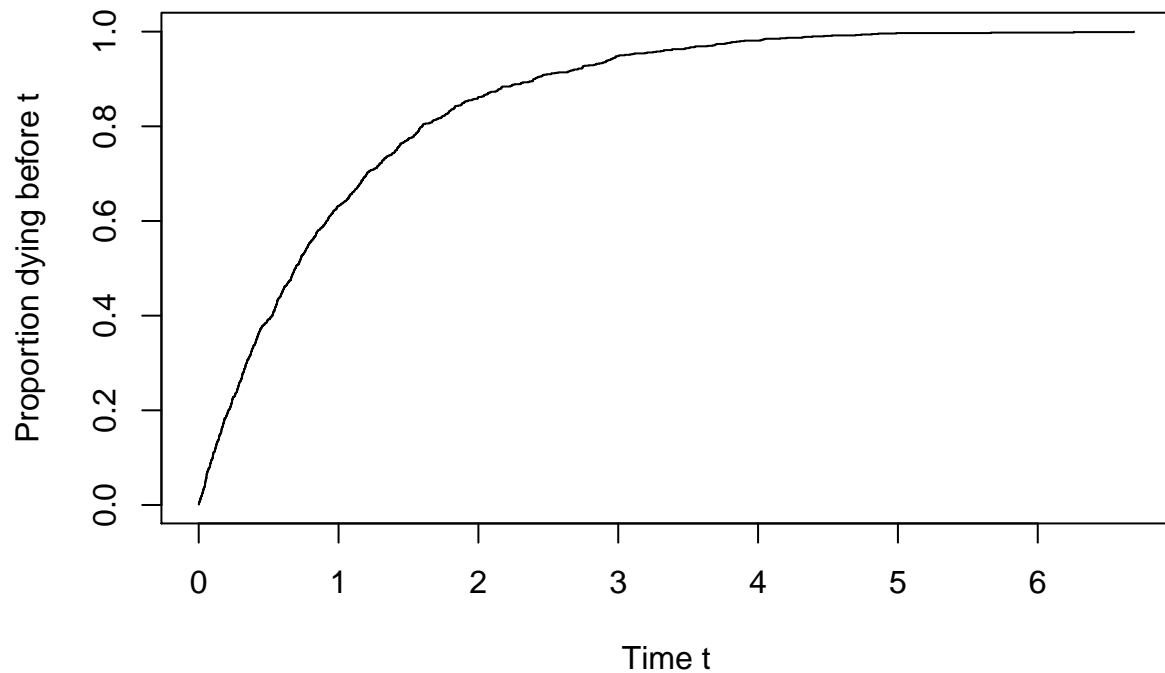# Delayed_entry.R

hputter

2021-02-23

```r
# Generate some survival data, from an exponential distribution with rate 1
set.seed(2021)
n <- 1000
tt <- rexp(n, rate=1)
# Without censoring, we can estimate the empirical distribution function
# (the empirical proportion of time points below t)
plot(sort(tt), (1:n)/n, type="s", xlab="Time t", ylab="Proportion dying before t")
```



```r
# Survival function
plot(sort(tt), 1 - (1:n)/n, type="s", xlab="Time t", ylab="Proportion surviving beyond t")
```
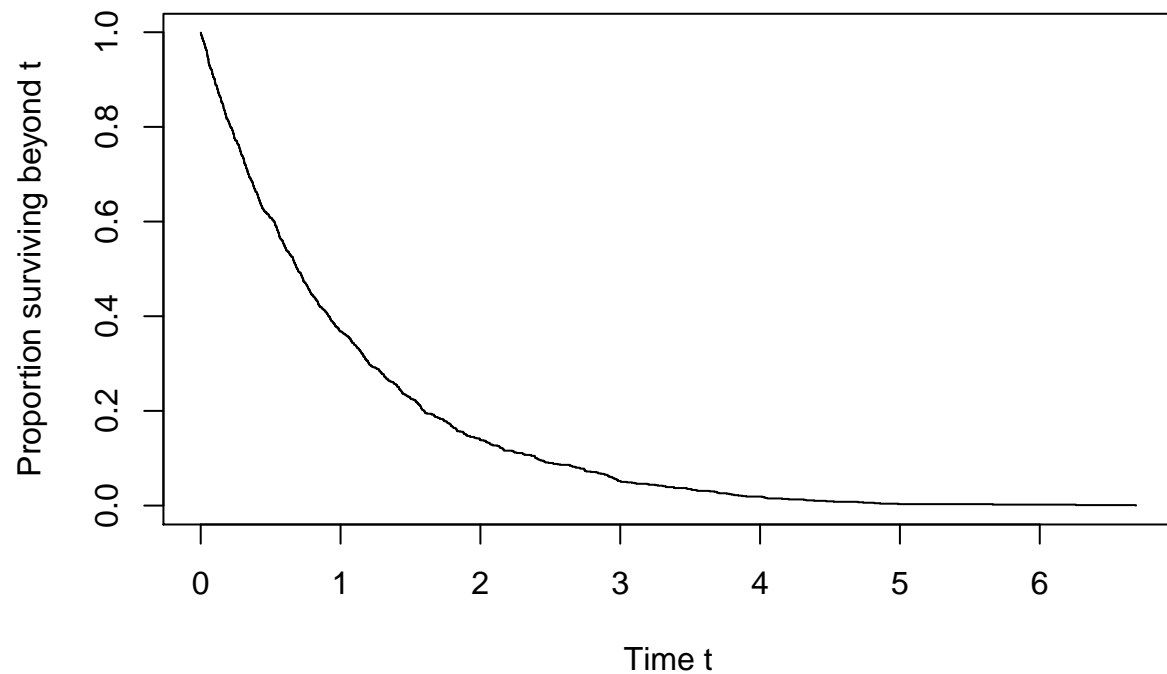
```
# Estimate this with survfit
library(survival)
status <- rep(1, n) # artificial, everyone has an event
sf <- survfit(Surv(tt, status) ~ 1)
plot(sf, xlab="Time", ylab="Survival")
```
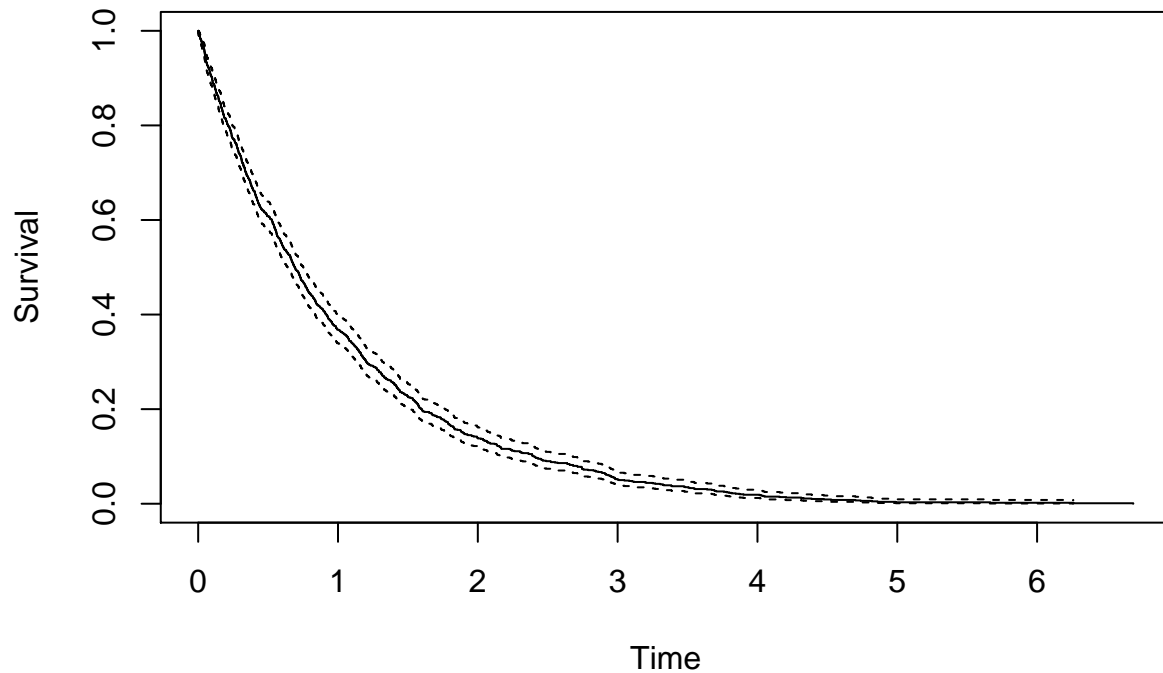


```
#
# Censoring
#
# Now add censoring, let's take uniform between 2 and 5
cc <- runif(n, min=2, max=5)
```

```
d <- data.frame(time = pmin(tt, cc), status = as.numeric(tt <= cc))
# Estimate again with survfit
sf <- survfit(Surv(time, status) ~ 1, data=d)
plot(sf, xlab="Time", ylab="Survival")
tseq <- seq(0, 5, by=0.01)
lines(tseq, exp(-tseq), type="l", lwd=2, lty=3, col="red")
```
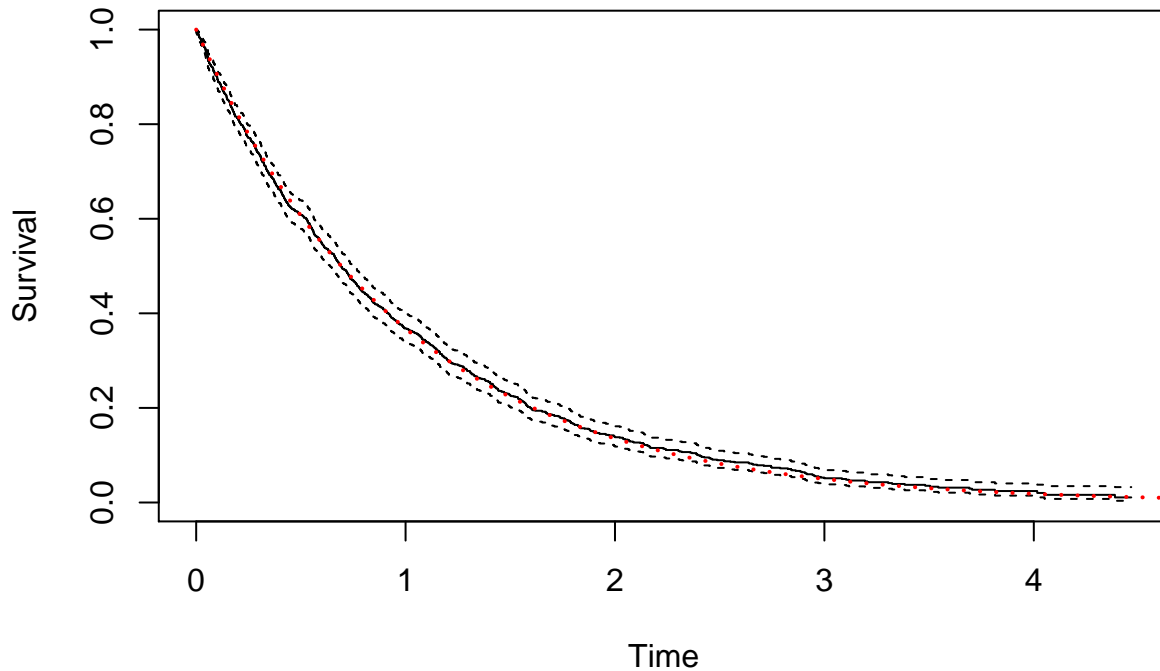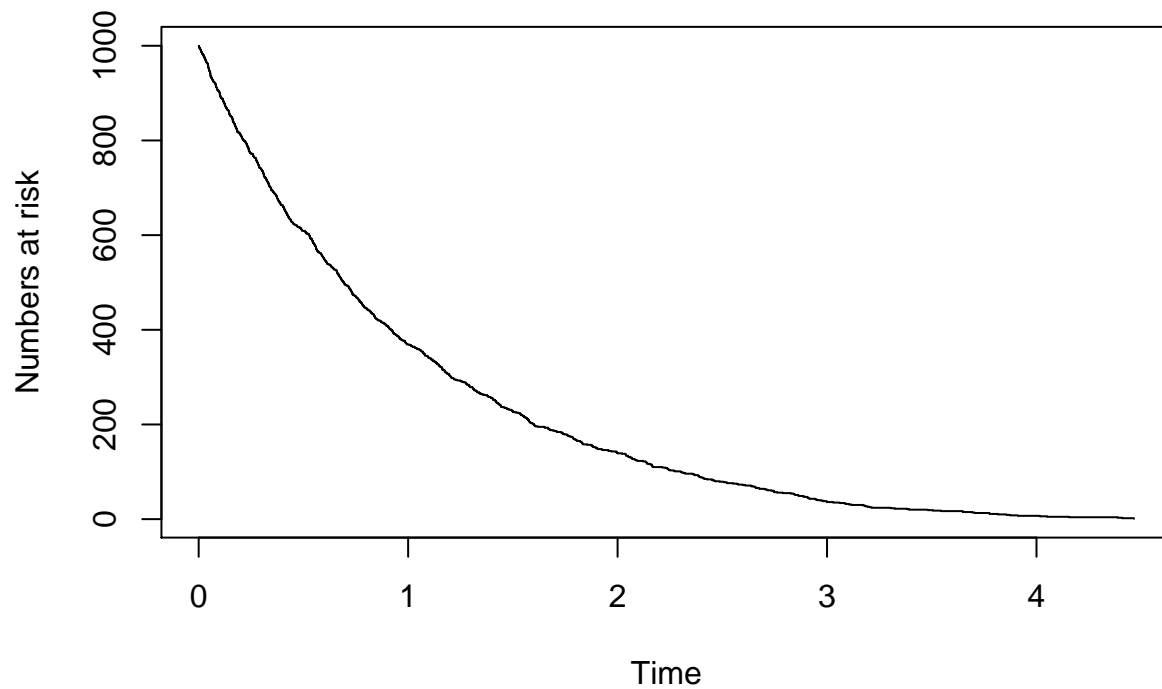


```
# Estimate the censoring distribution
sf0 <- survfit(Surv(time, status==0) ~ 1, data=d)
plot(sf, xlab="Time", ylab="Censoring")
tseq <- seq(2, 5, by=0.01)
lines(tseq, 1 - (tseq-2)/(5-2), type="l", lwd=2, lty=3, col="red")
```
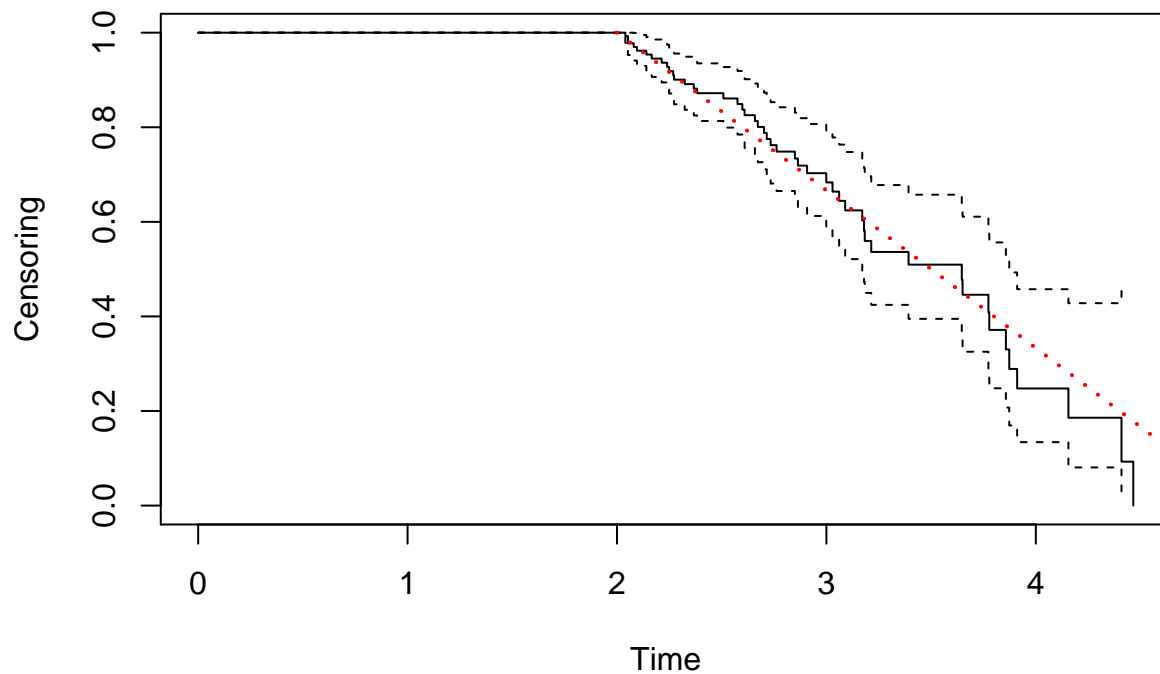
```
# Make a plot of the numbers at risk
names(sf)
```

```
##  [1] "n"         "time"      "n.risk"    "n.event"   "n.censor"  "surv"      "std.err"   "cumhaz"
## [11] "logse"     "conf.int"  "conf.type" "lower"     "upper"     "call"
```

```
plot(sf$time, sf$n.risk, type="s", xlab="Time", ylab="Numbers at risk")
```



```
dcomplete <- d # save
#
# Delayed entry
#
```

4

```
# Let's say that half of the subjects enter uniformly between 0 and 2
d$entry <- 0
idx <- sample(1:n, size=n/2, replace=FALSE) # those are the ones that enter late
d$entry[idx] <- runif(n/2, min=0, max=2)
# Those that would have an event before entry are not included
d <- subset(d, entry < time)
nrow(d)
```
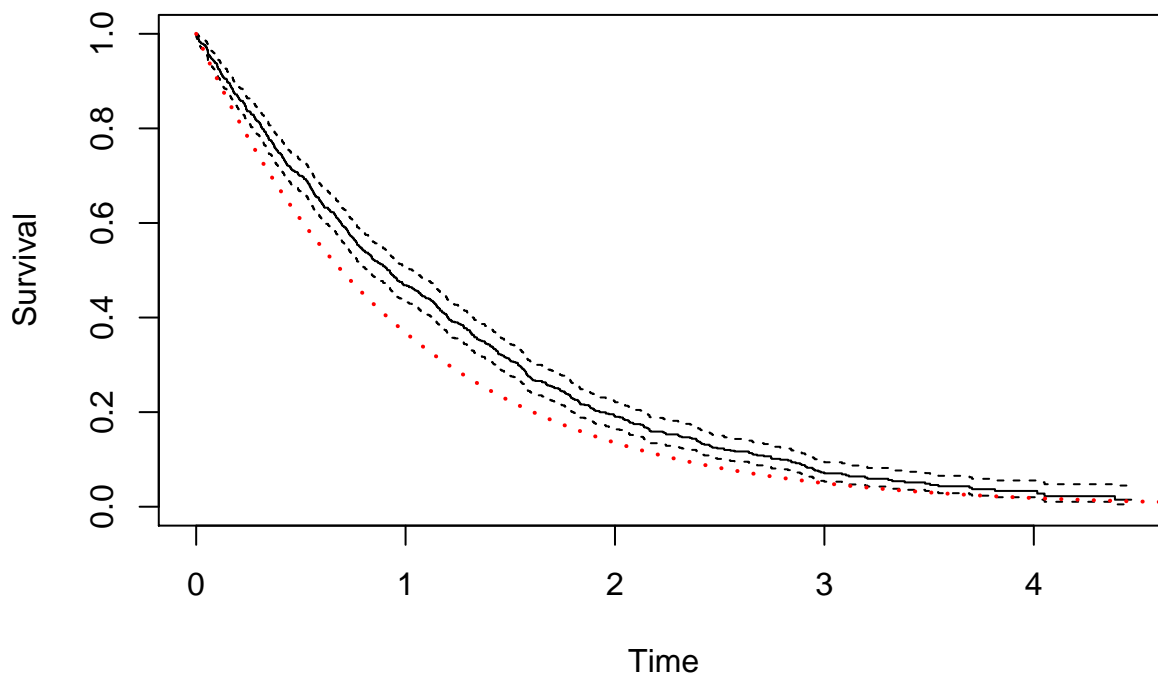
```
## [1] 726
```

```
# Wrong to ignore the delayed entry
sfwrong <- survfit(Surv(time, status) ~ 1, data=d)
plot(sfwrong, xlab="Time", ylab="Survival")
tseq <- seq(0, 5, by=0.01)
lines(tseq, exp(-tseq), type="l", lwd=2, lty=3, col="red")
```
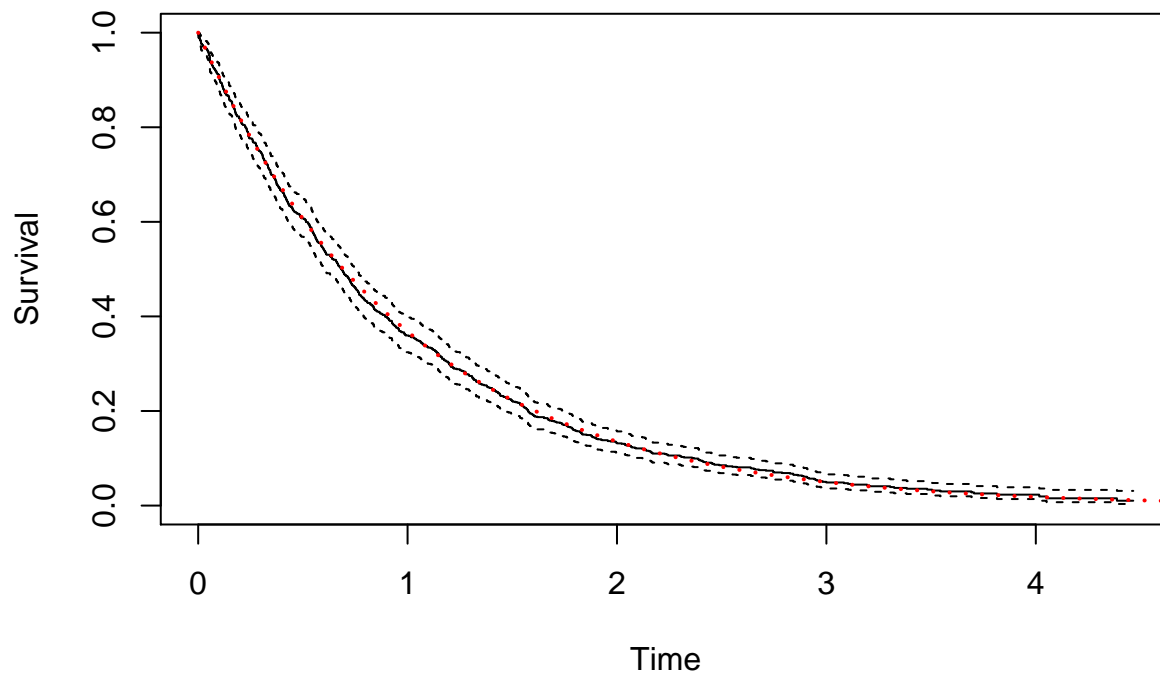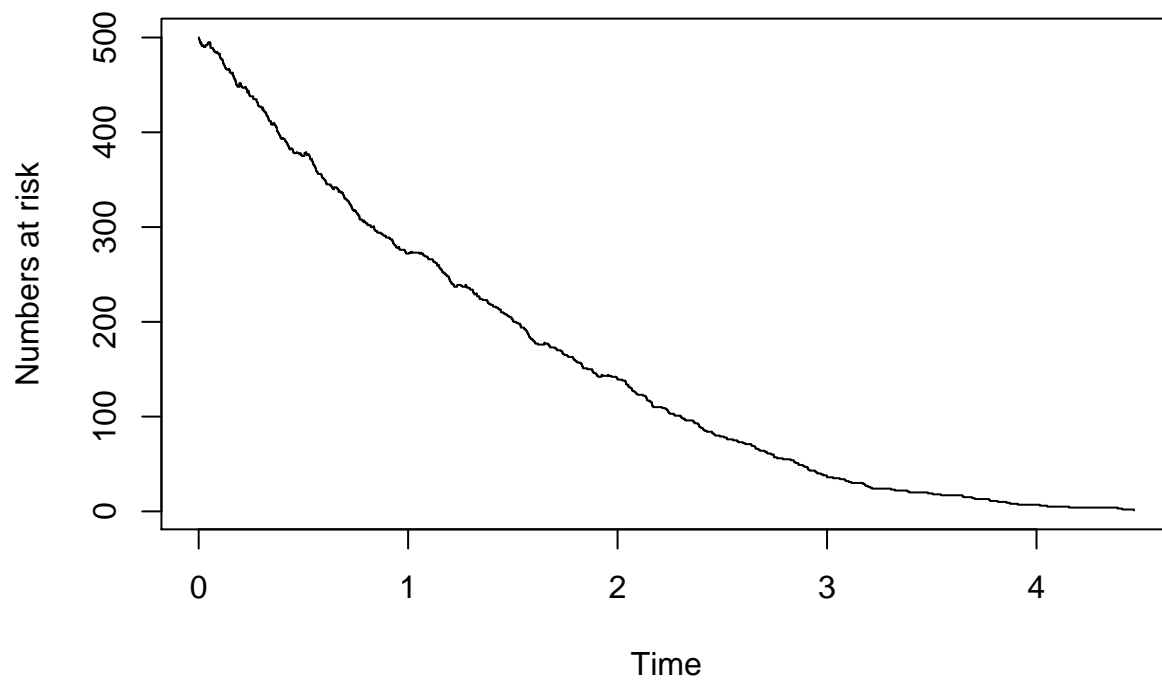


```
# But easy to incorporate the delayed entry
sfcorrect <- survfit(Surv(entry, time, status) ~ 1, data=d)
plot(sfcorrect, xlab="Time", ylab="Survival")
lines(tseq, exp(-tseq), type="l", lwd=2, lty=3, col="red")
```
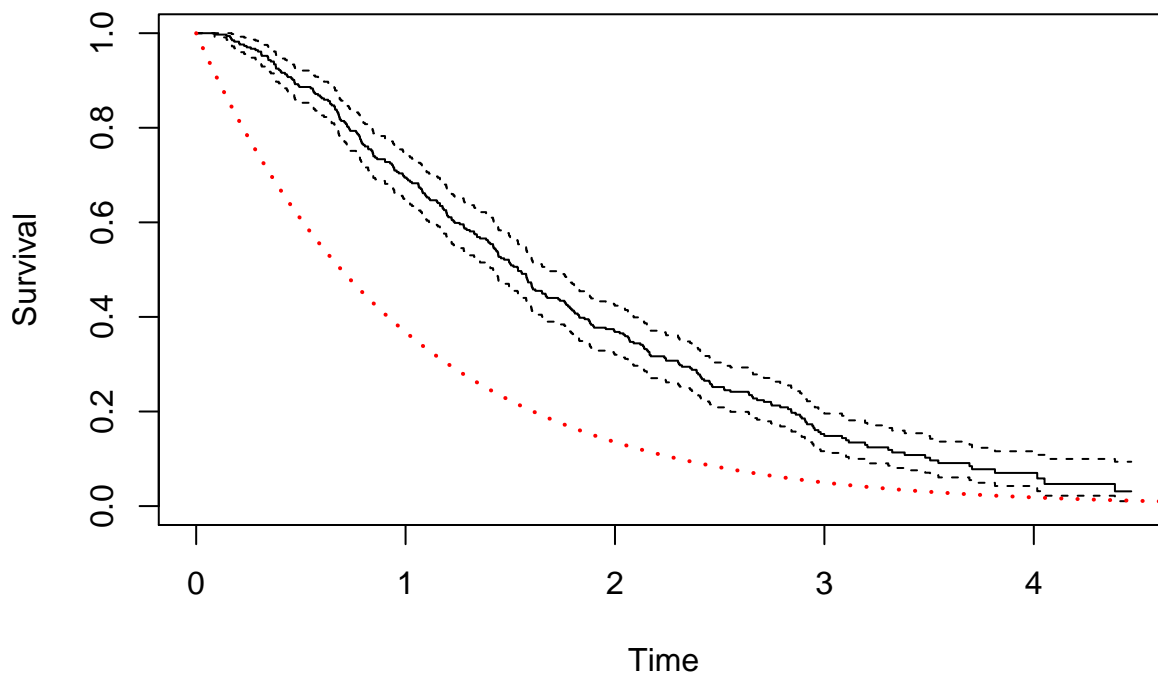
```
# Numbers at risk
plot(sfcorrect$time, sfcorrect$n.risk, type="s", xlab="Time", ylab="Numbers at risk")
```
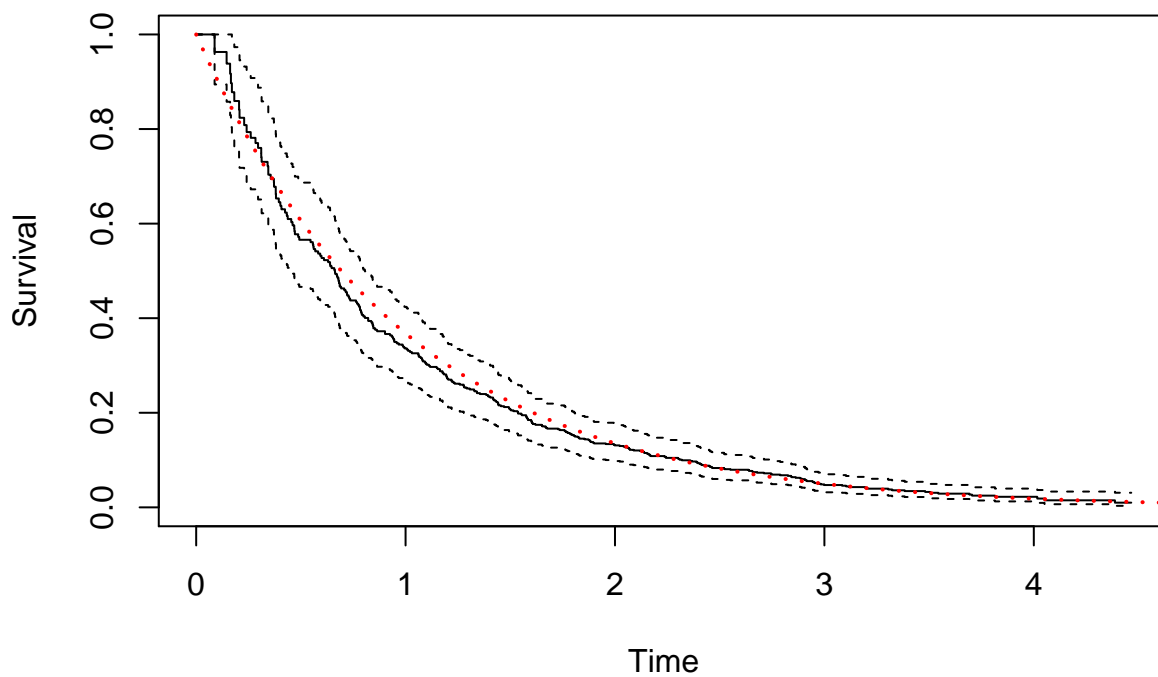


```
# Now let's say that all of the subjects enter uniformly between 0 and 3
d <- dcomplete
d$entry <- runif(n, min=0, max=3)
# Those that would have an event before entry are not included
d <- subset(d, entry < time)
nrow(d)
```

```
## [1] 334
```

```
# Wrong to ignore the delayed entry
sfwrong <- survfit(Surv(time, status) ~ 1, data=d)
plot(sfwrong, xlab="Time", ylab="Survival")
tseq <- seq(0, 5, by=0.01)
lines(tseq, exp(-tseq), type="l", lwd=2, lty=3, col="red")
```



```
# But easy to incorporate the delayed entry
sfcorrect <- survfit(Surv(entry, time, status) ~ 1, data=d)
plot(sfcorrect, xlab="Time", ylab="Survival")
lines(tseq, exp(-tseq), type="l", lwd=2, lty=3, col="red")
```

```
# Numbers at risk
plot(sfcorrect$time, sfcorrect$n.risk, type="s", xlab="Time", ylab="Numbers at risk")
```