Codes: A Real Time Monitoring Approach for Bivariate Event Data

Inez Maria Zwetsloot, Tahir Mahmood, Funmilola Mary Taiwo and Zezhong Wang

12 July 2021

This file contains codes to reproduce the results of our paper Zwetsloot et al. (2021).

```
library(knitr)      ## For Writing operations
library(dplyr)      ## For Data operations
library(VGAM)       ## For Lambert W function
library(tidyverse)  ## For Writing operations
```

1. Required Functions

1.1 Function for GBE ATS when $\delta \neq 1$

1.1.1. Function to generate GBE random numbers

```
# general random data from bivariate GBE model
rgbe = function(n, par){
  U = runif(n,min=0,max=1)
  M = rbinom(n,size=1,prob=par[3])
  V1 = rexp(n, rate=1)
  V2 = rexp(n, rate=1)
  V = V1 + M*V2
  X1 = par[1]*(U^par[3])*V
  X2 = par[2]*((1-U)^par[3])*V
  X = matrix(data = NA,nrow=2,ncol=n)
  X[1,] = X1
  X[2,] = X2
  return(X)
}
```

1.1.2. Function for $C(x_1, x_2)$

```
fC = function(x1,x2,par){
  t1 = par[1];t2 = par[2];del = par[3]
  C = (x1/t1)^(1/del) + (x2/t2)^(1/del)
  return(C)
}
```

1.1.3. Function for GBE limits

```
CL.Z =  function(par,ATS){
  t1=par[1];t2=par[2];del=par[3]
  C11=fC(1,1,par)
  ETBE0=0.5*(t1+t2-(1/(C11^del)))
```

```
  alpha = ETBE0/ATS
  ucl= -(1/(C11^del))*log(alpha)
  return(ucl)
}

CL.M =  function(par,ATS,vX){
  t1=par[1];t2=par[2];del=par[3]
  C11=fC(1,1,par)
  ETBE0=0.5*(t1+t2-(C11^(-del)))
  alpha = ETBE0/ATS
  z = min(vX)
  v = as.numeric(vX[1]<vX[2])

  if (del==1){
    if (v==1){
       ucl = z - t2*log(alpha)
    } else if (v==0){
       ucl = z - t1*log(alpha)
    }
  } else if (del<1){
    if (v==1){
        G1 = (del/(1-del))*(fC(z,z,par)^del)*((1-alpha)*exp(-(fC(z,z,par)^del)))^(-del/(1-del))
        G2 = (del/(1-del))*(fC(z,z,par)^del)*(alpha*exp(-(fC(z,z,par)^del)))^(-del/(1-del))
        WG1 = lambertW(G1,tolerance = 1e-10,maxit=50)
        WG2 = lambertW(G2,tolerance = 1e-10,maxit=50)
        ucl = ((t2*(1-del)/del*WG2)^(1/del)-(z*t2/t1)^(1/del))^del
    } else if (v==0){
        G1 = (del/(1-del))*(fC(z,z,par)^del)*((1-alpha)*exp(-(fC(z,z,par)^del)))^(-del/(1-del))
        G2 = (del/(1-del))*(fC(z,z,par)^del)*(alpha*exp(-(fC(z,z,par)^del)))^(-del/(1-del))
        WG1 = lambertW(G1,tolerance = 1e-10,maxit=50)
        WG2 = lambertW(G2,tolerance = 1e-10,maxit=50)
        ucl = ((t1*(1-del)/del*WG2)^(1/del)-(z*t1/t2)^(1/del))^del
    }
  }
  return(ucl)
}
```

1.1.4. Function based on simulations to compute ATS

```
GBE_ATS = function(ATS,par0,par1){
  R=10000;vTS=rep(0,R)
  for (i in 1:R){
    #run chart as long as there is no signal
    vSignal = c(FALSE,FALSE)
    while(all(!vSignal)){
     vX = rgbe(n=1,par1) #generate data from MOBW
        Z = min(vX);M = max(vX)

        #update signal vector for first passage time
      UCL.z=CL.Z(par0,ATS) #ucl for Z
      UCL.m=CL.M(par0,ATS,vX) #lcl for M
        vSignal[1] = Z > UCL.z
        vSignal[2] = M > UCL.m
```

```
        if(all(vSignal == FALSE)){
          vTS[i] =vTS[i]+M
      } else if(vSignal[1] == TRUE){
          vTS[i] =vTS[i]+Z
        } else if(vSignal[2] == TRUE){
          vTS[i] =vTS[i]+M
        }
    }
  }
   ATS = mean(vTS)
   return(ATS)
}
```

1.2. Function for GBE ATS when $\delta = 1$

```
GBE_ATS1=function(ATS,par0,par1){
  t1=par0[1];t2=par0[2]
  C11=fC(1,1,par0)
  ETBE0=0.5*(t1+t2-C11^(-1))
  alpha = ETBE0/ATS

  t1_=par1[1];t2_=par1[2]
  C11_=fC(1,1,par1)
  ETBE1=0.5*(t1_+t2_-C11_^(-1))

  num = 2- alpha^(C11_/C11)
  Pns1 = 1-alpha^(C11_/C11)
  P1_=1/(t1_*C11_)
  P2_=1/(t2_*C11_)
  denum = 1 - Pns1*(P1_*(1-alpha^(t2/t2_))+P2_*(1-alpha^(t1/t1_)))
  ats1=num/denum*ETBE1
return(ats1)
}
```

1.3. Function for MOBE ATS

```
MOBE_ATS=function(ATS,par0,par1){
l1= par0[2];l2=par0[3];l12=par0[1]
l=l1+l2+l12
ETBE0 = 0.5*(l2/l^2+l2/l/(l1+l12)+l1/l^2+l1/l/(l2+l12))+l12/l^2
alpha = ETBE0/(ATS)

l_1= par1[2];l_2=par1[3];l_12=par1[1]
l_=l_1+l_2+l_12
ETBE1 = 0.5*(l_2/l_^2+l_2/l_/(l_1+l_12)+l_1/l_^2+l_1/l_/(l_2+l_12))+l_12/l_^2

Noevent = (l_1+l_2)/l_*(2-alpha^(l_/l))+l_12/l_
Psignal = alpha^(l_/l)+((l_1/l_)*alpha^((l_2+l_12)/(l2+l12))+(l_2/l_)*alpha^((l_1+l_12)/(l1+l12)))*(1-a
ARL1 = Noevent/Psignal
ATS1 <- ARL1*ETBE1
return(ATS1)
}
```

1.4. Function for MOBW ATS

```
MOBW_ATS=function(ATS,par0,par1){
l1= par0[3];l2=par0[4];l12=par0[2]
l=l1+l2+l12
eta=par0[1]
ETBE0 = 0.5*gamma(1+1/eta)*((l2+l12)^(-1/eta)-(l2+l12)*l^(-1-1/eta)+(l1+l12)^(-1/eta)-(l1+l12)*l^(-1-1/
alpha=ETBE0/(2*ATS)

l_1= par1[3];l_2=par1[4];l_12=par1[2]
l_=l_1+l_2+l_12
eta_ =par1[1]
ETBE1 = 0.5*gamma(1+1/eta_)*((l_2+l_12)^(-1/eta_)-(l_2+l_12)*l_^(-1-1/eta_)+(l_1+l_12)^(-1/eta_)-(l_1+l_
L <- l_/l
L1 <- (l_1+l_12)/(l1+l12)
L2 <- (l_2+l_12)/(l2+l12)
alpha_=alpha^L+1-(1-alpha)^L

num = 1 + (l_1+l_2)/l_*(1-alpha_)
dum1 = l_1/l_*(1-(1-alpha)^L2+alpha^L2)
dum2 = l_2/l_*(1-(1-alpha)^L1+alpha^L1)
denum = alpha_ +(1-alpha_)*(dum1+dum2)
#print((1-alpha_)*(dum1+dum2))
ARL1= num/denum
ATS1 <- ARL1*ETBE1
return(ATS1)
}
```

2. Function to compute BTBE ATS values

```
BTBE_ATS=function(ATS,par0,par1,Dist){
  if(Dist=="GBE"){
    if(par0[3]!=1){
    print("ATS is based on 10,000 Simulations")
    ATS=GBE_ATS(ATS,par0,par1)
    }else if(par1[3]==1){
    print("ATS based on analytical expression")
    ATS=GBE_ATS1(ATS,par0,par1)
    }
  }else if (Dist=="MOBE"){
    ATS=MOBE_ATS(ATS,par0,par1)
  }else if (Dist=="MOBW"){
    ATS=MOBW_ATS(ATS,par0,par1)
  }
  return(ATS)
}
```

2.1. Examples of BTBE chart under GBE distribution

```
### When delta is equal to 1
ATS=200
par0=c(5,5,1)
```

```r
par1=c(5,5,1)
BTBE_ATS(ATS,par0,par1,Dist="GBE")
```

```
## [1] "ATS based on analytical expression"
```

```
## [1] 200
```

```r
### When delta is not equal to 1
ATS=200
par0=c(5,5,0.5)
par1=c(5,5,0.5)
BTBE_ATS(ATS,par0,par1,Dist="GBE")
```

```
## [1] "ATS is based on 10,000 Simulations"
```

```
## [1] 199.4755
```

2.2. Example of BTBE chart under MOBE distribution

```r
ATS=200
par0=c(1,5,5,0)
par1=c(1,5,5,0)
BTBE_ATS(ATS,par0,par1,Dist="MOBE")
```

```
## [1] 200
```

2.3. Example of BTBE chart under MOBW distribution

```r
ATS=200
par0=c(2,5,5,0)
par1=c(2,5,5,0)
BTBE_ATS(ATS,par0,par1,Dist="MOBW")
```

```
## [1] 200
```

References

Zwetsloot, Inez Maria, Tahir Mahmood, Funmilola Mary Taiwo, and Wang Zezhong. 2021. "A Real Time Monitoring Approach for Bivariate Event Data." *Submitted for Publication.*