

# Courbes de survie avec R

Karine Chalvet-Monfray

10 janvier 2019

Il faut connaître au moins deux informations pour tous les sujets de l'étude.

- ▶ Le temps de participation (tdp) = variable quantitative
- ▶ L'état à la fin du temps de participation vivant ou mort (etat) = variable dichotomique

```
d <- read.table("DATA/survie.txt", header = TRUE)
str(d)
```

```
## 'data.frame':    7 obs. of  2 variables:
## $ tdp : int  2 6 7 9 9 11 15
## $ etat: int  1 1 0 1 0 0 0
```

Il est à noter que la variable dichotomique est noté 0 et 1.

- ▶  $\text{etat} = 0$  : donnée censurée (sujet vivant à la fin du temps de participation)
- ▶  $\text{etat} = 1$  : donnée non-censurée (sujet mort à la fin du temps de participation)

Si nous reprenons l'exemple du cours...

```
head(d, 7)
```

```
##   tdp etat
## 1    2    1
## 2    6    1
## 3    7    0
## 4    9    1
## 5    9    0
## 6   11    0
## 7   15    0
```

## Comment je fais si je n'ai pas le temps de participation [1].

Prenons un autre exemple d'environ 600 sujets

```
d2<-read.table("DATA/d2019.txt",header=T)
str(d2$DTCONT)
```

```
## Factor w/ 465 levels "01/01/1980","01/01/1984",...: 113
```

```
d2$cont<-as.Date(d2$DTCONT,"%d/%m/%Y")
str(d2$cont)
```

```
## Date[1:605], format: "1976-05-11" "1980-05-14" "1978-12
```

```
d2$ddn<-as.Date(d2$DDN,"%d/%m/%Y")
```

## Comment je fais si je n'ai pas le temps de participation [2].

Prenons un autre exemple d'environ 600 sujet

Si nous reprenons l'exemple du cours...

```
d2$TDP<-d2$ddn-d2$cont  
str(d2$TDP)
```

```
##  'difftime' num [1:605] 5488 4462 4968 4257 ...  
##  - attr(*, "units")= chr "days"
```

## Pour réaliser une courbe de survie [1]. . .

- ▶ Il faut utiliser le package “survival” que l’on doit installer au préalable

```
library(survival)
```

- ▶ Créer un objet de données de survie avec la fonction `Surv(variable temps de participation, variable état)`

```
(ds <- Surv(d$tdp, d$etat))
```

```
## [1] 2 6 7+ 9 9+ 11+ 15+
```

## Pour réaliser une courbe de survie [2]. . .

- ▶ Estimer le taux survie avec la fonction `survfit()`. Attention “~ 1” signifie que l’on représente en fonction d’un groupe unique; ce terme est nécessaire. En cas d’absence, il y aura un message d’erreur.

```
(ts <- survfit(ds ~ 1))
```

```
## Call: survfit(formula = ds ~ 1)
```

```
##
```

```
##           n  events  median 0.95LCL 0.95UCL
```

```
##           7         3      NA         6      NA
```

## Pour réaliser une courbe de survie [3]...

- ▶ La fonction `summary` permet de visualiser les valeur des taux de survie et leur intervalle de confiance pour chaque temps où il y a un décès.

```
summary(ts)
```

```
## Call: survfit(formula = ds ~ 1)
```

```
##
```

```
##   time  n.risk  n.event  survival  std.err  lower 95% CI upper
```

```
##     2      7      1    0.857    0.132    0.633
```

```
##     6      6      1    0.714    0.171    0.447
```

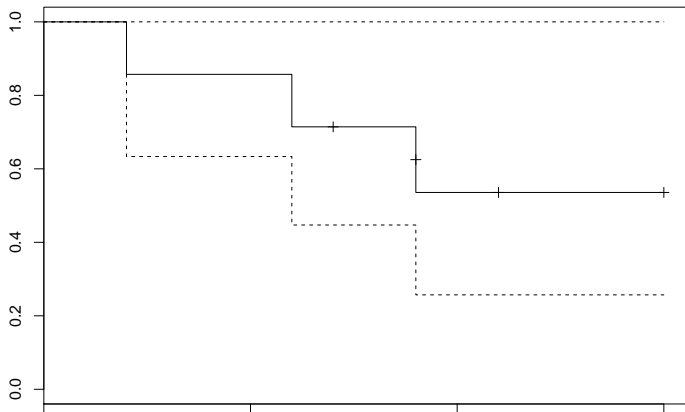
```
##     9      4      1    0.536    0.201    0.257
```



## Pour réaliser une courbe de survie [4]...

La fonction `plot` permet de faire une représentation de la courbe de survie avec son intervalle de confiance.

```
plot(ts, mark.time = TRUE)
```



Si nous reprenons l'exemple du cours...

d2

##	tdp	etat	groupe
## 1	2	1	A
## 2	6	1	A
## 3	7	0	A
## 4	9	1	A
## 5	9	0	A
## 6	11	0	A
## 7	15	0	A
## 8	2	1	B
## 9	4	1	B
## 10	7	0	B
## 11	8	1	B
## 12	9	1	B
## 13	11	1	B
## 14	12	0	B

## Pour comparer des courbes de survie [2]. . .

- ▶ Estimer le taux survie avec la fonction `survfit()` en fonction du facteur étudié “~ facteur”.

```
ds2 <- Surv(d2$tdp, d2$etat)
(ts2 <- survfit(ds2 ~ d2$groupe))
```

```
## Call: survfit(formula = ds2 ~ d2$groupe)
##
##           n events median 0.95LCL 0.95UCL
## d2$groupe=A 7      3     NA       6      NA
## d2$groupe=B 7      5      9       4      NA
```

## Pour comparer des courbes de survie [3]...

- ▶ La fonction `summary` permet de visualiser les valeur des taux de survie pour les deux groupes et leur intervalle de confiance pour chaque temps où il y a un décès.

```
summary(ts2)
```

```
## Call: survfit(formula = ds2 ~ d2$groupe)
```

```
##
```

```
##           d2$groupe=A
```

```
##   time  n.risk  n.event  survival  std.err  lower 95% CI upper
```

```
##     2      7      1    0.857    0.132    0.633
```

```
##     6      6      1    0.714    0.171    0.447
```

```
##     9      4      1    0.536    0.201    0.257
```

```
##
```

```
##           d2$groupe=B
```

```
##   time  n.risk  n.event  survival  std.err  lower 95% CI upper
```

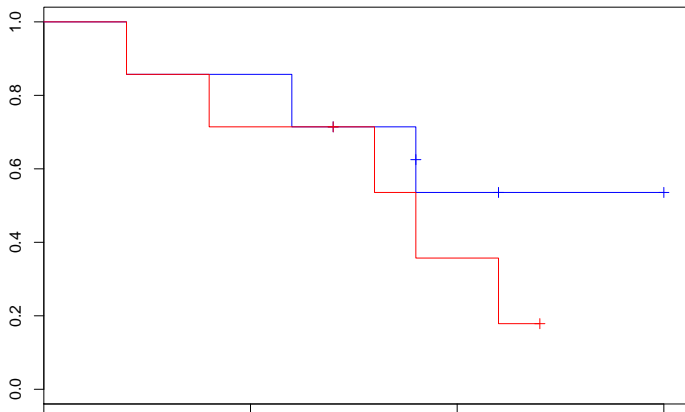
```
##     2      7      1    0.857    0.132    0.6334
```

```
##     4      6      1    0.714    0.171    0.4471
```

## Pour comparer des courbes de survie [4]...

La fonction `plot` permet de faire une représentation de la courbe de survie sans leur intervalle de confiance.

```
plot(ts2, mark.time = TRUE, col=c("blue", "red"))
```



## Pour tester la différence entre des courbes de survie [1]...

Le test du logrank est réalisable en utilisant la fonction `survdiff()`.  
La valeur du p indique si la différence est significative. Ici  $p=0.4$  (NS)

```
(dd2 <- survdiff(ds2 ~ d2$groupe))
```

```
## Call:
```

```
## survdiff(formula = ds2 ~ d2$groupe)
```

```
##
```

##		N	Observed	Expected	$(O-E)^2/E$	$(O-E)^2/V$
##	d2\$groupe=A	7	3	4.19	0.337	0.756
##	d2\$groupe=B	7	5	3.81	0.370	0.756

```
##
```

```
## Chisq= 0.8 on 1 degrees of freedom, p= 0.4
```

## Pour tester la différence entre des courbes de survie [2]. . .

Pour obtenir la valeur du HR associé, il est nécessaire d'utiliser la fonction `coxph()`. La valeur de HR correspond à  $\exp(\text{coef})$ . Ici HR=1.85 ; on retrouve  $p=0.4$  HR nest pas significativement différent de 1. Il n'y a pas de différence entre les groupes.

```
(dd2hr <- coxph(ds2 ~ d2$groupe))
```

```
## Call:  
## coxph(formula = ds2 ~ d2$groupe)  
##  
##           coef exp(coef) se(coef)    z    p  
## d2$groupeB  0.6         1.8     0.7 0.8 0.4  
##  
## Likelihood ratio test=0.7  on 1 df, p=0.4  
## n= 14, number of events= 8
```

## Pour tester la différence entre des courbes de survie [3]. . .

Pour obtenir la valeur du HR associé, il est nécessaire d'utiliser la fonction `coxph()`. La fonction `summary()` donne plus de renseignements par exemple l'intervalle de confiance. Ici  $HR = 1.85$  [0.44 ; 7.76].

```
summary(dd2hr)
```

```
## Call:
## coxph(formula = ds2 ~ d2$groupe)
##
##   n= 14, number of events= 8
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## d2$groupeB 0.614      1.847   0.732 0.84      0.4
##
##              exp(coef) exp(-coef) lower .95 upper .95
## d2$groupeB      1.85      0.541   0.44      7.76
##
```



## Pour tester la différence entre des courbes de survie [3]. . .

Pour obtenir la valeur du HR associé, il est nécessaire d'utiliser la fonction `coxph()`. La fonction `summary()` donne plus de renseignements par exemple l'intervalle de confiance. Ici  $HR = 1.85$  [0.44 ; 7.76].

```
summary(dd2hr)
```

```
## Call:
## coxph(formula = ds2 ~ d2$groupe)
##
##   n= 14, number of events= 8
##
##              coef exp(coef) se(coef)      z Pr(>|z|)
## d2$groupeB 0.614      1.847   0.732 0.84      0.4
##
##              exp(coef) exp(-coef) lower .95 upper .95
## d2$groupeB      1.85      0.541   0.44      7.76
##
```

## Pour comparer deux modèles...

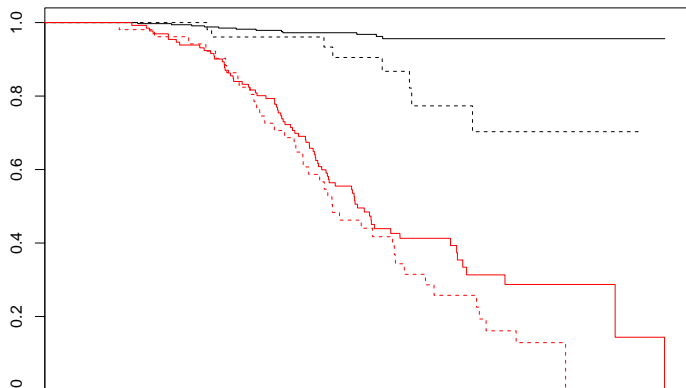
Repartons dans le jeu de données avec les 600 sujets. Etudions la survie en fonction de l'âge.

```
d2<-read.table("DATA/d2019.txt",header=T)
d2$cont<-as.Date(d2$DTCONT,"%d/%m/%Y");d2$ddn<-as.Date(d2$DTDDN,"%d/%m/%Y")
d2$TDP<-d2$ddn-d2$cont
ma<-coxph(Surv(d2$TDP, d2$DECES)~as.factor(d2$NAGEC))
msia<-coxph(Surv(d2$TDP,d2$DECES)~as.factor(d2$Maladie)+as.factor(d2$NAGEC))
anova(msia,ma,test='Chisq')
```

```
## Analysis of Deviance Table
## Cox model: response is Surv(d2$TDP, d2$DECES)
## Model 1: ~ as.factor(d2$Maladie) + as.factor(d2$NAGEC)
## Model 2: ~ as.factor(d2$NAGEC)
##      loglik Chisq Df P(>|Chi|)
## 1      -690
## 2      -782   184  1    <2e-16 ***
## ---
```

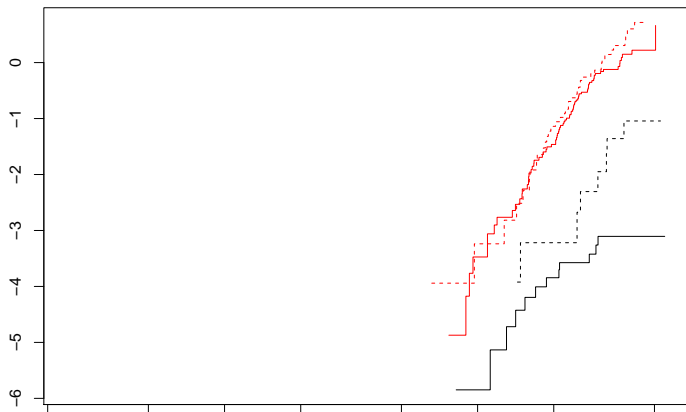
Pour comparer vérifier l'hypothèse des risques proportionnels [1] ...

```
fit <- survfit(Surv(d2$TDP, d2$DECES)~as.factor(d2$Maladie))  
plot(fit, col=c("black","black","red","red"),lty=c(1,2,1,2))
```



Pour comparer vérifier l'hypothèse des risques proportionnels [2]...

```
fit <- survfit(Surv(d2$TDP, d2$DECES)~as.factor(d2$Maladie),  
plot(fit,fun="cloglog",col=c("black","black","red","red"),
```



Qu'en pensez-vous ? Vérifié ou pas. . .