	Optional condition of each group, in the order of <code>n</code> : a character vector or factor, with the control named by <code>control</code> , or a 0/1 treatment indicator. Needed for a design with several treatments. The treatments are reported in the order of the factor's levels, or in the order they first appear in a character vector.
<code>pretested</code>	1 (or TRUE) for each pretested group and 0 for the others, in the order of <code>n</code> . Needed with <code>treat</code> .
<code>control</code>	The control condition, when <code>treat</code> is a character vector or factor.

Details

The four groups are given in the order used throughout the package: pretested treated (O2), pretested control (O4), unpretested treated (O5), and unpretested control (O6). The analysis is the 2 x 2 between-groups model on the posttest with a pooled error variance, so it reproduces the historical Tests A-D and the simple treatment effects (Tests B and C) of `fit_solomon_classic()`. Main effects are contrasts of unweighted cell means, which match the Type III sums of squares that statistical packages report for unbalanced cells.

Summary statistics limit the analysis. The pooled error variance assumes equal variances in the four groups; heteroskedasticity-consistent standard errors, covariate adjustment, and the analyses of the pretested groups (Tests E-G) need the individual data. Rounded published statistics reproduce published tests only to within rounding: for El Karkri et al. (2025a), the interaction F is 11.46 against the published 11.48.

Value

An object of class `solomon_summary_fit` with contrasts (Tests A-D, the pretest main effect, and the simple effects: estimate, standard error, t, degrees of freedom, p-value, confidence interval, F, and Type III sum of squares), an anova table, the cells, the error mean square and degrees of freedom, and the settings.

For a design with several treatments, an object of class `solomon_summary_ngroup` with contrasts (for each comparison of a treatment with the control and each of the four contrasts: estimate, standard error, t, p-value, Holm-adjusted p-value `p.adjusted`, degrees of freedom, and confidence interval), the omnibus tests in anova (Type III sum of squares, df, mean square, F, p-value), the cells in the package's group order, the conditions (control first), `adjust("holm")`, the error mean square and degrees of freedom, and the settings.

Designs with several treatments

A Solomon N-group design has k treatments and a control, each with and without a pretest: $2(k + 1)$ groups (Edmonds & Kennedy, 2017; Steyn, 2009). Give n , mean, and sd for every group, in any order, and say which group each value belongs to with `treat` (its condition) and `pretested` (1 for a pretested group), naming the control with `control`.

The analysis is the cell-means model of the posttest with a pooled error variance on $N - 2(k + 1)$ degrees of freedom. For each treatment against the control it gives the four Solomon contrasts with t tests and confidence intervals. They equal those of `fit_solomon_glm(y_post, treat, pretested, control = , robust = "none")` on the individual data, without the pretest as a covariate. The p-values of each contrast are adjusted across the k comparisons by Holm's (1979) procedure. The confidence intervals are not adjusted.

The omnibus F tests are those of the two-way ANOVA with Type III sums of squares: Condition (k df), Pretest (1 df), and Pretest x Condition (k df). Each is a Wald test of contrasts of the unweighted cell means, as in the four-group analysis. The result has class `solomon_summary_ngroup`. With two conditions, the result is the four-group analysis above.

References

Edmonds, W. A., & Kennedy, T. D. (2017). *An applied guide to research designs: Quantitative, qualitative, and mixed methods* (2nd ed.). SAGE Publications. <https://doi.org/10.4135/9781071802779>

- El Karkri, M., Quesada, A., & Romero-Ariza, M. (2025a). The dual impact of pretest sensitisation and the cognitive acceleration through science education programme in the Solomon four-group design. *Brain Sciences*, 16(1), Article 64. <https://doi.org/10.3390/brainsci16010064>
- Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2), 65–70. <https://www.jstor.org/stable/4615733>
- Mai, N. N., Takahashi, Y., & Oo, M. M. (2020). Testing the effectiveness of transfer interventions using Solomon four-group designs. *Education Sciences*, 10(4), Article 92. <https://doi.org/10.3390/educsci10040092>
- Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>
- Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

See Also

[solomon_effect_sizes\(\)](#) for effect sizes for meta-analysis, and [fit_solomon_glm\(\)](#) for the analysis of the individual data.

Examples

```
# El Karkri et al. (2025a), Table 8
solomon_from_summary(
  n = c(9, 25, 17, 37),
  mean = c(10.94, 7.80, 8.94, 9.35),
  sd = c(2.26, 2.29, 1.98, 2.11)
)

# A six-group design: two treatments and a control. Posttest statistics
# computed from the data of Mai et al. (2020); see mai2020.
solomon_from_summary(
  n = c(24, 23, 27, 22, 15, 22),
  mean = c(2.929167, 3.168116, 3.112346, 3.128788, 3.152184, 3.018548),
  sd = c(0.434203, 0.369613, 0.355440, 0.383150, 0.374069, 0.354758),
  treat = c("RP", "GS", "Control", "RP", "GS", "Control"),
  pretested = c(1, 1, 1, 0, 0, 0),
  control = "Control"
)
```

solomon1949

Solomon's (1949) spelling experiment

Description

Group sizes, means, and standard errors from the first published experiment with the Solomon design: spelling lessons in a fifth-grade and a sixth-grade class, analyzed with the three-group design (Solomon, 1949, Tables II and III, pp. 144–145). Numbers reported in the publication are reused with citation.

Usage

solomon1949

Format

A data frame with 6 rows, one per group and grade, and 11 variables:

grade 5 or 6.

group Experimental, Control I, or Control II.

pretested, treat Indicators (1 = yes); treat is the spelling lesson.

n Group size.

pre_mean, pre_se Pretest mean and its standard error (pretested groups only).

mean, se Posttest mean and its standard error.

change, change_se Mean improvement, as printed, and its standard error (missing for Control II, printed as "?").

The standard errors are the values printed with a plus-or-minus sign, which Solomon's notation (sigma_m, Table I) identifies as standard errors of the means.

Details

Design. In each class, pupils were assigned to three groups that were "roughly equated in spelling ability by means of teachers' judgments" (p. 144), not randomized.

- **Experimental group.** Pretested on a list of words, given a standard spelling lesson, and posttested on the same words.
- **Control Group I.** Pretested and posttested, without the lesson.
- **Control Group II.** Given the lesson and the posttest, without the pretest.

There was no fourth group; Solomon introduced it later in the article for field studies (p. 147).

Known result. `fit_solomon_1949()` reproduces the published values from these means: an inferred pretest of 3.0 and 5.7, improvements for Control Group II of 8.2 and 8.7, and interactions I = -2.2 (grade 5) and -3.1 (grade 6). The pretest reduced the effect of the lesson, so the usual two-group design would have underrated it (p. 145).

Caveats. The groups are small (8 to 10 pupils) and were not randomized. Solomon printed the standard error of Control Group II's improvement as "?", since it depends on the inferred pretest.

Source

Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>

See Also

`fit_solomon_1949()`, `elkarkri2025a`, `mai2020`

Examples

```
solomon1949
g6 <- solomon1949[solomon1949$grade == 6, ]
fit_solomon_1949(post_mean = g6$mean, pre_mean = g6$pre_mean[1:2], n = g6$n)
```

solomonR

solomonR: Analyze Solomon Four-Group Designs

Description

Provides tools for analyzing, teaching, and studying the Solomon four-group experimental design. The package includes the historical Tests A-I workflow, unified generalized linear models with robust covariance estimation, stratified randomization inference, Solomon-specific full-information maximum likelihood, and observed- and latent-variable structural equation models.

Details

The package distinguishes historically important procedures from contemporary recommendations and emphasizes explicit Solomon-specific estimands, structural pretest missingness, reproducible diagnostics, and modern inference.

Key historical references include Huck & Sandler (1973), Walton Braver & Braver (1988), the Sawilowsky and Markman methodological exchanges, and van Engelenburg (1999).

Lifecycle

Each exported function's help page, and the reference index of the package website, shows the function's lifecycle stage, in the stages of the lifecycle package (Henry & Wickham, 2026):

- **Stable.** The interface is settled. Any change goes through deprecation: the old name keeps working, with a warning, through v1.x, when the arguments that follow a former argument name are also named.
- **Experimental.** The function is tested, but its interface or defaults may change. `fit_solomon_sem()` has no simulation study yet at Solomon sample sizes. The issue #55 study found no measurement-invariance criterion that holds its false-rejection rate in Solomon-sized groups, which affects `fit_solomon_sem_latent()` and `invariance_solomon()`. The analysis of designs with several treatments in `fit_solomon_glm()` is experimental, because the rule for error control set before its simulation study (issue #45) was not met for two omnibus tests with three treatments and 10 participants per group. The pre-specified rules of their simulation studies were not met for `fit_solomon_mmrn()` (issue #57) and for `fit_solomon_mi()`, `tipping_point_solomon()`, and `plot_tipping_point()` (issue #82). `analysis_plan_solomon()` stays experimental until researchers have used it (issue #81).
- **Deprecated.** A former name that still works, with a warning. Its help page names the replacement.

solomonR 0.8.0 settled the interface (issue #83). The outcome arguments are `y_post` and `y_pre` everywhere, a fitted model is passed as `fit`, and the planning functions share one argument order: `n` (or `power`), `delta`, `sens`, `rho`, `sigma`, `alpha`. Each function that takes data vectors also takes an optional data data frame.

References

Campbell, D. T., & Stanley, J. C. (1966). *Experimental and quasi-experimental designs for research*. Rand McNally. (Original work published 1963)

Henry, L., & Wickham, H. (2026). *lifecycle: Manage the life cycle of your package functions* (Version 1.0.5) [R package]. <https://doi.org/10.32614/CRAN.package.lifecycle>

Huck, S. W., & Sandler, H. M. (1973). A note on the Solomon 4-group design: Appropriate statistical analyses. *The Journal of Experimental Education*, 42(2), 54–55. <https://doi.org/10.1080/00220973.1973.11011460>

Sawilowsky, S. S., Kelley, D. L., Blair, R. C., & Markman, B. S. (1994). Meta-analysis and the Solomon four-group design. *The Journal of Experimental Education*, 62(4), 361–376. <https://doi.org/10.1080/00220973.1994.11011460>

Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>

Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

van Engelenburg, G. (1999). *Statistical analysis for the Solomon four-group design* (Research Report 99-06). University of Twente. ERIC. <https://eric.ed.gov/?id=ED435692>

Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

steyn2005	Group statistics from Steyn’s (2005) eight-group study
-----------	--

Description

The sample size, pretest and posttest means, and standard deviations of the eight groups of a Solomon design with three treatments: a study of how information about one’s own ability changes self-efficacy perceptions, with 1,723 police trainees (Steyn, 2005, Table 5.59, pp. 151–152). It is the eight-group study that Steyn (2009) describes, and Steyn and Mynhardt (2008) report it in English. Numbers reported in the publication are reused with citation.

Usage

steyn2005

Format

- A data frame with 8 rows, one per group, and 8 variables:
- group** The group’s label in the thesis: EG1 to EG3 are the pretested treatment groups, KG1.1 to KG1.3 the unpretested treatment groups, KG2 the pretested control, and KG3 the unpretested control.
 - condition** Norms, Marking, Test, or Control.
 - pretested** 1 if the group was pretested, 0 if not.

n Sample size.

pre_mean, pre_sd Pretest mean and standard deviation (pretested groups only).

mean, sd Posttest mean and standard deviation.

Details

Design. Three treatments and a control, each with and without a pretest (pp. 103–105). Each treatment added a source of information about the participant's ability (p. 102):

- **Test.** Completing a 60-item cognitive test.
- **Marking.** Completing the test and marking one's own answers.
- **Norms.** Completing and marking the test, and receiving the test's norms.

The outcome is the total score on a 45-item questionnaire of self-efficacy perceptions (p. 94). The posttest followed the treatment after a break of 10 minutes (p. 107).

Assignment. Participants were not randomized individually. Fifty-six existing classes of a police training college were allocated to the eight groups, seven classes to each, in consultation with the college's management, and the thesis states that this allocation was not random (pp. 105–106). The classes had been formed from the order in which trainees reported for training. The thesis is not consistent on this point. Its introduction says that participants were divided at random into eight groups and that the groups were assigned at random to the conditions (p. 9), and it calls the allocation random again on p. 107. So does the English report, which does not mention the classes (Steyn & Mynhardt, 2008, pp. 566–567). The method chapter, followed here, gives the detail: the researcher judged the existing classes to be random groups and made no further random assignment (pp. 105–106), and balanced the groups by sex: each held five men's classes and two women's classes (Steyn, 2005, p. 106). The analyses in the thesis, and those below, treat participants as the units, so they do not allow for the classes. See [baseline_solomon\(\)](#) for nonrandomized designs and [validate_solomon\(\)](#) for clustered ones.

Known results. These statistics reproduce the analyses in the thesis:

- **One-way ANOVA of the eight posttest groups** (Table 5.60, p. 152): $F(7, 1715) = 4.545$.
- **Scheffé's (1953) tests** (Table 5.61, p. 153), within .001. Only the unpretested Test group differs from the two control groups ($p = .002$ and $p = .011$).
- **The 2 x 2 ANOVAs of each treatment against the control** (Tables 5.21, 5.34, and 5.47; pp. 128, 135, 142; Steyn & Mynhardt, 2008, Table 2, p. 569), within rounding. For Test: intervention $F = 21.3$, pretest $F = 4.5$ ($p = .033$), and interaction $F = 2.0$ ($p = .152$). For Marking: 9.3 ($p = .002$), 0.4 ($p = .538$), and 0.0 ($p = .929$). For Norms, the thesis ran this analysis with 213 participants in the unpretested group (Table 5.45, pp. 141–142), one fewer than in Table 5.59, so its error degrees of freedom are 854 and those computed from these statistics are 855. The F statistics for Norms (14.0, 0.9, and 0.1) are the same to one decimal.

Steyn and Mynhardt (2008) report only these three analyses, judged at the .01 level (p. 568), so they read the pretest effect for Test ($p = .033$) as no effect (p. 568); the thesis calls it significant at .05 (p. 128). Their means of the treated and untreated groups reproduce as unweighted averages of the two groups' means. Their effect size d divides the difference by what they call the mean standard deviation (p. 568): 13.00, 12.696, and 12.803 (Table 2, p. 569), the "Totaal" rows of the thesis (Tables 5.19, 5.32, and 5.45; pp. 127, 134, 142). For Norms, their means and that standard deviation follow the 213 participants of Table 5.45, not the 214 of Table 5.59 and of their own Table

1 (p. 567). Their text gives the Marking pretest p as .583 (p. 568); their Table 2 and these statistics give .538.

The treatments lowered the scores. The thesis analyzed the design as overlapping four-group designs, one with the treatments pooled and one for each treatment, and then as a one-way analysis of variance of the eight posttests (pp. 103–105). The joint model of `solomon_from_summary()` tests the Pretest $\times$ Condition interaction once, $F(3, 1715) = 1.00$, $p = .392$.

Language. The thesis is in Afrikaans. The condition labels are `solomonR`'s.

Source

Steyn, R. (2005). *Self-evaluasie en die vorming van selfdoeltreffendheidspersepsies* [Self-evaluation and the forming of self-efficacy perceptions] [Doctoral thesis, University of South Africa]. Unisa Institutional Repository. <https://hdl.handle.net/10500/1745>

References

Scheffé, H. (1953). A method for judging all contrasts in the analysis of variance. *Biometrika*, 40(1–2), 87–104. <https://doi.org/10.1093/biomet/40.1-2.87>

Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

Steyn, R., & Mynhardt, J. (2008). Factors that influence the forming of self-evaluation and self-efficacy perceptions. *South African Journal of Psychology*, 38(3), 563–573. <https://doi.org/10.1177/008124630803800310>

See Also

[solomon_from_summary\(\)](#), [fit_solomon_steyn\(\)](#), [mai2020](#), [lana1959](#)

Examples

```
steyn2005

# One model for all eight groups.
with(steyn2005, solomon_from_summary(n, mean, sd, treat = condition,
                                     pretested = pretested, control = "Control"))
```

stouffer_solomon

Stouffer's Z combiner (Walton Braver & Braver, 1988, Test I)

Description

[Stable] Combine one-tailed p -values that test the *same directional* hypothesis into a single Z . This is provided to reproduce the Walton Braver & Braver (1988) meta-analytic option (Test I) for the Solomon four-group design. Use cautiously and document assumptions about homogeneity; see the 1988–1990 exchanges for caveats.

Usage

```
stouffer_solomon(p)
```

Arguments

`p` numeric vector of one-tailed p-values (same direction)

Details

Walton Braver and Braver (1988, p. 152) convert the one-tailed p-value of each test to z and refer the combined z to a normal table. Their worked example reports $z = 2.05$ with $p = .040$ (p. 153), the two-tailed value, so `fit_solomon_classic()` judges Test I by `p_meta_two_tailed`. The one-tailed value is also returned.

Value

list with `z_meta`, `p_meta_two_tailed`, and `p_meta_one_tailed`

References

Sawilowsky, S. S., Kelley, D. L., Blair, R. C., & Markman, B. S. (1994). Meta-analysis and the Solomon four-group design. *The Journal of Experimental Education*, 62(4), 361–376. <https://doi.org/10.1080/00220973.1994.10800000>

Stouffer, S. A., Suchman, E. A., DeVinney, L. C., Star, S. A., & Williams, R. M., Jr. (1949). *The American soldier: Adjustment during army life* (Vol. 1). Princeton University Press.

Walton Braver, M. C., & Braver, S. L. (1988). Statistical treatment of the Solomon four-group design: A meta-analytic approach. *Psychological Bulletin*, 104(1), 150–154. <https://doi.org/10.1037/0033-2909.104.1.150>

Examples

```
stouffer_solomon(c(0.10, 0.11))

# Walton Braver and Braver's (1988, p. 153) example: the ANCOVA (p = .0993)
# and the t test (p = .2127), halved to one-tailed p-values in the
# direction of the effect, give z = 2.05 and p = .040.
stouffer_solomon(c(0.0993, 0.2127) / 2)
```

tipping_point_solomon *Tipping-point analysis for missing posttests*

Description

[Experimental] Repeats `fit_solomon_mi()` over a range of offsets added to the imputed posttests of chosen Solomon groups, and reports the smallest offset in each direction at which the conclusion about a contrast changes. White et al. (2011, "Perform Sensitivity Analyses" section, para. 1) suggest reporting "how large an amount should be added to or subtracted from imputed outcomes" without changing the interpretation, and Little et al. (2012, p. 1358) call a finding robust if it holds over the plausible offsets.

Usage

```
tipping_point_solomon(
  y_post,
  treat,
  pretested,
  y_pre = NULL,
  contrast = "ATE (avg over pretest)",
  groups = "all",
  deltas = NULL,
  alpha = 0.05,
  m = 100,
  robust = c("HC3", "none"),
  seed = NULL,
  data = NULL
)
```

Arguments

y_post	Numeric posttest scores, with NA for missing posttests.
treat	Treatment indicator coded 0/1 (or logical). Designs with several treatments are not supported; see fit_solomon_glm() .
pretested	Pretest indicator coded 0/1 (or logical).
y_pre	Optional numeric pretest scores, missing by design for unpretested participants.
contrast	The Solomon contrast to follow. Default is the average treatment effect.
groups	The groups whose imputed posttests are shifted: "all" (default), "treatment", "control", "pretested", "unpretested", or group numbers (1 pretested treatment, 2 pretested control, 3 unpretested treatment, 4 unpretested control).
deltas	Offsets to try, on the posttest scale. The default is 21 values from -1 to 1 pooled within-group standard deviations of the observed posttests. Zero is always included.
alpha	Significance level that defines the conclusion. Default 0.05.
m	Number of imputations for each offset. Default 100.
robust	Covariance for each completed-data analysis: "HC3" (default) or "none"; see fit_solomon_glm() .
seed	Optional random-number seed. When NULL, one is drawn so that every offset uses the same imputations.
data	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (y_post = post) or as strings (y_post = "post").

Details

Which groups. Offsets confined to some groups test different departures from missing at random. Offsets in the treatment groups (groups = "treatment") bear on the treatment effect. Offsets that differ between the pretested and unpretested groups, for example in the pretested treatment group alone (groups = 1), bear on the sensitization contrast.

Common random numbers. Every offset uses the same imputation draws, so the estimates change smoothly with the offset. The location of the tipping point still carries Monte Carlo error from the imputations: in the worked example on `mai2020`, the tipping point for sensitization ranged from 0.1 to 0.5 standard deviations across seeds with 100 imputations and was 0.3 with 2,000. Before reporting a tipping point, increase `m` or compare a few seeds.

Value

An object of class `solomon_tipping` with results (one row per offset: the offset in posttest units and in standard deviations, and the pooled estimate, interval, and p-value), `tipping` (the smallest negative and positive offsets at which the conclusion differs from the one under MAR, NA when it does not change within the range), and the settings used.

Lifecycle

Experimental, with `fit_solomon_mi()`, whose validation study it shares.

References

Little, R. J., D’Agostino, R., Cohen, M. L., Dickersin, K., Emerson, S. S., Farrar, J. T., Frangakis, C., Hogan, J. W., Molenberghs, G., Murphy, S. A., Neaton, J. D., Rotnitzky, A., Scharfstein, D., Shih, W. J., Siegel, J. P., & Stern, H. (2012). The prevention and treatment of missing data in clinical trials. *The New England Journal of Medicine*, 367(14), 1355–1360. <https://doi.org/10.1056/NEJMSr1203730>

White, I. R., Horton, N. J., Carpenter, J., & Pocock, S. J. (2011). Strategy for intention to treat analysis in randomised trials with missing outcome data. *BMJ*, 342, Article d40. <https://doi.org/10.1136/bmj.d40>

See Also

`fit_solomon_mi()`, `plot_tipping_point()`

Examples

```
d <- solomon_example
set.seed(82)
d$y_post[sample(nrow(d), 20)] <- NA
tipping_point_solomon(y_post, treat, pretested, y_pre, groups = "treatment",
  deltas = seq(-10, 10, by = 2.5), m = 10, seed = 1, data = d)
```

validate_solomon

Validate the structure and coding of a Solomon design

Description

[Stable] Checks that data can support a Solomon analysis before a model is fitted: equal input lengths, the coding of the design indicators, all cells present, enough observed outcomes per cell, and the distinction between structurally absent and incidentally missing values (see `check_solomon_missing()`). Problems are returned as a table of issues rather than stopping at the first one, so every problem is reported at once. Designs with several treatments are checked too (see `control`).

Usage

```
validate_solomon(
  y_post,
  treat,
  pretested,
  y_pre = NULL,
  min_cell_n = 2,
  cluster = NULL,
  control = NULL,
  data = NULL
)
```

Arguments

<code>y_post</code>	Numeric posttest scores.
<code>treat</code>	Treatment indicator coded 0/1 (or logical); or a factor or character vector of conditions, with the control named by <code>control</code> .
<code>pretested</code>	Pretest indicator coded 0/1 (or logical).
<code>y_pre</code>	Optional numeric pretest scores, missing by design for unpretested participants.
<code>min_cell_n</code>	Minimum number of observed posttest scores required in each cell (at least 2).
<code>cluster</code>	Optional cluster identifier (for example, class, school, or site), one value per participant.
<code>control</code>	The control condition when <code>treat</code> is a factor or character vector with more than two conditions, a Solomon N-group design; see fit_solomon_glm() . With two conditions the result is the same as with a 0/1 <code>treat</code> .
<code>data</code>	Optional data frame. When supplied, the other data arguments are looked up in it first, as bare column names (<code>y_post = post</code>) or as strings (<code>y_post = "post"</code>).

Details

Accepted coding: `pretested` must be numeric 0/1 or logical, and so must `treat` unless the control condition is named with `control`. Without `control`, factors and character codes are rejected so that group membership is never inferred from level order. The Solomon four-group design requires all four cells (Solomon, 1949): pretested treatment, pretested control, unpretested treatment, and unpretested control.

Designs with several treatments: give `treat` as a factor or character vector of conditions and name the control with `control`. A design with k treatments requires all $2(k + 1)$ cells, each treatment and the control with and without a pretest: six cells for two treatments and eight for three (Steyn, 2009). Every check below is then made for each of these cells. Levels of a factor `treat` that no participant has are left out of the design, with a note.

Severity:

- **error:** the data cannot support a Solomon analysis as supplied (unequal lengths, invalid coding, an empty cell, fewer than `min_cell_n` observed posttest scores in a cell, or no observed pretests among pretested participants when `y_pre` is supplied).

- **warning:** analyses can run, but some participants are excluded or values are inconsistent with the design (unassigned participants, incidental missingness, pretest scores recorded for unpretested participants).
- **note:** information for reporting, such as the range of cell sizes.

`min_cell_n` is a technical minimum, not a sample-size recommendation: at least two observations per cell are needed to estimate within-cell variability, and HC3 standard errors can be undefined for a cell with a single observation. Plan cell sizes with a power analysis.

Clustered designs: when participants belong to classes, schools, or sites, supply `cluster`. The number of clusters and the range of cluster sizes are then reported for each cell, together with whether treatment and pretesting were assigned to whole clusters or varied within them.

A cell that contains a single cluster is an error. Class and condition are then completely confounded, as when each Solomon condition is one intact class (El Karkri et al., 2025a): a difference between the classes cannot be separated from the treatment, pretest, or sensitization effect, and no between-cluster variability can be estimated. Cluster-robust inference (Pustejovsky & Tipton, 2018) needs several clusters in every cell. For example, Kvalem et al. (1996) randomized 124 school classes to the four Solomon conditions. When whole clusters are randomized, a cell with two or three clusters is a warning: Hayes and Moulton (2017, p. 128) regard four clusters per arm as an absolute minimum.

Value

An object of class `solomon_validation` with `valid` (TRUE when no errors were found), `issues` (severity, check, and message), `cells` (counts by cell, with clusters and cluster sizes when `cluster` is supplied), and `missing` (the `check_solomon_missing()` result). For a design with several treatments, `cells` has one row for each of the $2(k + 1)$ cells and `conditions` names the control and the treatments.

References

- El Karkri, M., Quesada, A., & Romero-Ariza, M. (2025a). The dual impact of pretest sensitisation and the cognitive acceleration through science education programme in the Solomon four-group design. *Brain Sciences*, 16(1), Article 64. <https://doi.org/10.3390/brainsci16010064>
- Hayes, R. J., & Moulton, L. H. (2017). *Cluster randomised trials* (2nd ed.). Chapman and Hall/CRC. <https://doi.org/10.4324/9781315370286>
- Kvalem, I. L., Sundet, J. M., Rivø, K. I., Eilertsen, D. E., & Bakketeig, L. S. (1996). The effect of sex education on adolescents' use of condoms: Applying the Solomon four-group design. *Health Education Quarterly*, 23(1), 34–47. <https://doi.org/10.1177/109019819602300103>
- Mai, N. N., Takahashi, Y., & Oo, M. M. (2020). Testing the effectiveness of transfer interventions using Solomon four-group designs. *Education Sciences*, 10(4), Article 92. <https://doi.org/10.3390/educsci10040092>
- Pustejovsky, J. E., & Tipton, E. (2018). Small-sample methods for cluster-robust variance estimation and hypothesis testing in fixed effects models. *Journal of Business & Economic Statistics*, 36(4), 672–683. <https://doi.org/10.1080/07350015.2016.1247004>
- Solomon, R. L. (1949). An extension of control group design. *Psychological Bulletin*, 46(2), 137–150. <https://doi.org/10.1037/h0062958>

Steyn, R. (2009). Re-designing the Solomon four-group: Can we improve on this exemplary model? *Design Principles and Practices: An International Journal—Annual Review*, 3(1), 383–394. <https://doi.org/10.18848/1833-1874/CGP/v03i01/37588>

See Also

[check_solomon_missing\(\)](#)

Examples

```
data(solomon_example)

# A valid design.
with(solomon_example, validate_solomon(y_post, treat, pretested, y_pre))

# An empty cell makes the design invalid.
no_group_3 <- solomon_example[!(solomon_example$pretested == 0 & solomon_example$treat == 1), ]
with(no_group_3, validate_solomon(y_post, treat, pretested, y_pre))

# One intact class per condition confounds class with condition.
one_class <- with(solomon_example, 2 * pretested + treat)
with(solomon_example, validate_solomon(y_post, treat, pretested, y_pre,
                                       cluster = one_class))

# A six-group design: two treatments and a control (Mai et al., 2020).
validate_solomon(post_behavior, condition, pretested, pre_behavior,
                 control = "Control", data = mai2020)
```

Index

- * **ANCOVA**
 - solomonR, [97](#)
- * **SEM**
 - solomonR, [97](#)
- * **Solomon**
 - solomonR, [97](#)
- * **datasets**
 - elkarkri2025a, [14](#)
 - jordaan2014, [53](#)
 - kvalem1996, [55](#)
 - lana1959, [56](#)
 - mai2020, [57](#)
 - solomon1949, [95](#)
 - solomon_demo, [89](#)
 - solomon_example, [92](#)
 - steyn2005, [98](#)
- * **four-group**
 - solomonR, [97](#)
- * **maximum-likelihood**
 - solomonR, [97](#)
- * **permutation**
 - solomonR, [97](#)
- analysis_plan_solomon, [3](#)
- analysis_plan_solomon(), [68](#), [69](#), [84](#), [97](#)
- baseline_solomon, [6](#)
- baseline_solomon(), [15](#), [59](#), [85](#), [99](#)
- check_solomon_assumptions, [8](#)
- check_solomon_missing, [10](#)
- check_solomon_missing(), [33](#), [34](#), [47](#), [50](#), [76](#), [103](#), [105](#), [106](#)
- compare_solomon_methods, [12](#)
- compare_solomon_methods(), [76](#), [90](#)
- elkarkri2025a, [14](#), [55–57](#), [59](#), [96](#)
- equivalence_solomon, [16](#)
- equivalence_solomon(), [72](#), [73](#), [78](#), [85](#)
- fisher_solomon, [18](#)
- fisher_solomon(), [55](#), [62](#), [85](#)
- fit_solomon_1949, [20](#)
- fit_solomon_1949(), [96](#)
- fit_solomon_classic, [22](#)
- fit_solomon_classic(), [14](#), [21](#), [22](#), [46](#), [47](#), [50](#), [54](#), [69](#), [83](#), [85](#), [94](#), [101](#)
- fit_solomon_glm, [26](#)
- fit_solomon_glm(), [5](#), [7](#), [9](#), [10](#), [13](#), [14](#), [16](#), [17](#), [19](#), [20](#), [22](#), [23](#), [32](#), [33](#), [35](#), [37–42](#), [45](#), [49–51](#), [58–62](#), [64](#), [65](#), [72](#), [73](#), [77–79](#), [85](#), [89](#), [92](#), [95](#), [97](#), [102](#), [104](#)
- fit_solomon_mi, [32](#)
- fit_solomon_mi(), [85](#), [97](#), [101](#), [103](#)
- fit_solomon_ml, [35](#)
- fit_solomon_ml(), [14](#), [16](#), [17](#), [72](#), [73](#), [78](#), [85](#)
- fit_solomon_mmrn, [37](#)
- fit_solomon_mmrn(), [4](#), [54](#), [85](#), [97](#)
- fit_solomon_sem, [40](#)
- fit_solomon_sem(), [14](#), [78](#), [85](#), [97](#)
- fit_solomon_sem_latent, [42](#)
- fit_solomon_sem_latent(), [52](#), [78](#), [85](#), [97](#)
- fit_solomon_steyn, [45](#)
- fit_solomon_steyn(), [30](#), [85](#), [100](#)
- invariance_solomon, [50](#)
- invariance_solomon(), [43](#), [44](#), [97](#)
- jordaan2014, [39](#), [53](#)
- kvalem1996, [15](#), [55](#), [59](#)
- lana1959, [56](#), [100](#)
- mai2020, [57](#), [96](#), [100](#)
- marginal_solomon, [59](#)
- marginal_solomon(), [19](#), [28](#), [29](#), [55](#), [84](#), [85](#)
- p_to_z, [63](#)
- perm_solomon, [63](#)
- perm_solomon(), [28](#), [55](#), [61](#), [70](#), [85](#)
- plan_solomon, [66](#)

`plan_solomon()`, [3](#), [4](#), [6](#), [71](#), [72](#), [84](#), [89](#)
`plot_classic_flow`, [69](#)
`plot_perm`, [70](#)
`plot_power_solomon`, [71](#)
`plot_power_solomon()`, [84](#)
`plot_sensitization`, [72](#)
`plot_sensitization()`, [76](#), [80](#), [81](#)
`plot_solomon`, [74](#)
`plot_solomon_change`, [75](#)
`plot_solomon_change()`, [80](#), [81](#)
`plot_solomon_design`, [76](#)
`plot_solomon_effects`, [78](#)
`plot_solomon_effects()`, [73](#)
`plot_solomon_gg`, [79](#)
`plot_solomon_means`, [80](#)
`plot_solomon_means()`, [74](#), [79](#)
`plot_tipping_point`, [81](#)
`plot_tipping_point()`, [97](#), [103](#)
`power_solomon`, [82](#)
`power_solomon()`, [66–69](#), [71](#), [72](#), [87](#), [89](#)

`report_solomon`, [84](#)
`report_solomon()`, [5](#), [6](#), [8](#), [50](#)

`simulate_solomon`, [86](#)
`solomon1949`, [22](#), [57](#), [95](#)
`solomon_demo`, [89](#), [92](#), [93](#)
`solomon_effect_sizes`, [90](#)
`solomon_effect_sizes()`, [95](#)
`solomon_example`, [87](#), [90](#), [92](#)
`solomon_from_summary`, [93](#)
`solomon_from_summary()`, [15](#), [54](#), [56](#), [57](#), [85](#),
[91](#), [100](#)
`solomonR`, [97](#)
`stats::pairwise.t.test()`, [48](#)
`steyn2005`, [50](#), [54](#), [98](#)
`stouffer_solomon`, [100](#)

`tipping_point_solomon`, [101](#)
`tipping_point_solomon()`, [4](#), [32](#), [34](#), [81](#), [82](#),
[85](#), [97](#)

`validate_solomon`, [103](#)
`validate_solomon()`, [12](#), [15](#), [27](#), [77](#), [99](#)