

# Package: TreatmentSelection (via r-universe)

November 4, 2024

**Type** Package

**Title** Evaluate Treatment Selection Biomarkers

**Version** 2.1.1

**Date** 2017-8-11

**Author** Marshall Brown and Holly Janes

**Maintainer** Marshall Brown <mbrown@fhcrc.org>

**Description** A suite of descriptive and inferential methods designed to evaluate one or more biomarkers for their ability to guide patient treatment recommendations. Package includes functions to assess the calibration of risk models; and plot, evaluate, and compare markers. Please see the reference Janes H, Brown MD, Huang Y, et al. (2014) <[doi:10.1515/ijb-2012-0052](https://doi.org/10.1515/ijb-2012-0052)> for further details.

**URL** <http://rpubs.com/mbrown/TreatmentSelection>

**License** GPL-3

**Imports** ggplot2, survival, binom, grid

**RoxygenNote** 6.0.1

**Repository** <https://mbrown.r-universe.dev>

**RemoteUrl** <https://github.com/mbrown/treatmentselection>

**RemoteRef** HEAD

**RemoteSha** 3cc9f213edd10e09f2801ea30b73bff66a4655ff

## Contents

TreatmentSelection-package	2
calibrate	3
calibrate.trtsel	4
compare	6
compare.trtsel	6
evaluate	9
evaluate.trtsel	10

is.trtsel . . . . .	12
plot.trtsel . . . . .	12
print.calibrate.trtsel . . . . .	15
print.compare.trtsel . . . . .	16
print.eval.trtsel . . . . .	16
print.trtsel . . . . .	17
print.trtsel_summary_measures . . . . .	17
surv_tsdata . . . . .	18
trtsel . . . . .	18
trtsel_measures . . . . .	22
tsdata . . . . .	23
tsdata_cc . . . . .	24
tsdata_scc . . . . .	25
<b>Index</b>	<b>27</b>

---

TreatmentSelection-package

*Evaluate markers used to guide patient treatment selection decisions*

---

## Description

A suite of descriptive and inferential methods designed to evaluate one or more biomarkers for their ability to guide patient treatment recommendations.

## Details

Package: TreatmentSelection  
 Type: Package  
 Version: 2.0.-  
 Date: 2016-07-28  
 License: GPL-2

## Author(s)

Marshall Brown and Holly Janes

Maintainer: Marshall Brown <mdbrown@fhcrc.org>

## References

Janes H, Brown MD, Huang Y, et al. An approach to evaluating and comparing biomarkers for patient treatment selection. *Int J Biostat.* 2014;10(1):99-121.

**See Also**

[trtsel\\_measures](#) for evaluating the performance of a user-specified marker-based treatment rule, [trtsel](#) for creating trtsel objects, [plot.trtsel](#) for plotting risk curves and more, [evaluate.trtsel](#) for evaluating marker performance, [calibrate.trtsel](#) for assessing model calibration, and [compare.trtsel](#) to compare two trtsel object.

**Examples**

```
# See individual function man pages for examples.  
  
?trtsel_measures  
?trtsel  
?plot.trtsel  
?evaluate.trtsel  
?compare.trtsel  
?calibrate.trtsel
```

---

calibrate	<i>method for calibrating trtsel objects</i>
-----------	--

---

**Description**

method for calibrating trtsel objects

**Usage**

```
calibrate(x, ...)
```

**Arguments**

x	object
...	not used

**See Also**

[calibrate.trtsel](#)

---

calibrate.trtsel      *assess model calibration of a trtsel object*

---

## Description

Assess calibration of fitted models for risk and treatment effect given marker. Plots are used to compare observed vs. fitted risks and treatment effects and Hosmer-Lemeshow goodness-of-fit tests are reported. An object of class "trtsel" must first be created using the function "trtsel" by supplying a data.frame containing marker, treatment, and event status information.

## Usage

```
## S3 method for class 'trtsel'
calibrate(x, ..., groups = 10, plot.type = "calibration",
  trt.names = c("Treatment", "No Treatment"), line.color = "black",
  point.color = "grey10", ci = 0.95, main = NULL, ylim = NULL,
  xlim = NULL, ylab = NULL, xlab = NULL)
```

## Arguments

x	An object of class "trtsel", created by using the function "trtsel."
...	ignored.
groups	Number of groups; observations are split into groups based on quantiles of predicted risk or treatment effect, depending on plot.type. For plot.type = "treatment effect", observations are split into groups based on quantile of predicted treatment effect; for plot.type= "calibration", "risk.t0", or "risk.t1", and for the Hosmer-Lemeshow test statistic, observations on each treatment are split into groups based on quantile of predicted risk. The default value is 10.
plot.type	Which type of plot to produce. Options are "calibration" (default), which plots average predicted vs. observed risks on a log scale; "risk.t1" and "risk.t0" which overlays observed risks on fitted risk curves for T = 1 and T = 0 subjects, respectively; and "treatment effect" which overlays observed treatment effects on the fitted treatment effect curve.
trt.names	A vector of length 2 indicating the names for the two treatment options, T= 1 and T = 0, respectively, for the plot legend. This option is only used when plot.type="calibration". The default value is c("Treatment", "No Treatment").
line.color	color for lines in calibration plots.
point.color	color for points in calibration plots.
ci	numeric value (default 0.95) specifying the level for confidence intervals.
main	The main title for the plot. Only applies if plot.type is specified.
ylim	The limits for the y-axis of the plot, in the form c(lower,upper). Only applies if plot.type is specified.
xlim	The limits for the x-axis of the plot, in the form c(lower,upper). Only applies if plot.type is specified.

ylab	A label for the y-axis. Default values depend on plot.type. Only applies if plot.type is specified.
xlab	A label for the x-axis. Default values depend on plot.type. Only applies if plot.type is specified.

### Value

A list with the following components:

HL.TestStat	Hosmer-Lemeshow test statistic for assessing fit of the risk models for the T = 0 and T = 1 groups.
p.value	P-values for the Hosmer-Lemeshow tests in each treatment group.
Df	Degrees of freedom for the chi-square distribution used to generate a p-value for the Hosmer-Lemeshow chi-square test. This equals "groups" - 2.
plot	If plot output was created, the ggplot plotting object.

### See Also

[trtsel](#) for creating trtsel objects, [plot.trtsel](#) for plotting risk curves and more, [evaluate.trtsel](#) for evaluating marker performance, and [compare.trtsel](#) to compare two trtsel object.

### Examples

```
data(tsdata)

#####
## Create trtsel objects
#####
trtsel.Y1 <- trtsel(event ~ Y1*trt,
                   treatment.name = "trt",
                   data = tsdata,
                   study.design = "RCT",
                   default.trt = "trt all")

trtsel.Y1

#####
## Assess model calibration
#####

cali.Y1 <- calibrate(trtsel.Y1, plot.type = "calibration")
cali.Y1

# A "treatment effect" plot
calibrate(trtsel.Y1, line.color = "coral", plot.type = "treatment effect")
```

---

compare	<i>method to compare trtsel objects</i>
---------	---

---

### Description

method to compare trtsel objects

### Usage

```
compare(x, ...)
```

### Arguments

x	object
...	not used

---

compare.trtsel	<i>compare the performance of two treatment selection markers</i>
----------------	---

---

### Description

Evaluates and compares the performance of two treatment selection markers measured in the same data. (Bias-corrected) summary measures for the performance of each marker are estimated and confidence intervals are provided. Differences in measures of performance between markers are estimated along with confidence intervals, and the results of tests comparing marker performance measures are returned. "Treatment effect curves" for each marker are overlaid on the same plot. An object of class 'trtsel' must first be created for each marker using the function 'trtsel' by supplying a data.frame containing marker, treatment, and event status information; treatment and event data must be identical for the two markers.

### Usage

```
## S3 method for class 'trtsel'
compare(x, ..., x2, bootstraps = 500, bias.correct = TRUE,
  alpha = 0.05, plot = TRUE, ci = "horizontal", fixed.values = NULL,
  offset = 0.01, conf.bands = TRUE, conf.bandsN = 100,
  model.names = c("Model 1", "Model 2"), xlab = NULL, ylab = NULL,
  xlim = NULL, ylim = NULL, main = NULL, annotate.plot = TRUE)
```

**Arguments**

x	An object of class "trtsel" created by using function "trtsel". This is created using data for the first marker. Note: event and treatment vectors provided to create this trtsel object must be identical to those used to create the x2 object.
...	ignored.
x2	An object of class "trtsel" created by using function "trtsel". This is created using data for the second marker.
bootstraps	Number of bootstrap replicates for creating confidence intervals and bands. Default value is 500. Set bootstraps=0 if no confidence intervals or bands are desired.
bias.correct	logical indicator of whether to bias-correct measures for over-optimism using bootstrap-bias correction. When the same data is used to fit and evaluate the model, performance measures are over-optimistic. Setting this equal to TRUE uses a bootstrap method to bias-correct performance measures.
alpha	(1-alpha)*100% confidence intervals are calculated. Default value is alpha = 0.05 which yields 95% CI's.
plot	Indication of whether a plot showing treatment effect curves for the two markers should be created. TRUE (default) or FALSE.
ci	If plot = TRUE, indication of whether horizontal or vertical confidence intervals be plotted. Character string of either "horizontal" (default) or "vertical." Only applies if plot = TRUE.
fixed.values	A numeric vector indicating fixed values on the x- or y-axes at which bootstrap confidence intervals are provided. If "fixed.values" are provided, point-wise confidence intervals will be printed (i.e. conf.bands will be taken as FALSE). This option applies to the plot only.
offset	If confidence intervals are to be plotted for specified fixed.values, offset is the amount of distance to offset confidence intervals so that they do not overlap on the plot. The default value is 0.01. Only applies if plot = TRUE.
conf.bands	Indication of whether pointwise confidence intervals are shown for the curve(s). TRUE (default) or FALSE. If "fixed.values" are input, this option is ignored and no confidence bands are produced. Only applies if plot = TRUE.
conf.bandsN	If conf.bands = TRUE, the number of points along the x- or y-axis at which to calculate the pointwise confidence intervals. The default is 100. Only applies if plot = TRUE.
model.names	A vector of length 2 indicating the names for the two markers in x, and x2, respectively, for the plot legend. The default value is c("Model 1", "Model 2").
xlab	A label for the x-axis. Default values depend on plot.type. Only applies if plot = TRUE.
ylab	A label for the y-axis. Default values depend on plot.type. Only applies if plot = TRUE.
xlim	The limits for the x-axis of the plot, in the form c(x1,x2). Only applies if plot = TRUE.
ylim	The limits for the y-axis of the plot, in the form c(y1,y2). Only applies if plot = TRUE.

<code>main</code>	The main title for the plot. Only applies if <code>plot = TRUE</code> .
<code>annotate.plot</code>	Only applies to comparison of two discrete markers. Should the plot be annotated with the marker group percentages? default is <code>TRUE</code> .

### Value

A list with components (see Janes et al. (2013) for a description of the summary measures and estimators):

<code>estimates.marker1</code>	Point estimates of the summary measures: <code>p.neg</code> , <code>p.pos</code> , <code>B.neg</code> , <code>B.pos</code> , <code>Theta</code> , <code>Var.Delta</code> , <code>TG</code> and event rates for marker 1.
<code>estimates.marker2</code>	Point estimates for the same summary measures for marker 2.
<code>estimates.diff</code>	Estimated difference in summary measures (marker 1 - marker 2).
<code>ci.marker1</code>	2x9 data.frame with confidence intervals for marker 1 performance measures.
<code>ci.marker2</code>	2x9 data.frame with confidence intervals for marker 2 performance measures.
<code>ci.diff</code>	2x9 data.frame with confidence intervals for differences in performance measures.
<code>p.values</code>	1x9 data.frame with p-values for each difference corresponding to a test of $H_0$ : marker 1 performance = marker 2 performance.
<code>bootstraps</code>	bootstraps value provided.

In addition, if `plot = TRUE`:

<code>plot</code>	ggplot object containing the generated plot.
<code>plot.ci.marker1</code>	A data.frame containing the bounds of the bootstrap-based confidence intervals for marker1, along with the <code>fixed.values</code> they are centered around, where applicable.
<code>plot.ci.marker2</code>	a data.frame containing the bounds of the bootstrap-based confidence intervals for marker 2, along with the <code>fixed.values</code> they are centered around, where applicable.

### Note

Plot output is only produced when comparing a continuous (discrete) marker to a continuous (discrete) marker because confidence bands are calculated differently for continuous vs. discrete markers. See the note under `?plot.trtsel`, for a description of the differences between how the bootstrap confidence bands are calculated.

### References

Janes, Holly; Brown, Marshall D; Pepe, Margaret; Huang, Ying; "An Approach to Evaluating and Comparing Biomarkers for Patient Treatment Selection" *The International Journal of Biostatistics*. Volume 0, Issue 0, ISSN (Online) 1557-4679, ISSN (Print) 2194-573X, DOI: 10.1515/ijb-2012-0052, April 2014

**See Also**

[trtsel](#) for creating trtsel objects, [plot.trtsel](#) for plotting risk curves and more, [evaluate.trtsel](#) for evaluating marker performance, and [calibrate.trtsel](#) for assessing model calibration..

**Examples**

```

data(tpdata)

#####
## Create trtsel objects
#####

trtsel.Y1 <- trtsel(event ~ Y1*trt,
                   treatment.name = "trt",
                   data = tpdata,
                   study.design = "RCT",
                   default.trt = "trt all")

trtsel.Y1

trtsel.Y2 <- trtsel(event ~ Y2*trt,
                   treatment.name = "trt",
                   data = tpdata,
                   default.trt = "trt all")

trtsel.Y2

#####
## Compare marker performance
#####

# Plot treatment effect curves with pointwise confidence intervals
## use more bootstraps in practice
compare(x = trtsel.Y1, x2 = trtsel.Y2,
        bootstraps = 10, plot = TRUE,
        ci = "horizontal", conf.bands = TRUE)

```

---

evaluate

*a method for evaluating a trtsel object*


---

**Description**

a method for evaluating a trtsel object

**Usage**

```
evaluate(x, ...)
```

**Arguments**

x	object
...	not used

**See Also**

[evaluate.trtsel](#)

---

evaluate.trtsel	<i>evaluate the performance of one or more biomarkers for their ability to guide patient treatment recommendations.</i>
-----------------	---

---

**Description**

Evaluates the ability of one or more biomarkers for their ability to guide patient treatment recommendations. (Bias-corrected) summary measures of marker performance are estimated and confidence intervals are provided. This function accepts as input an object of class "trtsel", created using the function "trtsel".

**Usage**

```
## S3 method for class 'trtsel'
evaluate(x, ..., bias.correct = TRUE, bootstraps = 1000,
         alpha = 0.05)
```

**Arguments**

x	An object of class "trtsel", created by using the function "trtsel."
...	ignored.
bias.correct	logical indicator of whether to bias-correct measures for over-optimism using bootstrap-bias correction. When the same data is used to fit and evaluate the model, performance measures are over-optimistic. Setting this equal to TRUE uses a bootstrap method to bias-correct performance measures. See details for more information.
bootstraps	Number of bootstrap replicates for creating confidence intervals for each performance measure. The default value is 1000. Set bootstraps = 0 if no confidence intervals are desired.
alpha	(1-alpha)*100% confidence intervals are calculated. Default value is alpha = 0.05 which yields 95% CI's. The same alpha is used for the two-sided type-I error for the test of H0: No decrease in event rate under marker-based treatment.

**Value**

A list with the following components (see Janes et al. (2013) for a description of the summary measures and estimators):

<code>test.Null</code>	List of results of a test of the null hypothesis H0: No decrease in event rate under marker-based treatment. Contains "reject" (logical; was H0 rejected), <code>p.value</code> , <code>z.statistic</code> , and <code>alpha</code>
<code>estimates</code>	<code>data.frame</code> of dimension 1x9 of summary measure estimates. Includes: <code>p.rec.trt</code> : proportion recommended treatment; <code>p.rec.notrt</code> : proportion recommended no treatment; <code>B.neg.emp</code> , <code>B.neg.mod</code> : Average benefit of no treatment among those recommended no trt, empirical and model-based estimates; <code>B.pos.emp</code> , <code>B.pos.mod</code> : Average benefit of treatment among those recommended trt, empirical and model-based estimates; <code>Theta.emp</code> , <code>Theta.mod</code> : Decrease in rate of outcomes under marker-based treatment, empirical and model-based estimates; <code>Var.Delta</code> : variance in estimated treatment effect; <code>TG</code> : Total gain. <code>ER.trt0.emp/mod</code> , <code>ER.trt1.emp/mod</code> , <code>ER.mkrbased.emp/mod</code> : Event rates under <code>trt = 0</code> , <code>trt = 1</code> or for marker-based treatment. <code>Marker.Thresh</code> : Marker positivity threshold—defined as the maximum marker value such that estimated treatment effect < "thresh" (=treatment effect used to define the treatment rule, this is set when creating the <code>trtsel</code> object). If all observations are marker negative (or all are positive), <code>Marker.Thresh</code> has value NA
<code>conf.intervals</code>	<code>data.frame</code> of dimension 2x9 with bootstrap-based confidence intervals for each summary measures. If <code>bootstraps = 0</code> or <code>boot = FALSE</code> , this component is NULL.

**Note**

The `evaluate` function used this way both fit and evaluates a risk model using the same data, which is known to bias performance measure estimates to be overly optimistic. To correct for this bias, the `evaluate` function by default sets `bias.correct=TRUE`. This implements bootstrap bias correction via the "refined bootstrap" method described in Efron and Tibshirani 1994. In short, we sample BB bootstrap datasets. For each, obtain a new treatment selection rule based on the re-fit model and calculate the difference in estimated performance of this rule using the bootstrap vs. original data. The average of these differences estimates the bias. We shift naive performance estimates and confidence intervals down by the estimated bias.

**See Also**

[trtsel](#) for creating `trtsel` objects, [plot.trtsel](#) for plotting risk curves and more, [calibrate.trtsel](#) for assessing model calibration, and [compare.trtsel](#) to compare two `trtsel` object.

**Examples**

```
data(tsdata)

#####
## Create trtsel objects
#####
```

```

trtsel.Y1 <- trtsel(event ~ Y1*trt,
                  treatment.name = "trt",
                  data = tsdata,
                  study.design = "RCT",
                  family = binomial(link = "logit"),
                  default.trt = "trt all")

trtsel.Y1

#####
## Evaluate marker performance
#####

# Marker Y1
estimates.Y1 <- evaluate(trtsel.Y1, bootstraps = 50)
estimates.Y1

```

---

is.trtsel

*is x a trtsel object?*

---

### Description

Function to check if an object is of class "trtsel"

### Usage

```
is.trtsel(x)
```

### Arguments

x                   Any R object.

### Value

Returns TRUE or FALSE depending on whether x is a "trtsel" object or not.

---

plot.trtsel

*plot risk curves, treatment effect curves or cdf of risk for a trtsel object.*

---

### Description

Plots a treatment selection marker. Estimated "risk curves" and "treatment effect curves" are plotted. An object of class "trtsel" must first be created using the function "trtsel" by supplying a data.frame containing marker, treatment, and event status information.

**Usage**

```
## S3 method for class 'trtsel'
plot(x, bootstraps = 500, plot.type = c("treatment effect",
  "risk", "cdf", "selection impact"), ci = c("default", "horizontal",
  "vertical", "none"), alpha = 0.05, fixed.values = NULL, offset = 0.01,
  conf.bands = TRUE, conf.bandsN = 100, trt.names = c("Treatment",
  "  No \nTreatment"), xlab = NULL, ylab = NULL, xlim = NULL,
  ylim = NULL, main = NULL, show.marker.axis = TRUE, ...)
```

**Arguments**

x	An object of class "trtsel", created by using the function "trtsel."
bootstraps	Number of bootstrap replicates for creating confidence intervals and bands. Default value is 500. Set bootstraps=0 if no confidence intervals or bands are desired.
plot.type	Type of plot to produce. Options are "risk" (default) for risk curves which show risk of adverse event as a function of marker and treatment, "treatment effect" for the treatment effect curve which shows the difference in marker-specific risk of adverse event with vs. without treatment, "cdf" for the cumulative distribution function of the treatment effect, or "selection impact" to show a plot of the estimated event rate if different proportions of the population are treated. Cdf plots cannot be generated for a discrete marker.
ci	Indication of whether horizontal or vertical confidence intervals are desired. Character string of either "horizontal" (default) or "vertical." For a discrete marker only vertical confidence bands can be produced. See note below for more information.
alpha	(1-alpha)*100% confidence intervals are calculated. Default value is alpha = 0.05 which yields 95% CI's.
fixed.values	A numeric vector indicating fixed values on the x- or y-axes at which bootstrap confidence intervals are provided. If "fixed.values" are provided, point-wise confidence intervals will be printed (i.e. conf.bands will be taken as FALSE). Only applies to a continuous marker.
offset	If confidence intervals are to be plotted for specified fixed.values, offset is the amount of distance to offset confidence intervals so that they do not overlap on the plot. The default value is 0.01. Only applies to a continuous marker.
conf.bands	Indication of whether pointwise confidence intervals are shown for the curve(s). TRUE (default) or FALSE. If "fixed.values" are input, this option is ignored and no confidence bands are produced.
conf.bandsN	If conf.bands = TRUE, the number of points along the x- or y-axis at which to calculate the pointwise confidence intervals. The default is 100.
trt.names	A vector of length 2 indicating the names for the two treatment options, T= 1 and T = 0, respectively, for the plot legend. This option is only used when plot.type="risk". The default value is c("Treatment", "No Treatment").
xlab	A label for the x-axis. Default values depend on plot.type.
ylab	A label for the y-axis. Default values depend on plot.type.

xlim	The limits for the x-axis of the plot, in the form c(x1,x2)
ylim	The limits for the y-axis of the plot, in the form c(y1,y2)
main	The main title for the plot.
show.marker.axis	For risk curve plots with a continuous marker, should an axis showing marker values be displayed? default is TRUE.
...	N/A

**Value**

Generates a plot. In addition, if the function call is assigned to a variable (as in "myplot <- plot(mytrtsel)"), plot.trtsel returns a list composed of:

plot	ggplot object containing the printed plot.
ci.bounds	A data.frame containing the bounds of the bootstrap-based confidence intervals, along with the fixed.values they are centered around, if applicable.

**Note**

For a discrete marker, vertical confidence intervals are generated by fixing the marker value and generating a bootstrap distribution for risk given marker. Bands are then generated based on the percentiles of the empirical bootstrap distribution. For a continuous marker, vertical confidence intervals are generated by fixing the empirical marker quantiles, and generating the empirical distribution of risk given marker percentile.

**See Also**

[trtsel](#) for creating trtsel objects, [evaluate.trtsel](#) for evaluating marker performance, [calibrate.trtsel](#) for assessing model calibration, and [compare.trtsel](#) to compare two trtsel object.

**Examples**

```
data(tsdata)

#####
## Create trtsel objects
#####

trtsel.Y1 <- trtsel(event ~ Y1*trt,
                   treatment.name = "trt",
                   data = tsdata,
                   study.design = "RCT",
                   default.trt = "trt all")

#####
## Use the plot function
```

```
#####

# Plot risk curves
plot(trtsel.Y1, main = "Marker Y1",
     plot.type = "risk", bootstraps = 10, #set higher in practice
     trt.names=c("trt","no trt"))

# Now with confidence intervals around fixed.values
plot(trtsel.Y1, main = "Marker Y1",
     plot.type = "risk", ci = "horizontal",
     fixed.values = c(.2, .5),
     offset = .01, bootstraps = 10,
     trt.names=c("trt","no trt"))

# Plot treatment effect curves
#with confidence intervals around fixed.values
plot(trtsel.Y1, main = "Marker Y1",
     plot.type = "treatment effect",
     ci = "vertical", conf.bands = FALSE,
     fixed.values = c(10, 20), bootstraps = 10)
```

---

```
print.calibrate.trtsel
```

```
print a trtsel object
```

---

## Description

S3 method for class "trtsel"

## Usage

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

## Arguments

x	object of class "calibrate.trtsel", output from the calibrate.trtsel function
...	ignored

---

`print.compare.trtsel` *print a trtsel object*

---

**Description**

S3 method for class "trtsel"

**Usage**

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

**Arguments**

<code>x</code>	object of class "compare.trtsel", output from the <code>compare.trtsel</code> function
<code>...</code>	ignored

---

`print.eval.trtsel` *print a trtsel object*

---

**Description**

S3 method for class "trtsel"

**Usage**

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

**Arguments**

<code>x</code>	object of class "eval.trtsel", output from the <code>eval.trtsel</code> function
<code>...</code>	ignored

---

print.trtsel	<i>print a trtsel object</i>
--------------	------------------------------

---

**Description**

S3 method for class "trtsel"

**Usage**

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

**Arguments**

x	object of class "trtsel", which can be created using function "trtsel." See ?trtsel for more information.
...	ignored

---

print.trtsel_summary_measures	<i>print output from 'trtsel_measures'</i>
-------------------------------	--

---

**Description**

S3 method for class "trtsel\_measures"

**Usage**

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

**Arguments**

x	object of class "trtsel_summary_measures", output from the estimate_measures function
...	ignored

---

`surv_tsdata`*sample data for R package TreatmentSelection*

---

### Description

Simulated data set consisting of two continuous markers, two discrete markers, treatment, and time-to-event status.

### Usage

```
data(surv_tsdata)
```

### Format

A data frame with 1000 simulated observations reflecting the following clinical context. Marker Y for identifying a subset of women with estrogen-receptor positive breast cancer who do not benefit from adjuvant chemotherapy over and above tamoxifen. The clinical outcome of interest, termed "event", is 5-year breast cancer recurrence or death. Data come from a hypothetical randomized trial where subjects are randomly assigned to tamoxifen alone or tamoxifen plus chemotherapy and followed to ascertain this outcome; markers Y1 and Y2 are measured at baseline.

`xi` a numeric vector specifying follow up time for right censored data.

`di` a numeric vector indicating event status (1 for recurrence or death within 5 years, 0 for censoring)

`Y` a numeric vector of marker values.

`trt` a numeric vector indicating treatment status.

### See Also

[trtsel](#) for creating trtsel objects, [plot.trtsel](#) for plotting risk curves and more, [evaluate.trtsel](#) for evaluating marker performance, [calibrate.trtsel](#) for assessing model calibration, and [compare.trtsel](#) to compare two trtsel object.

---

`trtsel`*create a trtsel object*

---

### Description

Creates an object of class "trtsel" given a data.frame containing marker, treatment, and adverse event status information. The functions "plot.trtsel", "evaluate.trtsel", and "calibrate.trtsel" can then be used to plot risk and treatment effect curves, estimate summary measures, and check model calibration. The function "compare.trtsel" can also be used to compare two treatment selection markers.

**Usage**

```
trtsel(formula, treatment.name, data, fittedrisk.t0 = NULL,
       fittedrisk.t1 = NULL, thresh = 0, prediction.time = NULL,
       study.design = c("RCT", "nested case-control",
                        "stratified nested case-control"), cohort.attributes = NULL, family,
       default.trt = c("trt all", "trt none"))
```

**Arguments**

formula	a 'formula' object consisting of outcome ~ markers and marker by treatment interactions for the treatment selection model to be evaluated. The outcome can be continuous, binary or a 'Surv' object for time-to-event outcomes. Binary variable should equal 1 for cases and 0 for controls. For continuous outcomes, we assume that larger values are associated with worse outcomes.
treatment.name	Name of the treatment variable in data.frame "data". The treatment variable must be coded 1 for "treated" and 0 for "un-treated."
data	data.frame object used to fit and evaluate the model.
fittedrisk.t0	Instead of providing a marker, fitted risks for T=0 and T=1 may be provided. This should be the column name of the fitted risk for T=0 that can be found in 'data'. If fitted risks are provided, a model will not be fit, and all bootstrap confidence intervals will be conditional on the model fit provided.
fittedrisk.t1	Name of for the fitted risks given treatment = 1.
thresh	The treatment effect threshold used to define the treatment rule: Do not treat if the marker-specific treatment effect is less than "thresh". This is a numeric constant with a default value of 0.
prediction.time	a landmark prediction time used only when the outcome is a time-to-event Surv object
study.design	Character string indicating the study design used to collect the data. The three options are "RCT" (default), "nested case-control", or "stratified nested case-control". A "RCT" design is simply a randomized trial comparing T = 0 to T = 1 with the marker measured at baseline. A nested case-control or stratified nested case-control study samples cases and controls for marker measurement, perhaps stratified on treatment assignment, from a randomized trial comparing T = 0 to T = 1. See Janes et al. (2013) for a full description of these designs. If a "nested case-control" or "stratified nested case-control" design is specified, "cohort.attributes" must be provided (see below).
cohort.attributes	If a "nested case-control" or "stratified nested case-control" design is specified, "cohort.attributes" must be provided. Order does matter. For the "nested case-control" design, specify the following attributes of the randomized trial "cohort" from which the case-control sample was selected:  cohort.attributes = c(cohort sample size, proportion treated in cohort (Pr(trt==1)), adverse event prevalance in cohort (Pr(event==1)),

fraction of cases sampled from cohort)

For the "stratified nested case-control" design, specify the following attributes of the randomized trial "cohort" from which the stratified case-control sample was selected:

```
cohort.attributes = c(cohort sample size,
Pr(trt==0 & event==0) in cohort,
Pr(trt==0 & event==1) in cohort,
Pr(trt==1 & event==0) in cohort,
fraction of cases with trt == 0 sampled from cohort,
fraction of cases with trt == 1 sampled from cohort )
```

family	function passed to 'glm' describing the error distribution and link function to be used in the model. Defaults to binomial(link = "logit") when the outcome is numeric with two values 0,1 and gaussian(link = "identity") otherwise. Ignored for time-to-event outcomes. For non RCT study designs, methods only exist for family = binomial(link = "logit").
default.trt	The default treatment assignment to compare with marker-based treatment. Can either be set at "trt all" (default) or "trt none". Use "trt all" if everyone is treated and the aim is to discover those who would benefit from no treatment, but use "trt none" if the common practice is to treat no-one and the goal is to discover those who would benefit from treatment.

## Value

An object of class "trtsel," which is a list containing:

model.fit	A list containing "coefficients" – a 4 x 4 matrix with columns for coefficient estimates, standard errors, t-statistics, and two-sided p-values. "cohort.attributes" – the vector of cohort.attributes provided "study.design" – character string of study.design provided
derived.data	A data.frame with event, trt, and marker information along with "fittedrisk.t0" (risk estimates given no treatment), "fittedrisk.t1" (risk estimates given treatment), "trt.effect" (treatment effect estimates), and "trt.rec" (indicator of trt recommendation) columns.
functions	For internal package use only

## References

Janes, Holly; Brown, Marshall D; Pepe, Margaret; Huang, Ying; "An Approach to Evaluating and Comparing Biomarkers for Patient Treatment Selection" The International Journal of Biostatistics. Volume 0, Issue 0, ISSN (Online) 1557-4679, ISSN (Print) 2194-573X, DOI: 10.1515/ijb-2012-0052, April 2014

## See Also

[plot.trtsel](#) for plotting risk curves and more, [evaluate.trtsel](#) for evaluating marker performance, [calibrate.trtsel](#) for assessing model calibration, and [compare.trtsel](#) to compare two trtsel object.

## Examples

```
data(tsdata)

#####
## Create trtsel objects
#####

trtsel.Y1 <- trtsel(event ~ Y1*trt,
                   treatment.name = "trt",
                   data = tsdata,
                   study.design = "RCT",
                   family = binomial("logit"),
                   default.trt = "trt all")

trtsel.Y1

trtsel.Y2 <- trtsel(event ~ Y2*trt,
                   treatment.name = "trt",
                   data = tsdata,
                   default.trt = "trt all")

trtsel.Y2

# calculate fitted risks using a logistic model
#(one can use any model here, the point is that the fitted risks are provided )
mymod <- glm(event~trt*Y2, data= tsdata, family = binomial("logit"))

tsdata$fitted.t0 <- predict(mymod, newdata=data.frame(trt = 0, Y2 = tsdata$Y2), type = "response")
tsdata$fitted.t1 <- predict(mymod, newdata=data.frame(trt = 1, Y2 = tsdata$Y2), type = "response")

#all bootstrapping done using this object will be conditional on the model fit.

myfitted.trtsel <- trtsel( event~trt, treatment.name = "trt",
                          data = tsdata,
                          fittedrisk.t0 = "fitted.t0",
                          fittedrisk.t1 = "fitted.t1",
                          default.trt = "trt all")
```

---

trtsel_measures	<i>a simple function to estimate performance measures for a rule used to select treatment.</i>
-----------------	--

---

### Description

Provides point estimates for summary measures to evaluate a rule used to select treatment.

### Usage

```
trtsel_measures(event, trt, trt.rule, trt.effect, time,
  default.trt = c("trt all", "trt none"), prediction.time = NULL,
  silent = FALSE)
```

### Arguments

event	vector for adverse event. Can be binary (1 is bad, 0 is good) or continuous (large numbers are worse). If failure time variable 'time' is set, event is used as the adverse event status indicator.
trt	binary trt status 1 for "treated" and 0 for "un-treated."
trt.rule	a binary treatment rule used to recommend treatment where 1 means recommend treatment and 0 means recommend no treatment.
trt.effect	estimated treatment effects.
time	the failure time for survival outcomes.
default.trt	The default treatment assignment to compare with marker-based treatment. Can either be set at "trt all" (default) or "trt none". Use "trt all" if everyone is treated and the aim is to discover those who would benefit from no treatment, but use "trt none" if the common practice is to treat no-one and the goal is to discover those who would benefit from treatment.
prediction.time	a landmark prediction time used only when the 'time' variable is set.
silent	suppress messages

### Examples

```
data(tsdata)
#The user must specify a vector of clinical outcomes,
#a vector of treatment assignments, and a vector of
#marker-based treatment recommendations based on the pre-specified rule.

#Here we let Y1_disc represent a user-specified treatment
#rule and evaluate its performance.

trtsel_measures(event = tsdata$event, trt = tsdata$trt, trt.rule = 1- tsdata$Y1_disc )

#We can also fit our own risk model using GLM, use this model
```

```

#to develop a marker-based treatment recommendation, and evaluate its performance.
#This allows us to obtain model-based estimates of performance:

mod <- glm(event~trt*Y1_disc, data = tsdata, family = binomial())

tsdata.0 <- tsdata;
tsdata.0$trt = 0
tsdata.1 <- tsdata;
tsdata.1$trt = 1
delta.hat <- predict(mod,
                     newdata= tsdata.0,
                     type = "response") -
predict(mod,
        newdata= tsdata.1,
        type = "response")

trtsel_measures(event = tsdata$event, trt = tsdata$trt,
                trt.rule = 1- tsdata$Y1_disc, trt.effect = delta.hat )

```

---

tsdata

*sample data for R package TreatmentSelection*


---

## Description

Simulated data set consisting of two continuous markers, two discrete markers, treatment, and event status.

## Usage

```
data(tsdata)
```

## Format

A data frame with 1000 simulated observations reflecting the following clinical context. Markers Y1 and Y2 are candidates for identifying a subset of women with estrogen-receptor positive breast cancer who do not benefit from adjuvant chemotherapy over and above tamoxifen. The clinical outcome of interest, termed "event", is 5-year breast cancer recurrence or death. Data come from a hypothetical randomized trial where subjects are randomly assigned to tamoxifen alone or tamoxifen plus chemotherapy and followed to ascertain this outcome; markers Y1 and Y2 are measured at baseline.

`trt` a numeric vector distinguishing chemotherapy-treated individuals (`trt = 1`) vs individuals treated with tamoxifen alone (`trt = 0`)

`event` a numeric vector indicating event status (1 for recurrence or death within 5 years, 0 otherwise)

`Y1` a numeric vector of values for marker 1. This marker has relatively weak performance.

`Y2` a numeric vector of values for marker 2. This marker has strong performance.

`Y1_disc` a discrete version of marker 1, coded 1 if  $Y1 > \text{mean}(Y1)$  and 0 otherwise.

`Y2_disc` a discrete version of marker 2, coded 1 if  $Y2 > 0$ , and 0 otherwise.

## References

Janes, Holly; Brown, Marshall D; Pepe, Margaret; Huang, Ying; "An Approach to Evaluating and Comparing Biomarkers for Patient Treatment Selection" The International Journal of Biostatistics. Volume 0, Issue 0, ISSN (Online) 1557-4679, ISSN (Print) 2194-573X, DOI: 10.1515/ijb-2012-0052, April 2014

## See Also

[trtsel](#) for creating trtsel objects, [plot.trtsel](#) for plotting risk curves and more, [evaluate.trtsel](#) for evaluating marker performance, [calibrate.trtsel](#) for assessing model calibration, and [compare.trtsel](#) to compare two trtsel object.

---

tsdata\_cc

*sample nested case-control data for R package TreatmentSelection*

---

## Description

Simulated case-cohort data set consisting of two continuous markers, two discrete markers, treatment, and event status.

## Usage

```
data(tsdata)
```

## Format

A data frame with 468 observations reflecting a 1:1 case control subcohort sample from the tsdata dataset. See ?tsdata for the clinical context these data were simulated under.

`trt` a numeric vector distinguishing chemotherapy-treated individuals (`trt = 1`) vs individuals treated with tamoxifen alone (`trt = 0`)

`event` a numeric vector indicating event status (1 for recurrence or death within 5 years, 0 otherwise)

`Y1` a numeric vector of values for marker 1. This marker has relatively weak performance.

`Y2` a numeric vector of values for marker 2. This marker has strong performance.

`Y1_disc` a discrete version of marker 1, coded 1 if  $Y1 > \text{mean}(Y1)$  and 0 otherwise.

`Y2_disc` a discrete version of marker 2, coded 1 if  $Y2 > 0$ , and 0 otherwise.

## Note

In order to evaluate the markers in this dataset, we need to provide the function `trtsel` with information from the full cohort `tsdata`. See the example below for help.

## References

Janes, Holly; Brown, Marshall D; Pepe, Margaret; Huang, Ying; "An Approach to Evaluating and Comparing Biomarkers for Patient Treatment Selection" *The International Journal of Biostatistics*. Volume 0, Issue 0, ISSN (Online) 1557-4679, ISSN (Print) 2194-573X, DOI: 10.1515/ijb-2012-0052, April 2014

## See Also

[trtsel](#) for creating trtsel objects, [plot.trtsel](#) for plotting risk curves and more, [evaluate.trtsel](#) for evaluating marker performance, [calibrate.trtsel](#) for assessing model calibration, and [compare.trtsel](#) to compare two trtsel object.

---

tsdata_scc	<i>sample stratified nested case-control data for R package TreatmentSelection</i>
------------	--

---

## Description

Simulated stratified case-control data set consisting of two continuous markers, two discrete markers, treatment, and event status.

## Usage

```
data(tsdata)
```

## Format

A data frame with 468 observations reflecting a 1:1 stratified case control subcohort sample from the tsdata dataset. Observations are stratified on treatment status. See ?tsdata for the clinical context these data were simulated under.

`trt` a numeric vector distinguishing chemotherapy-treated individuals (`trt = 1`) vs individuals treated with tamoxifen alone (`trt = 0`)

`event` a numeric vector indicating event status (1 for recurrence or death within 5 years, 0 otherwise)

`Y1` a numeric vector of values for marker 1. This marker has relatively weak performance.

`Y2` a numeric vector of values for marker 2. This marker has strong performance.

`Y1_disc` a discrete version of marker 1, coded 1 if  $Y1 > \text{mean}(Y1)$  and 0 otherwise.

`Y2_disc` a discrete version of marker 2, coded 1 if  $Y2 > 0$ , and 0 otherwise.

## Note

In order to evaluate the markers in this dataset, we need to provide the function `trtsel` with information from the full cohort `tsdata`. See the example below for help.

**References**

Janes, Holly; Brown, Marshall D; Pepe, Margaret; Huang, Ying; "An Approach to Evaluating and Comparing Biomarkers for Patient Treatment Selection" The International Journal of Biostatistics. Volume 0, Issue 0, ISSN (Online) 1557-4679, ISSN (Print) 2194-573X, DOI: 10.1515/ijb-2012-0052, April 2014

**See Also**

[trtsel](#) for creating trtsel objects, [plot.trtsel](#) for plotting risk curves and more, [evaluate.trtsel](#) for evaluating marker performance, [calibrate.trtsel](#) for assessing model calibration, and [compare.trtsel](#) to compare two trtsel object.

# Index

## \* datasets

surv\_tsdata, 18

tsdata, 23

tsdata\_cc, 24

tsdata\_scc, 25

## \* package

TreatmentSelection-package, 2

calibrate, 3

calibrate.trtsel, 3, 4, 9, 11, 14, 18, 21,  
24–26

compare, 6

compare.trtsel, 3, 5, 6, 11, 14, 18, 21, 24–26

evaluate, 9

evaluate.trtsel, 3, 5, 9, 10, 10, 14, 18, 21,  
24–26

is.trtsel, 12

plot(plot.trtsel), 12

plot.trtsel, 3, 5, 9, 11, 12, 18, 21, 24–26

print.calibrate.trtsel, 15

print.compare.trtsel, 16

print.eval.trtsel, 16

print.trtsel, 17

print.trtsel\_summary\_measures, 17

surv\_tsdata, 18

TreatmentSelection

(TreatmentSelection-package), 2

TreatmentSelection-package, 2

trtsel, 3, 5, 9, 11, 14, 18, 18, 24–26

trtsel\_measures, 3, 22

tsdata, 23

tsdata\_cc, 24

tsdata\_scc, 25