

## TD 4 : Etude d'un jeu de données

### Exercice 1

Les données que vous allez analyser proviennent d'un échantillon d'une étude de survie menée dans le comté d'Evans en Géorgie aux Etats-Unis de 1967 à 1980. Les durées de survie (en années) sont recueillies pour 2 groupes, chacun de taille 25. Le groupe 1 est constitué de patients qui n'ont pas de maladies chroniques connues, tandis que dans le groupe 2 les patients sont suivis pour une ou plusieurs maladies chroniques.

```
evans <- data.frame(T1 =  
c(12.3,5.4,8.2,2.5,11.7,10,5.7,9.8,2.6,11,9.2,12.1,6.6,2.2,1.8,10.2,10.7,  
11.1,5.3,3.5,9.2,12.2,8.7,3.8,3,5.8,2.9,8.4,8.3,9.1,4.2,4.1,1.8,3.1,11.4,  
2.4,1.4,5.9,1.6,2.8,4.9,3.5,6.5,9.9,3.6,5.2,8.8,7.8,4.7,3.9),  
Indicator = c(0,1,1,0,rep(1,7),0,rep(1,13),rep(1,25)),  
CHR = c(rep(0,25),rep(1,25)))
```

On soumet le code R suivant :

code 1

```
## Loading required package: ggplot2  
## Loading required package: ggpubr  
## Call: survfit(formula = evans.s ~ CHR, data = evans, type = "kaplan-  
meier",  
##      conf.type = "plain")  
##  
##              CHR=0  
##   time n.risk n.event survival std.err lower 95% CI upper 95% CI  
##   1.8     25      1  0.9600  0.0392    0.8832    1.000  
##   2.2     24      1  0.9200  0.0543    0.8137    1.000  
##   2.6     22      1  0.8782  0.0660    0.7489    1.000  
##   3.0     21      1  0.8364  0.0749    0.6895    0.983  
##   3.5     20      1  0.7945  0.0820    0.6338    0.955  
##   3.8     19      1  0.7527  0.0877    0.5808    0.925  
##   5.3     18      1  0.7109  0.0923    0.5301    0.892  
##   5.4     17      1  0.6691  0.0959    0.4812    0.857  
##   5.7     16      1  0.6273  0.0986    0.4341    0.820  
##   6.6     15      1  0.5855  0.1005    0.3885    0.782  
##   8.2     14      1  0.5436  0.1016    0.3444    0.743  
##   8.7     13      1  0.5018  0.1021    0.3018    0.702  
##   9.2     12      2  0.4182  0.1007    0.2208    0.616  
##   9.8     10      1  0.3764  0.0990    0.1824    0.570  
##  10.0      9      1  0.3345  0.0964    0.1456    0.523  
##  10.2      8      1  0.2927  0.0930    0.1105    0.475
```

```
## 10.7      7      1  0.2509  0.0886      0.0773      0.425
## 11.0      6      1  0.2091  0.0831      0.0462      0.372
## 11.1      5      1  0.1673  0.0763      0.0177      0.317
## 11.7      4      1  0.1255  0.0677      0.0000      0.258
## 12.2      2      1  0.0627  0.0558      0.0000      0.172
##
##                               CHR=1
## time n.risk n.event survival std.err lower 95% CI upper 95% CI
## 1.4    25      1    0.96  0.0392    0.8832    1.000
## 1.6    24      1    0.92  0.0543    0.8137    1.000
## 1.8    23      1    0.88  0.0650    0.7526    1.000
## 2.4    22      1    0.84  0.0733    0.6963    0.984
## 2.8    21      1    0.80  0.0800    0.6432    0.957
## 2.9    20      1    0.76  0.0854    0.5926    0.927
## 3.1    19      1    0.72  0.0898    0.5440    0.896
## 3.5    18      1    0.68  0.0933    0.4971    0.863
## 3.6    17      1    0.64  0.0960    0.4518    0.828
## 3.9    16      1    0.60  0.0980    0.4080    0.792
## 4.1    15      1    0.56  0.0993    0.3654    0.755
## 4.2    14      1    0.52  0.0999    0.3242    0.716
## 4.7    13      1    0.48  0.0999    0.2842    0.676
## 4.9    12      1    0.44  0.0993    0.2454    0.635
## 5.2    11      1    0.40  0.0980    0.2080    0.592
## 5.8    10      1    0.36  0.0960    0.1718    0.548
## 5.9     9      1    0.32  0.0933    0.1371    0.503
## 6.5     8      1    0.28  0.0898    0.1040    0.456
## 7.8     7      1    0.24  0.0854    0.0726    0.407
## 8.3     6      1    0.20  0.0800    0.0432    0.357
## 8.4     5      1    0.16  0.0733    0.0163    0.304
## 8.8     4      1    0.12  0.0650    0.0000    0.247
## 9.1     3      1    0.08  0.0543    0.0000    0.186
## 9.9     2      1    0.04  0.0392    0.0000    0.117
## 11.4     1      1    0.00    NaN      NaN      NaN
```

### Question 1

Décrire ce que produit chaque ligne de ce code R, en précisant bien le rôle de chacune des options utilisées. (On identifiera le nom de la variable des durées de vie et de la variable indicatrice du décès)

```
library(survminer)
library(survival)# des jeux de données de La Littérature
library(bpcp)# données de Freireich

evans.s <- Surv(time = evans$T1,event = evans$Indicator)#"Surv" met Les
données du dataframe au format données de survie
#La commande Surv permet de créer des variables de survie.
km <- survfit(evans.s~CHR,data=evans,type="kaplan-
meier",conf.type="plain")#La commande survfit permet de calculer La fonction
de survie.
```

```
#Variable durées de vie : time (T1)variable de durée réellement observée
#Variable indicatrice du décès : event (Indicator)indicatrice qui vaut 0 ou 1
: indicatrice de décès
summary(km) #description statistique de km
```

```
## Call: survfit(formula = evans.s ~ CHR, data = evans, type = "kaplan-
meier",
```

```
##   conf.type = "plain")
```

```
##
```

```
##           CHR=0
```

##	time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
##	1.8	25	1	0.9600	0.0392		0.8832		1.000
##	2.2	24	1	0.9200	0.0543		0.8137		1.000
##	2.6	22	1	0.8782	0.0660		0.7489		1.000
##	3.0	21	1	0.8364	0.0749		0.6895		0.983
##	3.5	20	1	0.7945	0.0820		0.6338		0.955
##	3.8	19	1	0.7527	0.0877		0.5808		0.925
##	5.3	18	1	0.7109	0.0923		0.5301		0.892
##	5.4	17	1	0.6691	0.0959		0.4812		0.857
##	5.7	16	1	0.6273	0.0986		0.4341		0.820
##	6.6	15	1	0.5855	0.1005		0.3885		0.782
##	8.2	14	1	0.5436	0.1016		0.3444		0.743
##	8.7	13	1	0.5018	0.1021		0.3018		0.702
##	9.2	12	2	0.4182	0.1007		0.2208		0.616
##	9.8	10	1	0.3764	0.0990		0.1824		0.570
##	10.0	9	1	0.3345	0.0964		0.1456		0.523
##	10.2	8	1	0.2927	0.0930		0.1105		0.475
##	10.7	7	1	0.2509	0.0886		0.0773		0.425
##	11.0	6	1	0.2091	0.0831		0.0462		0.372
##	11.1	5	1	0.1673	0.0763		0.0177		0.317
##	11.7	4	1	0.1255	0.0677		0.0000		0.258
##	12.2	2	1	0.0627	0.0558		0.0000		0.172

```
##
```

```
##           CHR=1
```

##	time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
##	1.4	25	1	0.96	0.0392		0.8832		1.000
##	1.6	24	1	0.92	0.0543		0.8137		1.000
##	1.8	23	1	0.88	0.0650		0.7526		1.000
##	2.4	22	1	0.84	0.0733		0.6963		0.984
##	2.8	21	1	0.80	0.0800		0.6432		0.957
##	2.9	20	1	0.76	0.0854		0.5926		0.927
##	3.1	19	1	0.72	0.0898		0.5440		0.896
##	3.5	18	1	0.68	0.0933		0.4971		0.863
##	3.6	17	1	0.64	0.0960		0.4518		0.828
##	3.9	16	1	0.60	0.0980		0.4080		0.792
##	4.1	15	1	0.56	0.0993		0.3654		0.755
##	4.2	14	1	0.52	0.0999		0.3242		0.716
##	4.7	13	1	0.48	0.0999		0.2842		0.676
##	4.9	12	1	0.44	0.0993		0.2454		0.635
##	5.2	11	1	0.40	0.0980		0.2080		0.592

##	5.8	10	1	0.36	0.0960	0.1718	0.548
##	5.9	9	1	0.32	0.0933	0.1371	0.503
##	6.5	8	1	0.28	0.0898	0.1040	0.456
##	7.8	7	1	0.24	0.0854	0.0726	0.407
##	8.3	6	1	0.20	0.0800	0.0432	0.357
##	8.4	5	1	0.16	0.0733	0.0163	0.304
##	8.8	4	1	0.12	0.0650	0.0000	0.247
##	9.1	3	1	0.08	0.0543	0.0000	0.186
##	9.9	2	1	0.04	0.0392	0.0000	0.117
##	11.4	1	1	0.00	NaN	NaN	NaN

## Question 2

A l'aide des sorties produites par le code 1 données ci-après, répondre aux questions suivantes :

- a) Donner la signification de n.risk, n.event, survival, std.err, lower 95% CI et upper 95% CI.
  - n.risk :le vecteur contenant pour tout  $i$  n.risk[i] = le nombre de sujets exposés au risque de decès au temps  $t(i)$
  - n.event :le vecteur contenant pour tout  $i$  n.event[i] = le nombre de decès observés au temps  $t(i)$
  - survival :le vecteur contenant pour tout  $i$  survival[i] = estimateur de Kaplan-Meier  $S_{\hat{}}(t_i)$  de la fonction de survie  $S(t_i)$  au temps de decès  $t(i)$
  - std.err :le vecteur contenant pour tout  $i$  std.err[i] = erreur standard de l'estimateur de Kaplan Meier au temps  $t(i)$
  - lower 95% CI :le vecteur contenant pour tout  $i$  lower[i] = borne inf de l'intervalle de confiance de niveau
  - upper 95% CI :le vecteur contenant pour tout  $i$  upper[i] = borne sup de l'intervalle de confiance de niveau
- b) Retrouver les valeurs des durées qui correspondent à des censures dans chacun des deux groupes. Combien y a t'il de durées censurées dans chaque groupe ?

```

evans$T1[evans$Indicator==0 & evans$CHR==0]
## [1] 12.3 2.5 12.1
evans$T1[evans$Indicator==0 & evans$CHR==1]
## numeric(0)

```

Dans le groupe CHR = 0, nous avons comme censures 3 valeurs qui sont : - 12.3 - 2.5 - 12.1

Dans le groupe CHR = 1, nous avons aucunes censures.

- c) Quatre valeurs numériques ont été effacées dans la sortie du groupe CHR=0 et remplacées par XXXXXX : retrouver ces valeurs par le calcul (résultats donnés à 10<sup>-4</sup> près) en donnant les formules utilisées.

$$survival(t(j)) = \hat{S}(t(j)) = \prod_{i=1}^j \left(1 - \frac{m_i}{n_i}\right) \quad std.err(\hat{S}(t(i))) = \hat{\sigma}(\hat{S}(t(i))) = \hat{S}(t(i)) \sqrt{\sum_{j=1}^i \frac{m_j}{n_j(n_j - m_j)}} \\ lower(t) = \hat{S}(t) - \hat{\sigma}(\hat{S}(t)) t_{1-\frac{\alpha}{2}} \quad upper(t) = \hat{S}(t) + \hat{\sigma}(\hat{S}(t)) t_{1-\frac{\alpha}{2}}$$

```
#S_hat
survival<-function(a,table){
  index<-which(table$time==a)
  S_hat<-table$surv[index-1]*(1-(table$n.event[index]/table$n.risk[index]))
  Sum <- 0
  for (i in 1:index){
    Sum <- Sum + table$n.event[i]/(table$n.risk[i]*(table$n.risk[i]-
    table$n.event[i]))
  }
  std <- S_hat * sqrt(Sum)
  quantile <- qnorm(0.05/2,mean=0,sd = 1,lower.tail=TRUE)
  upper <- S_hat-std*quantile
  lower <- S_hat+std*quantile
  return(list('valeur1'=S_hat, 'valeur2'=std, 'valeur3'=lower, 'valeur4'=upper))
}
valeur1<-survival(2.6,km)$valeur1
valeur1

## [1] 0.8781818

valeur2<-survival(2.6,km)$valeur2
valeur2

## [1] 0.06596755

valeur3<-survival(2.6,km)$valeur3
valeur3

## [1] 0.7488878

valeur4<-survival(2.6,km)$valeur4
valeur4

## [1] 1.007476
```

Question 3.

Dans la fonction survfit, on modifie l'option conf.type="log-log". Expliquer ce que produit cette option. Refaire les calculs de la Question 2. c) dans ce cas.

La formule conf.type="log-log" (log-hazard scale) pour les intervalles de confiance pour  $S(t)$  est obtenue par double exponentiation à partir des intervalles de confiance pour  $\log(\Lambda(t))$  :

$$S(t) \in \left[ \exp \left( -\exp \left( \log(-\log(\hat{S}(t))) \pm z_{1-\alpha/2} \cdot \frac{\widehat{se}(\hat{\Lambda}(t))}{\hat{\Lambda}(t)} \right) \right) \right]$$

$$survival(t(j)) = \hat{S}(t(j)) = \prod_{i=1}^j \left( 1 - \frac{m_i}{n_i} \right) \text{std.err}(Shat(t(i)) = \hat{\sigma}(\hat{S}(t(i))) =$$

$$\hat{S}(t(i)) \sqrt{\sum_{j=1}^i \frac{m_j}{n_j(n_j - m_j)}} \text{std.err}_{\log-\log}(Shat(t(i)) = \hat{\sigma}_{\log-\log}(\hat{S}(t(i))) =$$

$$\sqrt{\frac{1}{\log^2(\hat{S}(t(i)))} \sum_{j=1}^i \frac{m_j}{n_j(n_j - m_j)}} \text{lower}(t) = \log(-\log(\hat{S}(t))) - \hat{\sigma}_{\log-\log}(\hat{S}(t)) t_{1-\frac{\alpha}{2}} \text{upper}(t) =$$

$$\log(-\log(\hat{S}(t))) + \hat{\sigma}_{\log-\log}(\hat{S}(t)) t_{1-\frac{\alpha}{2}}$$

```
km1 <- survfit(evans.s~CHR,data=evans,type="kaplan-meier",conf.type="log-
log")
#permet de calculer la table de survie avec log-log
summary(km1)#affiche la table de survie et des informations supplémentaires

## Call: survfit(formula = evans.s ~ CHR, data = evans, type = "kaplan-
meier",
##      conf.type = "log-log")
##
##                CHR=0
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##   1.8     25      1   0.9600  0.0392   0.74839      0.994
##   2.2     24      1   0.9200  0.0543   0.71639      0.979
##   2.6     22      1   0.8782  0.0660   0.66798      0.959
##   3.0     21      1   0.8364  0.0749   0.62045      0.935
##   3.5     20      1   0.7945  0.0820   0.57447      0.909
##   3.8     19      1   0.7527  0.0877   0.53007      0.881
##   5.3     18      1   0.7109  0.0923   0.48717      0.851
##   5.4     17      1   0.6691  0.0959   0.44566      0.819
##   5.7     16      1   0.6273  0.0986   0.40549      0.786
##   6.6     15      1   0.5855  0.1005   0.36660      0.752
##   8.2     14      1   0.5436  0.1016   0.32896      0.716
##   8.7     13      1   0.5018  0.1021   0.29255      0.679
##   9.2     12      2   0.4182  0.1007   0.22350      0.602
##   9.8     10      1   0.3764  0.0990   0.19093      0.562
##  10.0      9      1   0.3345  0.0964   0.15978      0.520
##  10.2      8      1   0.2927  0.0930   0.13014      0.477
##  10.7      7      1   0.2509  0.0886   0.10219      0.433
##  11.0      6      1   0.2091  0.0831   0.07618      0.386
##  11.1      5      1   0.1673  0.0763   0.05245      0.338
##  11.7      4      1   0.1255  0.0677   0.03157      0.287
##  12.2      2      1   0.0627  0.0558   0.00553      0.229
##
##                CHR=1
##  time n.risk n.event survival std.err lower 95% CI upper 95% CI
##   1.4     25      1    0.96  0.0392   0.74839      0.994
##   1.6     24      1    0.92  0.0543   0.71639      0.979
```

##	1.8	23	1	0.88	0.0650	0.67256	0.960
##	2.4	22	1	0.84	0.0733	0.62806	0.937
##	2.8	21	1	0.80	0.0800	0.58445	0.911
##	2.9	20	1	0.76	0.0854	0.54205	0.884
##	3.1	19	1	0.72	0.0898	0.50089	0.855
##	3.5	18	1	0.68	0.0933	0.46093	0.825
##	3.6	17	1	0.64	0.0960	0.42215	0.794
##	3.9	16	1	0.60	0.0980	0.38449	0.761
##	4.1	15	1	0.56	0.0993	0.34794	0.727
##	4.2	14	1	0.52	0.0999	0.31249	0.692
##	4.7	13	1	0.48	0.0999	0.27813	0.656
##	4.9	12	1	0.44	0.0993	0.24490	0.619
##	5.2	11	1	0.40	0.0980	0.21282	0.581
##	5.8	10	1	0.36	0.0960	0.18195	0.542
##	5.9	9	1	0.32	0.0933	0.15237	0.502
##	6.5	8	1	0.28	0.0898	0.12421	0.460
##	7.8	7	1	0.24	0.0854	0.09762	0.417
##	8.3	6	1	0.20	0.0800	0.07284	0.372
##	8.4	5	1	0.16	0.0733	0.05020	0.325
##	8.8	4	1	0.12	0.0650	0.03026	0.277
##	9.1	3	1	0.08	0.0543	0.01391	0.225
##	9.9	2	1	0.04	0.0392	0.00289	0.170
##	11.4	1	1	0.00	NaN	NA	NA

- c) Quatre valeurs numériques ont été effacées dans la sortie du groupe CHR=0 et remplacées par XXXXXX : retrouver ces valeurs par le calcul (résultats donnés à 10<sup>-4</sup> près) en donnant les formules utilisées.

```
survival_log<-function(a,table){
  index<-which(table$time==a)
  S_hat<-table$surv[index-1]*(1-(table$n.event[index]/table$n.risk[index]))
  Sum<-0

  for (i in 1:index){
    Sum <- Sum+ table$n.event[i]/(table$n.risk[i]*(table$n.risk[i]-
    table$n.event[i]))
  }
  std_log<-S_hat * sqrt(Sum)

  quantile<-qnorm(0.05/2,mean=0,sd = 1,lower.tail=TRUE)
  upper<-S_hat^(exp((std_log)*quantile))
  lower<-S_hat^(exp((std_log)*(-quantile)))

  return(list('valeur1'=S_hat, 'valeur2'=std_log, 'valeur3'=lower, 'valeur4'=upper
  ))
}

valeur1<-survival_log(2.6,km1)$valeur1
valeur1

## [1] 0.8781818
```

```

valeur2<-survival_log(2.6,km1)$valeur2
valeur2

## [1] 0.06596755

valeur3<-survival_log(2.6,km1)$valeur3
valeur3

## [1] 0.8625767

valeur4<-survival_log(2.6,km1)$valeur4
valeur4

## [1] 0.8921272

```

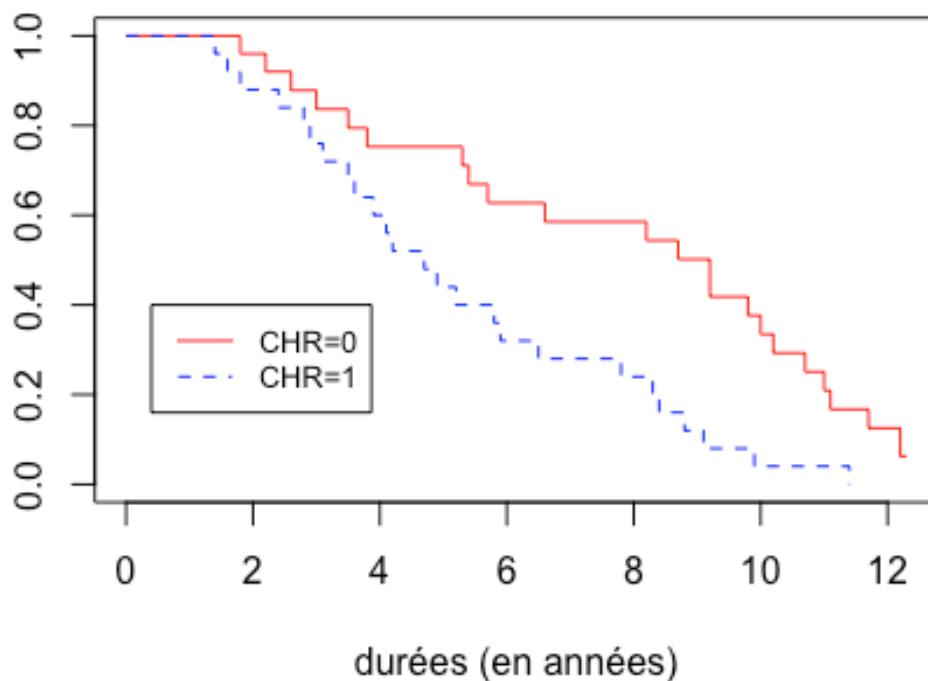
Question 4.

On soumet la commande R suivante :

```

plot(km,lty=1:2,xlab="durées (en années)",col=c('red','blue'))
legend(0.4,0.4,legend=c("CHR=0","CHR=1"),col=c("red","blue"),lty=1:2,cex=0.8)

```



Elle produit le graphique ci-dessus. Compléter la légende qui a été effacée. Que représentent ces courbes ? D'après ce graphique, dans quel groupe peut-on supposer qu'il y a une meilleure survie ? justifier votre réponse.



Ces courbes représentent la fonction de survie en fonction du groupe. En rouge on a la survie pour le groupe CHR=0 et en bleu la survie pour le groupe CHR=1. On peut supposer que dans le groupe 0 la survie est meilleure car la courbe de l'estimateur de survie du groupe 0 est supérieur à celle du groupe 1.

Question 5.

On soumet le code R suivant :

code 2

```
survdifff(evans.s~CHR,data=evans,rho=1)

## Call:
## survdifff(formula = evans.s ~ CHR, data = evans, rho = 1)
##
##           N Observed Expected (O-E)^2/E (O-E)^2/V
## CHR=0  25      9.72     14.9       1.77       6.47
## CHR=1  25     15.34     10.2       2.58       6.47
##
##  Chisq= 6.5  on 1 degrees of freedom, p= 0.01

survdifff(evans.s~CHR,data=evans,rho=0)

## Call:
## survdifff(formula = evans.s ~ CHR, data = evans, rho = 0)
##
##           N Observed Expected (O-E)^2/E (O-E)^2/V
## CHR=0  25      22     31.3       2.74       9.12
## CHR=1  25      25     15.7       5.45       9.12
##
##  Chisq= 9.1  on 1 degrees of freedom, p= 0.003
```

a) Quel est le test mis en œuvre par la fonction survdifff : on précisera l'hypothèse nulle.

La fonction survdifff du package survival permet le test d'une différence de survie statistiquement significative entre plusieurs sous-groupes.

- rho = 1 : permet d'effectuer un test de type Wilcoxon
- rho = 0 : permet d'effectuer un test de type log-rank

H0 : pas de différence de survie entre les deux groupes étudiés H1 : différence de survie entre les deux groupes étudiés

b) Expliquer l'effet de l'option rho=0 ou rho=1

- rho = 0 : permet d'effectuer un test de type log-rank c'est à dire que tous les individus ont le même poids égale à 1.
- rho = 1 : permet d'effectuer un test de type Wilcoxon généralisé c'est à dire que Les individus dont le decès est precoces ont plus de poids. Il montre une difference entre les courbes de survies sur les survies de courtes durées.

- c) Commenter les résultats produits par le code 2. En particulier, après avoir donné la règle de décision de chacun des tests, vous expliquerez la différence observée entre les deux statistiques de tests et leur p-value respective, selon que l'on utilise l'option rho=0 ou rho=1

A FAIIIRREEEEEEEEE!!!!!!!

#### Question 6

On soumet le code R suivant :

code 3

```
cox <- coxph(evans.s~CHR, data=evans, method = "efron")#ajuste un modèle de
Cox
summary(cox)

## Call:
## coxph(formula = evans.s ~ CHR, data = evans, method = "efron")
##
##      n= 50, number of events= 47
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## CHR 0.9254      2.5229   0.3161  2.927  0.00342 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##      exp(coef) exp(-coef) lower .95 upper .95
## CHR      2.523      0.3964      1.358      4.688
##
## Concordance= 0.607 (se = 0.039 )
## Likelihood ratio test= 8.64 on 1 df,  p=0.003
## Wald test               = 8.57 on 1 df,  p=0.003
## Score (logrank) test = 9.05 on 1 df,  p=0.003
```

- a) Rappeler la définition du modèle de Cox. On précisera bien quelle(s) est(sont) le(s) covariable(s) ici.

Un modèle de régression adapté au cas d'une variable à expliquer censurée comme la variable de durée de survie. Un modèle de Cox va permettre de relier la durée de survie d'un patient à plusieurs covariables.

Formule générale du modèle de Cox :

$$h(t, X_i) = h_0(t)e^{\beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k}$$

avec :

- $h$ : risque
- $h_0$ : risque de base
- $t$ : temps

- $X_i$ : variables explicatives ou prédictives
- $\beta_i$ : coefficients de la régression

Les covariables sont l'appartenance d'un patient au groupe CHR = 0 et CHR = 1.

- b) Retrouver par le calcul (résultats donnés à  $10^{-4}$  près) les cinq valeurs numériques qui ont été remplacées par XXXXXX

```
c<-log(2.5229)
c

## [1] 0.925409

z<-c/0.3161
z

## [1] 2.927583

exp(c)

## [1] 2.5229

exp(-c)

## [1] 0.3963693

W<-z^2
W

## [1] 8.570743
```

- c) Que signifie l'option method=efron ? Quelle autre option connaissez-vous ? Laquelle est-il préférable d'utiliser ici ? (on ne demande pas de donner l'expression explicite des formules)

L'option method spécifie la méthode employée pour traiter les ex aequo. S'il n'y a aucun ex aequo, dans les données, alors toutes les méthodes sont équivalentes.

Le logiciel R utilise par défaut la méthode d'Efron qui est plus précise et néanmoins moins gourmande en temps de calcul que la méthode de Breslow.

- d) Peut-on dire que le coefficient associé à la variable CHR dans le modèle est significativement différent de 0 au risque  $\alpha = 5\%$  ? Que peut-on alors en déduire en ce qui concerne le risque de décès dans le groupe 2 par rapport au groupe 1 ? Vous justifierez vos réponses à l'aide des sorties produites par le code 3.
- e) Peut-on interpréter la valeur de  $\exp(\text{coef})$  ? Si oui donner cette interprétation.
- f) Quel lien peut-on faire entre les sorties du code 2 et du code 3 ?