Explicit formula for the net benefit

Brice Ozenne, Julien Péron

August 5, 2018

Contents

1 Parameter of interest

Let consider two independent real valued random variables X and Y. We are interested in:

$$\Delta = \mathbb{P}[Y > X] - \mathbb{P}[X > Y]$$

In the examples we will use a sample size of:

```
n <- 1e4
```

and use the following R packages

```
library(BuyseTest)
library(riskRegression)
library(survival)
```

2 Binary variable

2.1 Theory

$$\mathbb{P}[Y > X] = \mathbb{P}[Y = 1, X = 0]$$

Using the independence between Y and X:

$$\mathbb{P}[Y > X] = \mathbb{P}[Y = 1]\,\mathbb{P}[X = 0] = \mathbb{P}[Y = 1]\,(1 - \mathbb{P}[X = 1]) = \mathbb{P}[Y = 1] - \mathbb{P}[Y = 1]\,\mathbb{P}[X = 1]$$

By symmetry:

$$\mathbb{P}[X > Y] = \mathbb{P}[X = 1] - \mathbb{P}[Y = 1]\,\mathbb{P}[X = 1]$$

So

$$\Delta = \mathbb{P}[Y = 1] - \mathbb{P}[X = 0]$$

2.2 In R

Settings:

```
prob1 <- 0.4
prob2 <- 0.2
```

Simulate data:

```
set.seed(10)
df <- rbind(data.frame(tox = rbinom(n, prob = prob1, size = 1), group = "C"),
        data.frame(tox = rbinom(n, prob = prob2, size = 1), group = "T"))
```

Buyse test:

```
BuyseTest(group ~ bin(tox), data = df, method.inference = "none", trace = 0)
```

```
endpoint threshold   delta   Delta
     tox       0.5 -0.1981 -0.1981
```

Expected:

```
prob2 - prob1
```

```
[1] -0.2
```

3 Continuous variable

3.1 Theory

Let's consider two independent normally distributed variables with common variance:

- $X \sim \mathcal{N}\left(\mu_X, \sigma^2\right)$
- $Y \sim \mathcal{N}\left(\mu_Y, \sigma^2\right)$

Denoting $d = \frac{\mu_Y - \mu_X}{\sigma}$:

- $X^* \sim \mathcal{N}\left(0, 1\right)$
- $Y^* \sim \mathcal{N}\left(d, 1\right)$

$$\mathbb{P}\left[Y > X\right] = \mathbb{E}\left[\mathbb{1}_{Y > X}\right] = \mathbb{E}\left[\mathbb{1}_{Y* > X*}\right] = \mathbb{E}\left[\mathbb{1}_{Z > 0}\right]$$

where $Z \sim \mathcal{N}\left(d, 2\right)$ so $\mathbb{P}\left[Y > X\right] = \Phi(\frac{d}{\sqrt{2}})$

By symmetry

$$\Delta = 2 * \Phi(\frac{d}{\sqrt{2}}) - 1$$

3.2 In R

Settings:

```
meanX <- 0
meanY <- 2
sdXY <- 1
```

Simulate data:

```
set.seed(10)
df <- rbind(data.frame(tox = rnorm(n, mean = meanX, sd = sdXY), group = "C"),
        data.frame(tox = rnorm(n, mean = meanY, sd = sdXY), group = "T"))
```

Buyse test:

```
BuyseTest(group ~ cont(tox), data = df, method.inference = "none", trace = 0)
```

```
endpoint threshold  delta  Delta
     tox     1e-12 0.8359 0.8359
```

Expected:

```
d <- (meanY-meanX)/sdXY
2*pnorm(d/sqrt(2))-1
```

```
[1] 0.8427008
```

4 Survival

4.1 Theory

For a given cumulative density function $F(x)$ and a corresponding probability density function $f(x)$ we define the hazard by:

$$\begin{aligned}\lambda(t) &= \left.\frac{\mathbb{P}\left[t \leq T \leq t+h | T \geq t\right]}{h}\right|_{h\to 0^+} \\ &= \left.\frac{\mathbb{P}\left[t \leq T \leq t+h\right]}{\mathbb{P}\left[T \geq t\right] h}\right|_{h\to 0^+} \\ &= \frac{f(t)}{1-F(t)}\end{aligned}$$

Let now consider two times to events following an exponential distribution:

- $X \sim Exp(\alpha_1)$. The corresponding hazard function is $\lambda(t) = \alpha_1$.
- $Y \sim Exp(\alpha_2)$. The corresponding hazard function is $\lambda(t) = \alpha_2$.

So the hazad ratio is $HR = \frac{\lambda_2}{\lambda_1}$. Note that if we use a cox model we will have:

$$\lambda(t) = \lambda_0(t)\exp(\beta \mathbb{1}_{group})$$

where $\exp(\beta)$ is the hazard ratio.

$$\begin{aligned}\mathbb{P}\left[Y > X\right] &= \int_0^\infty \alpha_1 \exp(-\alpha_1 x) \int_x^\infty \alpha_2 \exp(-\alpha_2 y) dy dx \\ &= \int_0^\infty \alpha_1 \exp(-\alpha_1 x) [\exp(-\alpha_2 y)]_\infty^x dx \\ &= \int_0^\infty \alpha_1 \exp(-\alpha_1 x) \exp(-\alpha_2 x) dx \\ &= \frac{\alpha_1}{\alpha_1+\alpha_2}[\exp(-(\alpha_1+\alpha_2)x)]_\infty^0 \\ &= \frac{\alpha_1}{\alpha_1+\alpha_2} \\ &= \frac{1}{1+HR}\end{aligned}$$

So:

$$\Delta = 2\frac{1}{1+HR} - 1 = \frac{1-HR}{1+HR}$$

4.2 In R

Settings:

```
alphaX <- 2
alphaY <- 1
```

Simulate data:

```
set.seed(10)
df <- rbind(data.frame(time = rexp(n, rate = alphaX), group = "C", event = 1),
        data.frame(time = rexp(n, rate = alphaY), group = "T", event = 1))
```

Buyse test:

```
BuyseTest(group ~ tte(time, censoring = event), data = df,
      method.inference = "none", trace = 0, method.tte = "Gehan")
```

```
endpoint threshold  delta  Delta
    time     1e-12 0.3403 0.3403
```

Expected:

```
e.coxph <- coxph(Surv(time,event)~group,data = df)
HR <- as.double(exp(coef(e.coxph)))
c("HR" = alphaY/alphaX, "Delta" = 2*alphaX/(alphaY+alphaX)-1)
c("HR.cox" = HR, "Delta" = (1-HR)/(1+HR))
```

```
       HR     Delta
0.5000000 0.3333333
   HR.cox     Delta
0.4918256 0.3406392
```

5 Competing risks

5.1 Theory

5.1.1 General case (no censoring)

Let consider:

- X_E^* the time to the occurrence of the event of interest in the control group.
- Y_E^* the time to the occurrence of the event of interest in the treatment group.
- X_{CR}^* the time to the occurrence of the competing event of interest in the control group.
- Y_{CR}^* the time to the occurrence of the competing event of interest in the treatment group.

Let denote $\varepsilon_X = 1 + \mathbb{1}_{X_E^* > X_{CR}^*}$ the event type indicator in the control group and $\varepsilon_Y = 1 + \mathbb{1}_{Y_E^* > Y_{CR}^*}$ the event type indicator in treatment group ($= 1$ when the cause of interest is realised first and 2 when the competing risk is realised first).

For each subject either the event of interest or the competing event is realized. We now define:

$$X = \left\{ \begin{array}{l} X_E^* \text{ if } \varepsilon_X = 1 \\ +\infty \text{ if } \varepsilon_X = 2 \end{array} \right. \quad \text{and } Y = \left\{ \begin{array}{l} Y_E^* \text{ if } \varepsilon_Y = 1 \\ +\infty \text{ if } \varepsilon_Y = 2 \end{array} \right.$$

i.e. when the event of interest is not realized we say that the time to event is infinite.

We thus have:

$$\begin{aligned}
\mathbb{P}\left[Y > X\right] =& \mathbb{P}\left[Y > X | \varepsilon_X = 1, \varepsilon_Y = 1\right] \mathbb{P}\left[\varepsilon_X = 1, \varepsilon_Y = 1\right] \\
&+ \mathbb{P}\left[Y > X | \varepsilon_X = 1, \varepsilon_Y = 2\right] \mathbb{P}\left[\varepsilon_X = 1, \varepsilon_Y = 2\right] \\
&+ \mathbb{P}\left[Y > X | \varepsilon_X = 2, \varepsilon_Y = 1\right] \mathbb{P}\left[\varepsilon_X = 2, \varepsilon_Y = 1\right] \\
&+ \mathbb{P}\left[Y > X | \varepsilon_X = 2, \varepsilon_Y = 2\right] \mathbb{P}\left[\varepsilon_X = 2, \varepsilon_Y = 2\right] \\
=& \mathbb{P}\left[Y > X | \varepsilon_X = 1, \varepsilon_Y = 1\right] \mathbb{P}\left[\varepsilon_X = 1, \varepsilon_Y = 1\right] \\
&+ 1 * \mathbb{P}\left[\varepsilon_X = 1, \varepsilon_Y = 2\right] \\
&+ 0 * \mathbb{P}\left[\varepsilon_X = 2, \varepsilon_Y = 1\right] \\
&+ 0 * \mathbb{P}\left[\varepsilon_X = 2, \varepsilon_Y = 2\right]
\end{aligned}$$

So $\mathbb{P}\left[X > Y\right] = \mathbb{P}\left[X > Y | \varepsilon_X = 1, \varepsilon_Y = 1\right] \mathbb{P}\left[\varepsilon_X = 1, \varepsilon_Y = 1\right] + \mathbb{P}\left[\varepsilon_X = 1, \varepsilon_Y = 2\right]$ and:

$$\begin{aligned}
\Delta =& \Big(\mathbb{P}\left[X > Y | \varepsilon_X = 1, \varepsilon_Y = 1\right] - \mathbb{P}\left[X < Y | \varepsilon_X = 1, \varepsilon_Y = 1\right]\Big) \mathbb{P}\left[\varepsilon_X = 1, \varepsilon_Y = 1\right] \\
&+ \mathbb{P}\left[\varepsilon_X = 1, \varepsilon_Y = 2\right] - \mathbb{P}\left[\varepsilon_X = 2, \varepsilon_Y = 1\right]
\end{aligned}$$

5.1.2 General case (censoring, method: Gehan)

In case of censoring we can use an inverse probability weighting approach. Let denote $\delta_{c,X}$ (resp. $\delta_{c,Y}$) the indicator of no censoring relative to $\tilde{X}$ (resp $\tilde{Y}$), $\tilde{X}_E$ and $\tilde{Y}_E$ the censored event time. We can use inverse probability weighting to compute the net benefit:

$$\begin{aligned}
\Delta^{IPW} &= \frac{\delta_{c,\tilde{X}}\delta_{c,\tilde{Y}}}{\mathbb{P}\left[\delta_{c,\tilde{X}}\right]\mathbb{P}\left[\delta_{c,\tilde{Y}}\right]}(\mathbb{1}_{\tilde{Y}>\tilde{X}} - \mathbb{1}_{\tilde{Y}<\tilde{X}}) \\
&= \begin{cases} \frac{1}{\mathbb{P}\left[\delta_{c,\tilde{X}}\right]\mathbb{P}\left[\delta_{c,\tilde{Y}}\right]}(\mathbb{1}_{Y>X} - \mathbb{1}_{Y<X}), \text{ if no censoring} \\ 0, \text{ if censoring} \end{cases}
\end{aligned}$$

This is equivalent to weight the informative pairs (i.e. favorable, unfavorable and neutral) by the inverse of the complement of the probability of being uninformative. This is what is done by the argument `correction.tte` of `BuyseTest`. This works whenever the censoring mechanism is independent of the event times and we have a consistent estimate of $\mathbb{P}[\delta_c]$ since:

$$\begin{aligned}
\mathbb{E}\left[\Delta^{IPW}\right] &= \mathbb{E}\left[\mathbb{E}\left[\frac{\delta_{c,\tilde{X}}\delta_{c,\tilde{Y}}}{\mathbb{P}\left[\delta_{c,\tilde{X}}\right]\mathbb{P}\left[\delta_{c,\tilde{Y}}\right]}(\mathbb{1}_{\tilde{Y}>\tilde{X}} - \mathbb{1}_{\tilde{Y}<\tilde{X}})\middle|\tilde{X},\tilde{Y}\right]\right] \\
&= \mathbb{E}\left[\mathbb{E}\left[\frac{\delta_{c,\tilde{X}}\delta_{c,\tilde{Y}}}{\mathbb{P}\left[\delta_{c,\tilde{X}}\right]\mathbb{P}\left[\delta_{c,\tilde{Y}}\right]}\middle|\tilde{X},\tilde{Y}\right]\right]\mathbb{E}\left[\mathbb{1}_{Y>X} - \mathbb{1}_{Y<X}\right] \\
&= \frac{\mathbb{E}\left[\delta_{c,\tilde{X}}\delta_{c,\tilde{Y}}\right]}{\mathbb{P}\left[\delta_{c,\tilde{X}}\right]\mathbb{P}\left[\delta_{c,\tilde{Y}}\right]}\Delta = \frac{\mathbb{E}\left[\delta_{c,\tilde{X}}\right]\mathbb{E}\left[\delta_{c,\tilde{Y}}\right]}{\mathbb{P}\left[\delta_{c,\tilde{X}}\right]\mathbb{P}\left[\delta_{c,\tilde{Y}}\right]}\Delta \\
&= \Delta
\end{aligned}$$

where we used the law of total expectation (first line) and the independence between the censoring mecanisms.

5.1.3 Exponential distribution (no censoring)

Now let's assume that:

- $X_E \sim Exp(\alpha_{E,X})$.
- $Y_E \sim Exp(\alpha_{E,Y})$.
- $X_{CR} \sim Exp(\alpha_{CR,X})$.
- $Y_{CR} \sim Exp(\alpha_{CR,Y})$.

Then:

$$\begin{aligned}
\mathbb{P}[Y_E > X_E] &= \mathbb{P}[Y_E > X_E|\varepsilon_X = 1, \varepsilon_Y = 1]\,\mathbb{P}[\varepsilon_X = 1, \varepsilon_Y = 1] + \mathbb{P}[\varepsilon_X = 1, \varepsilon_Y = 2] \\
&= \frac{1}{(\alpha_{E,X} + \alpha_{CR,X})(\alpha_{E,Y} + \alpha_{CR,Y})}\left(\alpha_{E,X}\alpha_{E,Y}\frac{\alpha_{E,X}}{\alpha_{E,X} + \alpha_{E,Y}} + \alpha_{E,X}\alpha_{CR,Y}\right)
\end{aligned}$$

Just for comparison let's compare to the cumulative incidence. First we only consider one group and two competing events whose times to event follow an exponential distribution:

- $T_E \sim Exp(\alpha_E)$. The corresponding hazard function is $\lambda(t) = \alpha_E$.
- $T_{CR} \sim Exp(\alpha_{CR})$. The corresponding hazard function is $\lambda(t) = \alpha_{CR}$.

The cumulative incidence function can be written:

$$\begin{aligned}
CIF_1(t) &= \int_0^t \lambda_1(s) S(s_-) ds \\
&= \int_0^t \alpha_E \exp(-(\alpha_E + \alpha_{CR}) * s_-) ds \\
&= \frac{\alpha_E}{\alpha_E + \alpha_{CR}} \left[\exp(-(\alpha_E + \alpha_{CR}) * s_-)\right]_t^0 \\
&= \frac{\alpha_E}{\alpha_E + \alpha_{CR}} \left(1 - \exp(-(\alpha_E + \alpha_{CR}) * t_-)\right)
\end{aligned}$$

where $S(t)$ denote the event free survival and s_- denotes the right sided limit.

Then applying this formula in the case of two groups gives:

$$\begin{aligned}
CIF_1(t|group = X) &= \frac{\alpha_{E,X}}{\alpha_{E,X} + \alpha_{CR,X}} \left(1 - \exp(-(\alpha_{E,X} + \alpha_{CR,X}) * t_-)\right) \\
CIF_1(t|group = Y) &= \frac{\alpha_{E,Y}}{\alpha_{E,Y} + \alpha_{CR,Y}} \left(1 - \exp(-(\alpha_{E,Y} + \alpha_{CR,Y}) * t_-)\right)
\end{aligned}$$

5.2 In R

5.2.1 BuyseTest (no censoring)

Setting:

```
alphaE.X <- 2
alphaCR.X <- 1
alphaE.Y <- 3
alphaCR.Y <- 2
```

Simulate data:

```
set.seed(10)
df <- rbind(data.frame(time1 = rexp(n, rate = alphaE.X), time2 = rexp(n, rate = alphaCR
    .X), group = "1"),
        data.frame(time1 = rexp(n, rate = alphaE.Y), time2 = rexp(n, rate = alphaCR.Y),
     group = "2"))
df$time <- pmin(df$time1,df$time2) ## first event
df$event <- (df$time2<df$time1)+1 ## type of event
```

BuyseTest:

```r
e.BT <- BuyseTest(group ~ tte(time, censoring = event), data = df,
          method.inference = "none", method.tte = "Gehan",
          trace = 0)
summary(e.BT, percentage = TRUE)
```

```
       Generalized pairwise comparison with 1 prioritized endpoint

> statistic       : net chance of a better outcome (delta: endpoint specific, Delta: global)
> null hypothesis : Delta == 0
> treatment groups: 1 (control) vs. 2 (treatment)
> censored pairs  : uninformative pairs

> results
endpoint threshold total favorable unfavorable neutral uninf   delta   Delta
    time     1e-12   100      41.6       45.12   13.28     0 -0.0352 -0.0352
```

Note that without censoring one can get the same results by treating time as a continuous variable that take value ∞ when the competing risk is observed:

```r
df$timeXX <- df$time
df$timeXX[df$event==2] <- max(df$time)+1
e.BT.bis <- BuyseTest(group ~ cont(timeXX), data = df,
          method.inference = "none", trace = 0)
summary(e.BT.bis, percentage = TRUE)
```

```
       Generalized pairwise comparison with 1 prioritized endpoint

> statistic       : net chance of a better outcome (delta: endpoint specific, Delta: global)
> null hypothesis : Delta == 0
> treatment groups: 1 (control) vs. 2 (treatment)
> results
endpoint threshold total favorable unfavorable neutral uninf   delta   Delta
  timeXX     1e-12   100      41.6       45.12   13.28     0 -0.0352 -0.0352
```

Expected:

```r
weight <- (alphaE.X+alphaCR.X)*(alphaE.Y+alphaCR.Y)
exp <- list()
exp$favorable <- 1/weight*(alphaE.X*alphaE.Y*alphaE.X/(alphaE.X+alphaE.Y)+(alphaE.X*
    alphaCR.Y))
exp$unfavorable <- 1/weight*(alphaE.X*alphaE.Y*alphaE.Y/(alphaE.X+alphaE.Y)+(alphaE.Y*
    alphaCR.X))
exp$neutral <- alphaCR.X*alphaCR.Y/weight

100*unlist(exp)
```

```
favorable unfavorable     neutral
 42.66667    44.00000    13.33333
```

5.2.2 BuyseTest (with censoring)

Simulate data:

```
df$eventC <- df$event
df$eventC[rbinom(n, size = 1, prob = 0.2)==1] <- 0
```

BuyseTest (biased):

```
e.BTC <- BuyseTest(group ~ tte(time, censoring = eventC), data = df,
           method.inference = "none", method.tte = "Gehan",
           trace = 0)
summary(e.BTC, percentage = TRUE)
```

```
       Generalized pairwise comparison with 1 prioritized endpoint

> statistic       : net chance of a better outcome (delta: endpoint specific, Delta: global)
> null hypothesis : Delta == 0
> treatment groups: 1 (control) vs. 2 (treatment)
> censored pairs  : uninformative pairs

> results
endpoint threshold total favorable unfavorable neutral uninf   delta   Delta
    time     1e-12   100      31.1       35.15    8.65  25.1 -0.0406 -0.0406
```

BuyseTest (unbiased):

```
e.BTCC <- BuyseTest(group ~ tte(time, censoring = eventC), data = df,
           method.inference = "none", method.tte = "Gehan corrected",
           trace = 0)
summary(e.BTCC, percentage = TRUE)
```

```
       Generalized pairwise comparison with 1 prioritized endpoint

> statistic       : net chance of a better outcome (delta: endpoint specific, Delta: global)
> null hypothesis : Delta == 0
> treatment groups: 1 (control) vs. 2 (treatment)
> censored pairs  : uninformative pairs
                    IPW for uninformative pairs

> results
endpoint threshold total favorable unfavorable neutral uninf   delta   Delta
    time     1e-12   100     41.52       46.94   11.54     0 -0.0542 -0.0542
```

5.2.3 Cumulative incidence

Settings:

```
alphaE <- 2
alphaCR <- 1
```

Simulate data:

```
set.seed(10)
df <- data.frame(time1 = rexp(n, rate = alphaE), time2 = rexp(n, rate = alphaCR), group
     = "1", event = 1)
df$time <- pmin(df$time1,df$time2)
df$event <- (df$time2<df$time1)+1
```

Cumulative incidence (via risk regression):

```
e.CSC <- CSC(Hist(time, event) ~ 1, data = df)
vec.times <- unique(round(exp(seq(log(min(df$time)),log(max(df$time)),length.out = 12))
    ,2))
e.CSCpred <- predict(e.CSC, newdata = data.frame(X = 1), time = vec.times , cause = 1)
```

Expected vs. calculated:

```
cbind(time = vec.times,
      CSC = e.CSCpred$absRisk[1,],
      manual = alphaE/(alphaE+alphaCR)*(1-exp(-(alphaE+alphaCR)*(vec.times)))
      )
```

```
     time    CSC     manual
[1,] 0.00 0.0000 0.00000000
[2,] 0.01 0.0186 0.01970298
[3,] 0.02 0.0377 0.03882364
[4,] 0.05 0.0924 0.09286135
[5,] 0.14 0.2248 0.22863545
[6,] 0.42 0.4690 0.47756398
[7,] 1.24 0.6534 0.65051069
[8,] 3.70 0.6703 0.66665659
```

Could also be obtained treating the outcome as binary:

```
mean((df$time<=1)*(df$event==1))
```

```
[1] 0.6375
```