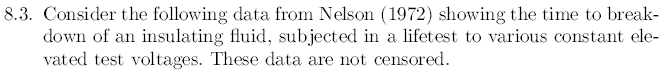
Chapter 8. Fitting Parametric Regression Models

library(survival)

 ### Carga de los datos

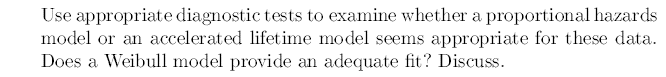
Se cargan los datos al entorno en formato de DataFrame, se crean 4 factores bajo la covariable KV (Kilo Volts), 30,32,34,46 que se refieren a cada uno de los grupos de estudio.

voltages <- read.csv("voltages.csv", header = T, sep = ",")  
voltages$KV <- as.factor(voltages$K)

Se visualizan los datos y el formato de cada columna del dataFrame.

str(voltages)

## 'data.frame': 60 obs. of 3 variables:  
## $ time: num 7.74 17.05 20.46 21.02 22.66 ...  
## $ KV : Factor w/ 4 levels "30","32","34",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ cens: int 1 1 1 1 1 1 1 1 1 1 ...



Estimación de la sobrevida vida Kaplan-Meier (EKM)

Para verificar si los datos hacen parte de un modelo acelerado o hacen parte de un modelo de datos de modelo de tiempo de vida acelerada, se creará un modelo de Kaplan-Meier para estimar el valor de los datos de supervivencia.

km.voltages <- survfit(Surv(time,cens)~KV, data=voltages)  
#summary(km.voltages)

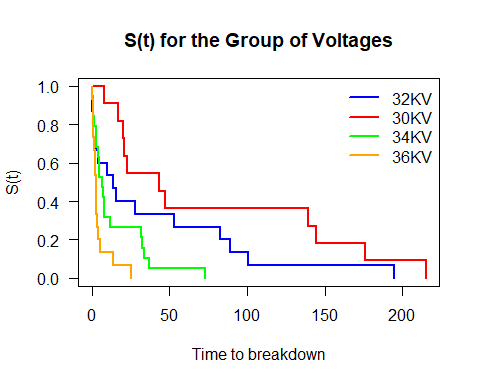
### 

### Gráficos de la estimación de la sobrevida vía Kaplan-Meier (EKM)

### Diagnostico 1.

Una primera forma de verificar si los datos se ajustan a un modelo de riesgos proporcionales lo podemos hacer mediante la visualización de la estimación de la sobrevida vía EKM.

plot(km.voltages, conf.int=F,col=c("red", "blue", "green", "orange"), xlab = "Time to breakdown", ylab = "S(t)", main="S(t) for the Group of Voltages", mark.time = TRUE, lwd = 2, las=1)  
legend("topright", legend=c("32KV","30KV","34KV","36KV"), lty = 1, lwd = 2, col=c("blue", "red", "green", "orange"), bty="n")



Como se puede ver los estimadores para los grupos 32,34,36, se cruzan entre sí, lo cual es un primer indicativo que NO se pueda cumplir el modelo de riesgos proporcionales.

### Gráficos de la estimación de la sobrevida vía Kaplan-Meier (EKM)

### Diagnostico 2.

Para verificar que efectivamente no se cumple los gráficos No se ajusten al modelo, ajustamos los datos a cada una de las diferentes distribuciones.

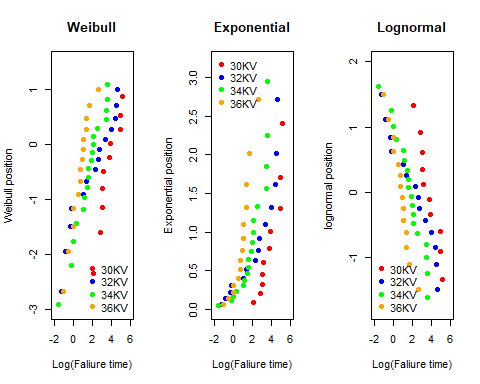
Dicho lo anterior extraemos los valores de la Supervivencia estimada y la ajustados a los modelos probabilísticos Weibull, exponencial y logarítmico.

St <- km.voltages$surv  
logtime <- log(km.voltages$time)  
weibull <- log(-log(St))  
exponential <- -log(St)  
lognormal <- qnorm(St)  
  
voltages <- cbind(voltages,St,logtime,weibull,exponential,lognormal)

### 

### Gráficos ajuste a los modelos Probabilísticos

par(mfrow = c(1,3))  
  
### Weibull  
plot(logtime[1:10],weibull[1:10],col="red", pch = 19 , ylab= "Weibull position", xlab="Log(Faliure time)", xlim=c(-2,6), ylim= c(-3,1.5), main = "Weibull")  
points(logtime[12:25],weibull[12:25],col="blue", pch = 19)  
points(logtime[27:44],weibull[27:44],col="green", pch = 19)  
points(logtime[46:59],weibull[46:59],col="orange", pch = 19)  
legend("bottomright", legend=c("30KV", "32KV", "34KV", "36KV"), col = c("red","blue","green","orange"), pch = 19, bty = "n")  
  
### Exponential  
plot(logtime[1:10],exponential[1:10],col="red", pch = 19 , ylab= "Exponential position", xlab="Log(Faliure time)", xlim=c(-2,6), ylim= c(0,3.2), main = "Exponential")  
points(logtime[12:25],exponential[12:25],col="blue", pch = 19)  
points(logtime[27:44],exponential[27:44],col="green", pch = 19)  
points(logtime[46:59],exponential[46:59],col="orange", pch = 19)  
legend("topleft", legend=c("30KV", "32KV", "34KV", "36KV"), col = c("red","blue","green","orange"), pch = 19, bty = "n")  
  
### lognormal  
plot(logtime[1:10],lognormal[1:10],col="red", pch = 19 , ylab= "lognormal position", xlab="Log(Faliure time)", xlim=c(-2,6), ylim= c(-1.8,2), main = "Lognormal")  
points(logtime[12:25],lognormal[12:25],col="blue", pch = 19)  
points(logtime[27:44],lognormal[27:44],col="green", pch = 19)  
points(logtime[46:59],lognormal[46:59],col="orange", pch = 19)  
legend("bottomleft", legend=c("30KV", "32KV", "34KV", "36KV"), col = c("red","blue","green","orange"), pch = 19, bty = "n")



Una vez más se observa claramente que las curvas para los modelos estimados se cruzan entre sí, indicando nuevamente la ausencia de un modelo de riesgos proporcionales.

### 

### Diagnostico 3

Como siguiente paso crearemos una función para el diagnóstico 3 visualizaremos el riesgo empírico y el riesgo acumulado para lo que se construye la siguiente función.

### Los siguientes ciclos calculan todos los datos para el riesgo empírico, para facilitar el cálculo los datos censurados se rellenan con ceros.  
  
CumulativeHazard <- function(len,k,time,Cens){  
   
 Emp.h <- c()  
   
 for(i in 1:len){  
   
 if(Cens[i] == 0){  
   
 Emp.h[i]= 0  
  
   
 }else{  
   
 Emp.h[i] = 1/k[i]  
  
 }  
   
}  
  
cum.H <- cumsum(Emp.h)  
  
 for(i in 1:len){  
   
 if(Cens[i] == 0){  
   
 cum.H[i]= 0  
  
 }  
 }  
  
  
 return (as.data.frame(cbind(time,Cens, k, Emp.h, cum.H)))  
   
}

Para facilitar el cálculo dividiremos el dataset en 4 grupos, un para cada nivel de voltaje

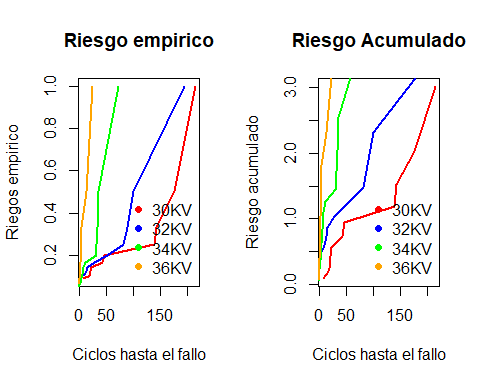
V30 <- subset(voltages, voltages$KV == 30)  
V32 <- subset(voltages, voltages$KV == 32)  
V34 <- subset(voltages, voltages$KV == 34)  
V36 <- subset(voltages, voltages$KV == 36)

Ahora llamaremos a la función para el calculo de los riesgos de cada grupo de voltajes

HV30 <- CumulativeHazard(length(V30$time), seq(length(V30$time),1),V30$time, V30$cens)  
HV32 <- CumulativeHazard(length(V32$time), seq(length(V32$time),1),V32$time, V32$cens)  
HV34 <- CumulativeHazard(length(V34$time), seq(length(V34$time),1),V34$time, V34$cens)  
HV36 <- CumulativeHazard(length(V36$time), seq(length(V36$time),1),V36$time, V36$cens)

### Graficos de ajuste las funciones de h(t) y H(t)

par(mfrow=c(1,2))  
  
plot(HV30$time,HV30$Emp.h, col="red",type="l", lwd = 2, main = "Riesgo empirico", ylab = "Riegos empirico", xlab = "Ciclos hasta el fallo")  
lines(HV32$time,HV32$Emp.h, col="blue", lwd = 2)  
lines(HV34$time,HV34$Emp.h, col="green", lwd = 2)  
lines(HV36$time,HV36$Emp.h, col="orange", lwd = 2)  
  
legend("bottomright", legend=c("30KV", "32KV", "34KV", "36KV"), col = c("red","blue","green","orange"), pch = 19, bty = "n")  
  
plot(HV30$time,HV30$cum.H, col="red",type="l", lwd = 2, main = "Riesgo Acumulado", ylab = "Riesgo acumulado", xlab = "Ciclos hasta el fallo")  
lines(HV32$time,HV32$cum.H, col="blue", lwd = 2)  
lines(HV34$time,HV34$cum.H, col="green", lwd = 2)  
lines(HV36$time,HV36$cum.H, col="orange", lwd = 2)  
  
legend("bottomright", legend=c("30KV", "32KV", "34KV", "36KV"), col = c("red","blue","green","orange"), pch = 19, bty = "n")

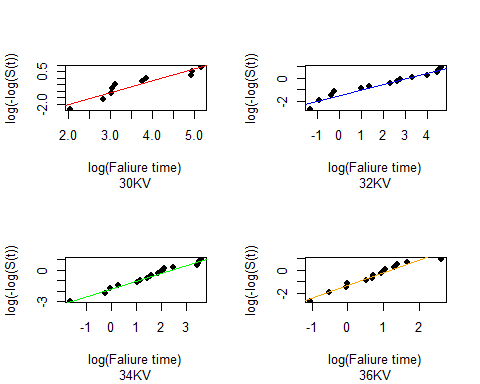


Se comprueba con el Diagnostico 3, no se sigue el modelo de riesgos proporcionales, por el contrario, pareciera que se sigue un modelo de vida acelerado, como es de esperarse para los modelos paramétricos de supervivencia pues estos no necesitan ser modelos de riesgos proporcionales. Muchos modelos paramétricos son modelos de vida acelerados

### Ajuste al modelo Weibull

Finalmente, y como lo solicita el ejercicio se ajustan los datos al modelo Weibull, con el fin de verificar si proporciona el ajuste correcto

lm30 <- lm(weibull[1:10] ~ logtime[1:10])  
lm32 <- lm(weibull[12:25] ~ logtime[12:25])  
lm34 <- lm(weibull[27:44] ~ logtime[27:44])  
lm36 <- lm(weibull[46:59] ~ logtime[46:59])  
  
par(mfrow=c(2,2))  
  
plot(weibull[1:10] ~ logtime[1:10], pch=16, xlab="log(Faliure time)", sub="30KV",ylab="log(-log(S(t))")  
abline(lm30, col ="red")  
  
plot(weibull[12:25] ~ logtime[12:25], pch=16, xlab="log(Faliure time)", sub="32KV",ylab="log(-log(S(t))")  
abline(lm32, col ="blue")  
  
plot(weibull[27:44] ~ logtime[27:44], pch=16, xlab="log(Faliure time)", sub="34KV",ylab="log(-log(S(t))")  
abline(lm34, col ="green")  
  
plot(weibull[46:59] ~ logtime[46:59], pch=16, xlab="log(Faliure time)", sub="36KV",ylab="log(-log(S(t))")  
abline(lm36, col ="orange")



Gráficamente podemos comprobar que el modelo se ajusta bastante bien a una distribución Weibull, pues gran parte de los datos se ajustan a la regresión.

Weibull\_voltages<-survreg(Surv(voltages$time, voltages$cens) ~ voltages$KV, dist='weibull')  
alpha<-1/Weibull\_voltages$scale  
  
Weibull\_voltages

## Call:  
## survreg(formula = Surv(voltages$time, voltages$cens) ~ voltages$KV,   
## dist = "weibull")  
##   
## Coefficients:  
## (Intercept) voltages$KV32 voltages$KV34 voltages$KV36   
## 4.2470237 -0.7736785 -1.7471331 -2.8693640   
##   
## Scale= 1.304872   
##   
## Loglik(model)= -232.7 Loglik(intercept only)= -246  
## Chisq= 26.74 on 3 degrees of freedom, p= 6.67e-06   
## n= 60

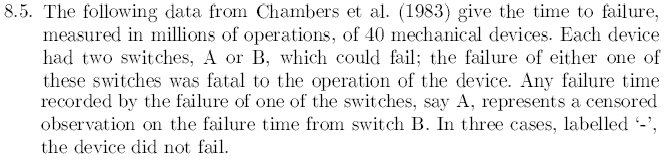
alpha

## [1] 0.7663588

Weibull\_voltages$loglik

## [1] -246.0295 -232.6581

Para una distribución ji-cuadrado, con p = 0.28, se evidencia que los datos siguen una distribución weibull, con un alpha = 0.7664.



### Se cargan los datos de los Switches

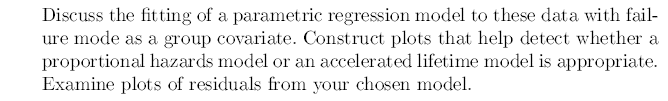
Se cargan los datos al entorno en formato de DataFrame, se crea una variable tipo factores dummy llamada grupo que divide los Switches entre los que fallaron en A y los que fallaron en B

##### Visualización de los Datos

swt <- read.csv("switches.csv", header=T, sep =";")  
swt$Group <- as.factor(swt$Group)  
head(swt)

## time A B Group  
## 1 1.151 0 1 B  
## 2 1.170 0 1 B  
## 3 1.248 0 1 B  
## 4 1.331 0 1 B  
## 5 1.381 0 1 B  
## 6 1.499 1 0 A

knitr::include\_graphics("8\_51.PNG")



Separación de los datos

Para facilitar el análisis se separan los datos en dos subset de datos diferentes, los switches que fallaron en A y los que fallaron en B.

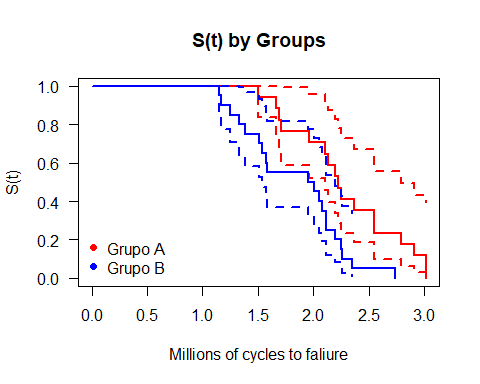
GroupA <- subset(swt, Group == "A")  
GroupB <- subset(swt, Group == "B")

### Diagnostico 1

Una primera forma de verificar si los datos se ajustan a un modelo de riesgos proporcionales lo podemos hacer mediante la visualización de la estimación de la sobrevida vía EKM.

km.GA <- survfit(Surv(time,A)~1, data = GroupA)  
km.GB <- survfit(Surv(time,B)~1, data = GroupB)

plot(km.GA, conf.int=T,col=c("red"), xlab = "Millions of cycles to faliure", ylab = "S(t)", main="S(t) by Groups", mark.time = TRUE, lwd = 2, las=1)  
lines(km.GB, conf.int=T,col=c("blue"), mark.time = TRUE, lwd = 2, las=1)  
legend("bottomleft", legend=c("Grupo A", "Grupo B"), col = c("red","blue"), pch = 19, bty = "n")



Gracias al estimador Kaplan meier vemos que los Switches pertenecientes al grupo A tiene un mejor desempeño que los del grupo B. También se observa que los estimadores no se cruzan entre si sugiriendo un modelo de riesgos proporcionales

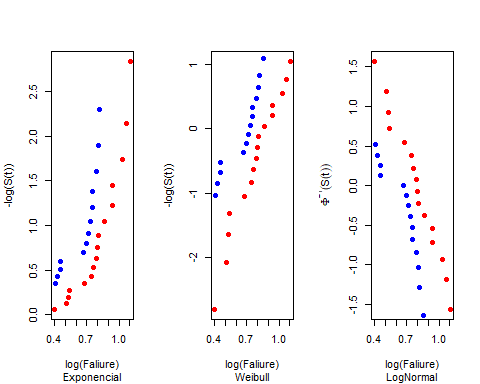
### Diagnostico 2

A continuación, se procede ajustar el modelo a un modelo probabilístico Weibull, Exponencial, lognormal.

######  
  
logtimeA <- log(GroupA$time)  
StA <- km.GA$surv   
weibullA <- log(-log(StA))  
exponentialA <- -log(StA)  
lognormalA <- qnorm(StA)  
  
GroupA <- cbind(GroupA,StA,logtimeA,weibullA,exponentialA,lognormalA)  
  
  
######  
  
logtimeB <- log(GroupB$time)  
StB <- km.GB$surv   
weibullB <- log(-log(StB))  
exponentialB <- -log(StB)  
lognormalB <- qnorm(StB)  
  
GroupB <- cbind(GroupB,StB,logtimeB,weibullB,exponentialB,lognormalB)

### Graficos de los modelos probabilisticos

par(mfrow=c(1,3))  
  
plot(logtimeA, exponentialA, pch=16, xlab="log(Faliure)", sub="Exponencial",ylab="-log(S(t))", col ="red")  
points(logtimeB, exponentialB,pch=16, col="blue")  
  
plot(logtimeA, weibullA, pch=16, xlab="log(Faliure)", sub="Weibull",ylab="-log(S(t))", col ="red")  
points(logtimeB, weibullB,pch=16, col="blue")  
  
plot(logtimeA, lognormalA , pch=16, xlab="log(Faliure)", sub="LogNormal",ylab= expression(Phi^-1 \* (S(t))), col ="red")  
points(logtimeB, lognormalB ,pch=16, col="blue")



Visualmente se puede decir que el modelo sigue una distribución Weibull, y parece ajustarse aun modelo de reisgos proporcionales.

### Verificación via Cox-snell

Ahora diagnosticaremos la calidad del modelo elegido frente al conjunto de datos.

### Función para el calculo de los residuales CoxSnell

### Function to calculate the CoxSnell residual  
  
### Modified from   
### https://blogs2.datall-analyse.nl/2016/02/19/rcode\_martingale\_residuals\_aft\_model/  
### Author:Stefan Gelissen  
### Based on: Modelling survival data in medical research, Collett’s 2003  
  
CoxSnellResidual <- function (standRes, weight=1, dist){  
   
 standRes <- standRes[rep(seq\_len(length(standRes)), weight)]  
   
 if (dist=="lognormal") {  
   
 csr <- -log(1-pnorm(standRes))  
 }else if (dist=="weibull") {  
   
 csr <- -log(exp(-exp(standRes)))  
   
 }  
}

### Calculos de los valores de la regresión

Se obtiene la regresión para ambos grupos de Switch

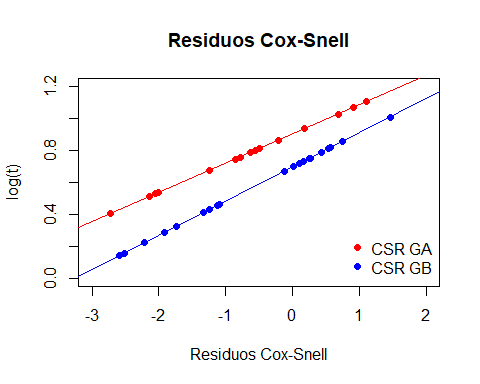
reg\_weibullA<-survreg(Surv(time, A)~1, dist='weibull',data = GroupA)  
  
reg\_weibullB<-survreg(Surv(time, B)~1, dist='weibull',data = GroupB)

### Residuos estandarizados

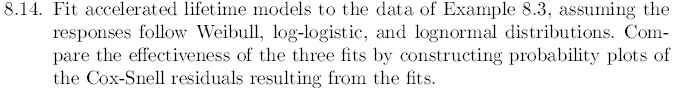
lfA <- predict(reg\_weibullA, type="lp")  
sderr <- (logtimeA-lfA)/reg\_weibullA$scale  
cxsnA <- CoxSnellResidual(standRes=sderr, dist="weibull")  
  
lfB <- predict(reg\_weibullB, type="lp")  
sderr <- (logtimeB-lfB)/reg\_weibullB$scale  
cxsnB <- CoxSnellResidual(standRes=sderr, dist="weibull")

### Calculo de los residuales CoxSnell

### CS Weibull  
  
plot(log(cxsnA), logtimeA,col="red", pch = 19 , ylab= "log(t)", xlab="Residuos Cox-Snell", main = "Residuos Cox-Snell", xlim=c(-3,2), ylim=c(0,1.2))  
abline(lm(logtimeA~log(cxsnA)), col="red")  
  
points(log(cxsnB),logtimeB,col="blue", pch = 19)  
abline(lm(logtimeB~log(cxsnB)), col="blue")  
  
legend("bottomright", legend=c("CSR GA", "CSR GB"), col = c("red","blue"), pch = 19, bty = "n")



Vemos que los residuales Cox-Snell se ajustan a la recta a la perfección por lo que podemos asegurar que le modelo sigue una distribución Weibull, pero no podemos asegurar que se ajusta a un modelo proporcional pues no formar un Angulo cercano a 45° a pesar que el intercepto es cercano a 0, y por ende al ser Weibull también a un modelo de vida acelerado.



### Ajuste de los datos a una regresión

Para este ejercicio nuevamente se tomas los datos en el dataset de voltajes y se ajustan aun modelo a un weibull o a uno lognormal, haciendo uso de los residuos Cox\_snell

wsr <- survreg(Surv(time, cens)~ KV, data = voltages, dist="weibull")  
  
lnsr <- survreg(Surv(time, cens) ~ KV, data = voltages, dist="lognormal")

### Calculo de los residuos estandarizados lognormal

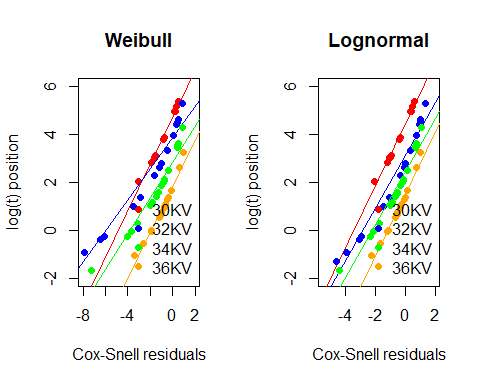
lfw <- predict(wsr, type="lp")  
sderr <- (log(voltages$time)-lfw)/wsr$scale  
cxsnw <- CoxSnellResidual(standRes=sderr, dist="lognormal")

### Calculo de los residuos estandarizados Weibull

lfln <- predict(lnsr, type="lp")  
sderr <- (log(voltages$time)-lfln)/lnsr$scale  
cxsnln <- CoxSnellResidual(standRes=sderr, dist="lognormal")

### Visualización de los residuales Cox-Snell para Weibull o para log normal.

par(mfrow = c(1,2))  
  
### Weibull  
  
plot(log(cxsnw[1:11]), logtime[1:11],col="red", pch = 19 , ylab= "log(t) position", xlab="Cox-Snell residuals", main = "Weibull", xlim=c(-8,2), ylim=c(-2,6))  
abline(lm(logtime[1:11]~log(cxsnw[1:11])), col="red")  
  
points(log(cxsnw[12:26]),logtime[12:26],col="blue", pch = 19)  
abline(lm(logtime[12:26]~log(cxsnw[12:26])), col="blue")  
  
points(log(cxsnw[27:45]),logtime[27:45],col="green", pch = 19)  
abline(lm(logtime[27:45]~log(cxsnw[27:45])), col="green")  
  
points(log(cxsnw[46:60]),logtime[46:60],col="orange", pch = 19)  
abline(lm(logtime[46:60]~log(cxsnw[46:60])), col="orange")  
  
legend("bottomright", legend=c("30KV", "32KV", "34KV", "36KV"), col = c("red","blue","green","orange"), pch = 19, bty = "n")  
  
  
### Lognormal  
  
plot(log(cxsnln[1:11]), logtime[1:11],col="red", pch = 19 , ylab= "log(t) position", xlab="Cox-Snell residuals", main = "Lognormal", xlim=c(-5.5,2), ylim=c(-2,6))  
abline(lm(logtime[1:11]~log(cxsnln[1:11])), col="red")  
  
points(log(cxsnln[12:26]),logtime[12:26],col="blue", pch = 19)  
abline(lm(logtime[12:26]~log(cxsnln[12:26])), col="blue")  
  
points(log(cxsnln[27:45]),logtime[27:45],col="green", pch = 19)  
abline(lm(logtime[27:45]~log(cxsnln[27:45])), col="green")  
  
points(log(cxsnln[46:60]),logtime[46:60],col="orange", pch = 19)  
abline(lm(logtime[46:60]~log(cxsnln[46:60])), col="orange")  
  
legend("bottomright", legend=c("30KV", "32KV", "34KV", "36KV"), col = c("red","blue","green","orange"), pch = 19, bty = "n")



Mediante la visualización de los residuales Cox-snell podemos ver que nuestro modelo se ajusta mejor a una distribución Log normal que a una Weibull y no sigue un modelo de riesgos proporcionales, pues su intercepto no es cercano a 0 y su recta no forma 45°