

# Package ‘sensitivymult’

May 9, 2026

**Type** Package

**Title** Sensitivity Analysis for Observational Studies with Multiple Outcomes

**Version** 1.0.2

**Author** Paul R. Rosenbaum

**Maintainer** Paul R. Rosenbaum <rosenbaum@wharton.upenn.edu>

**Description** Sensitivity analysis for multiple outcomes in observational studies. For instance, all linear combinations of several outcomes may be explored using Scheffe projections in the `comparison()` function; see Rosenbaum (2016, Annals of Applied Statistics) <[doi:10.1214/16-AOAS942](https://doi.org/10.1214/16-AOAS942)>. Alternatively, attention may focus on a few principal components in the `principal()` function. The package includes parallel methods for individual outcomes, including tests in the `senm()` function and confidence intervals in the `senmCI()` function.

**Imports** stats

**License** GPL-2

**LazyData** TRUE

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2017-08-29 21:49:06 UTC

## Contents

<code>amplify</code>	2
<code>artcog</code>	3
<code>comparison</code>	7
<code>mcorev</code>	11
<code>planScheffe</code>	12
<code>principal</code>	14
<code>senm</code>	20
<code>senmCI</code>	23
<code>separable1v</code>	26
<code>tbmetaphase</code>	27
<code>teeth</code>	28

---

 amplify

*Amplification of sensitivity analysis in observational studies.*


---

### Description

Uses the method in Rosenbaum and Silber (2009) to interpret a value of the sensitivity parameter  $\gamma$ . Each value of  $\gamma$  amplifies to a curve  $(\lambda, \delta)$  in a two-dimensional sensitivity analysis, the inference being the same for all points on the curve. That is, a one-dimensional sensitivity analysis in terms of  $\gamma$  has a two-dimensional interpretation in terms of  $(\lambda, \delta)$ .

### Usage

```
amplify(gamma, lambda)
```

### Arguments

gamma	$\gamma > 1$ is the value of the sensitivity parameter, for instance the parameter in <code>senmv</code> . <code>length(gamma)&gt;1</code> will generate an error.
lambda	lambda is a vector of values $> \gamma$ . An error will result unless <code>lambda[i] &gt; gamma &gt; 1</code> for every <code>i</code> .

### Details

A single value of  $\gamma$ , say  $\gamma = 2.2$  in the example, corresponds to a curve of values of  $(\lambda, \delta)$ , including (3, 7), (4, 4.33), (5, 3.57), and (7, 3) in the example. An unobserved covariate that is associated with a  $\lambda = 3$  fold increase in the odds of treatment and a  $\delta = 7$  fold increase in the odds of a positive pair difference is equivalent to  $\gamma = 2.2$ .

The curve is  $\gamma = (\lambda * \delta + 1) / (\lambda + \delta)$ . Amplify is given one  $\gamma$  and a vector of  $\lambda$ s and solves for the vector of  $\delta$ s. The calculation is elementary.

This interpretation of  $\gamma$  is developed in detail in Rosenbaum and Silber (2009), and it makes use of Wolfe's (1974) family of semiparametric deformations of an arbitrary symmetric distribution.

Strictly speaking, the amplification describes matched pairs, not matched sets. The `senm` function views a  $k$ -to-1 matched set with  $k$  controls matched to one treated individual as a collection of  $k$  correlated treated-minus-control matched pair differences; see Rosenbaum (2007). For matched sets, it is natural to think of the amplification as describing any one of the  $k$  matched pair differences in a  $k$ -to-1 matched set.

The curve has asymptotes that the function `amplify` does not compute:  $\gamma$  corresponds with  $(\lambda, \delta) = (\gamma, \infty)$  and  $(\infty, \gamma)$ .

A related though distinct idea is developed in Gastwirth et al (1998). The two approaches agree when the outcome is binary, that is, for McNemar's test.

### Value

Returns a vector of values of  $\delta$  of length(`lambda`) with names `lambda`.

**Note**

The amplify function is also in the sensitivitymv package where a different example is used.

**Author(s)**

Paul R. Rosenbaum

**References**

- Gastwirth, J. L., Krieger, A. M., Rosenbaum, P. R. (1998) Dual and simultaneous sensitivity analysis for matched pairs. *Biometrika*, 85, 907-920.
- Rosenbaum, P. R. (2007). Sensitivity analysis for m-estimates, tests and confidence intervals in matched observational studies. *Biometrics* 63 456-64. (R package sensitivitymv) <doi:10.1111/j.1541-0420.2006.00717.x>
- Rosenbaum, P. R. (2016) Using Scheffe projections for multiple outcomes in an observational study of smoking and periodontal disease. *Annals of Applied Statistics*, 10, 1447-1471. <doi:10.1214/16-AOAS942>.
- Rosenbaum, P. R. and Silber, J. H. (2009) Amplification of sensitivity analysis in observational studies. *Journal of the American Statistical Association*, 104, 1398-1405. <doi:10.1198/jasa.2009.tm08470>
- Rosenbaum, P. R. (2015). Two R packages for sensitivity analysis in observational studies. *Observational Studies*, v. 1. (Free on-line.)
- Wolfe, D. A. (1974) A characterization of population weighted symmetry and related results. *Journal of the American Statistical Association*, 69, 819-822.

**Examples**

```
data(teeth)
attach(teeth)
# The following calculation reproduces the comparison
# in expression (5.1) of Rosenbaum (2016, p. 1464)
comparison(cbind(either4low,either4up),smoker,
  mset,c(.714,.286),gamma=2.2,trim=2.5,Scheffe=TRUE)
# The parameter gamma=2.2 is given alternative interpretations
# in Rosenbaum (2016, p. 1465) as follows:
amplify(2.2,c(3,4,5,6,7))
detach(teeth)
```

## Description

The R package contains a simulated data set similar to actual data from 2009-2011 Irish Longitudinal Study of Aging (TILDA) used in Rosenbaum (2017). Additionally, in the documentation below, instructions are given for constructing the actual data set after downloading a file from ICPSR. The simulated data may be used to try the methods in this package. The actual data may be used to replicate the calculations in Rosenbaum (2017). Please be careful to distinguish the simulated data (with continuous outcomes) and the actual data (with integer outcomes), as scientific conclusions should not be based on the simulated data.

Instructions for creating the actual data are in the example section, but are not executed because you must obtain a file from ICPSR.

The simulated data were built from the actual data by calculating the trivariate within group means and the pooled within group covariance matrix. Then a data set of the same size was sampled from the trivariate Normal distribution, using the actual means and covariance matrix as the population parameters for the simulation. The simulation used the `mvtnorm` package. Although the data set consists of matched triples, in the simulated version, the matched sets are independent of the outcomes.

There are 219 matched triples containing one individual with arthritis (`arthritis=1`) and two without (`arthritis=0`). There are three measures of cognitive performance, words, wordsdelay and animals.

## Usage

```
data("artcog")
```

## Format

A data frame with 657 observations on the following 5 variables.

`arthritis` 1 if osteoarthritis, 0 if no arthritis

`words` Individuals are read a list of words and are immediately asked to recall as many as they can.

`wordsdelay` After a delay and another task, individuals are again asked to recall as many of the words as they can.

`animals` Individuals are asked to name all of the animals they can think of.

`mset` Indicator of the matched set: 1, 2, ..., 219.

## Details

A theory that NSAIDs reduce the risk of Alzheimer's disease has often been examined by comparing elderly people with and without arthritis, reasoning that many people with arthritis have consumed NSAIDs in quantity for a long period; see McGeer et al. (1996). This comparison does not ask a person with Alzheimer's disease to recall past use of NSAIDs.

Triples were matched for age, sex, education and mother's education.

Everyone is 75 years old or older.

## Source

Simulated data with a script for obtaining the actual data from the Irish Longitudinal Study of Aging 2009-2011.

## References

- Irish Longitudinal Study of Aging. <http://tilda.tcd.ie/>
- McGeer, P. L., Schulzer, M., and McGeer, E. G. (1996). Arthritis and anti-inflammatory agents as possible protective factors for alzheimer's disease. *Neurology*, 47, 425-432.
- Rosenbaum, P. R. (2017) Combining planned and discovered comparisons in observational studies. Manuscript. (The artcog example is discussed in this manuscript.)

## Examples

```
# data(artcog) returns the simulated example.
data(artcog)
# Three correlated outcomes.
cor(artcog[,2:4])
# See documentation for principal() for use of this example.

# The code below constructs the actual data, as distinct
# from the simulated example. The lengthy list of numbers
# assembles the 219 matched triples, or 657 = 3*219 rows,
# from the larger TILDA data set.

## Not run:
# Obtain from ICPSR the R data file
# ICPSR_34315-1IrishAging/34315-0001-Data.rda
# http://www.icpsr.umich.edu/icpsrweb/ICPSR/studies/34315?q=34315

# The data should have 8504 rows and 1992 columns

d<-da34315.0001
attach(d)

wordsC<-PH118
wordsI<-PH119
wordsC[wordsC<0]<-0
wordsI[wordsI<0]<-0
words<-wordsC+wordsI

wordsdelayC<-PH712
wordsdelayC[is.na(wordsdelayC)]<-0
wordsdelayC[wordsdelayC<=-1]<-0
wordsdelayI<-PH713
wordsdelayI[is.na(wordsdelayI)]<-0
wordsdelayI[wordsdelayI<=-1]<-0
wordsdelay<-wordsdelayC+wordsdelayI

animals<-PH125

arthritis<-PH301_03
osteoA<-PH304_1
z<-rep(NA,dim(d)[1])
z[arthritis==0]<-0
z[(arthritis==1)&(osteoA==1)]<-1
```

```
detach(d)
```

```
artcog<-data.frame(z,words,worddelay,animals)
```

```
who <- c(91, 4408, 7754, 129, 4716, 8383, 135, 6066,  
8028, 280, 894, 5300, 288, 151, 667, 298, 4889, 5977,  
333, 1100, 3707, 480, 696, 8148, 568, 372,  
7578, 584, 1852, 8057, 589, 3799, 6567, 590, 7422,  
8419, 609, 2825, 8272, 669, 1197, 8471, 684, 141,  
1847, 687, 2416, 7591, 771, 5239, 6986, 782,  
4857, 7654, 850, 885, 2239, 892, 2717, 7788, 929, 248,  
4740, 975, 1965, 8242, 1036, 6459, 7973, 1059, 1541,  
5901, 1103, 6518, 8264, 1160, 4798, 7330,  
1168, 4678, 7319, 1180, 152, 2735, 1191, 3740,  
7260, 1199, 26, 5209, 1252, 2615, 3251, 1444, 4790,  
7298, 1549, 898, 7630, 1587, 4418, 7122, 1596, 5875,  
8489, 1604, 3594, 7246, 1614, 3189, 7052, 1646,  
5415, 6828, 1708, 1634, 7029, 1760, 1950, 7815, 1840,  
5860, 8334, 1843, 6054, 7331, 1849, 5617, 8046, 1854,  
2890, 7703, 1885, 5846, 7247, 1896, 4365, 7803,  
1898, 3952, 4187, 1977, 544, 940, 1987, 768, 960,  
2029, 5363, 6293, 2161, 10, 4432, 2270, 5620, 7132,  
2330, 445, 1301, 2372, 1014, 1138, 2379, 3906,  
6183, 2386, 6226, 7203, 2417, 2458, 6616, 2437, 6262,  
7178, 2442, 3840, 8024, 2443, 4955, 5834, 2455, 1969,  
5967, 2457, 6962, 7560, 2466, 986, 2895, 2498, 2461,  
5876, 2522, 1837, 4803, 2618, 7279, 7764, 2734, 4005,  
4477, 2747, 221, 3837, 2763, 4440, 7863, 2765,  
6173, 7377, 2799, 7711, 7822, 2820, 2676, 7288, 2853,  
3035, 7518, 2914, 3142, 6891, 2952, 3081, 4908, 2969,  
3077, 6837, 3013, 747, 7614, 3107, 1754, 6564,  
3178, 2242, 4377, 3192, 260, 4530, 3246, 3019, 6478,  
3313, 4710, 7271, 3389, 356, 1796, 3481, 99, 491,  
3571, 658, 1410, 3693, 4341, 7624, 3694, 522,  
7702, 3704, 6532, 7171, 3705, 4973, 7131, 3806, 2163,  
5400, 3848, 4811, 7097, 3850, 2154, 5773, 3851, 3547,  
7613, 3862, 3357, 3370, 3877, 6186, 7990, 3913,  
455, 2883, 3931, 3548, 3699, 3933, 3210, 6164, 3935,  
4712, 7813, 3940, 5598, 7826, 3964, 2129, 8005, 3997,  
49, 1537, 4000, 3915, 5392, 4044, 3014, 6130,  
4052, 5208, 7213, 4186, 1586, 4249, 4264, 7058, 7182,  
4324, 3950, 7507, 4343, 3701, 6359, 4358, 567, 1020,  
4387, 2919, 4011, 4389, 5851, 7125, 4409, 3310,  
8100, 4427, 767, 2108, 4439, 1263, 6024, 4447, 3814,  
8373, 4478, 3493, 6743, 4479, 939, 2621, 4537, 1264,  
7942, 4608, 1797, 2987, 4633, 976, 1814, 4641,  
274, 1116, 4697, 4718, 7008, 4750, 2842, 5787, 4791,  
4386, 6966, 4812, 2817, 5640, 4815, 845, 5430, 4856,  
2288, 2289, 4887, 2182, 4874, 4942, 460, 4300,  
4945, 565, 3644, 4946, 487, 3369, 4953, 4352, 6709,  
4956, 2731, 3387, 4958, 4436, 6460, 4964, 3388, 6692,  
5078, 278, 963, 5110, 842, 4842, 5166, 600,
```

```

1530, 5199, 1775, 6210, 5204, 4993, 8477, 5210, 2646,
5563, 5291, 2957, 7777, 5325, 4881, 7053, 5342, 4385,
6444, 5377, 3957, 4319, 5384, 3144, 7757, 5385,
2813, 3054, 5386, 3636, 6185, 5474, 2507, 5085, 5488,
4278, 5675, 5584, 2606, 5359, 5599, 3180, 7037, 5605,
2459, 5304, 5637, 2581, 3621, 5641, 2781, 4302,
5805, 6424, 7227, 5870, 492, 5847, 5909, 1750, 5158,
5923, 3199, 6492, 6039, 4347, 4762, 6048, 5332, 7320,
6080, 1992, 2830, 6091, 5213, 7045, 6099, 5167,
6511, 6135, 5177, 6944, 6172, 2983, 6455, 6176, 2319,
3737, 6189, 5525, 7257, 6196, 4423, 6893, 6256, 2639,
5740, 6322, 1427, 2435, 6370, 7321, 7385, 6371,
1212, 2423, 6417, 205, 1674, 6462, 2393, 2882, 6463,
2170, 4765, 6496, 1630, 5048, 6519, 3058, 7498, 6901,
5237, 7508, 6984, 3819, 6548, 7042, 2961, 6445,
7057, 5457, 7984, 7061, 5401, 6049, 7093, 502, 3847,
7094, 4717, 6348, 7096, 825, 7844, 7099, 2188, 8337,
7251, 7423, 7576, 7269, 2616, 6401, 7270, 2394,
5039, 7273, 2337, 4941, 7300, 2241, 4934, 7316, 2604,
6369, 7355, 2113, 3880, 7402, 1456, 2378, 7473, 6368,
7243, 7592, 5583, 7892, 7615, 855, 6924, 7684,
6412, 6822, 7852, 405, 7077, 7862, 3623, 3990, 7879,
2447, 6334, 7913, 3927, 5299, 7930, 5289, 5844, 7983,
297, 7772, 8006, 3869, 6930, 8009, 2729, 6480,
8081, 4700, 6560, 8109, 62, 8061, 8130, 3351, 4381,
8149, 2854, 6513, 8157, 5220, 7559, 8184, 1523, 4195,
8185, 1459, 3820, 8188, 117, 1050, 8206, 2513,
3954, 8335, 2352, 4435, 8346, 5109, 8207)

```

```

artcog<-artcog[who,]
mset<-as.numeric(gl(219,3))
artcog<-cbind(artcog,mset)

rm(z,words,wordsdelay,animals,mset)

## End(Not run)

```

---

comparison

*Sensitivity Analysis for a Comparison Involving Several Outcomes in an Observational Study.*

---

## Description

For multiple outcomes in an observation study, computes a weighted combination of M-statistics, one for each outcome, and performs either a one-sided randomization test or an analysis of sensitivity to departures from random assignment. Each matched set contains one treated individual and one or more controls. The method is described in Rosenbaum (2016). For one outcome, use the function `senm()`.

**Usage**

```
comparison(y, z, mset, w, gamma = 1, inner = 0, trim = 3, lambda = 0.5,
           TonT = FALSE, apriori = FALSE, Scheffe = FALSE)
```

**Arguments**

y	A matrix of responses with no missing data. Different columns of y are different variables. If present, the column names of y are used to label output.
z	Treatment indicators, z=1 for treated, z=0 for control with length(z)==dim(y)[1].
mset	Matched set indicators, 1, 2, ..., sum(z) with length(mset)==dim(y)[1]. The vector mset may contain integers or may be a factor.
w	Vector of weights to be applied to the M-statistics for the several outcomes with length(w)==dim(y)[2]. At least one weight must be nonzero. See Details for discussion of scaling.
gamma	gamma is the sensitivity parameter $\Gamma$ , where $\Gamma \geq 1$ . Setting $\Gamma = 1$ is equivalent to assuming ignorable treatment assignment given the matched sets, and it performs a within-set randomization test.
inner	inner and trim together define the $\psi$ -function for the M-statistic. The default values yield a version of Huber's $\psi$ -function, while setting inner = 0 and trim = Inf uses the mean within each matched set. The $\psi$ -function is an odd function, so $\psi(w) = -\psi(-w)$ . For $w \geq 0$ , the $\psi$ -function is $\psi(w) = 0$ for $0 \leq w \leq \text{inner}$ , is $\psi(w) = \text{trim}$ for $w \geq \text{trim}$ , and rises linearly from 0 to trim for $\text{inner} < w < \text{trim}$ . An error will result unless $0 \leq \text{inner} \leq \text{trim}$ . Taking $\text{trim} < \text{Inf}$ limits the influence of outliers; see Huber (1981). Taking $\text{trim} < \text{Inf}$ and inner = 0 uses Huber's psi function. Taking $\text{trim} = \text{Inf}$ does no trimming and is similar to a weighted mean; see TonT. Taking inner > 0 often increases design sensitivity; see Rosenbaum (2013).
trim	inner and trim together define the $\psi$ -function for the M-statistic. See inner.
lambda	Before applying the $\psi$ -function to treated-minus-control differences, the differences are scaled by dividing by the lambda quantile of all within set absolute differences. Typically, lambda = 1/2 for the median. The value of lambda has no effect if trim=Inf and inner=0. See Maritz (1979) for the paired case and Rosenbaum (2007) for matched sets. An error will result unless $0 < \text{lambda} < 1$ .
TonT	TonT refers to the effect of the treatment on the treated; see Rosenbaum and Rubin (1985, equation 1.1.1) The default is TonT=FALSE. If TonT=FALSE, then the total score in matched set i is divided by the number ni of individuals in set i, as in expression (8) in Rosenbaum (2007). This division by ni has few consequences when every matched set has the same number of individuals, but when set sizes vary, dividing by ni is intended to increase efficiency by weighting inversely as the variance; see the discussion in section 4.2 of Rosenbaum (2007). If TonT=TRUE, then the division is by ni-1, not by ni, and there is a further division by the total number of matched sets to make it a type of mean. If TonT=TRUE and trim=Inf, then the statistic is the mean over matched sets of the

	treated minus mean-control response, so it is weighted to estimate the average effect of the treatment on the treated.
apriori	If Scheffe=FALSE and apriori=TRUE, then the weights are assumed to have been chosen a priori, and a one-sided, uncorrected P-value is reported for gamma=1 or an upper bound on the one-sided, uncorrected P-value is reported for gamma>1. In either case, this is a Normal approximation based on the central limit theorem and equals 1-pnorm(deviate).
Scheffe	If Scheffe=TRUE, then the weights are assumed to have been chosen after looking at the data. In this case, the P-value or P-value bound is corrected using Scheffe projections. The approximate corrected P-value or P-value bound is 1-pchisq(max(0,deviate)^2,dim(y)[2]). If Scheffe=FALSE and apriori=FALSE, then the deviate is returned, but no P-value is given. See Rosenbaum (2016). A Scheffe correction entitles you to look in both tails, which you do by considering both $w$ and $-w$ . See the planScheffe() function for a combination of an apriori and Scheffe comparisons.

## Details

If  $y$  has  $k$  columns for  $k$  outcomes, then comparison computes  $k$  M-statistics, one for each outcome, combines them into a single statistic using weights  $w$ , and computes a one-sided, upper-tailed deviate for a randomization test or a sensitivity analysis, as described in Rosenbaum (2016). The  $k$  individual statistics for the  $k$  outcomes separately are as described in Rosenbaum (2007).

When  $\text{trim} < \text{Inf}$ , outcomes are scaled using by the  $\lambda$  quantile of the absolute differences before applying the  $\psi$ -function. In this sense, when  $\text{trim} < \text{Inf}$ , the test statistics share a common scaling before they are combined using the weights in  $w$ . Weights  $w=c(1,1)$  give equal emphasis to two outcomes that may be recorded in different units, inches or pounds or whatever.

When  $\text{inner}=0$  and  $\text{trim}=\text{Inf}$ , the  $\psi$ -function is the identity, and no scaling is done. In this case, the weights refer to the original variables in their unscaled original units, inches or pounds or whatever. Because of this, the weights  $w$  have a different meaning with  $\text{trim}=\text{Inf}$  or  $\text{trim} < \text{Inf}$ . In the comparison() function,  $\text{inner} > 0$  is not permitted if  $\text{trim}=\text{Inf}$ .

If one has a single a priori choice of weights,  $w$ , then the one-sided P-value (for  $\text{gamma}=1$ ) or the one-sided upper bound on the P-value (for  $\text{gamma} > 1$ ) is approximately 1-pnorm(deviate).

If one considers every possible choice of weights,  $w$ , then the P-value (for  $\text{gamma}=1$ ) or the upper bound on the P-value (for  $\text{gamma} > 1$ ) uses a Scheffe correction and is approximately 1-pchisq(max(0,deviate^2),dim(y)[2]); see Rosenbaum (2016).

Matched sets of unequal size are weighted using weights that would be efficient in a randomization test under a simple model with additive set and treatment effects and errors with constant variance; see Rosenbaum (2007).

The upper bound on the P-value is based on the separable approximation described in Gastwirth, Krieger and Rosenbaum (2000); see also Rosenbaum (2007).

## Value

deviate	The upper bound on the standardized deviate that is used to approximate P-values using the Normal or chi-square distribution; see apriori and Scheffe in the arguments.
---------	---

aprioriPval	If Scheffe=FALSE and apriori=TRUE, the deviate is compared to the upper tail of the Normal distribution to produce either a P-value for gamma=1 or an upper bound on the P-value for gamma>1.
ScheffePval	If Scheffe=TRUE, the deviate is compared to the upper tail of the chi-square distribution to produce either a P-value for gamma=1 or an upper bound on the P-value for gamma>1.
weights	The weights possibly relabeled with colnames of y.

**Note**

For confidence intervals for individual outcomes, use function `senmCI()`.

**Author(s)**

Paul R. Rosenbaum.

**References**

- Huber, P. (1981) *Robust Statistics*. New York: John Wiley. (M-estimates based on M-statistics.)
- Maritz, J. S. (1979). A note on exact robust confidence intervals for location. *Biometrika* 66 163–166. (Introduces exact permutation tests based on M-statistics by redefining the scaling parameter.)
- Rosenbaum, P. R. (2007). Sensitivity analysis for m-estimates, tests and confidence intervals in matched observational studies. *Biometrics* 63 456-64. (R package `sensitivitymv`) <doi:10.1111/j.1541-0420.2006.00717.x>
- Rosenbaum, P. R. (2013). Impact of multiple matched controls on design sensitivity in observational studies. *Biometrics* 69 118-127. (Introduces inner trimming, `inner>0`.) <doi:10.1111/j.1541-0420.2012.01821.x>
- Rosenbaum, P. R. (2015). Two R packages for sensitivity analysis in observational studies. *Observational Studies*, v. 1. (Free on-line.)
- Rosenbaum, P. R. (2016) Using Scheffe projections for multiple outcomes in an observational study of smoking and periodontal disease. *Annals of Applied Statistics*, 10, 1447-1471. <doi:10.1214/16-AOAS942>
- Rosenbaum, P. R., & Rubin, D. B. (1985). The bias due to incomplete matching. *Biometrics*, 41, 103-116.
- Scheffe, H. (1953) A method for judging all contrasts in the analysis of variance. *Biometrika*, 40, 87-104.

**Examples**

```
data(teeth)
attach(teeth)
# The following calculation reproduces the comparison in
# expression (5.1) of Rosenbaum (2016, p. 1464)
comparison(cbind(teeth$either4low, teeth$either4up), teeth$smoker,
           teeth$mset, c(.714, .286), gamma=2.2, trim=2.5, Scheffe=TRUE)
#
# The following example reproduces the deviate for lower teeth
```

```
# mentioned on line 4 of Rosenbaum (2016, p. 1466) as a one-sided
# test with w picked a priori as w=c(1,0):
comparison(cbind(either4low,either4up),smoker,mset,c(1,0),
  trim=2.5,gamma=2.2,apriori=TRUE)
# Because the previous comparison implicitly involves just one outcome, it
# could be done more simply with senm() as follows:
senm(either4low,smoker,mset,trim=2.5,gamma=2.2)
# Had one done all comparisons including the comparison for lower teeth,
# then one would need to adjust for multiple testing:
comparison(cbind(either4low,either4up),smoker,mset,c(1,0),
  trim=2.5,gamma=2.2,Scheffe=TRUE)
detach(teeth)
```

---

mscorev

*Computes M-scores for M-tests and estimates.*


---

## Description

Of limited interest to most users, function `mscorev()` computes the M-scores used by functions `senm()`, `senmCI()`, `comparison()`, and `principal()` that perform Huber-Maritz M-tests. The function is also in the package `sensitivitymv`.

## Usage

```
mscorev(yamat, inner = 0, trim = 2.5, qu = 0.5, TonT = FALSE)
```

## Arguments

<code>yamat</code>	<code>yamat</code> is a matrix as described in the documentation for <code>senm()</code> .
<code>inner</code>	<code>inner</code> is the parameter described in the documentation for <code>senm()</code> . The inner parameter is discussed in Rosenbaum (2013).
<code>trim</code>	<code>trim</code> is the parameter described in the documentation for <code>senm()</code> . Note that the default for <code>mscorev</code> is <code>trim = 2.5</code> , but other functions in this package reset the default to <code>trim = 3</code> . This is for consistency with the <code>sensitivitymv</code> package which also contains the <code>mscorev</code> function.
<code>qu</code>	<code>qu</code> is the lambda parameter described in the documentation for <code>senm()</code> .
<code>TonT</code>	If <code>TonT=FALSE</code> , then the total score in set (row) <i>i</i> is divided by the number <i>ni</i> of individuals in row <i>i</i> , as in expression (8) in Rosenbaum (2007). If <code>TonT=TRUE</code> , then the division is by <i>ni</i> -1, not by <i>ni</i> , and there is a further division by the total number of matched sets. See the discussion of <code>TonT</code> in the documentation for <code>senm()</code> .

## Value

Generally, a matrix with the same dimensions as `yamat` containing the M-scores. Exception: if a matched set does not contain at least one treated subject and at least one control, then that set will not appear in the result, and the result will have fewer rows than `yamat`. However, if a matched set has several controls but no treated subject, then these controls will contribute to the estimate of the scale parameter, typically the median absolute pair difference.

**Note**

The example reproduces Table 3 in Rosenbaum (2007).

**Author(s)**

Paul R. Rosenbaum

**References**

- Rosenbaum, P. R. (2007) Sensitivity analysis for m-estimates, tests and confidence intervals in matched observational studies. *Biometrics*, 2007, 63, 456-464. <doi:10.1111/j.1541-0420.2006.00717.x>
- Rosenbaum, P. R. (2013). Impact of multiple matched controls on design sensitivity in observational studies. *Biometrics* 69 118-127. (Introduces inner trimming.) <doi:10.1111/j.1541-0420.2012.01821.x>
- Rosenbaum, P. R. (2015). Two R packages for sensitivity analysis in observational studies. *Observational Studies*, v. 1. (Free on-line.)

**Examples**

```
# The example reproduces Table 3 in Rosenbaum (2007).
data(tbmetaphase)
mscorev(tbmetaphase, trim=1)
```

---

planScheffe

*Combining One Planned Comparison and a Scheffe Correction For All Comparisons.*

---

**Description**

The function planScheffe() computes the critical values for a level alpha test that combines one planned linear combination of a K-dimensional multivariate Normal outcome and consideration of all possible combinations correcting for multiple testing using a Scheffe projection.

**Usage**

```
planScheffe(K, alpha = 0.05)
```

**Arguments**

K                    An integer  $\geq 2$  giving the number of outcomes to be compared.  
alpha                The level of the test, with  $0 < \alpha < 1$ .

**Details**

Although the calculation uses the multivariate Normal distribution, a typical application uses K test statistics that are asymptotically Normal.

The method is based on Rosenbaum (2017). The example below reproduces some of the comparisons in that manuscript.

**Value**

critical	critical is a vector with two elements, a and c. The null hypothesis is rejected at level alpha if either the Normal deviate for the planned comparison is $\geq a$ or if the square of the Normal deviate for any comparison is $\geq b$ . Then the probability of a false rejection is $\leq \alpha$ .
alpha	alpha is a vector with three elements, a, c and joint. The value of joint should equal the input value of alpha aside from numerical errors of computation: it is the probability of a false rejection using the joint test that rejects if either of the two critical values in critical is exceeded. In contrast, a is the probability that the planned deviate will be $\geq \text{critical}[1]$ when the null hypothesis is true. Also, c is the probability that at least one comparison will have a squared deviate $\geq \text{critical}[2]$ when the null hypothesis is true.

**Note**

The method is based on Rosenbaum (2017).

The functions `comparison()` and `principal()` may be used to calculate the standardized deviates that are compared to the critical values from `planScheffe`. Those functions have options for an a priori comparison or consideration of all possible comparisons with a Scheffe correction. The function `planScheffe` provides a third option: one planned comparison plus all possible comparisons.

**Author(s)**

Paul R. Rosenbaum.

**References**

- Miller, R. G., Jr. (1981) *Simultaneous Statistical Inference* (2nd edition). New York: Springer. Section 2.2, pp. 48-67 discusses Scheffe projections.
- Rosenbaum, P. R. (2016) Using Scheffe projections for multiple outcomes in an observational study of smoking and periodontal disease. *Annals of Applied Statistics*, 10, 1447-1471. <doi:10.1214/16-AOAS942>
- Rosenbaum, P. R. (2017) Combining planned and discovered comparisons in observational studies. Manuscript.
- Scheffe, H. (1953) A method for judging all contrasts in the analysis of variance. *Biometrika*, 40, 87-104.

**Examples**

```
# Please READ the documentation for artcog, and in particular
# the distinction between simulated and actual data.
# The dontrun section refers to the actual data and
# reproduces results in Rosenbaum (2017).

planScheffe(2,alpha=0.05)
# Interpretation of this output follows.
# Suppose there is a bivariate Normal outcome. We specify
# one a priori linear combination of its two coordinates.
```

```

# We test that the expectation is (0,0) with known covariance
# matrix. We compute the standardized difference for the
# a priori contrast, rejecting if it is >=1.895. We also
# reject if we can find any linear combination of the two
# coordinates whose squared standardized difference is
# >=7.077. The chance that we falsely reject a true
# null hypothesis is 0.05. The chance of a false rejection
# using the a priori comparison is 0.029. The chance of
# false rejection using any linear combination is 0.029.
#
# The a priori comparison could be the first principal
# component. Using the principal() function with
# w=c(1,0) gives the deviate for the first principal
# component. Exploring every w=c(w1,w2) gives the
# deviates that are squared for comparison with 7.077.

## Not run:
# For this illustration, obtain the actual data,
# as described in the documentation for artcog.
# An illustration from Rosenbaum (2017) follows.
data(artcog)
attach(artcog)
# The comparison using the first principal component:
principal(cbind(words,wordsdelay,animals),arthritis,mset,
          w=c(1,0),gamma=1.396,detail=TRUE)
# The resulting deviate, 1.900 is slightly greater than 1.895,
# so the hypothesis of no effect would be rejected at 0.05 even if
# we allow for a bias in treatment assignment of gamma=1.396.
principal(cbind(words,wordsdelay,animals),arthritis,mset,
          w=c(1,-.075),gamma=1.396,detail=TRUE)
# The comparison w=c(1,-.075) yields a slightly larger
# deviate, 1.907, but  $1.907^2 < 7.077$ , so this ad hoc
# comparison would not lead to rejection.
detach(artcog)
# Interpret gamma:
amplify(1.396,c(2,3))
amplify(1.4,c(2,3))

## End(Not run)

```

---

principal

*Sensitivity Analysis for Principal Components of M-Scores for Several Outcomes in an Observational Study.*


---

## Description

For  $k$  outcomes in a matched observational study, `principal()` computes the M-scores for the outcomes one at a time, computes the principal components of the M-scores, and uses some of the larger principal components as the outcomes in a sensitivity analysis. The user controls the number of components using `w`: (i) if `is.null(w)==TRUE`, then the first principal component is used

in a one-sided test, (ii) if  $\text{length}(w)=1$ , then  $w=1$  and  $w=-1$  both use the first principal component, but direct attention to the upper or lower tails, respectively, (iii) if  $\text{length}(w)>1$ , then the first  $\text{length}(w)$  principal components are used with weights  $w$ ; e.g.,  $w=c(1,1)$  adds the first two principal components together. Setting `Scheffe=TRUE` with  $\text{length}(w)=2$  permits the user to test every linear combination of the first two principal components – that is, every with  $\text{length}(w)=2$  – while controlling the family-wise error rate. Every matched set contains one treated subject and one or more controls.

## Usage

```
principal(y,z,mset,w=NULL,gamma=1,inner=0,trim=3,lambda=0.5,
          TonT=FALSE,apriori=FALSE,Scheffe=FALSE,detail=FALSE,
          cor=FALSE)
```

## Arguments

<code>y</code>	A matrix of responses with no missing data. Different columns of <code>y</code> are different variables, and there are $k=\text{dim}(y)[2]$ variables. If present, the column names of <code>y</code> are used to label output.
<code>z</code>	Treatment indicators, $z=1$ for treated, $z=0$ for control with $\text{length}(z)=\text{dim}(y)[1]$ .
<code>mset</code>	Matched set indicators, 1, 2, ..., $\text{sum}(z)$ with $\text{length}(mset)=\text{dim}(y)[1]$ . The vector <code>mset</code> may contain integers or may be a factor.
<code>w</code>	A vector of weights to be applied to principal components. There are $k=\text{dim}(y)[2]$ variables in <code>y</code> . (i) If $\text{is.null}(w)=\text{TRUE}$ or if $\text{length}(w)=1$ with $w \neq 0$ , then the test is applied to the first principal component of the $n$ vars M-test scores, and no adjustment for multiple testing is needed. (ii) If $\text{length}(w)>1$ , then <code>w</code> determines a comparison among the first $\text{length}(w)$ principal components of the $k$ M-scores. At least one weight must be nonzero. The meaning of the weights is affected by whether <code>cor=FALSE</code> (the default) or <code>cor=TRUE</code> . If <code>Scheffe=TRUE</code> , the dimensionality of the Scheffe correction is $\text{length}(w)$ , so $w=c(1,1)$ is different from $w=(1,1,0)$ , because the latter implies the investigator may consider the third principal component in some comparison, even though $w=(1,1,0)$ ignores it.
<code>gamma</code>	<code>gamma</code> is the sensitivity parameter $\Gamma$ , where $\Gamma \geq 1$ . Setting $\Gamma = 1$ is equivalent to assuming ignorable treatment assignment given the matched sets, and it performs a within-set randomization test.
<code>inner</code>	<code>inner</code> and <code>trim</code> together define the $\psi$ -function for the M-statistic. The default values yield a version of Huber's $\psi$ -function, while setting <code>inner = 0</code> and <code>trim = Inf</code> uses the mean within each matched set. The $\psi$ -function is an odd function, so $\psi(w) = -\psi(-w)$ . For $w \geq 0$ , the $\psi$ -function is $\psi(w) = 0$ for $0 \leq w \leq \text{inner}$ , is $\psi(w) = \text{trim}$ for $w \geq \text{trim}$ , and rises linearly from 0 to <code>trim</code> for $\text{inner} < w < \text{trim}$ . An error will result unless $0 \leq \text{inner} \leq \text{trim}$ .
<code>trim</code>	<code>inner</code> and <code>trim</code> together define the $\psi$ -function for the M-statistic. See <code>inner</code> . Unlike other functions in this package, <code>principal()</code> requires <code>trim &lt; Inf</code> . When <code>trim &lt; Inf</code> , the M-statistics for different outcomes have been scaled so their magnitudes are comparable, but this would not be true for <code>trim = Inf</code> . If you would

like to do analogous calculations without trimming, then give the comparison() function principal component scores rather than data for y, set trim=Inf and inner=0, and the w in that function will be applied to the principal component scores that you supplied.

lambda	<p>Before applying the <math>\psi</math>-function to treated-minus-control differences, the differences are scaled by dividing by the lambda quantile of all within set absolute differences. Typically, lambda = 1/2 for the median. The value of lambda has no effect if trim=Inf and inner=0. See Maritz (1979) for the paired case and Rosenbaum (2007) for matched sets.</p> <p>An error will result unless <math>0 &lt; \text{lambda} &lt; 1</math>.</p>
TonT	<p>TonT refers to the effect of the treatment on the treated; see Rosenbaum and Rubin (1985, equation 1.1.1) The default is TonT=FALSE. If TonT=FALSE, then the total score in matched set i is divided by the number ni of individuals in set i, as in expression (8) in Rosenbaum (2007). This division by ni has few consequences when every matched set has the same number of individuals, but when set sizes vary, dividing by ni is intended to increase efficiency by weighting inversely as the variance; see the discussion in section 4.2 of Rosenbaum (2007). If TonT=TRUE, then the division is by ni-1, not by ni, and there is a further division by the total number of matched sets to make it a type of mean. If TonT=TRUE and trim=Inf, then the statistic is the mean over matched sets of the treated minus mean-control response, so it is weighted to estimate the average effect of the treatment on the treated.</p>
apriori	<p>If Scheffe=FALSE and apriori=TRUE, then the weights are assumed to have been chosen a priori, and a one-sided, uncorrected P-value is reported for gamma=1 or an upper bound on the one-sided, uncorrected P-value is reported for gamma&gt;1. In either case, this is a Normal approximation based on the central limit theorem and equals 1-pnorm(deviate).</p>
Scheffe	<p>If Scheffe=TRUE, then the weights are assumed to have been chosen after looking at the data. In this case, the P-value or P-value bound is corrected using Scheffe projections. The approximate corrected P-value or P-value bound is 1-pchisq(max(0,deviate)^2,length(w)). If Scheffe=FALSE and apriori=FALSE, then the deviate is returned, but no P-value is given. See Rosenbaum (2016). Note carefully that length(w) determines the extent of the correction for multiplicity, so that, for example, w=c(1,0) focuses attention on the first principal component but allows for consideration of all comparisons using the first two principal components. See the discussion of w above and the examples below. A Scheffe correction entitles you to look in both tails, which you do by considering both w and -w. See the planScheffe() function for a combination of an apriori and Scheffe comparisons.</p>
detail	<p>If detail=TRUE, then some detail from the princomp() function in the stats package is returned.</p>
cor	<p>If cor=FALSE, the principal components of the M-scores are computed from the covariance matrix, but if cor=TRUE, then they are computed from the correlation matrix. Because the columns of y were scaled using lambda and scored by the same <math>\psi</math>-function, they are on a common scale, and it is reasonable to compute the principal components from the covariance matrix with cor=FALSE, the default. Setting cor=TRUE standardizes the columns of y twice, so perhaps it</p>

is pointless. In any event, because the weights  $w$  are applied to the principal components, and the latter are affected by  $\text{cor}$ , it follows that the meaning of  $w$  is affected by the value of  $\text{cor}$ .

### Details

If  $y$  has  $k$  columns for  $k$  outcomes, then `comparison` computes  $k$  M-scores, one for each outcome, computes principal components from these scores, combines the scores into a single comparison using  $w$ , and computes a one-sided, upper-tailed deviate for a randomization test or a sensitivity analysis, as described in Rosenbaum (2007, 2016).

Outcomes are scaled using by the  $\lambda$  quantile of the absolute differences before applying the  $\psi$ -function. In this sense, when `trim<Inf`, the M-scores share a common scaling before principal components are computed.

Taking `Scheffe=TRUE` and `w=(w1,w2)` for all  $w_1$  and  $w_2$  considers all comparisons based on the first two principal components.

Matched sets of unequal size are weighted using weights that would be efficient in a randomization test under a simple model with additive set and treatment effects and errors with constant variance; see Rosenbaum (2007).

The upper bound on the P-value is based on the separable approximation described in Gastwirth, Krieger and Rosenbaum (2000); see also Rosenbaum (2007).

### Value

<code>deviate</code>	The upper bound on the standardized deviate that is used to approximate P-values using the Normal or chi-square distribution; see <code>apriori</code> and <code>Scheffe</code> in the arguments.
<code>aprioriPval</code>	If <code>Scheffe=FALSE</code> and <code>apriori=TRUE</code> , the deviate is compared to the upper tail of the Normal distribution to produce either a P-value for $\gamma=1$ or an upper bound on the P-value for $\gamma>1$ .
<code>ScheffePval</code>	If <code>Scheffe=TRUE</code> , the deviate is compared to the upper tail of the chi-square distribution to produce either a P-value for $\gamma=1$ or an upper bound on the P-value for $\gamma>1$ .
<code>weights</code>	The weights are returned.

### Note

For confidence intervals for individual outcomes, use function `senmCI()`.

Under Fisher's hypothesis of no treatment effect, the principal components of the outcomes are unaffected by the treatment, so they may be used in randomization tests of no effect. However, this logic does not permit confidence intervals for the magnitude of effect on a principal component.

The `principal()` function computes principal components of M-scores, not of the outcomes themselves. This has various implications. The M-scores share a common, resistant scaling, so it is reasonable to consider principal components of the covariance matrix of the M-scores. In contrast, M-tests computed from principal components of outcomes are not resistant to outliers because the components themselves are not resistant to outliers. M-scores add to zero within each matched set; see the example for the `mscorev()` function. In this specific and limited sense, variation among

M-scores reflects variation within matched sets rather than variation between matched sets. For example, if the matched sets had been exactly matched for age, then the M-scores would be uncorrelated with age. In contrast, principal components of outcomes reflect both variation within and variation between matched sets. For instance, principal components of outcomes might be correlated with age even if the sets had been matched exactly for age. When the matched set size is variable, the M-scores incorporate variable weights, and the principal components are affected by these weights. For this reason, principal components of M-scores are more interpretable when every matched set has the same size, say matched pairs or matching 1-to-2, and they may be difficult to interpret if the set sizes vary widely, say 1-1 mixed with 1-5. In thinking about the relationship between outcomes and their M-scores, it can be helpful to examine the small, univariate example for the `mscorev()` function.

### Author(s)

Paul R. Rosenbaum.

### References

- Huber, P. (1981) *Robust Statistics*. New York: John Wiley. (M-estimates based on M-statistics.)
- Maritz, J. S. (1979). A note on exact robust confidence intervals for location. *Biometrika* 66 163–166. (Introduces exact permutation tests based on M-statistics by redefining the scaling parameter.)
- Rosenbaum, P. R. (2007). Sensitivity analysis for m-estimates, tests and confidence intervals in matched observational studies. *Biometrics* 63 456-64. (R package `sensitivitymv`) <doi:10.1111/j.1541-0420.2006.00717.x>
- Rosenbaum, P. R. (2013). Impact of multiple matched controls on design sensitivity in observational studies. *Biometrics* 69 118-127. (Introduces inner trimming, `inner>0`.) <doi:10.1111/j.1541-0420.2012.01821.x>
- Rosenbaum, P. R. (2015). Two R packages for sensitivity analysis in observational studies. *Observational Studies*, v. 1. (Free on-line.)
- Rosenbaum, P. R. (2016) Using Scheffe projections for multiple outcomes in an observational study of smoking and periodontal disease. *Annals of Applied Statistics*, 10, 1447-1471. <doi:10.1214/16-AOAS942>
- Rosenbaum, P. R. (2017) Combining planned and discovered comparisons in observational studies. Manuscript.
- Rosenbaum, P. R., & Rubin, D. B. (1985). The bias due to incomplete matching. *Biometrics*, 41, 103-116.

### Examples

```
# Please READ the documentation for artcog, and in particular
# the distinction between simulated and actual data.
# The dontrun section refers to the actual data and
# reproduces results in Rosenbaum (2017).
# The example immediately below uses the simulated data,
# and is simply a numerical illustration.
```

```
data(artcog)
attach(artcog)
```

```

# Randomization test using the first principal component of the simulated data.
principal(cbind(words,wordsdelay,animals),arthritis,mset,w=1,apriori=TRUE,detail=TRUE)

# Randomization test exploring a contrast of the first two principal components.
principal(cbind(words,wordsdelay,animals),arthritis,mset,w=c(1,-.1),Scheffe=TRUE)

# Sensitivity analysis using the first principal component of the simulated data.
principal(cbind(words,wordsdelay,animals),arthritis,mset,w=1,gamma=1.2,apriori=TRUE)
amplify(1.2,c(1.5,2))

## Not run:
# For this illustration, obtain the actual data,
# as described in the documentation for artcog.
# An illustration from Rosenbaum (2017) follows.
data(artcog)
attach(artcog)
# A randomization test using the first principal component for the three memory scores.
# The loadings show that the first component gives positive weight to each memory score.
principal(cbind(words,wordsdelay,animals),arthritis,mset,w=1,apriori=TRUE,detail=TRUE)
#
# The comparison above is insensitive to a bias of gamma=1.45
principal(cbind(words,wordsdelay,animals),arthritis,mset,w=1,gamma=1.45,apriori=TRUE,detail=TRUE)
#
# gamma=1.45 is an unobserved covariate that more than triples the odd of a poor memory score
# and more than doubles the odds of arthritis.
amplify(1.45,c(2,3,4))
#
# Although the first principal component is insensitive to a bias of gamma=1.45, each
# of the three individual variables is sensitive to a bias of gamma=1.45
senm(words,arthritis,mset,gamma=1.45)
senm(wordsdelay,arthritis,mset,gamma=1.45)
senm(animals,arthritis,mset,gamma=1.45)
#
# Although not particularly useful or enlightening in this one example, we can
# explore all weighted combinations of the first two principle components,
# correcting for multiple testing using Scheffe projections for dimension 2.
# This would be more interesting in an example with 50 outcomes, where we
# might want to reduce the dimensionality to 2 or 3 from 50, rather than to 1.
# We will do calculations for gamma=1.25. A gamma=1.25 is an unobserved
# covariate that doubles the odds of arthritis and doubles
# the odds of a worse memory score.
amplify(1.25,2)
# The deviate is the same but the corrected P-value is different if w=1 or w=c(1,0),
# because the former is doing a single one-sided test, while the latter is anticipating
# consideration of all possible combinations of the first two components.
principal(cbind(words,wordsdelay,animals),arthritis,mset,w=1,gamma=1.25,apriori=TRUE,detail=TRUE)
principal(cbind(words,wordsdelay,animals),arthritis,mset,w=c(1,0),gamma=1.25,Scheffe =TRUE)
# A weighted combination of the first two principal components, w=c(1,-.1), is ever so
# slightly less sensitive than using the first component alone.
principal(cbind(words,wordsdelay,animals),arthritis,mset,w=c(1,-.1),gamma=1.25,Scheffe =TRUE)
detach(artcog)

```

```
## End(Not run)
```

---

senm	<i>Sensitivity Analysis for a Matched Comparison in an Observational Study.</i>
------	---

---

## Description

Each matched set contains one treated individual and one or more controls. Uses Huber's M-statistic as the basis for the test, for instance, a mean. Performs either a randomization test or an analysis of sensitivity to departures from random assignment. For confidence intervals, use function `senmCI()`. The method is described in Rosenbaum (2007,2013). The `senm()` function is intended as a convenience for a user of the `comparison()` function in the `sensitivymult` package. The function `senm()` in the `sensitivymult` package is essentially the same as `senmv()` in the `sensitivymv` package, except the format of the input to `senm()` resembles the format of the input to `comparison()`. In particular, in the `sensitivymv` package, the rows of `y` are matched sets, not people, whereas in `sensitivymult` the rows of `y` are people with treated and control people identified by `z` and matched sets identified by `mset`.

## Usage

```
senm(y, z, mset, gamma = 1, inner = 0, trim = 3, lambda = 1/2,
      tau = 0, alternative="greater", TonT = FALSE)
```

## Arguments

<code>y</code>	A vector of responses with no missing data.
<code>z</code>	Treatment indicator, $z=1$ for treated, $z=0$ for control with $\text{length}(z)=\text{length}(y)$ .
<code>mset</code>	Matched set indicator, 1, 2, ..., $\text{sum}(z)$ with $\text{length}(mset)=\text{length}(y)$ . Matched set indicators should be either integers or a factor.
<code>gamma</code>	<code>gamma</code> is the sensitivity parameter $\Gamma$ , where $\Gamma \geq 1$ . Setting $\Gamma = 1$ is equivalent to assuming ignorable treatment assignment given the matched sets, and it performs a within-set randomization test.
<code>inner</code>	<code>inner</code> and <code>trim</code> together define the $\psi$ -function for the M-statistic. The default values yield a version of Huber's $\psi$ -function, while setting <code>inner = 0</code> and <code>trim = Inf</code> uses the mean within each matched set. The $\psi$ -function is an odd function, so $\psi(w) = -\psi(-w)$ . For $w \geq 0$ , the $\psi$ -function is $\psi(w) = 0$ for $0 \leq w \leq \text{inner}$ , is $\psi(w) = \text{trim}$ for $w \geq \text{trim}$ , and rises linearly from 0 to <code>trim</code> for <code>inner</code> < $w$ < <code>trim</code> . An error will result unless $0 \leq \text{inner} \leq \text{trim}$ . Taking <code>trim</code> < <code>Inf</code> limits the influence of outliers; see Huber (1981). Taking <code>trim</code> < <code>Inf</code> and <code>inner = 0</code> uses Huber's $\psi$ function. Taking <code>trim = Inf</code> does no trimming and is similar to a weighted mean; see <code>TonT</code> . Taking <code>inner</code> > 0 often increases design sensitivity; see Rosenbaum (2013).
<code>trim</code>	<code>inner</code> and <code>trim</code> together define the $\psi$ -function for the M-statistic. See <code>inner</code> .

lambda	<p>Before applying the <math>\psi</math>-function to treated-minus-control differences, the differences are scaled by dividing by the lambda quantile of all within set absolute differences. Typically, lambda = 1/2 for the median. The value of lambda has no effect if trim=Inf and inner=0. See Maritz (1979) for the paired case and Rosenbaum (2007) for matched sets.</p> <p>An error will result unless <math>0 &lt; \text{lambda} &lt; 1</math>.</p>
tau	<p>The null hypothesis asserts that the treatment has an additive effect, tau. By default, tau=0, so by default the null hypothesis is Fisher's sharp null hypothesis of no treatment effect.</p>
alternative	<p>If alternative="greater", the null hypothesis of a treatment effect of tau is tested against the alternative of a treatment effect larger than tau. If alternative="less", the null hypothesis of a treatment effect of tau is tested against the alternative of a treatment effect smaller than tau. In particular, alternative="less" is equivalent to: (i) alternative="greater", (ii) y replaced by -y, and (iii) tau replaced by -tau. See the note for discussion of two-sided sensitivity analyses.</p>
TonT	<p>TonT refers to the effect of the treatment on the treated; see Rosenbaum and Rubin (1985, equation 1.1.1) The default is TonT=FALSE. If TonT=FALSE, then the total score in matched set i is divided by the number ni of individuals in set i, as in expression (8) in Rosenbaum (2007). This division by ni has few consequences when every matched set has the same number of individuals, but when set sizes vary, dividing by ni is intended to increase efficiency by weighting inversely as the variance; see the discussion in section 4.2 of Rosenbaum (2007). If TonT=TRUE, then the division is by ni-1, not by ni, and there is a further division by the total number of matched sets to make it a type of mean. If TonT=TRUE and trim=Inf, then the statistic is the mean over matched sets of the treated minus mean-control response, so it is weighted to estimate the average effect of the treatment on the treated. See the examples.</p>

### Details

For the given  $\Gamma$ , senm() computes the upper bound on the 1-sided P-value testing the null hypothesis of an additive treatment effect tau against the alternative hypothesis of a treatment effect larger than tau. By default, senm() tests the null hypothesis of no treatment effect against the alternative of a positive treatment effect. The P-value is an approximate P-value based on a Normal approximation to the null distribution; see Rosenbaum (2007).

Matched sets of unequal size are weighted using weights that would be efficient in a randomization test under a simple model with additive set and treatment effects and errors with constant variance; see Rosenbaum (2007).

The upper bound on the P-value is based on the separable approximation described in Gastwirth, Krieger and Rosenbaum (2000); see also Rosenbaum (2007).

### Value

pval	The upper bound on the 1-sided P-value.
deviate	The deviate that was compared to the Normal distribution to produce pval.
statistic	The value of the M-statistic.

expectation	The maximum expectation of the M-statistic for the given $\Gamma$ .
variance	The maximum variance of the M-statistic among treatment assignments that achieve the maximum expectation. Part of the separable approximation.

### Note

The function `senm()` performs 1-sided tests. One approach to a 2-sided,  $\alpha$ -level test does both 1-sided tests at level  $\alpha/2$ , and rejects the null hypothesis if either 1-sided test rejects. Equivalently, a bound on the two sided P-value is the smaller of 1 and twice the smaller of the two 1-sided P-values. This approach views a 2-sided test as two 1-sided tests with a Bonferroni correction; see Cox (1977, Section 4.2). In all cases, this approach is a valid large sample test: a true null hypothesis is falsely rejected with probability at most  $\alpha$  if the bias in treatment assignment is at most  $\Gamma$ ; so, this procedure is entirely safe to use. For a randomization test,  $\Gamma = 1$ , this Bonferroni procedure is not typically conservative. For large  $\Gamma$ , this Bonferroni procedure tends to be somewhat conservative.

Related packages are `sensitivitymv`, `sensitivitymw`, `sensitivityfull` and `sensitivity2x2kk`.

### Author(s)

Paul R. Rosenbaum.

### References

- Cox, D. R. (1977). The role of significance tests (with Discussion). *Scand. J. Statist.* 4, 49-70.
- Huber, P. (1981) *Robust Statistics*. New York: John Wiley. (M-estimates based on M-statistics.)
- Maritz, J. S. (1979). A note on exact robust confidence intervals for location. *Biometrika* 66 163–166. (Introduces exact permutation tests based on M-statistics by redefining the scaling parameter.)
- Rosenbaum, P. R. (2007). Sensitivity analysis for m-estimates, tests and confidence intervals in matched observational studies. *Biometrics* 63 456-64. (R package `sensitivitymv`) <doi:10.1111/j.1541-0420.2006.00717.x>
- Rosenbaum, P. R. (2013). Impact of multiple matched controls on design sensitivity in observational studies. *Biometrics* 69 118-127. (Introduces inner trimming.) <doi:10.1111/j.1541-0420.2012.01821.x>
- Rosenbaum, P. R. (2014). Weighted M-statistics with superior design sensitivity in matched observational studies with multiple controls. *J. Am. Statist. Assoc.* 109 1145-1158. (R package `sensitivitymw`) <doi:10.1080/01621459.2013.879261>
- Rosenbaum, P. R. (2015). Two R packages for sensitivity analysis in observational studies. *Observational Studies*, v. 1. (Free on-line.)
- Rosenbaum, P. R. (2016) Using Scheffe projections for multiple outcomes in an observational study of smoking and periodontal disease. *Annals of Applied Statistics*, 10, 1447-1471. <doi:10.1214/16-AOAS942>
- Rosenbaum, P. R., & Rubin, D. B. (1985). The bias due to incomplete matching. *Biometrics*, 41, 103-116.

### Examples

```
data(teeth)
attach(teeth)
```

```
# The following example reproduces the deviate for lower teeth
# mentioned on line 4 of Rosenbaum (2016, p. 1466).
senm(either4low,smoker,mset,trim=2.5,gamma=2.2)
# The calculation above is equivalent to using comparison()
# with weights w=c(0,1) so upper teeth are ignored.
comparison(cbind(either4up,either4low),smoker,mset,c(0,1),trim=2.5,gamma=2.2)
# The following example illustrates the permutational t-test
# which uses the mean of the pair differences as a test statistic.
senm(either4low,smoker,mset,trim=Inf,TonT=TRUE)
dif<-either4low[smoker==1]-either4low[smoker==0] # Matched pair differences
mean(dif) # Equals the test statistic above
detach(teeth)
```

---

senmCI

*Sensitivity Analysis for a Confidence Interval.*


---

## Description

Each matched set contains one treated individual and one or more controls. Uses Huber's M-statistic as the basis for a confidence interval for an additive constant treatment effect,  $\tau$ . Produces either a randomization based confidence interval or an analysis of sensitivity to departures from random assignment. Also produces a point estimate for randomization inference or an interval of point estimates for a sensitivity analysis. For tests, use function `senm()`. The method is described in Rosenbaum (2007,2013).

## Usage

```
senmCI(y, z, mset, gamma=1, inner=0, trim=3, lambda=1/2,
       alpha=0.05, twosided=TRUE, upper=TRUE, TonT=FALSE)
```

## Arguments

<code>y</code>	A vector of responses with no missing data.
<code>z</code>	Treatment indicators, $z=1$ for treated, $z=0$ for control with $\text{length}(z)=\text{length}(y)$ .
<code>mset</code>	Matched set indicators, 1, 2, ..., $\text{sum}(z)$ with $\text{length}(mset)=\text{length}(y)$ . Matched set indicators should be either integers or a factor.
<code>gamma</code>	$\gamma$ is the sensitivity parameter $\Gamma$ , where $\Gamma \geq 1$ . Setting $\Gamma = 1$ is equivalent to assuming ignorable treatment assignment given the matched sets, and it returns a randomization-based confidence interval.
<code>inner</code>	<code>inner</code> and <code>trim</code> together define the $\psi$ -function for the M-statistic. The default values yield a version of Huber's $\psi$ -function, while setting <code>inner = 0</code> and <code>trim = Inf</code> uses the mean within each matched set. The $\psi$ -function is an odd function, so $\psi(w) = -\psi(-w)$ . For $w \geq 0$ , the $\psi$ -function is $\psi(w) = 0$ for $0 \leq w \leq \text{inner}$ , is $\psi(w) = \text{trim}$ for $w \geq \text{trim}$ , and rises linearly from 0 to <code>trim</code> for <code>inner</code> < $w$ < <code>trim</code> . An error will result unless $0 \leq \text{inner} \leq \text{trim}$ . Taking <code>trim</code> < <code>Inf</code> limits the influence of outliers; see Huber (1981). Taking <code>inner</code> > 0 often increases design sensitivity; see Rosenbaum (2013).

trim	inner and trim together define the $\psi$ -function for the M-statistic. See inner.
lambda	Before applying the $\psi$ -function to treated-minus-control differences, the differences are scaled by dividing by the lambda quantile of all within set absolute differences. Typically, lambda = 1/2 for the median. The value of lambda has no effect if trim=Inf and inner=0. See Maritz (1979) for the paired case and Rosenbaum (2007) for matched sets. An error will result unless $0 < \text{lambda} < 1$ .
alpha	The coverage rate of the confidence interval is $1 - \alpha$ . If the bias in treatment assignment is at most $\Gamma$ , then the confidence interval will cover the true $\tau$ with probability at least $1 - \alpha$ .
twosided	If twosided==TRUE, then a two-sided $1 - \alpha$ confidence interval is constructed. If twosided==FALSE, then a one-sided $1 - \alpha$ confidence interval is constructed. The two sided interval is the intersection of two one-sided $1 - \alpha/2$ intervals.
upper	If twosided==TRUE, then upper is ignored. If twosided==FALSE and upper=TRUE, then an upper $1 - \alpha$ confidence interval is constructed. If twosided==FALSE and upper=FALSE, then a lower $1 - \alpha$ confidence interval is constructed.
TonT	TonT refers to the effect of the treatment on the treated; see Rosenbaum and Rubin (1985, equation 1.1.1) The default is TonT=FALSE. If TonT=FALSE, then the total score in matched set $i$ is divided by the number $n_i$ of individuals in set $i$ , as in expression (8) in Rosenbaum (2007). This division by $n_i$ has few consequences when every matched set has the same number of individuals, but when set sizes vary, dividing by $n_i$ is intended to increase efficiency by weighting inversely as the variance; see the discussion in section 4.2 of Rosenbaum (2007). If TonT=TRUE, then the division is by $n_i - 1$ , not by $n_i$ , and there is a further division by the total number of matched sets to make it a type of mean. If TonT=TRUE and trim=Inf, then the statistic is the mean over matched sets of the treated minus mean-control response, so it is weighted to estimate the average effect of the treatment on the treated.

### Details

For the given  $\Gamma$ , senmCI() inverts the test in the function senm() to produce the confidence interval. That is, it tests every  $\tau$  and retains the values not rejected at level  $\alpha$ .

The test is a large sample approximation based on a Normal approximation to the null distribution; see Rosenbaum (2007).

If TonT=FALSE, matched sets of unequal size are weighted using weights that would be efficient in a randomization test under a simple model with additive set and treatment effects and errors with constant variance; see Rosenbaum (2007).

The upper bound on the P-value is based on the separable approximation described in Gastwirth, Krieger and Rosenbaum (2000); see also Rosenbaum (2007).

### Value

PointEstimates The interval of point estimates of  $\tau$ . If gamma=1, then the interval is a single point estimate.

ConfidenceInterval  
The confidence interval for  $\tau$ .

description      Reminder of the coverage rate, gamma, and type of interval.

### Note

In a sensitivity analysis, a one-sided confidence interval is not conservative; however, two-sided intervals formed as the intersection of two one-sided  $1 - \alpha/2$  intervals are somewhat conservative. See the discussion of two-sided tests in the documentation for `senm()`.

### Author(s)

Paul R. Rosenbaum.

### References

- Huber, P. (1981) *Robust Statistics*. New York: John Wiley. (M-estimates based on M-statistics.)
- Maritz, J. S. (1979). A note on exact robust confidence intervals for location. *Biometrika* 66 163–166. (Introduces exact permutation tests based on M-statistics by redefining the scaling parameter.)
- Rosenbaum, P. R. (1993). Hodges-Lehmann point estimates of treatment effect in observational studies. *Journal of the American Statistical Association*, 88, 1250-1253. (Introduces sensitivity analysis for point estimates.)
- Rosenbaum, P. R. (2007). Sensitivity analysis for m-estimates, tests and confidence intervals in matched observational studies. *Biometrics* 63 456-64. (R package `sensitivymv`) <doi:10.1111/j.1541-0420.2006.00717.x>
- Rosenbaum, P. R. (2013). Impact of multiple matched controls on design sensitivity in observational studies. *Biometrics* 69 118-127. (Introduces inner trimming.) <doi:10.1111/j.1541-0420.2012.01821.x>
- Rosenbaum, P. R. (2015). Two R packages for sensitivity analysis in observational studies. *Observational Studies*, v. 1. (Free on-line.)
- Rosenbaum, P. R. (2016) Using Scheffe projections for multiple outcomes in an observational study of smoking and periodontal disease. *Annals of Applied Statistics*, 10, 1447-1471. DOI: 10.1214/16-AOAS942.
- Rosenbaum, P. R., & Rubin, D. B. (1985). The bias due to incomplete matching. *Biometrics*, 41, 103-116.

### Examples

```
data(teeth)
attach(teeth)
#
# Note: Computing confidence intervals takes a few moments
# The calls to senmCI() are commented to meet time requirements
# for cran examples. Remove the comment symbol to run them.
#
# The calculations that follow reproduce the intervals from
# section 5.1 of Rosenbaum (2016, p. 1466)
# senmCI(either4low,smoker,mset,trim=2.5,gamma=1.5)
# senmCI(either4up,smoker,mset,trim=2.5,gamma=1.5)
# Confidence interval using the mean by inverting the
# permutational t-test.
```

```
# senmCI(either4low,smoker,mset,trim=Inf,TonT=TRUE)
dif<-either4low[smoker==1]-either4low[smoker==0] # Matched pair differences
mean(dif) # Equals the point estimate above
t.test(dif) # But permutational t-interval and t-interval differ
# Sensitivity analysis using the mean difference
# senmCI(either4low,smoker,mset,gamma=1.5,trim=Inf,TonT=TRUE)
detach(teeth)
```

---

separable1v

*Asymptotic separable calculations internal to other functions.*


---

### Description

Of limited interest to most users, this general purpose function is internal to other functions in the sensitivitymult package. It is the same function as in the sensitivitymv package, version 1.3. The function performs the asymptotic separable calculations described in Gastwirth, Krieger and Rosenbaum (2000), as used in section 4 of Rosenbaum (2007). See the sensitivitymv package for an example.

### Usage

```
separable1v(yamat, gamma = 1)
```

### Arguments

yamat	yamat is a matrix whose rows are matched sets and whose columns are matched individuals. The first column describes treated individuals. Other columns describe controls. If matched sets contain variable numbers of controls, NAs fill in empty spaces in yamat; see the documentation for senmv. In senmv, the matrix yamat is created by mscorev. Instead, if there were no NAs and ranks within rows were used in yamat, then separable1v would perform a sensitivity analysis for the stratified Wilcoxon two-sample test. Applied directly to data, it performs a sensitivity analysis for the permutational t-test.
gamma	gamma is the value of the sensitivity parameter; see the documentation for the senmv function in the sensitivitymv package. One should use a value of gamma $\geq 1$ .

### Value

pval	Approximate upper bound on the one-sided P-value.
deviate	Deviate that is compared to the upper tail of the standard Normal distribution to obtain the P-value.
statistic	Value of the test statistic.
expectation	Maximum null expectation of the test statistic for the given value of gamma.
variance	Among null distributions that yield the maximum expectation, variance is the maximum possible variance for the given value of gamma. See Rosenbaum (2007, Section 4) and Gastwirth, Krieger and Rosenbaum (2000).

**Author(s)**

Paul R. Rosenbaum

**References**

Gastwirth, J. L., Krieger, A. M., and Rosenbaum, P. R. (2000) Asymptotic separability in sensitivity analysis. *Journal of the Royal Statistical Society B* 2000, 62, 545-556. <DOI:10.1111/1467-9868.00249>

Rosenbaum, P. R. (2007) Sensitivity analysis for m-estimates, tests and confidence intervals in matched observational studies. *Biometrics*, 2007, 63, 456-464. <doi:10.1111/j.1541-0420.2006.00717.x>

---

tbmetaphase

*Genetic damage from drugs used to treat TB*

---

**Description**

This is a matched comparison of the effects of two drug sequences, namely HRZ and H2R2Z2, for the treatment of tuberculosis. HRZ is a higher dose sequence than H2R2Z2. The outcome is a measure of genetic damage, namely the frequency of aberrant metaphases two months after treatment. Individuals were matched for the frequency of aberrant metaphases before treatment. 15 individuals treated with HRZ are matched to 1 or 2 controls treated with H2R2Z2. Each row is one matched set. If a set is a pair, the third element in a row is NA. The data are originally from Rao, Gupta and Thomas (1991) and were used as an example in Rosenbaum (2007, Table 3). Data are used to illustrate the `senmv` function in the `sensitivitymv` package.

**Usage**

```
data(tbmetaphase)
```

**Format**

A data frame with 15 observations on the following 3 variables.

HRZ Aberrant metaphases for individual treated with HRZ.

H2R2Z2.1 Aberrant metaphases for first matched individual treated with H2R2Z2.

H2R2Z2.2 Aberrant metaphases for second matched individual treated with H2R2Z2. For matched pairs, this is NA.

**References**

Rao, V. V. N. G., Gupta, E. V. V and Thomas, I. M. Chromosomal aberrations in tuberculosis patients before and after treatment with short-term chemotherapy. *Mutation Research* 1991, 259, 13-19.

Rosenbaum, P. R. Sensitivity analysis for m-estimates, tests and confidence intervals in matched observational studies. *Biometrics*, 2007, 63, 456-464.

**Examples**

```
data(tbmetaphase)
```

---

teeth	<i>Smoking and Periodontal Disease.</i>
-------	---

---

**Description**

Data from NHANES 2011-2012 concerning periodonal disease in 441 matched pairs of smokers and nonsmokers.

**Usage**

```
data("teeth")
```

**Format**

A data frame with 882 observations on the following 4 variables.

`mset` Matched pair indicator: 1, 2, ..., 441.

`smoker` Treatment indicator: 1 if current smoker, 0 if never smoker

`either4up` Measure of periodontal disease for upper teeth; see Details.

`either4low` Measure of periodontal disease for lower teeth; see Details.

`cigspersday` Cigarettes smoked per day. Zero for nonsmokers.

**Details**

Smoking is believed to cause periodontal disease; see Tomar and Asma (2000). Using more recent data from NHANES 2011-2012, the data describe 441 matched pairs of a daily smoker and a never smoker. Daily smokers smoked every day of the last 30 days. Never smokers smoked fewer than 100 cigarettes in their lives, do not smoke now, and had no tobacco use in the previous five days.

Pairs are matched for education, income, age, gender and black race.

Measurements were made for up to 28 teeth, 14 upper, 14 lower, excluding 4 wisdom teeth. Pocket depth and loss of attachment are two complementary measures of the degree to which the gums have separated from the teeth; see Wei, Barker and Eke (2013). Pocket depth and loss of attachment are measured at six locations on each tooth, providing the tooth is present. A measurement at a location was taken to exhibit disease if it had either a loss of attachment  $\geq 4$ mm or a pocked depth  $\geq 4$ mm, so each tooth contributes a score from 0 to 6. Upper and lower are the number of measurements exhibiting disease for upper and lower teeth.

This example is from Rosenbaum (2016) where more information may be found.

**Source**

National Health and Nutrition Examination Survey (NHANES), 2011-2012. <https://www.cdc.gov/nchs/nhanes/>

## References

- Rosenbaum, P. R. (2016). Using Scheffe projections for multiple outcomes in an observational study of smoking and periodontal disease. *Annals of Applied Statistics*, 10, 1447-1471. <doi:10.1214/16-AOAS942>
- Tomar, S. L. and Asma, S. (2000). Smoking attributable periodontitis in the United States: Findings from NHANES III. *J. Periodont.* 71, 743-751.
- Wei, L., Barker, L. and Eke, P. (2013). Array applications in determining periodontal disease measurement. SouthEast SAS User's Group. (SESUG2013) Paper CC-15, [analytics.ncsu.edu/sesug/2013/CC-15.pdf](http://analytics.ncsu.edu/sesug/2013/CC-15.pdf).

## Examples

```

data(teeth)
attach(teeth)
# The following calculation reproduces the comparison
# in expression (5.1) of Rosenbaum (2016, p. 1464)
comparison(cbind(either4low,either4up),smoker,
  mset,c(.714,.286),gamma=2.2,trim=2.5,Scheffe=TRUE)
# Note that Rosenbaum (2016) used trim=2.5, but comparison()
# has a default of trim=3.
# The parameter gamma=2.2 is given alternative interpretations
# in Rosenbaum (2016, p. 1465) as follows:
amplify(2.2,c(3,4,5,6,7))
# The calculation (Rosenbaum 2016, p. 1465) for lower teeth alone is:
comparison(cbind(either4low,either4up),
  smoker,mset,c(1,0),gamma=2.2,trim=2.5,apriori=TRUE)
# Because w = c(1,0) ignores upper teeth, it may also be done as follows.
# Remove the comment sign to execute senmCI which is a little slow.
# senm(either4low,smoker,mset,gamma=2.2,trim=2.5)
# The calculations that follow reproduce the intervals from
# section 5.1 of Rosenbaum (2016, p. 1466)
# Remove the comment sign to execute senmCI which is a little slow.
# senmCI(teeth$either4low,teeth$smoker,teeth$mset,trim=2.5,gamma=1.5)
# The example that follows uses inner=0.5 as in Rosenbaum (2016,
# p. 1466, section 5.2):
comparison(cbind(either4low,either4up),smoker,
  mset,c(.714,.286),gamma=2.2,inner=.5,trim=2.5,Scheffe=TRUE)
comparison(cbind(either4low,either4up),smoker,
  mset,c(.714,.286),gamma=2.37,inner=.5,trim=2.5,Scheffe=TRUE)
detach(teeth)

```

# Index

- \* **Alzheimers disease**
    - artcog, 3
  - \* **Amplification**
    - amplify, 2
  - \* **Amplify**
    - amplify, 2
  - \* **Causal inference**
    - amplify, 2
    - artcog, 3
    - comparison, 7
    - planScheffe, 12
    - principal, 14
    - senm, 20
    - senmCI, 23
    - teeth, 28
  - \* **Confidence interval**
    - senmCI, 23
  - \* **Matched pairs**
    - teeth, 28
  - \* **Matched sets**
    - artcog, 3
  - \* **Matching with multiple controls**
    - artcog, 3
  - \* **Matching**
    - planScheffe, 12
    - senm, 20
  - \* **Multiple comparisons**
    - planScheffe, 12
  - \* **Observational study**
    - amplify, 2
    - artcog, 3
    - comparison, 7
    - planScheffe, 12
    - principal, 14
    - senm, 20
    - senmCI, 23
    - teeth, 28
  - \* **Principal Components**
    - principal, 14
  - \* **Sensitivity analysis**
    - amplify, 2
    - artcog, 3
    - comparison, 7
    - planScheffe, 12
    - principal, 14
    - senm, 20
    - senmCI, 23
    - teeth, 28
  - \* **Sheffe projection**
    - comparison, 7
    - planScheffe, 12
    - principal, 14
  - \* **datasets**
    - artcog, 3
    - tbmetaphase, 27
    - teeth, 28
  - \* **htest**
    - comparison, 7
    - planScheffe, 12
    - principal, 14
    - senm, 20
    - senmCI, 23
  - \* **rbounds**
    - senm, 20
  - \* **robust**
    - comparison, 7
    - planScheffe, 12
    - principal, 14
    - senm, 20
    - senmCI, 23
- amplify, 2  
artcog, 3  
comparison, 7  
mscorev, 11  
planScheffe, 12

principal, [14](#)

senm, [20](#)

senmCI, [23](#)

separable1v, [26](#)

tbmetaphase, [27](#)

teeth, [28](#)