

# Examen

Antonio, H. F.

2022-12-09

## Contents

<b>Problema 1</b>	<b>1</b>
Solución 1 . . . . .	1
Solución 2 . . . . .	2
Solución 3 . . . . .	2
<b>Problema 2</b>	<b>3</b>
Solución 1 . . . . .	3
Solución 2 . . . . .	4
Solución 3 . . . . .	5
Solución 4 . . . . .	6
Solución 5 . . . . .	7
Solución 6 . . . . .	8

## Problema 1

### Solución 1

Función de supervivencia

```
fSup <-function(x,k) {  
  exp(-x*(1+x)^(-k))  
}
```

Función de Riesgo

```
fHaz <-function(x,k) {  
  (1+x)^(-k-1)*(1+(1-k)*x)  
}
```

Finalmente la función de verosimilitud

```
lLik <- function(k, t, c) {  
  eMV <-0  
  for(i in 1:10)  
    eMV <- eMV + c[i]*log(fHaz(t[i],k))+ log(fSup(t[i],k))  
  return(eMV)  
}
```

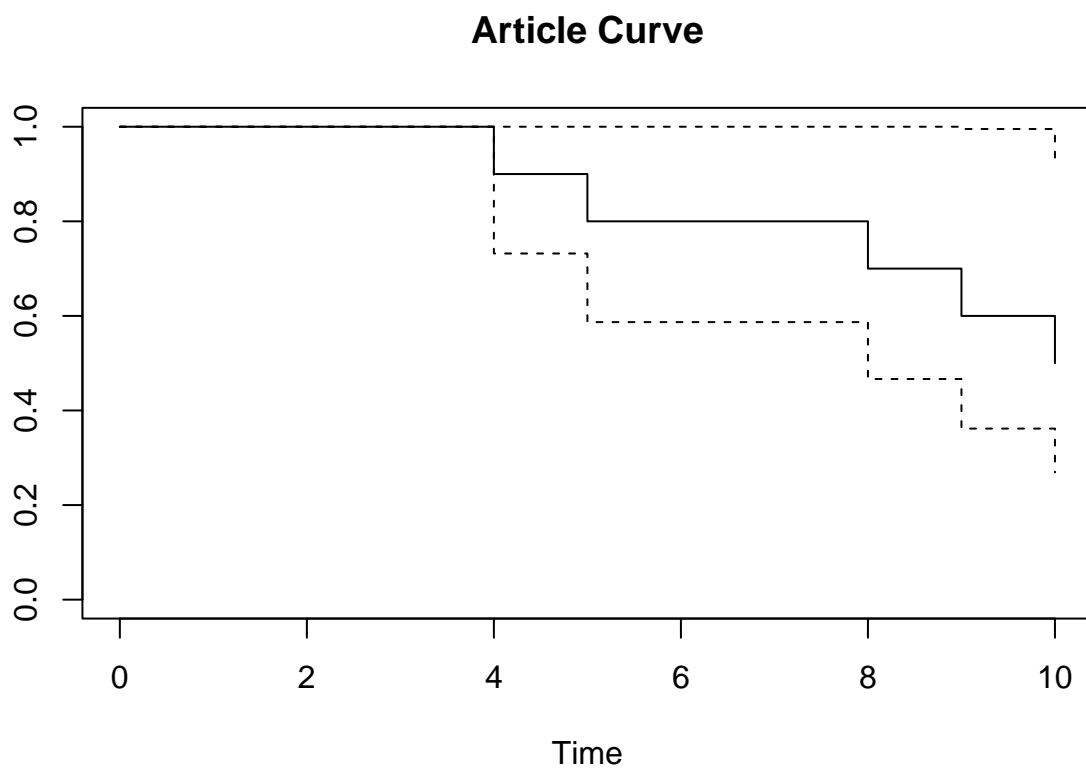
Obtenemos el estimador de k

```
kOpt <- optimize(lLik, c(0,1), maximum=TRUE, t = tRat, c = rCen);  
kOpt
```

```
## $maximum  
## [1] 0.8569638  
##  
## $objective  
## [1] -27.89282
```

## Solución 2

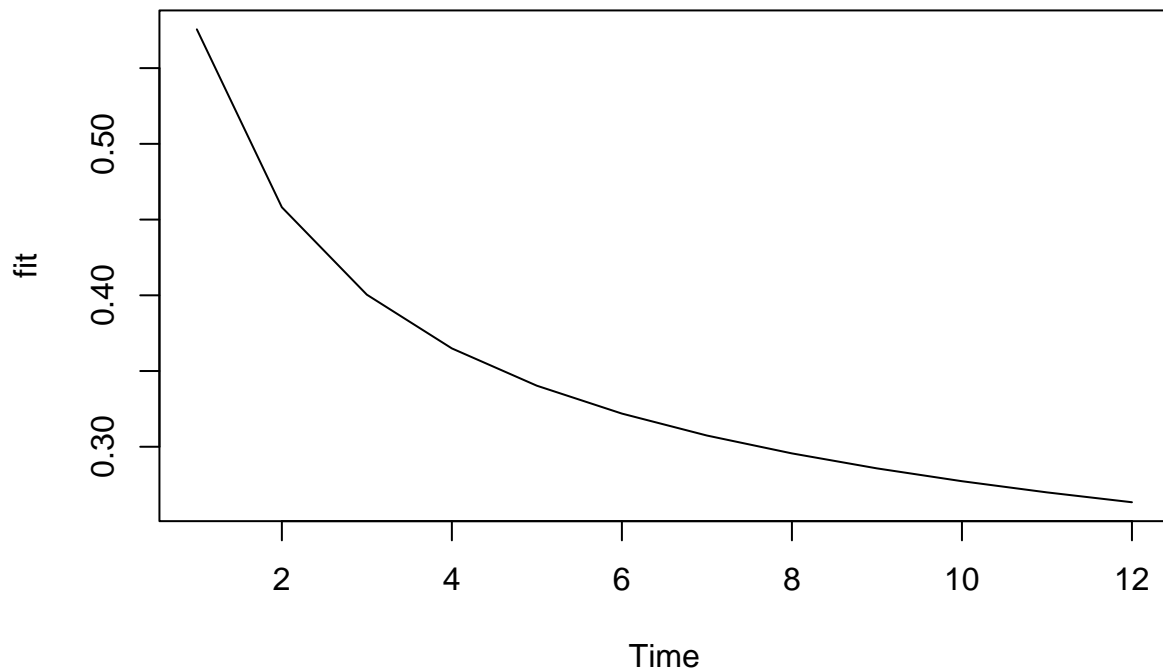
```
formula <- Surv(tRat,rCen) ~ 1  
mCox1 <- survival::survfit(formula)  
plot(mCox1, xlab = "Time")  
title("Article Curve")
```



## Solución 3

```
fit <- fSup(1:12,0.85638)  
plot(fit, xlab = "Time", type = "l")  
title("Article Curve")
```

## Article Curve



## Problema 2

### Solución 1

```
formula <- Surv(T3,D3) ~ Z1
survdif(formula, data = burn)
```

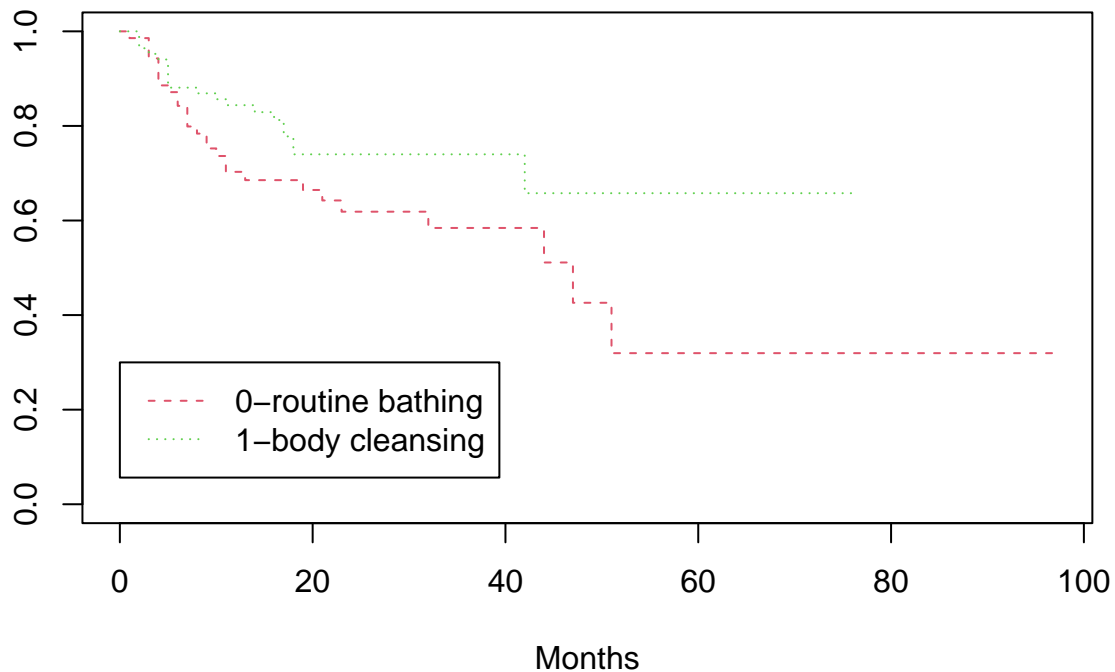
```
## Call:
## survdif(formula = formula, data = burn)
##
##      N Observed Expected (O-E)^2/E (O-E)^2/V
## Z1=0 70      28     21.4      2.07      3.79
## Z1=1 84      20     26.6      1.66      3.79
##
## Chisq= 3.8  on 1 degrees of freedom, p= 0.05
```

No se rechaza la hipótesis de que los tratamientos sean diferentes, por lo tanto, no hacen ninguna diferencia.

Gráficas de Kaplan-Meier

```
formula <- Surv(T3,D3) ~ Z1
fit <- survival::survfit(formula, data = burn)
plot(fit, lty = 2:3, xlab = "Months", col = 2:3)
legend(0,0.3, c("0-routine bathing", "1-body cleansing"), lty = 2:3, col = 2:3)
title("Kaplan-Meier Curves")
```

## Kaplan–Meier Curves



De las anteriores curvas podemos ver aunque no se cruzan las curvas, cuando aumentan los meses las curvas parecen separarse.

## Solución 2

Ajustamos el modelo de Cox

```
burn$Z1 <- as.factor(burn$Z1)
burn$Z2 <- as.factor(burn$Z2)
burn$Z3 <- as.factor(burn$Z3)
burn$Z5 <- as.factor(burn$Z5)
burn$Z6 <- as.factor(burn$Z6)
burn$Z7 <- as.factor(burn$Z7)
burn$Z8 <- as.factor(burn$Z8)
burn$Z9 <- as.factor(burn$Z9)
burn$Z10 <- as.factor(burn$Z10)
burn$Z11 <- as.factor(burn$Z11)
formula <- Surv(T3,D3) ~ Z1 + Z2 + Z3 + Z4 + Z5 + Z6 + Z7 + Z8 + Z9 + Z10 + Z11
mCox1 <- coxph(formula, data = burn)
mCox1
```

```
## Call:
## coxph(formula = formula, data = burn)
##
##           coef exp(coef) se(coef)      z      p
## Z11 -0.651754  0.521131  0.323330 -2.016 0.0438
## Z21 -0.556911  0.572976  0.405182 -1.374 0.1693
```

```
## Z31    2.149127  8.577367  1.040139  2.066 0.0388
## Z4     0.002041  1.002043  0.009843  0.207 0.8357
## Z51    -0.014035  0.986063  0.370920 -0.038 0.9698
## Z61     0.541461  1.718516  0.430265  1.258 0.2082
## Z71    -0.055650  0.945870  0.507956 -0.110 0.9128
## Z81    -0.171817  0.842133  0.393707 -0.436 0.6625
## Z91    -0.324566  0.722841  0.373905 -0.868 0.3854
## Z101   0.228682  1.256943  0.372930  0.613 0.5397
## Z112   1.527828  4.608156  1.128623  1.354 0.1758
## Z113   2.192439  8.957029  1.130097  1.940 0.0524
## Z114   0.949734  2.585021  1.036308  0.916 0.3594
##
## Likelihood ratio test=27.29  on 13 df, p=0.01136
## n= 154, number of events= 48
```

Se observa del ajuste que la solución de baño ahora es significativo a un  $\alpha = 0.5$ , lo que indica que aumenta el tiempo en que reaparezca una infección.

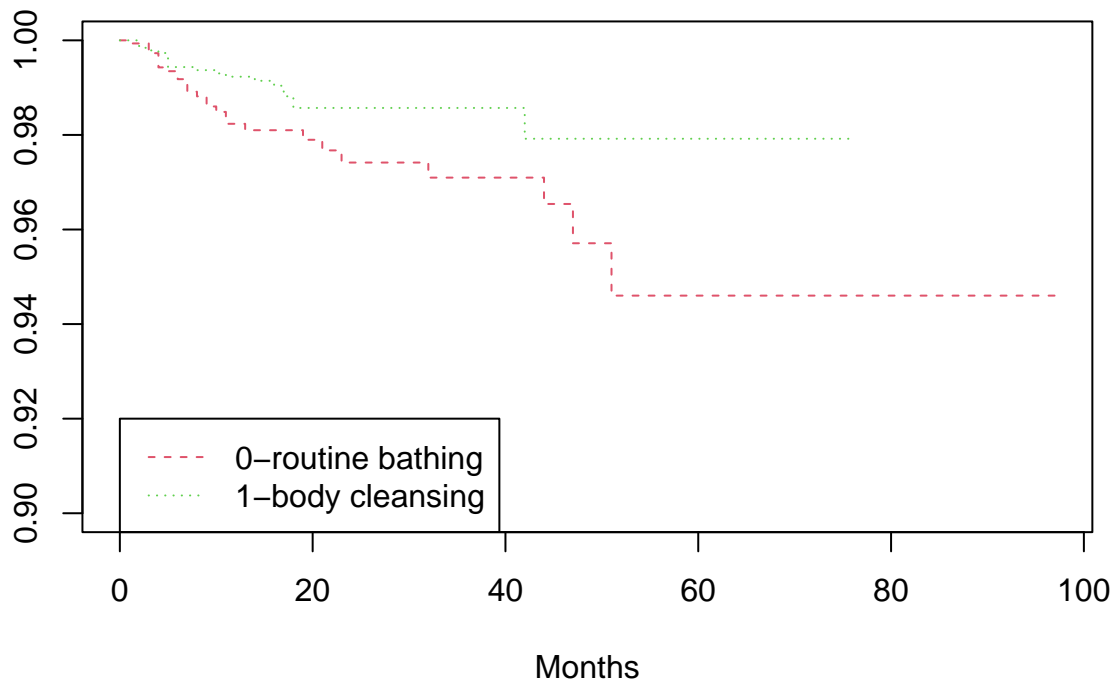
### Solución 3

```
formula <- Surv(T3,D3) ~ strata(Z1) + Z2 + Z3 + Z4 + Z5 + Z6 + Z7 + Z8 + Z9 + Z10 + Z11
mCox2 <- coxph(formula, data = burn)
mCox2
```

```
## Call:
## coxph(formula = formula, data = burn)
##
##              coef exp(coef)    se(coef)      z      p
## Z21  -0.5256685  0.5911601  0.4065962 -1.293 0.1961
## Z31   2.1192745  8.3250951  1.0422270  2.033 0.0420
## Z4     0.0011815  1.0011822  0.0099237  0.119 0.9052
## Z51    0.0008772  1.0008776  0.3713477  0.002 0.9981
## Z61    0.5579726  1.7471268  0.4313489  1.294 0.1958
## Z71   -0.0250822  0.9752297  0.5104246 -0.049 0.9608
## Z81   -0.1962773  0.8217843  0.3914169 -0.501 0.6161
## Z91   -0.2947557  0.7447135  0.3723399 -0.792 0.4286
## Z101   0.2653687  1.3039116  0.3747441  0.708 0.4789
## Z112   1.5072270  4.5141955  1.1324432  1.331 0.1832
## Z113   2.1630491  8.6976173  1.1312081  1.912 0.0559
## Z114   0.9567817  2.6033047  1.0365957  0.923 0.3560
##
## Likelihood ratio test=22.75  on 12 df, p=0.02989
## n= 154, number of events= 48
```

```
plot(survfit(mCox2), ylim=c(0.9,1), lty = 2:3, xlab = "Months", col = 2:3)
legend(0,0.92, c("0-routine bathing", "1-body cleansing"), lty = 2:3, col = 2:3)
title("Cox regression")
```

## Cox regression



De la gráfica observamos que las líneas base nunca se cruzan, por lo tanto se puede suponer que son riesgos proporcionales. Comprobamos si los riesgos son proporcionales

```
cox.zph(mCox2)
```

```
##          chisq df      p
## Z2      1.64e+00 1 0.201
## Z3      2.71e+00 1 0.100
## Z4      9.44e-04 1 0.975
## Z5      2.47e+00 1 0.116
## Z6      7.51e-01 1 0.386
## Z7      4.43e-05 1 0.995
## Z8      3.61e-01 1 0.548
## Z9      4.02e+00 1 0.045
## Z10     5.24e+00 1 0.022
## Z11     1.03e+01 3 0.016
## GLOBAL  1.87e+01 12 0.096
```

Se observa que de manera global se puede ajustar un modelo de riesgos proporcionales, aunque se observa que las variables Z9, Z10, Z11, no cumplen este supuesto.

## Solución 4

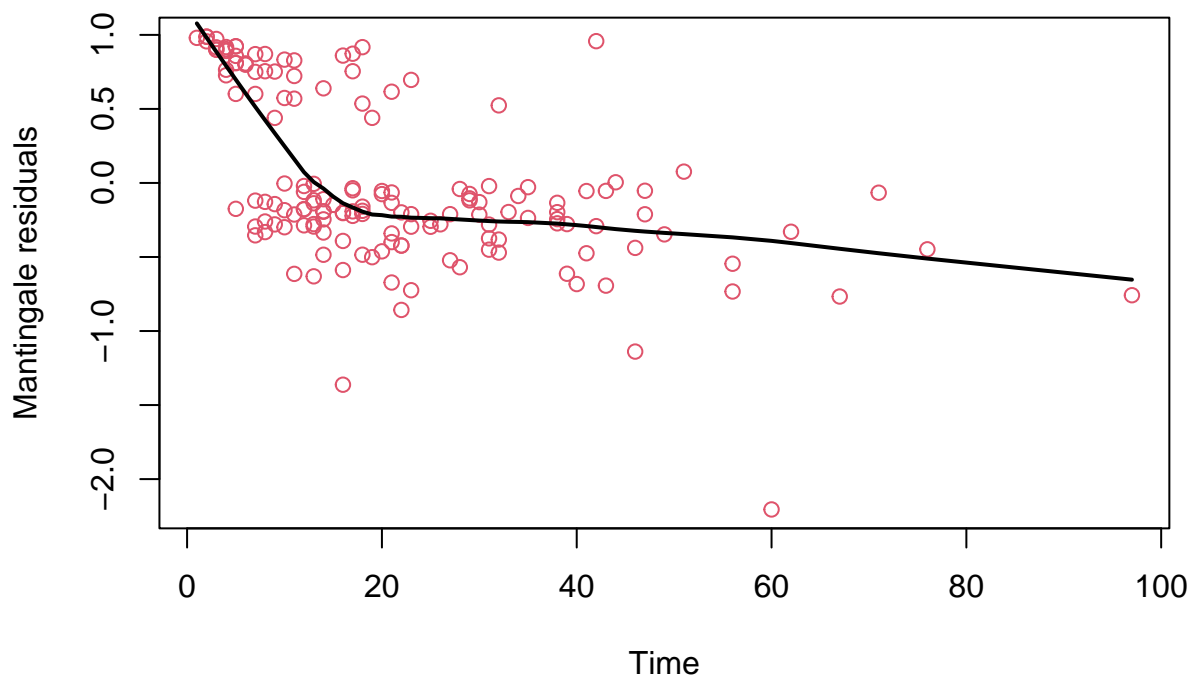
```
formula <- Surv(T3,D3) ~ strata(Z1) + Z2 + Z3 + Z4 + Z5 + Z6 + Z7 + Z8 + Z9 + Z10 + Z11
mCox2 <- coxph(formula, data = burn)
mCox2
```

```
## Call:
## coxph(formula = formula, data = burn)
##
##              coef exp(coef)    se(coef)      z      p
## Z21  -0.5256685  0.5911601  0.4065962 -1.293 0.1961
## Z31   2.1192745  8.3250951  1.0422270  2.033 0.0420
## Z4    0.0011815  1.0011822  0.0099237  0.119 0.9052
## Z51   0.0008772  1.0008776  0.3713477  0.002 0.9981
## Z61   0.5579726  1.7471268  0.4313489  1.294 0.1958
## Z71  -0.0250822  0.9752297  0.5104246 -0.049 0.9608
## Z81  -0.1962773  0.8217843  0.3914169 -0.501 0.6161
## Z91  -0.2947557  0.7447135  0.3723399 -0.792 0.4286
## Z101  0.2653687  1.3039116  0.3747441  0.708 0.4789
## Z112  1.5072270  4.5141955  1.1324432  1.331 0.1832
## Z113  2.1630491  8.6976173  1.1312081  1.912 0.0559
## Z114  0.9567817  2.6033047  1.0365957  0.923 0.3560
##
## Likelihood ratio test=22.75  on 12 df, p=0.02989
## n= 154, number of events= 48
```

## Solucion 5

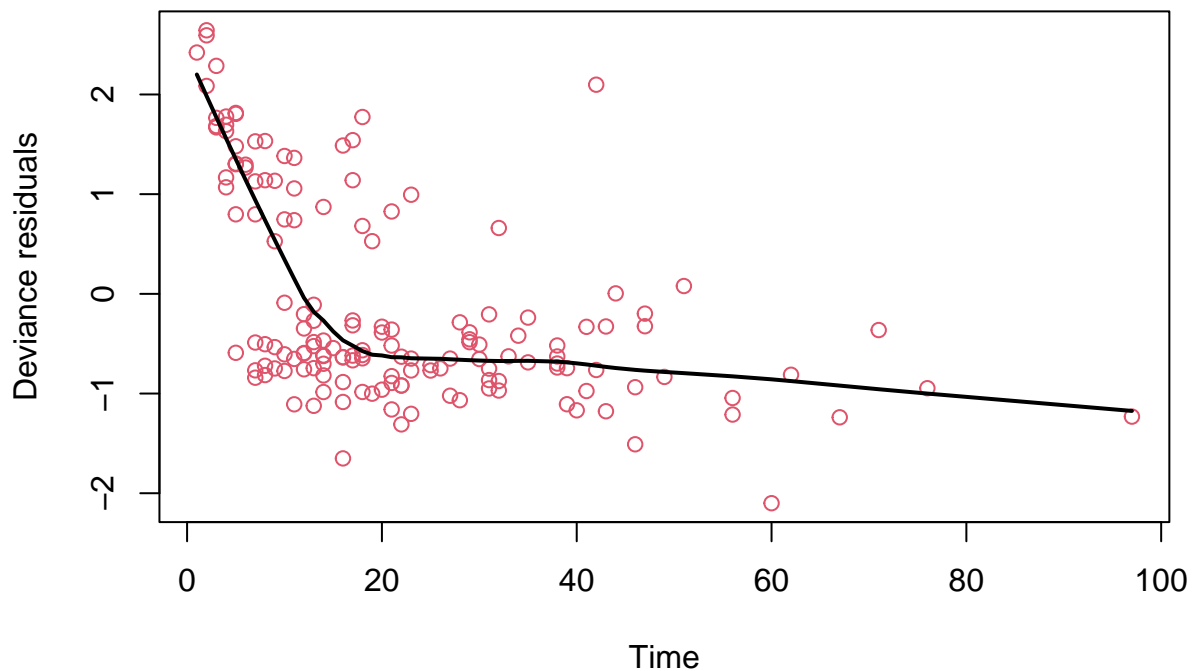
Residuales Martingala contra el tiempo de infección

```
rCox2 <- residuals(mCox1, type = "martingale")
plot(burn$T3, rCox2, col = 2, ylab = "Mantingale residuals", xlab = "Time")
lines(lowess(burn$T3, rCox2), col = 1, lwd=2)
```



Residuales de desviación contra el tiempo de infección

```
rCox2 <- residuals(mCox1, type = "deviance")
plot(burn$T3, rCox2, col = 2, ylab = "Deviance residuals", xlab = "Time")
lines(lowess(burn$T3, rCox2), col = 1, lwd=2)
```



## Solución 6

Se propone un modelo paramétrico y buscamos cuál ajusta mejor.

```
formula <- Surv(T3,D3) ~ Z1 + Z2 + Z3 + Z4 + Z5 + Z6 + Z7 + Z8 + Z9 + Z10 + Z11
dist <- c("weibull", "exponential", "gaussian", "logistic", "lognormal", "loglogistic")
b <- 0

for(i in dist){
  a <- survreg(formula,data=burn,dist=i)
  if(a$loglik[1]<b){
    b <- a$loglik[1]
    c <- a$dist
  }
}
c

## [1] "logistic"
```

El mejor modelo de acuerdo con la logverosimilitud es un logistic.