

Supervivencia con destino conocido

- Destino conocido

Organismos sésiles: plantas, corales, animales de laboratorio.

a) Curvas de supervivencia: Tiempos de germinación, recursos fluctuantes (sunflecks).



Datos censados

- Radiotransmisores: Individuos que a partir de cierto momento desconocemos si siguen vivos dada la falla del aparato.



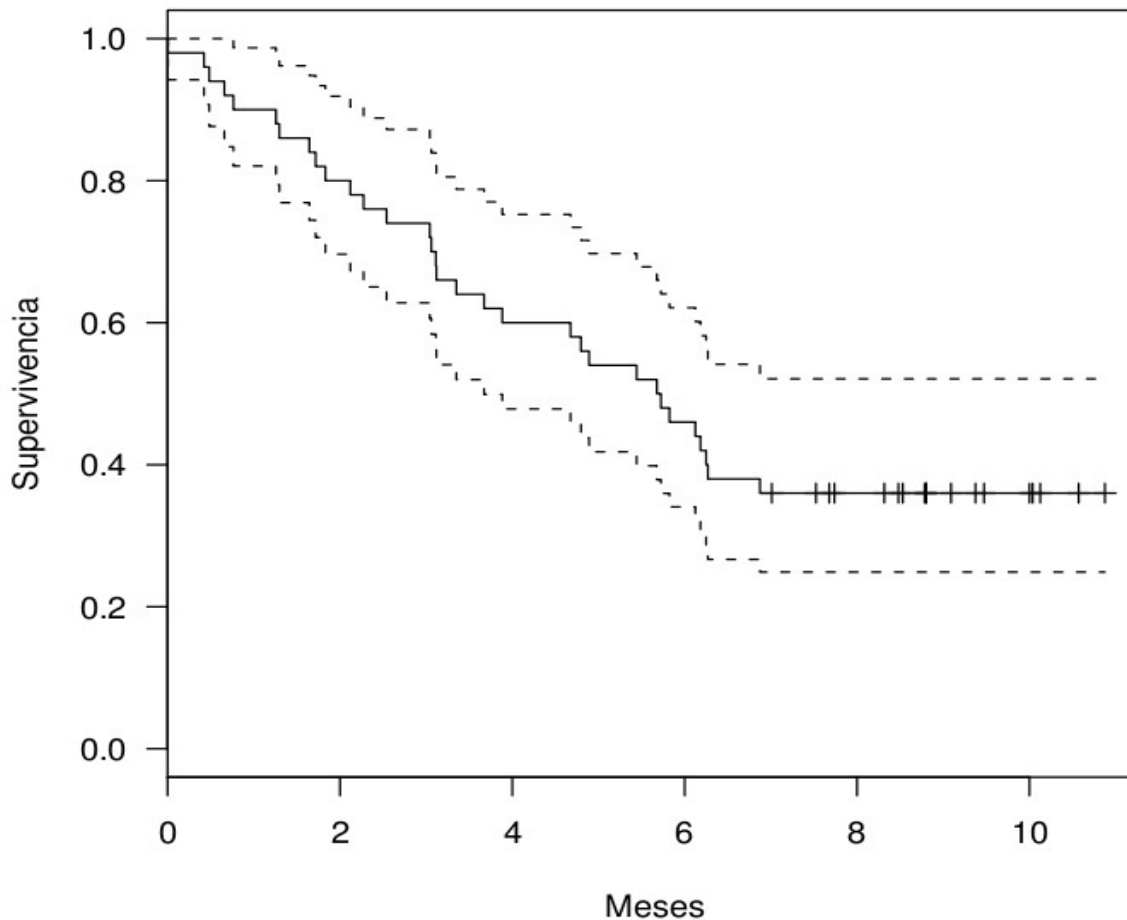
Código

Supervivencia: datos censados

Lambda = 0.1

Estocasticidad en el fallo de los radiotransmisores

Gráfica Kaplan-Meier, no paramétrico



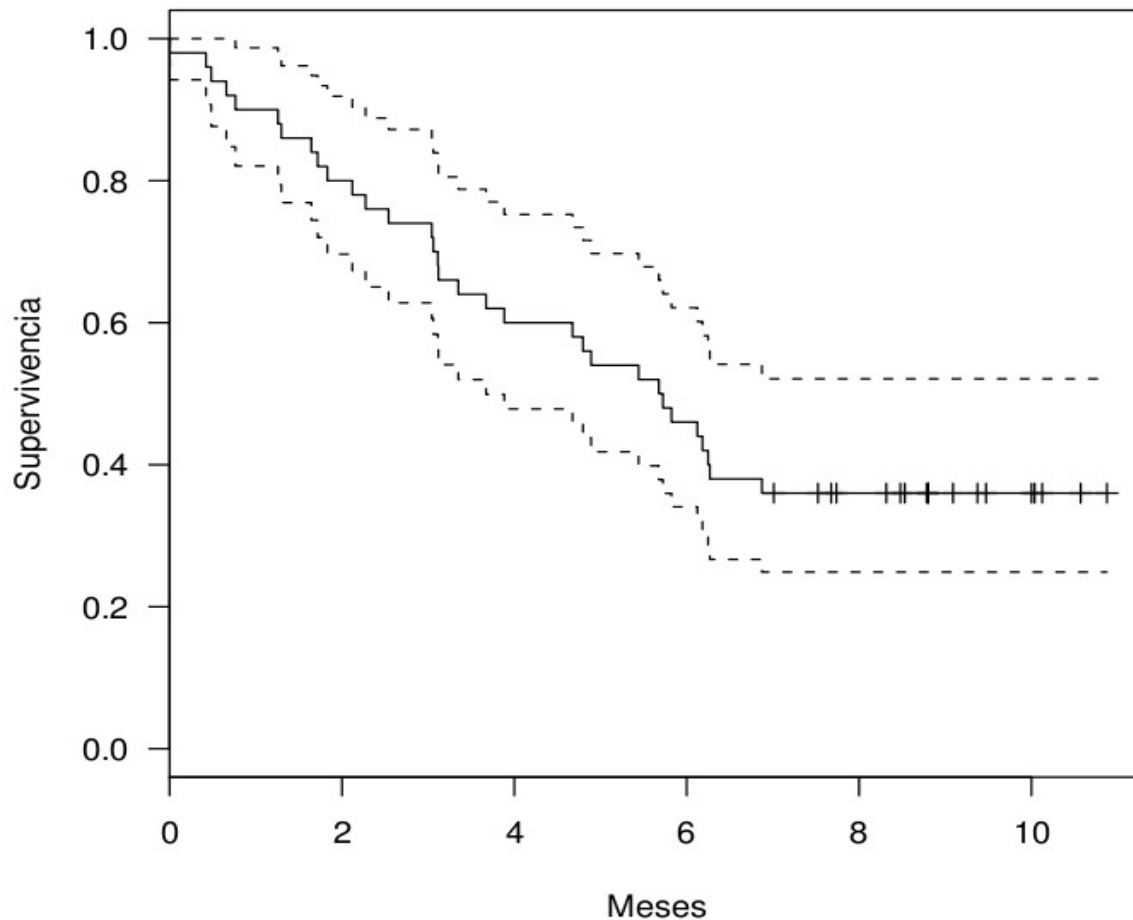
$$\hat{S}_t = \prod_{i=1}^t \left(\frac{n_i - d_i}{n_i} \right)$$

Supervivencia: datos censados

Implementación en MARK y en R

Estocasticidad en el fallo de los radiotransmisores

Gráfica Kaplan-Meier, no paramétrico



Análisis de supervivencia

- Kaplan-Meier
- Paramétrico: Exponencial, Weibull, Gamma, Lognormal.
- Semiparamétrico: Regresión de Cox
- Implementación en R

Kaplan-Meier MARK

- Historias de encuentro, visitas pares de datos

10101011 1;

- 10100000 3;

- 10101100 48;

Posibles covariables individuales.

- 10101011 1 3.2;

Ejemplo blkduck.inp

Kaplan Meier

- R, paquete survival
- Base de datos gehan
- 42 pacientes de leucemia, 2 tratamientos: placebo y medicina.
- `attach(gehan)`
- `Surv(time, cens)`
- `gehan.surv <- survfit(Surv(time, cens) ~ treat, data = gehan,
 conf.type = "log-log")`
- `summary(gehan.surv)`

Kaplan Meier

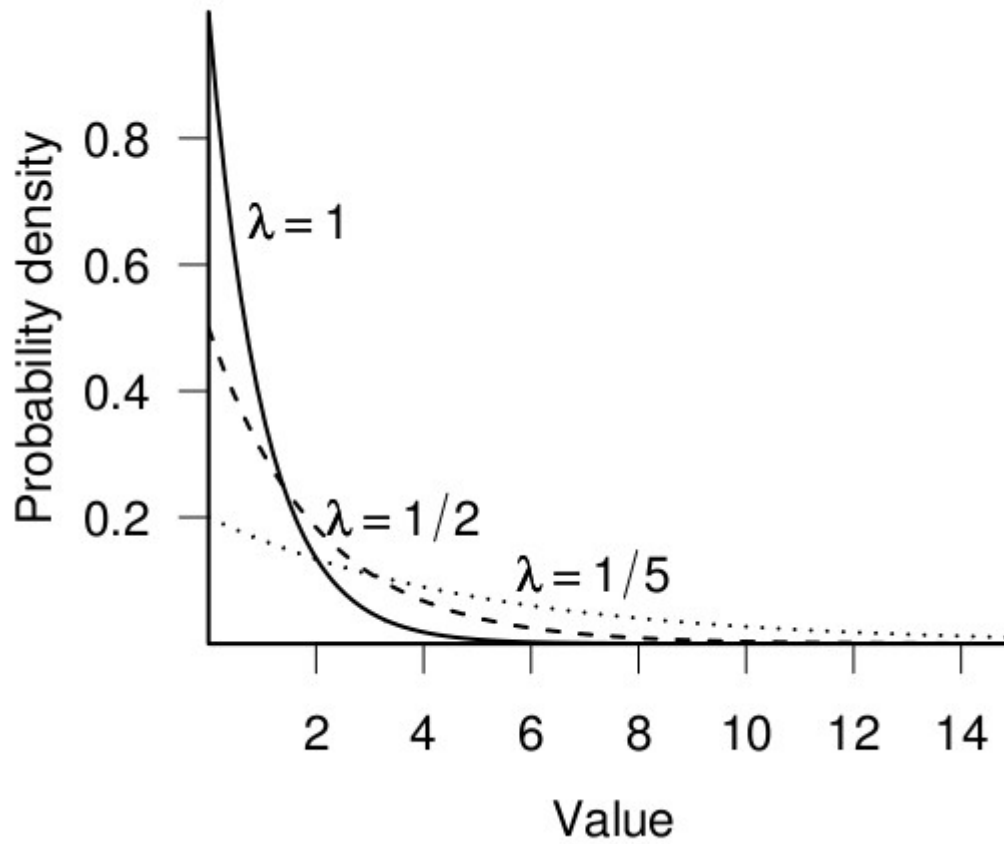
- **Gráfico:**
- `plot(gehan.surv, conf.int = T, lty = 3:2, log = T,`
- `xlab = "tiempo de remisión (semanas)", ylab`
`= "Supervivencia")`
- `lines(gehan.surv, lty = 1:2, lwd = 2, cex = 2)`
- `legend(25, 0.1 , c("control", "6-MP"), lty = 1:2,`
`lwd = 2)`

Kaplan-Meier

- Estadístico
- Prueba de rangos logarítmicos (log rank test)
- Sólo cuando no hay otras covariables
- `survdif(Surv(time, cens) ~ treat, data = gehan)`

Análisis paramétricos

- Exponencial y Weibull



$$\mu = \frac{1}{\lambda}$$

**Tasa de mortalidad : Lambda
Lambda = 0.15**

**Tasa de riesgo de mortalidad
constante: Exponencial,
variable Weibull**

Análisis paramétricos

- Implementación en R, datos de leucemia
- `survreg(Surv(time) ~ ag + log(wbc), data = leuk, dist= "exponential")`
- `survreg(Surv(time, cens) ~ factor(pair) + treat, data = gehan, dist = "exponential")`
- Selección de modelos con diferente distribución probabilística

Regresión Cox

- Regresión de tipo exponencial

$$\log h_i(t) = \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik}$$

$$h_i(t) = \exp(\alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik})$$

- Regresión Cox

$$h_i(t) = h_0(t) \exp(\beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik})$$

Regresión de Cox

- `coxph(Surv(time, cens) ~ treat, data = gehan, method = "exact")`
- `motor.cox <- coxph(Surv(time, cens) ~ temp, motors)`
- `plot(survfit(motor.cox, newdata = data.frame(temp=200), conf.type = "log-log"))`
- `cox.zph(motor.cox)`