

#### R code

```
> install.packages("BuyseTest", quiet = TRUE)
> library(BuyseTest)
```

#### R output

```
Loading required package: Rcpp
BuyseTest version 3.0.6
```

#### R code

```
> set.seed(10)
> data <- simBuyseTest(100, n.strata = 2)
> head(data)
```

#### R output

	id	treatment	eventtime	status	toxicity	score	strata
	<num>	<fctr>	<num>	<num>	<fctr>	<num>	<fctr>
1:	1	C	0.17392093	1	yes	-2.1250686	a
2:	2	C	0.16255166	0	yes	0.5211787	a
3:	3	C	0.08302502	1	yes	-0.0464229	b
4:	4	C	0.22204972	0	no	-1.1494717	b
5:	5	C	0.11669726	1	no	0.6293383	a
6:	6	C	0.11885540	1	yes	-0.7264715	a

#### R code

```
> e.BT <- BuyseTest(treatment ~ tte(eventtime, status = status),
  data = data)
```

#### R output

##### Generalized Pairwise Comparisons

##### Settings

- 2 groups : Control = C and Treatment = T
  - 1 endpoint:
- | priority | endpoint  | type          | operator            | event        |
|----------|-----------|---------------|---------------------|--------------|
| 1        | eventtime | time to event | higher is favorable | status (0 1) |
- right-censored pairs: probabilistic score based on the survival curves

##### Point estimation and calculation of the iid decomposition

##### Estimation of the estimator's distribution

- method: moments of the U-statistic

##### Gather the results in a S4BuyseTest object

# R code

```
> summary(e.BT)
```

# R output

Generalized pairwise comparisons with 1 endpoint

```
- statistic      : net treatment benefit (delta: endpoint specific, Delta: global)
- null hypothesis : Delta == 0
- confidence level: 0.95
- inference       : H-projection of order 1 after atanh transformation
- treatment groups: T (treatment) vs. C (control)
- censored pairs  : probabilistic score based on the survival curves
- results
endpoint total(%) favorable(%) unfavorable(%) neutral(%) uninf(%) Delta CI [2.5% ; 97.5%] p.value
eventtime    100      57.39      42.61      0      0 0.1479 [-0.0293;0.3161] 0.1019
```

# R code

```
> confint(e.BT, statistic = "winRatio")
```

# R output

	estimate	se	lower.ci	upper.ci	null	p.value
eventtime	1.347081	0.2450411	0.9430953	1.924118	1	0.1014458

# R code

```
> e.BThalf <- BuyseTest(treatment ~ tte(eventtime, status),
                        data = data, add.halfNeutral = TRUE, trace = FALSE)
> model.tables(e.BThalf, statistic = "favorable")
```

# R output

	endpoint	total	favorable	unfavorable	neutral	uninf	Delta	lower.ci	upper.ci	p.value
1 eventtime	100	57.39388	42.60612	0	0	0.5739388	0.4852354	0.6581263	0.1019135	

# R code

```
> coef(e.BThalf, statistic = "winRatio")
```

# R output

```
[1] 1.347081
```

# R code

```
> e.MBT <- BuyseTest(treatment ~ tte(eventtime, status, threshold = 1) + bin(toxicity, operator = "gt"),
                     data = data, trace = 0)
> model.tables(e.MBT)
```

# R output

	endpoint	threshold	total	favorable	unfavorable	neutral	uninf	delta	Delta	lower.ci	upper.ci
1	eventtime	1e+00	100.0	10.2	2.55	87.2	0	0.0768	0.0768	-0.00928	0.162
3	toxicity	1e-12	87.2	18.8	24.72	43.7	0	-0.0590	0.0178	-0.13396	0.169

# R code

```
plot(e.MBT)
```

# R code

```
plot(e.MBT, type = "racetrack")
```

# R code

```
> e.BTindiv <- BuyseTest(treatment ~ tte(eventtime, status = status),
                        data = data, keep.pairScore = TRUE)
> getPairScore(e.BTindiv)
```

# R output

	index.C	index.T	favorable	unfavorable	neutral	uninf	weight
1:	1	101	0.9192694	0.08073064	0	0.000000e+00	1
2:	2	101	0.5695583	0.43044167	0	1.110223e-16	1
3:	3	101	1.0000000	0.00000000	0	0.000000e+00	1
4:	4	101	0.4969601	0.50303994	0	0.000000e+00	1
5:	5	101	1.0000000	0.00000000	0	0.000000e+00	1
---							
9996:	96	200	0.2858328	0.71416716	0	0.000000e+00	1
9997:	97	200	0.8120919	0.18790807	0	0.000000e+00	1
9998:	98	200	0.6171644	0.38283561	0	0.000000e+00	1
9999:	99	200	0.6171644	0.38283561	0	0.000000e+00	1
10000:	100	200	0.4596044	0.54039560	0	0.000000e+00	1

# R code

```
> eBT.perm <- BuyseTest(treatment ~ cont(score), data = data,
                        method.inference = "varexact permutation")
> model.tables(eBT.perm)
```

# R output

	endpoint	total	favorable	unfavorable	neutral	uninf	Delta	p.value
1	score	100	53.67	46.33	0	0	0.0734	0.3698664

# R code

```
> wilcox.test(score ~ treatment, data = data, correct = FALSE)$p.value
```

# R output

```
0.3698664
```

R code

```
> rbind(confint(e.BTindiv, transformation = TRUE),
        confint(e.BTindiv, transformation = FALSE))
```

R output

	estimate	se	lower.ci	upper.ci	null	p.value
eventtime	0.1478776	0.08897931	-0.02931684	0.3160612	0	0.10150573
eventtime1	0.1478776	0.08897931	-0.02651861	0.3222739	0	0.09652625

R code

```
> NTB <- coef(e.BTindiv)
> sigma.NTB <- sqrt(crossprod(getIid(e.BTindiv)))
> sigmaTrans.NTB <- sigma.NTB/(1-NTB^2)
> c(estimate = NTB, se = sigmaTrans.NTB, p.value = 2*(1-pnorm(NTB/sigma.NTB)),
    pTrans.value = 2*(1-pnorm(atanh(NTB)/sigmaTrans.NTB)))
```

R output

estimate	se	p.value	pTrans.value
0.14787764	0.09096860	0.09652625	0.10150573

R code

```
BuyseTest.options(method.inference = "permutation", n.resampling = 1000,
                  statistic = "winRatio")
```

R code

```
> data("prodige", package = "BuyseTest")
> head(prodige)
```

R output

id	treatment	OS	statusOS	PFS	statusPFS	toxicity	sex
<num>	<fctr>	<num>	<num>	<num>	<num>	<num>	<fctr>
1:	1	C	0.0349	1	0.0349	0	F
2:	2	C	2.2790	0	2.2052	1	F
3:	3	C	0.2008	1	0.2008	0	M
4:	4	C	0.3418	1	0.3418	0	F

R code

```
> e.BR <- BuyseTest(treatment ~ tte(OS, statusOS, threshold = 6)
                    + cont(toxicity, threshold = 2)
                    + tte(OS, statusOS, threshold = 1)
                    + cont(toxicity), data = prodige)
> plot(e.BR)
```

R code

```
> summary(e.BR)
```

# R code

```
> M.threshold <- cbind(OS_t6 = c(3:5,3:5),
                        toxicity_t2 = c(2,2,2,3,3,3),
                        OS_t2 = 1,
                        toxicity = 0)
> M.threshold
```

# R output

	OS_t6	toxicity_t2	OS_t2	toxicity
[1,]	3	2	1	0
[2,]	4	2	1	0
[3,]	5	2	1	0
[4,]	3	3	1	0
[5,]	4	3	1	0
[6,]	5	3	1	0

# R code

```
> eBR.Se <- sensitivity(e.BR, band = TRUE,
                        threshold = M.threshold)
> plot(eBR.Se)
```

# R code

```
> e.NTB_Gehan <- BuyseTest(treatment ~ tte(OS, statusOS), scoring.rule = "Gehan",
                           data = prodige, keep.pairScore = TRUE, trace = FALSE)
> getPairScore(e.NTB_Gehan)[1:2,]
```

# R output

	index.C	index.T	favorable	unfavorable	neutral	uninf	weight
1:	1	403	1	0	0	0	1
2:	2	403	0	0	0	1	1

# R code

```
> e.NTB_Peron <- BuyseTest(treatment ~ tte(OS, statusOS), scoring.rule = "Peron",
                           data = prodige, keep.pairScore = TRUE, trace = FALSE)
> getPairScore(e.NTB_Peron)[1:2,]
```

# R output

	index.C	index.T	favorable	unfavorable	neutral	uninf	weight
1:	1	403	1.0000000	0.000000	0	0.00000000000	1
2:	2	403	0.5286551	0.47068	0	0.0006648516	1

# R code

```
> e.NTB_Latta <- BuyseTest(treatment ~ tte(OS, statusOS), scoring.rule = "Peron",
                           data = prodige, trace = FALSE,
                           model.tte = prodlm(Hist(OS, statusOS) ~ 1, data = prodige))
```

# R code

```
> e.NTB_restricted <- BuyseTest(treatment ~ tte(OS, statusOS, restriction = 24),
                                scoring.rule = "Peron", data = prodige)
```

$$U - \Delta = \underbrace{\frac{1}{m} \sum_{i=1}^m h_E(i)}_{\text{Experimental group}} + \underbrace{\frac{1}{n} \sum_{j=1}^n h_C(j)}_{\text{Control group}} + \underbrace{\frac{1}{mn} \sum_{i=1}^n \sum_{j=1}^m h_{EC}(i, j)}_{\text{Second order term}}$$

where for  $i \in \{1, \dots, m\}$ ,  $h_E(i) = \mathbb{E}[\mathbb{1}_{Y_i > X_j} - \mathbb{1}_{X_i > Y_j} \mid X_i] - \Delta$   
 $j \in \{1, \dots, n\}$ ,  $h_C(j) = \mathbb{E}[\mathbb{1}_{Y_i > X_j} - \mathbb{1}_{X_i > Y_j} \mid Y_j] - \Delta$

$$\widehat{\sigma}_U \underbrace{\approx}_{\text{First order}} \frac{1}{m^2} \sum_{i=1}^m h_E^2(i) + \frac{1}{n^2} \sum_{j=1}^n h_C^2(j)$$

$$p^{\mathcal{P}} = \frac{1}{1+P} \left\{ 1 + \sum_{p=1}^P \mathbb{1}_{|\Delta^{\mathcal{P}(p)}| \geq |\Delta|} \right\}$$

$$\mathbb{E} \left[ \mathbb{1}_{\textcolor{brown}{X}_i > \textcolor{blue}{Y}_j} | \widetilde{X}_{\textcolor{brown}{i}}, \varepsilon_{\textcolor{brown}{i}}, \widetilde{Y}_{\textcolor{blue}{j}}, \eta_j \right] = \begin{cases} 0.75 & \text{for } i = i_1 \\ 1 & \text{for } i = i_2 \end{cases}$$

$$\left(\widetilde{X}_{\textcolor{brown}{i}_1}, \widetilde{X}_{\textcolor{brown}{i}_2}, \widetilde{Y}_{\textcolor{blue}{j}}, \varepsilon_{\textcolor{brown}{i}_1}, \varepsilon_{\textcolor{brown}{i}_2}, \eta_j\right) = (4.7, 6.1, 1.5, 1, 1, 0)$$