



## *Apêndice E - Algoritmo de Turnbull*

### Capítulo 8 - Modelos de Regressão Paramétricos

1. Preparar data = arquivo de dados com as colunas left e right  
left = limite inferior do intervalo  
right = limite superior do intervalo
2. Usar os comandos a seguir para obter p e A

```
cria.tau <- function(data){  
  l <- data$left  
  r <- data$right  
  tau <- sort(unique(c(l,r[is.finite(r)])))  
  return(tau)  
}  
  
S.ini <- function(tau){  
  m<-length(tau)  
  ekm<-survfit(Surv(tau[1:m-1],rep(1,m-1))~1)  
  So<-c(1,ekm$surv)  
  p <- -diff(So)  
  return(p)  
}  
  
cria.A <- function(data,tau){  
  tau12 <- cbind(tau[-length(tau)],tau[-1])  
  interv <- function(x,inf,sup) ifelse(x[1]>=inf & x[2]<=sup,1,0)  
  A <- apply(tau12,1,interv,inf=data$left,sup=data$right)  
  id.lin.zero <- which(apply(A==0, 1, all))  
  if(length(id.lin.zero)>0) A <- A[-id.lin.zero, ]  
  return(A)  
}
```

---

3. Carregar a função Turnbull.R no R usando os comandos

```
> source("https://docs.ufpr.br/~giolo/Livro/ApendiceE/Turnbull.R (https://docs.ufpr.br/~giolo  
/Livro/ApendiceE/Turnbull.R)")
```

4. Executar o algoritmo de Turnbull usando os comandos

```
> Turnbull(p,A)
```

---

- Função Turnbull.R mencionada em 3.

```
Turnbull <- function(p, A, data, eps=1e-3,
                    iter.max=200, verbose=FALSE){
  n<-nrow(A)
  m<-ncol(A)
  Q<-matrix(1,m)
  iter <- 0
  repeat {
    iter <- iter + 1
    diff<- (Q-p)
    maxdiff<-max(abs(as.vector(diff)))
    if (verbose)
      print(maxdiff)
    if (maxdiff<eps | iter>=iter.max)
      break
    Q<-p
    C<-A%%p
    p<-p*((t(A)%%(1/C))/n)
  }
  cat("Iterations = ", iter,"\n")
  cat("Max difference = ", maxdiff,"\n")
  cat("Convergence criteria: Max difference < 1e-3","\n")
  dimnames(p)<-list(NULL,c("P Estimate"))
  surv<-round(c(1,1-cumsum(p)),digits=5)
  right <- data$right
  if(any(!(is.finite(right)))){
    t <- max(right[is.finite(right)])
    return(list(time=tau[tau<t],surv=surv[tau<t]))
  }
  else
    return(list(time=tau,surv=surv))
}
```