Package ‘sanon’

May 9, 2026

Type Package

Title Stratified Analysis with Nonparametric Covariable Adjustment

Version 1.6

Date 2020-05-27

Author Atsushi Kawaguchi

Maintainer Atsushi Kawaguchi `<kawa_a24@yahoo.co.jp>`

Depends R (>= 3.5)

Description There are several functions to implement the method for analysis in a randomized clinical trial with strata with following key features. A stratified Mann-Whitney estimator addresses the comparison between two randomized groups for a strictly ordinal response variable. The multivariate vector of such stratified Mann-Whitney estimators for multivariate response variables can be considered for one or more response variables such as in repeated measurements and these can have missing completely at random (MCAR) data. Non-parametric covariance adjustment is also considered with the minimal assumption of randomization. The p-value for hypothesis test and confidence interval are provided.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2020-05-27 14:20:03 UTC

Contents

`sanon-package` *The Stratified Analysis with Nonparametric covariable adjustment Package*

Description

A Package for Implementation of the method in Kawaguchi, Koch, and Wang (2011)

Author(s)

Atsushi Kawaguchi. `<kawa_a24@yahoo.co.jp>`

References

Kawaguchi A., Koch, G. G. (2015). sanon: An R Package for Stratified Analysis with Nonparametric Covariable Adjustment. Journal of Statistical Software, 67(9), 1-37. doi:10.18637/jss.v067.i09

Kawaguchi A., Koch, G. G., Wang, X. (2011). Stratified Multivariate Mann-Whitney Estimators for the Comparison of Two Treatments with Randomization Based Covariance Adjustment. Statistics in Biopharmaceutical Research, Vol. 3, No. 2, 217-231.

See Also

`sanon`

`catecovar` *Identify Categorical Covariables*

Description

This is a special function used in the context of `sanon`. It identifies categorical covariables when they appear on the right hand side of a formula.

Usage

```
catecovar(x, ref = NULL)
```

Arguments

`x`	variable name
`ref`	character for the reference group for the categorical covariable.

Details

In the `sanon`, the categorical covariable is converted into a dummy variable. The reference group is specified in the `ref` argument.

`coef.sanon` *Extract Model Coefficients*

Description

coef is a generic function which extracts model coefficients from objects returned by modeling functions. coefficients is an alias for it.

Usage

```
## S3 method for class 'sanon'
coef(object, ...)
```

Arguments

`object`	an object of class "`sanon`", usually, a result of a call to `sanon`
`...`	further arguments passed to or from other methods.

Details

All object classes which are returned by model fitting functions should provide a coef method or use the default one.

Value

Coefficients extracted from the model object object.

Examples

```
##### Example 3.1 Randomized Clinical Trial of Chronic Pain #####
data(cpain)
out1 = sanon(response ~ grp(treat, ref="placebo") + strt(center) + strt(diagnosis), data=cpain)
coef(out1)
coefficients(out1)

##### Example 3.2 Randomized Clinical Trial of Respiratory Disorder #####
data(resp)
P = rbind(rep(0, 4), diag(4), rep(0, 4))
out23 = sanon(cbind(baseline, visit1, visit2, visit3, visit4) ~ grp(treatment, ref="P")
```

```
 + strt(center) + strt(sex) + covar(age), data=resp, P=P)
# each four visits
coef(out23)
coefficients(out23)
```

`confint.sanon` *Confidence Intervals for Model Parameters*

Description

Computes confidence intervals for one or more parameters in a fitted model.

Usage

```
## S3 method for class 'sanon'
confint(object, parm = NULL, level = 0.95, ...)

## S3 method for class 'confint.sanon'
print(x, ...)
```

Arguments

`object`, `x`	an object of class "`sanon`", usually, a result of a call to `sanon`
`parm`	a specification of which parameters are to be given confidence intervals, either a vector of numbers or a vector of names. If missing, all parameters are considered.
`level`	the confidence level required.
`...`	further arguments passed to or from other methods.

Details

Confidence intervals for adjusted parameters in the weighted least squares are computed based on an asymptotic normal.

Value

`ci`	A matrix (or vector) with columns giving Mann-Whiteney estimates and their lower and upper confidence limits for each parameter with estimates. The interval will be labelled as Lower for (1 - level)/2 limit and Upper for 1 - (1 - level)/2 limit (by default 0.025 and 0.975).
`level`	Confidence level
`advarnames`	Adjust variable names in the weighted least squares method

Examples

```
##### Example 3.1 Randomized Clinical Trial of Chronic Pain #####
data(cpain)
out1 = sanon(response ~ grp(treat, ref="placebo") + strt(center) + strt(diagnosis), data=cpain)
confint(out1)

##### Example 3.2 Randomized Clinical Trial of Respiratory Disorder #####
data(resp)
P = rbind(rep(0, 4), diag(4), rep(0, 4))
out23 = sanon(cbind(baseline, visit1, visit2, visit3, visit4) ~ grp(treatment, ref="P")
 + strt(center) + strt(sex) + covar(age), data=resp, P=P)
# each four visits
confint(out23)
```

`contrast` *Contrast for Model Parameters*

Description

Inference by contrast of parameters in a fitted model.

Usage

```
contrast(
  object,
  C = diag(length(object$b)),
  confint = FALSE,
  level = 0.95,
  ...
)

## S3 method for class 'contrast'
print(x, ...)
```

Arguments

`object, x`	an object of class "sanon", usually, a result of a call to `sanon`.
`C`	contrast matrix. The number of column should be same as the length of `b` in outputs of `sanon`.
`confint`	logical value for whether the confidence interval is computed (only if C has one row).
`level`	the confidence level required (only if C has one row).
`...`	further arguments passed to or from other methods.

Details

This function provide the inference based on contrast after applying the function `sanon`. The contrast matrix C should be defined by the user. If the the number of row of C = 1, the confidence interval for the estimator is produced.

Value

`C`	contrast matrix
`Cb`	contrast estimates
`VCb`	variance and covariance matrix of `Cb`
`se`	standard error of `Cb`
`level`	confidence level
`UL`	upper confidence limit (only if the number of row of C = 1, otherwise `NULL`)
`LL`	lower confidence limit (only if the number of row of C = 1, otherwise `NULL`)
`Q`	test statistic
`df`	degree of freedom
`p`	p-value

Examples

```
##### Example 3.2 Randomized Clinical Trial of Respiratory Disorder #####
data(resp)
P = rbind(rep(0, 4), diag(4), rep(0, 4))
out23 = sanon(cbind(baseline, visit1, visit2, visit3, visit4) ~ grp(treatment, ref="P")
 + strt(center) + strt(sex) + covar(age), data=resp, P=P)

# Homogeneity of the xi_k across the four visits
contrast(out23, C=cbind(diag(3), rep(-1, 3)))

# Comparison between treatments for the average of the xi_k across the 4 visits
contrast(out23, C=matrix(rep(1, 4)/4, ncol=4))
```

`covar` *Identify Covariables*

Description

This is a special function used in the context of `sanon`. It identifies covariables when they appear on the right hand side of a formula.

Usage

```
covar(x)
```

Arguments

`x`	variable name

`cpain` *Chronic Pain Data*

Description

The data are from a multicenter randomized clinical trial to compare test and control treatments for the management of chronic pain, and they have had previous consideration in Stokes et al. (2000, chap. 13).

Usage

```
data(cpain)
```

Format

A data frame with 193 observations and 4 variables

Details

- `treat` a factor with levels `active` and `placebo` for treatment
- `response` a factor with five levels `poor`, `fair`, `moderate`, `good` and `excel` for pain status after treatment for 4 weeks
- `center` a factor with two levels `I` and `II` for two centers
- `diagnosis` a factor with four levels `A`, `B`, `C`, and `D` for diagnoses

References

Stokes, M. E., Davis, C. S., and Koch, G. G. (2000), Categorical Data Analysis using the SAS System, Cary: SAS Publishing.

`grp` *Identify Group Variables*

Description

This is a special function used in the context of `sanon`. It identifies group variables when they appear on the right hand side of a formula.

Usage

```
grp(x, ref = NULL)
```

Arguments

`x`	variable name
`ref`	character for the reference group for treatment group.

`heartburn` *Relief of heartburn Data*

Description

The data are from two period cross-over design clinical trial for relief of heartburn, and listings of the data appear in Koch, Gitomer, Skalland, and Stokes (1983). The variables are as follows:

Usage

```
data(heartburn)
```

Format

A data frame with 60 observations and 9 variables.

Details

`center` a factor vector for two centers

`sequence` a factor with levels `AP` and `PA` for sequence groups

`age` a numeric vector for age

`sex` a factor for sex with levels `female` and `male`

`freq` a numeric vector for weekly frequency of condition from previous medical history

`MD1` a numeric vector for time to relief from first dose during period 1

`MD2` a numeric vector for time to relief from first dose during period 2

`res1` a factor vector for relief status for period 1 (R = relief from first dose within 15 min, NF = no relief from first dose within 15 min)

`ref2` a factor vector for relief status for period 2 with same categories as `res1`

References

Koch G, Gitomer S, Skalland L, Stokes M (1983). "Some non-parametric and categorical data analyses for a change-over design study and discussion of apparent carry-over effects." Statistics in Medicine, 2(3), 397–412.

`resp` *Respiratory Disorder Data*

Description

The data are from a randomized clinical trial to compare a test treatment to placebo for a respiratory disorder, and listings of the data appear in Stokes et al. (2000, chap. 15, pp. 495-496) and Koch et al. (1990). The variables are as follows:

Usage

```
data(resp)
```

Format

A data frame with 111 observations and 9 variables.

Details

`center`	a factor vector for two centers
`treatment`	a factor with levels `A` and `P` for active and placebo treatments, respectively
`sex`	a factor with levels `F` and `M` for female and male, respectively
`age`	a numeric vector for age
`baseline`	a numeric vector for patient global ratings of symptom control according to 5 categories (4 = excellent, 3 = good, 2 = fair, 1 = poor, 0 = terrible) at baseline measurement
`visit1`	a numeric vector for patient global ratings of symptom control at visit 1 with same categories as `baseline`
`visit2`	a numeric vector for patient global ratings of symptom control at visit 2 with same categories as `baseline`
`visit3`	a numeric vector for patient global ratings of symptom control at visit 3 with same categories as `baseline`
`visit4`	a numeric vector for patient global ratings of symptom control at visit 4 with same categories as `baseline`

References

Stokes, M. E., Davis, C. S., and Koch, G. G. (2000), Categorical Data Analysis using the SAS System, Cary: SAS Publishing.

Koch, G. G., Carr, G. J., Amara, I. A., Stokes, M. E., and Uryniak, T. J. (1990), "Categorical Data Analysis," in Statistical Methodology in Pharmaceutical Sciences, ed. D. A. Berry, New York: Marcel Dekker, pp. 291-475.

`sanon` *Non-Parametric Covariable Adjustment for Stratified Rank Measures of Association*

Description

This is a function for computing a stratified multivariate Mann-Whitney estimator that addresses the comparison between two randomized groups for a strictly ordinal response variable. Response variables may have some missing completely at random (MCAR) values for some patients. Nonparametric covariable adjustment is considered through the difference estimates between mean covariable and the weighted least squares method. Although such estimators can be computed directly as weighted linear combinations of within-stratum Mann-Whitney estimators, consistent estimation of their covariance matrix is done using methods for multivariate U-statistics.

Usage

```
sanon(outcome, ...)

## S3 method for class 'formula'
sanon(formula, data = list(), ...)

## Default S3 method:
sanon(
  outcome,
  group,
  strt = NULL,
  covar = NULL,
  catecovar = NULL,
  ref = NULL,
  covref = NULL,
  P = NULL,
  res.na.action = "default",
  ...
)

## S3 method for class 'sanon'
print(x, ...)
```

Arguments

`outcome`	vector of observations of length n, or a matrix with n rows for the response (or outcome) variables
`...`	further arguments passed to or from other methods.
`formula`	a formula object, with the response on the left of a ~ operator, and the terms on the right.
`data`	a data.frame in which to interpret the variables named in the formula.

`group`	numeric vector of observations of length n for treatment group. The reference group can be specified in `ref`.
`strt`	numeric or factor vector of observations of length n, or a matrix with n rows for strata.
`covar`	numeric or factor vector of observations of length n, or a matrix with n rows for covariable.
`catecovar`	numeric or factor vector of observations of length n, or a matrix with n rows for categorical covariable.
`ref`	character for the reference group for treatment group in `group`.
`covref`	character vector for the reference group for categorical covariables in `catecovar`.
`P`	a matrix for weighted least squares estimation.
`res.na.action`	character for setting NA actions. "default", "LOCF1", "LOCF2", "replace", and "remove" are available. default is "default". see the details.
`x`	an object of class `"sanon"`, usually, a result of a call to `sanon`

Details

`sanon` has two specifications for the input, variable and formula based. In the variable based input, one can specify R objects to outcome, group, and strata variables, and covariable. In the formula based input, the formula consists of variable names in a data.frame. The strata and group variables, and covariable are recognized by functions `strt`, `grp`, `covar`, and `catecovar`. `outcome` can be contained missing values, which should be coded by `NA`. Five options for the management of missing values can be specifed in the argument `res.na.action`; `"default"` = the method in Kawaguchi et al. (2011), `"LOCF1"` and `"LOCF2"` = last observation carried forward with respect to kernels of U-statistics and observed velues, repsectively, `"replace"` = missing values are managed as tied with all other values in the same stratum, and `"remove"` = the complete cases analaysis. For `res.na.action` = `"LOCF1"` or `"LOCF2"`, the order in the outcome is considered as the time order in imputing. if the baseline measurement is missing, then the corresponding subject is removed. `outcome` can be also multiple (repeatly measured). If more than two strata are specified, these are taking a cross-classification. The group variable can be specifies its reference group in the argument `ref` in the `sanon` or in the function `grp`.

Value

`N`	Sample size
`Nna`	tne number of subjects with missing values
`nhik`	Sample size in each strata, group, and response
`nik`	Sample size in each group and response
`xi`	(multivariate) Mann-Whitney estimate(s) that addresses the comparison between two randomized groups
`g`	the difference estimates between mean covariable
`f`	a vector consisting of `xi` and `g`
`Vf`	estimated covariance matrix of `f`
`b`	fully adjustmented estimators for all covariables and the strata

`Vb`	covariance matrix of b
`se`	standard error of b
`Q`	test statistics for b
`p`	p-value for b
`outnames`	outcome or response names
`covarnames`	covariable names
`advarnames`	variable names adjusting in the weighted least squares
`bnames`	variable names of adjusted in the weighted least squares
`reslevels`	levels for response variables
`grouplevels`	levels for the group variable
`strtout`	resulting (cross-classification) strata
`strtlevels`	resulting (cross-classification) strata levels
`strtnames`	resulting (cross-classification) strata names
`matP`	design matrix used in the weighted least squares

References

Kawaguchi A., Koch, G. G. (2015). sanon: An R Package for Stratified Analysis with Nonparametric Covariable Adjustment. Journal of Statistical Software, 67(9), 1-37. doi:10.18637/jss.v067.i09

Kawaguchi, A., Koch, G. G., Wang, X. (2011): Stratified Multivariate Mann-Whitney Estimators for the Comparison of Two Treatments with Randomization Based Covariance Adjustment. Statistics in Biopharmaceutical Research, Vol. 3, No. 2, 217-231.

Examples

```
##### Example 3.1 Randomized Clinical Trial of Chronic Pain #####
data(cpain)
out11 = sanon(response ~ grp(treat, ref="placebo") + strt(center) + strt(diagnosis), data=cpain)
out11
summary(out11)

# R objects are also available
attach(cpain)
out12 = sanon(outcome=response, group=treat,
strt=cbind(center, diagnosis), ref="placebo")
out12
summary(out12)

##### Example 3.2 Randomized Clinical Trial of Respiratory Disorder #####
data(resp)
out21 = sanon(cbind(baseline, visit1, visit2, visit3, visit4)
~ grp(treatment, ref="P") + strt(center) + strt(sex) + covar(age), data=resp)
out21
summary(out21)

# the matrix P can be specified
```

```
P = rbind(rep(0, 4), diag(4), rep(0, 4))
out22 = sanon(cbind(baseline, visit1, visit2, visit3, visit4)
~ grp(treatment, ref="P") + strt(center) + strt(sex) + covar(age), data=resp, P=P)
out22
summary(out22)
```

`sebor` *Seborrheic Dermatitis Data*

Description

The data are from a randomized clinical trial to compare a test treatment to placebo for a seborrheic dermatitis, and listings of the data appear in Ramaswamy, Koch, and Amara (1997). The variables are as follows:

Usage

```
data(sebor)
```

Format

A data frame with 167 observations and 8 variables.

Details

`center` a factor vector for eight centers

`treat` a factor with levels `placebo` and `test` for placebo and test treatments, resectively

`score1` a numeric vector for patient global scores for the face according to 6 categories (0 = cleared, 1 = excellent improvement, 2 = moderate improvement, 3 = slight improvement, 4 = no change, 5 = exacerbation)

`score2` a numeric vector for patient global scores for the scalp with same categories as `score1`

`score3` a numeric vector for patient global scores for the chest with same categories as `score1`

`severity1` a numeric vector for the baseline desease severity for the face according to 3 categories (1 = mild, 2 = moderate, 3 = severe)

`severity2` a numeric vector for the baseline desease severity for the scalp with same categories as `severity1`

`severity3` a numeric vector for the baseline desease severity for the chest with same categories as `severity1`

References

Ramaswamy R, Koch G, Amara I (1997). "Application of rank analysis of covariance methods to analysis of multiple anatomical regions with treatment for seborrheic dermatitis." Journal of Biopharmaceutical Statistics, 7(3), 403–416.

`skin` *Skin Condition Data*

Description

The data are from a randomized clinical trial to compare a test treatment to placebo for skin conditions, and listings of the data appear in Stanish, Gillings, Koch (1978a, b). The variables are as follows:

Usage

```
data(skin)
```

Format

A data frame with 172 observations and 6 variables.

Details

`center` a factor vector for two centers

`treat` a factor with levels `A` and `P` for active and placebo treatments, skinectively

`stage` a numeric vector for initial severity of the skin condition according to 3 categories (3 = fair, 4 = poor, 5 = exacerbation) at baseline measurement

`res1` a numeric vector for extent of improvement at visit 1 according to 5 categories (1 = rapidly improving, 2 = slowly improving, 3 = stable, 4 = slowly worsening, 5 = rapidly worsening)

`res2` a numeric vector for extent of improvement at visit 2 with same categories as `res1`

`res3` a numeric vector for extent of improvement at visit 3 with same categories as `res1`

References

Stanish W, Gillings D, Koch G (1978a). "An application of multivariate ratio methods for the analysis of a longitudinal clinical trial with missing data." Biometrics, 34(2), pp. 305–317.

Stanish WM, Koch GG, Landis JR (1978b). "A computer program for multivariate ratio analysis (MISCAT)." Computer Programs in Biomedicine, 8(3-4), 197–207.

`strt` *Identify Stratification Variables*

Description

This is a special function used in the context of `sanon`. It identifies stratification variables when they appear on the right hand side of a formula.

Usage

```
strt(x)
```

Arguments

`x`	variable name

`summary.sanon` *Summarizing Weighted Least Squares Fits*

Description

summary method for class "sanon".

Usage

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

## S3 method for class 'summary.sanon'
print(x, ...)
```

Arguments

`object, x`	an object of class `"sanon"`, usually, a result of a call to `sanon`
`...`	further arguments passed to or from other methods.

Details

This function provide the p value for the hypothesis test of coefficient in the model of weighted least squares method. Note that the estimates in the output are for the (xi_k - 0.5).

Value

`coefficients`	a p x 4 matrix with columns for the estimated coefficient, its standard error, chi-squared statistic and corresponding (two-sided) p-value.
`advarnames`	adjust variable names in weighted least squares method

Examples

```
##### Example 3.1 Randomized Clinical Trial of Chronic Pain #####
data(cpain)
sum1 = summary(sanon(response ~ grp(treat, ref="placebo") + strt(center) + strt(diagnosis)
, data=cpain))
sum1

##### Example 3.2 Randomized Clinical Trial of Respiratory Disorder #####
data(resp)
sum22 = summary(sanon(cbind(baseline, visit1, visit2, visit3, visit4)
~ grp(treatment, ref="P") + strt(center) + strt(sex) + covar(age), data=resp))
sum22
```

`vcov.sanon` *Calculate Variance-Covariance Matrix for a Fitted Model Object*

Description

Returns the variance-covariance matrix of the main parameters of a fitted model object.

Usage

```
## S3 method for class 'sanon'
vcov(object, ...)
```

Arguments

`object`	an object of class "sanon", usually, a result of a call to `sanon`
`...`	further arguments passed to or from other methods.

Details

This is a generic function.

Value

Coefficients extracted from the model object object.

Examples

```
##### Example 3.1 Randomized Clinical Trial of Chronic Pain #####
data(cpain)
out1 = sanon(response ~ grp(treat, ref="placebo") + strt(center) + strt(diagnosis), data=cpain)
vcov(out1)

##### Example 3.2 Randomized Clinical Trial of Respiratory Disorder #####
data(resp)
P = rbind(rep(0, 4), diag(4), rep(0, 4))
```

```
out23 = sanon(cbind(baseline, visit1, visit2, visit3, visit4) ~ grp(treatment, ref="P")
 + strt(center) + strt(sex) + covar(age), data=resp, P=P)
# each four visits
vcov(out23)
```

Index