

# Capítulo 5

## Modelos de Confiabilidade



Gustavo Mello Reis  
José Ivo Ribeiro Júnior

Universidade Federal de Viçosa  
Departamento de Informática  
Setor de Estatística

Viçosa 2007

## 1. Introdução

Nesse estudo, serão abordadas as funções de confiabilidade  $[R(t)]$ , de desconfiabilidade  $[F(t)]$ , densidade de probabilidade  $[f(t)]$  e de taxa de falha  $[h(t)]$ , estimadas a partir de modelos não paramétricos ou paramétricos.

A análise de confiabilidade, no R, é realizada por meio do pacote survival, que após instalado, deverá ser ativado com o comando `library(survival)`.

## 2. Estimação Não Paramétrica

O estimador não paramétrico é aquele, para o qual não há a necessidade de especificar uma distribuição para a variável tempo de falha (T), podendo ser utilizado na ausência ou na presença de censuras.

### 2.1. Estimador de Kaplan-Meier

Como exemplo, considere os dados contidos na tabela 1, onde o ensaio terminou quando 45 itens haviam falhados. Como pode ser observado, o R reconhece o número 1 como sendo falha e o número 0 (zero) como sendo censura. Do mesmo modo, outros indicadores podem ser usados, como exemplo, T (falha) e F (censura).

Tabela 1. Dados dos tempos (h) de falhas

tempo	tipo	tempo	tipo	tempo	tipo	tempo	tipo
151	1	727	1	1329	1	2729	0
164	1	785	1	1334	1	2729	0
336	1	801	1	1379	1	2729	0
365	1	811	1	1380	1	2729	0
403	1	816	1	1633	1	2729	0
454	1	867	1	1769	1	2729	0
455	1	893	1	1827	1	2729	0
473	1	930	1	1831	1	2729	0
538	1	937	1	1849	1	2729	0
577	1	976	1	2016	1	2729	0
592	1	1008	1	2282	1	2729	0
628	1	1040	1	2415	1	2729	0
632	1	1051	1	2430	1	2729	0

647	1	1060	1	2686	1	2729	0
675	1	1183	1	2729	1	2729	0

A forma como os dados estão, apresentados é apenas ilustrativa. Ao entrar com os dados no R, deve-se organizá-los em duas colunas, uma contendo o tempo e a outra contendo o tipo de falha. Desse modo os dados foram organizados no arquivo “conf.csv”, que será lido da seguinte forma:

```
dados.conf<-read.csv2("conf.csv", dec= ".")
attach(dados.conf)
```

As estimativas de confiabilidades de Kaplan-Meier, são obtidas da seguinte forma:

```
mod.np<-survfit(Surv(tempo, tipo))
summary(mod.np) # Resumo do modelo ( coluna survival =  $\hat{R}(t)$  )
```

```
Call: survfit(formula = Surv(tempo, tipo))

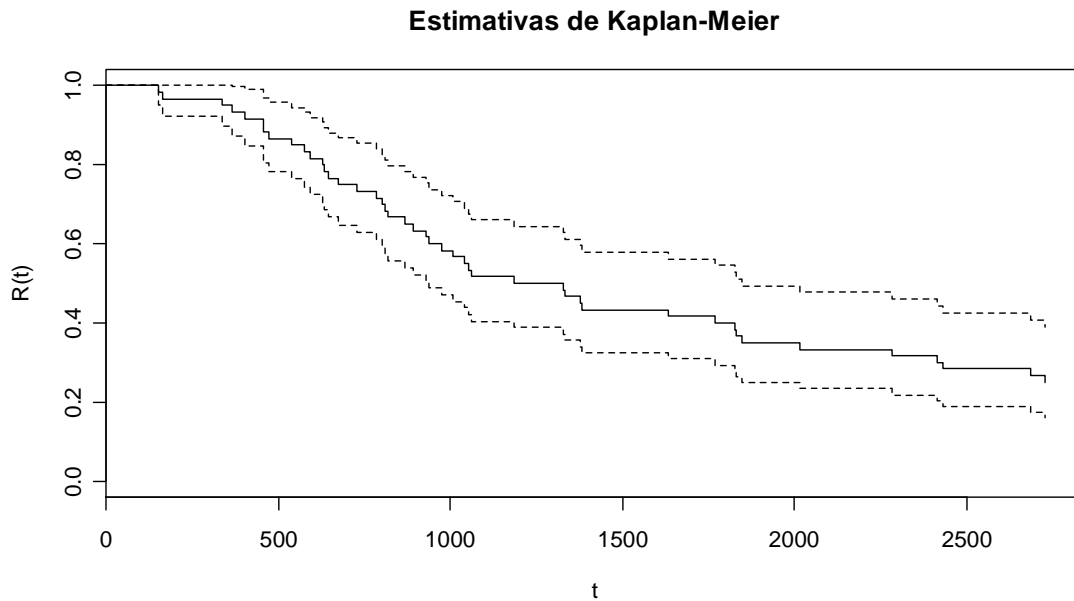
time n.risk n.event survival std.err lower 95% CI upper 95% CI
151   60     1    0.983  0.0165    0.951    1.000
164   59     1    0.967  0.0232    0.922    1.000
336   58     1    0.950  0.0281    0.896    1.000
365   57     1    0.933  0.0322    0.872    0.999
403   56     1    0.917  0.0357    0.849    0.989
454   55     1    0.900  0.0387    0.827    0.979
455   54     1    0.883  0.0414    0.806    0.968
473   53     1    0.867  0.0439    0.785    0.957
538   52     1    0.850  0.0461    0.764    0.945
577   51     1    0.833  0.0481    0.744    0.933
592   50     1    0.817  0.0500    0.724    0.921
628   49     1    0.800  0.0516    0.705    0.908
632   48     1    0.783  0.0532    0.686    0.895
647   47     1    0.767  0.0546    0.667    0.882
675   46     1    0.750  0.0559    0.648    0.868
727   45     1    0.733  0.0571    0.630    0.854
785   44     1    0.717  0.0582    0.611    0.840
801   43     1    0.700  0.0592    0.593    0.826
811   42     1    0.683  0.0601    0.575    0.812
816   41     1    0.667  0.0609    0.557    0.797
```

867	40	1	0.650	0.0616	0.540	0.783
893	39	1	0.633	0.0622	0.522	0.768
930	38	1	0.617	0.0628	0.505	0.753
937	37	1	0.600	0.0632	0.488	0.738
976	36	1	0.583	0.0636	0.471	0.722
1008	35	1	0.567	0.0640	0.454	0.707
1040	34	1	0.550	0.0642	0.437	0.691
1051	33	1	0.533	0.0644	0.421	0.676
1060	32	1	0.517	0.0645	0.405	0.660
1183	31	1	0.500	0.0645	0.388	0.644
1329	30	1	0.483	0.0645	0.372	0.628
1334	29	1	0.467	0.0644	0.356	0.612
1379	28	1	0.450	0.0642	0.340	0.595
1380	27	1	0.433	0.0640	0.324	0.579
1633	26	1	0.417	0.0636	0.309	0.562
1769	25	1	0.400	0.0632	0.293	0.545
1827	24	1	0.383	0.0628	0.278	0.528
1831	23	1	0.367	0.0622	0.263	0.511
1849	22	1	0.350	0.0616	0.248	0.494
2016	21	1	0.333	0.0609	0.233	0.477
2282	20	1	0.317	0.0601	0.218	0.459
2415	19	1	0.300	0.0592	0.204	0.442
2430	18	1	0.283	0.0582	0.189	0.424
2686	17	1	0.267	0.0571	0.175	0.406
2729	16	1	0.250	0.0559	0.161	0.388

Para a construção do gráfico de  $R(t)$  em função de  $t$ , em forma de escada, deve-se utilizar o comando plot da seguinte forma:

```
plot(mod.np, main= "Estimativas de Kaplan-Meier", ylab= "R(t)", xlab= "t")
```

Figura 1. Estimativas de confiabilidade de Kaplan-Meier



### 3. Estimação Paramétrica

Embora exista uma série de distribuições de probabilidades utilizadas nas análise de dados de confiabilidade, apenas algumas delas (normal, log-normal, exponencial e de Weibull), por serem adequadas a várias situações práticas, serão estudadas. Por isso é importante entender que cada modelo pode gerar estimadores diferentes para o mesmo parâmetro desconhecido. Desta forma, a escolha de um modelo adequado para descrever o tempo de falha deve ser feita com bastante cuidado.

#### 3.1. Método de Mínimos Quadrados

O método de mínimos quadrados (least squares method) permite obter estimativas dos parâmetros para modelos que podem ser linearizados, cujos dados podem ser completos (não há censuras) ou com poucas censuras.

Assim, após a obtenção dos tempos de falhas ( $t$ ), a função de confiabilidade [ $R(t) = 1 - F(t)$ ] deve ser linearizada e plotada no eixo y, sendo o eixo x constituído pelos tempos de falhas ou pelos logaritmos naturais dos tempos de falhas. Posteriormente, uma equação de regressão linear é ajustada aos pontos plotados no gráfico, onde as estimativas de mínimos quadrados são obtidas com base na

minimização da soma de quadrados das distâncias dos pontos em relação à reta ajustada.

Esse método também pode ser chamado de posição da regressão em Y (rank regression on Y), pois as estimativas dos parâmetros são baseadas na função de desconfiabilidade estimada [S(t)] obtida com base no cálculo da posição da mediana ou da média ou de Kaplan-Meier modificado ou não.

Para o cálculo de S(t), considere que os valores da amostra  $t_1, t_2, \dots, t_r, t_{r+1}, \dots, t_n$ , arranjados em ordem crescente, onde  $t_i$  é o tempo de falha de ordem i observado. Assim,  $t_1$  é o menor tempo e  $t_r$  é o maior tempo de falha de n observações que falharam ou censuraram. Neste caso, podem ser usados quaisquer um dos quatro métodos mencionados:

$$S(t_i) = \frac{i - 0,3}{n + 0,4} \text{ (posição da mediana segundo Bernard);}$$

$$S(t_i) = \frac{i}{n + 1} \text{ (posição da média segundo Herd-Johnson);}$$

$$S(t_i) = \frac{i - 0,5}{n} \text{ (posição de Kaplan-Meier modificado por Hazen);}$$

$$S(t_i) = \frac{i}{n} \text{ (posição de Kaplan-Meier).}$$

Para calcular S(t<sub>i</sub>) no R, utiliza-se o comando `ppoints(n, a)` onde “n” indica o número total de observações e “a” recebe valores que irão indicar o método a ser utilizado. Para a = 0.3, utiliza-se o primeiro método, para a = 0, utiliza-se o segundo método e para a = 0.5, utiliza-se o terceiro método.

Se  $y_i = \beta_0 + \beta_1 x_i + e_i$ , então as estimativas de mínimos quadrados dos parâmetros  $\beta_0$  e  $\beta_1$  são dadas por:

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X};$$

$$\hat{\beta}_1 = \frac{\sum_{i=1}^r X_i Y_i - \frac{\sum_{i=1}^r X_i \sum_{i=1}^r Y_i}{r}}{\sum_{i=1}^r X_i^2 - \frac{\left(\sum_{i=1}^r X_i\right)^2}{r}}.$$

Assim, a equação da regressão estimada é dada por:

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i.$$

Como exemplo para todas as distribuições estudadas, considere os dados contidos no arquivo “mquad.csv” que possui nove falhas e uma censura. O arquivo de dados será lido da seguinte forma:

```
dados.mq<-read.csv2("mquad.csv",dec=".")
```

```
dados.mq
```

```
tempo    tipo
   10     1
   25     1
   35     1
   45     1
   50     1
   55     1
   65     1
   75     1
   90     1
   90     0
```

```
attach(dados.mq)
```

### 3.1.1. Distribuição Normal

De acordo com o modelo normal, têm-se:

$$y_i = \Phi^{-1}[S(t_i)];$$

$$x_i = t_i;$$

$$\bar{T} = -\hat{\beta}_0 s_T \text{ (estimador de } \mu_T);$$

$$s_T = \frac{1}{\hat{\beta}_1} \text{ (estimador de } \sigma_T).$$

Para calcular  $S(t_i)$  pelo método da posição da mediana segundo Bernard, no R, tem-se:

```
s.t<-ppoints(10, 0.3)
```

```
s.t
```

```
[1] 0.06730769 0.16346154 0.25961538 0.35576923 0.45192308 0.54807692
[7] 0.64423077 0.74038462 0.83653846 0.93269231
```

O último valor (0,9326931) deve ser desconsiderado, pois se refere à censura.

Para eliminar o último valor de s.t e calcular  $y_i$  no R, tem-se:

```
s.t<-s.t[1:9]
```

```
Ynorm<-qnorm(s.t)
```

```
Ynorm
```

```
[1]-1.4961469 -0.9803304 -0.6445316 -0.3697907 -0.1208042  0.1208042  
[7]  0.3697907  0.6445316  0.9803304
```

Com esses dados, são obtidos os valores de  $S(t_i)$  e  $y_i$  (tabela 2).

Tabela 2. Dados da distribuição normal

tempo	tipo	$S(t_i)$	$Y_i$
10	1	0.06730769	-1.4961469
25	1	0.16346154	-0.9803304
35	1	0.25961538	-0.6445316
45	1	0.35576923	-0.3697907
50	1	0.45192308	-0.1208042
55	1	0.54807692	-0.1208042
65	1	0.64423077	0.3697907
75	1	0.74038462	0.6445316
90	1	0.83653846	0.9803304
90	0	--	--

Para estimar  $\beta_0$  e  $\beta_1$ , será criado um vetor no R contendo os valores de tempo (excluindo o tempo correspondente à censura) e depois será feita a regressão de Y em função do tempo, da seguinte forma:

```
tempo.f<-c(10,25,35,45,50,55,65,75,90)
```

```
mod.norm<-lm(Ynorm~tempo.f)
```

```
summary(mod.norm)
```

```
Call:  
lm(formula = Ynorm ~ tempo.f)
```

```
Residuals:
```

```

      Min          1Q      Median          3Q          Max
-0.124934 -0.044614 -0.001480  0.045434  0.128105

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.755617    0.061971  -28.33 1.76e-08 ***
tempo.f      0.031788    0.001122   28.33 1.76e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.07895 on 7 degrees of freedom
Multiple R-Squared:  0.9914,    Adjusted R-squared:  0.9901
F-statistic: 802.5 on 1 and 7 DF,  p-value: 1.756e-08

```

Como pode ser observado  $\hat{\beta}_0 = -1.755617$  e  $\hat{\beta}_1 = 0.031788$ . Assim a estimativa dos parâmetros  $\sigma_T$  e  $\mu_T$ , sendo calculadas da seguinte forma:

```
dp<- 1/0.031788
```

```
dp
```

```
[1] 31.45841
```

```
m<- -(-1.755617*dp)
```

```
m
```

```
[1] 55.22892
```

Portanto, a estimativa da função de confiabilidade é dada por:

$$\hat{R} = 1 - \hat{\Phi}\left(\frac{t - 55,22892}{31,45841}\right).$$

### 3.1.2. Distribuição Log-normal

De acordo com o modelo log-normal, têm-se:

$$y_i = \Phi^{-1}[S(t_i)];$$

$$x_i = y_i = \ln(t_i);$$

$$\bar{Y} = -\hat{\beta}_0 s_Y \text{ (estimador de } \mu_Y);$$

$$s_Y = \frac{1}{\hat{\beta}_1} \text{ (estimador de } \sigma_Y), \text{ em que } Y = \ln(T).$$

Como  $S(t_i)$  e  $y_i$  já foram calculados anteriormente, o vetor  $x_i = \ln(t_i)$  será obtido da seguinte forma no R:

```
X<-log(tempo.f)
```

Para estimar  $\mu_Y$  e  $\sigma_Y$ , será montado o modelo e depois será utilizado a função `summary` para obter os estimadores  $\hat{\beta}_0$  e  $\hat{\beta}_1$ , da seguinte forma:

```
mod.ln<-lm(Y~X)
summary(mod.ln)
```

```
Call:
lm(formula = Y ~ X)

Residuals:
    Min       1Q   Median       3Q      Max
-0.2623950 -0.2041550 -0.0001219  0.1707134  0.3225779

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -4.4386     0.4918  -9.025 4.19e-05 ***
X              1.1378     0.1292   8.809 4.90e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.2442 on 7 degrees of freedom
Multiple R-Squared:  0.9173,    Adjusted R-squared:  0.9054
F-statistic: 77.59 on 1 and 7 DF,  p-value: 4.904e-05
```

Como pode ser observado,  $\hat{\beta}_0 = -4,4386$  e  $\hat{\beta}_1 = 1,1378$ . Assim, as estimativas dos parâmetros  $\mu_Y$  e  $\sigma_Y$ , serão calculadas da seguinte forma:

```
sy<- 1/1.1378
```

```
sy
```

```
[1] 0.878889
```

```
my<- -(-4.4386*sy)
```

```
my
```

```
[1] 3.901037
```

Portanto, a estimativa da função de confiabilidade é dada por:

$$\hat{R} = 1 - \hat{\Phi}\left(\frac{t - 3,901037}{0,878889}\right)$$

### 3.1.3. Distribuição Exponencial

De acordo com o modelo exponencial, têm-se:

$$y_i = \ln\{-\ln[1-S(t_i)]\};$$

$$x_i = \ln(t_i);$$

$$\hat{\alpha} = e^{-\hat{\beta}_0} \text{ (estimador de } \alpha\text{)}.$$

Como  $S(t_i)$  e  $x_i$  já foram calculados anteriormente,  $y_i = \ln\{-\ln[1-S(t_i)]\}$  será calculado da seguinte forma:

```
Yexp<- log(-log(1-s.t))
```

O estimador  $\hat{\beta}_0$  é calculado da seguinte forma:

```
B0<-mean(Yexp)-1*mean(X)
```

```
B0
```

```
[1] -4.446482
```

Assim a estimativa do parâmetro  $\alpha$  será calculada da seguinte forma:

```
exp(-B0)
```

```
[1] 85.3262
```

Portanto, a estimativa da função de confiabilidade é dada por:

$$\hat{R} = e^{-\frac{t}{85,3262}}$$

### 3.1.4. Distribuição de Weibull

De acordo com o modelo de Weibull, têm-se:

$$y_i = \ln\{-\ln[1-S(t_i)]\};$$

$$x_i = \ln(t_i);$$

$$\hat{\alpha} = e^{-\frac{\hat{\beta}_0}{\hat{\beta}_1}} \text{ (estimador de } \alpha\text{);}$$

$$\hat{\delta} = \hat{\beta}_1 \text{ (estimador de } \delta\text{).}$$

Como  $S(t_i)$ ,  $y_i$  e  $x_i$ , já foram calculados anteriormente, para obter os estimadores  $\hat{\beta}_0$  e  $\hat{\beta}_1$ , será montado o seguinte modelo:

```
mod.wei<-lm(Yexp~X)
```

```
summary(mod.wei)
```

```
Call:
lm(formula = Yexp ~ X)

Residuals:

    Min       1Q   Median       3Q      Max
-0.21048 -0.20471  0.07452  0.12877  0.25266

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -6.44400     0.37820  -17.04 5.88e-07 ***
X             1.53197     0.09933   15.42 1.16e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1878 on 7 degrees of freedom
Multiple R-Squared:  0.9714,    Adjusted R-squared:  0.9673
F-statistic: 237.9 on 1 and 7 DF,  p-value: 1.163e-06
```

Como pode ser observado,  $\hat{\beta}_0 = -6,44400$  e  $\hat{\beta}_1 = 1,53197$ . Como  $\hat{\delta} = \hat{\beta}_1$ , logo:

$\hat{\delta} = 1,53197$ . Já a estimativa do parâmetro  $\alpha$  será calculada da seguinte forma:

```
a1<-exp(6.444 / 1.53197)
```

```
a1
```

```
[1] 67.11105
```

Portanto, a estimativa da função de confiabilidade em dada por:

$$\hat{R} = e^{-\left(\frac{t}{67,111049}\right)^{1,53197}}$$

### 3.2. Método de Máxima Verossimilhança

O método de máxima verossimilhança (maximum likelihood) permite obter estimativas dos parâmetros para modelos que podem ou não ser linearizados, cujos dados podem ser completos ou com presenças de censuras, sendo o mais recomendado para a aplicação em dados provenientes de falhas e de censuras que ocorrem ao longo do tempo.

A função existente no R responsável por estimar os parâmetros por este método (survreg), exige que seja criado um outro vetor de valores. Como essa outra variável não tem interesse no momento, será criado um vetor com valores constantes, para que este não tenha nenhuma influência no resultado.

Como exemplo para todas as distribuições, considere os dados contidos no arquivo “estpara.csv” que possui nove falhas e uma censura. A variável  $x = 1$  é utilizada somente com o objetivo de se ter uma variável regressora. O arquivo de dados será lido da seguinte forma:

```
dados.para<-read.csv2("estpara.csv",dec=".")
```

```
dados.para
```

tempo	tipo	x
10	1	1
25	1	1
35	1	1
45	1	1
50	1	1
55	1	1
65	1	1
75	1	1
90	1	1
90	0	1

```
attach(dados.para)
```

#### 3.2.1. Distribuição Normal

Primeiro, deve-se montar o modelo da seguinte forma:

```
mod.norm<-survreg(Surv(tempo,tipo) ~ x, dist = "gaussian")
```

Para ver as estimativas, deve-se usar a função summary. A média estimada está no item "Intercept" e o desvio padrão estimado no item "Scale".

```
summary(mod.norm)
```

```
Call:
survreg(formula = Surv(tempo, tipo) ~ x, dist = "gaussian")

              Value Std. Error      z      p
(Intercept) 55.32      8.822  6.27 3.60e-10
x              0.00      0.000   NaN   NaN
Log(scale)   3.32      0.242 13.71 8.28e-43

Scale= 27.6

Gaussian distribution
Loglik(model)= -43.8  Loglik(intercept only)= -43.8
      Chisq= 0 on 1 degrees of freedom, p= 1
Number of Newton-Raphson Iterations: 3
n= 10
```

Como pode-se observar, os estimadores de máxima verossimilhança  $\bar{T} = 55,32$  e  $s_T = 27,6$ , são relativamente próximos aos de mínimos quadrados.

### 3.2.2. Distribuição Log-Normal

Do mesmo modo, deve-se montar o modelo da seguinte forma:

```
mod.lnorm<-survreg(Surv(tempo,tipo) ~ x, dist= "lognormal")
```

```
summary(mod.lnorm)
```

```
Call:
survreg(formula = Surv(tempo, tipo) ~ x, dist = "lognormal")

              Value Std. Error      z      p
(Intercept)  3.867      0.222 17.42 5.45e-68
x              0.000      0.000   NaN   NaN
Log(scale)  -0.366      0.241 -1.52 1.28e-01

Scale= 0.693

Log Normal distribution
Loglik(model)= -44.3  Loglik(intercept only)= -44.3
```

```
Chisq= 0 on 1 degrees of freedom, p= 1
Number of Newton-Raphson Iterations: 3
n= 10
```

Os estimadores de máxima verossimilhança  $\bar{Y} = 3,867$  e  $s_Y = 0,693$  ( $Y=\ln T$ ), são relativamente próximos aos de mínimos quadrados.

### 3.2.3. Distribuição Exponencial

Neste caso, deve-se aplicar o exponencial no parâmetro estimado pelo R (item Intercept), para se obter a estimativa da média.

```
mod.exp<-survreg(Surv(tempo,tipo) ~ x, dist= "exponential")
summary(mod.exp)
```

```
Call:
survreg(formula = Surv(tempo, tipo) ~ x, dist = "exponential")

              Value Std. Error      z      p
(Intercept)  4.09         0.333 12.3 1.12e-34
x              0.00         0.000  NaN   NaN

Scale fixed at 1

Exponential distribution
Loglik(model)= -45.8   Loglik(intercept only)= -45.8
      Chisq= 0 on 1 degrees of freedom, p= 1
Number of Newton-Raphson Iterations: 4
n= 10
```

```
exp(4.09) # Aplicando o exponencial para obter  $\hat{\alpha}$ 
```

```
[1] 59.73989
```

Na estimação por mínimos quadrados, obteve-se  $\hat{\alpha}=85,3262$ .

### 3.2.4. Distribuição de Weibull

Assim como para a distribuição exponencial, deve-se aplicar o exponencial no parâmetro estimado pelo R (item Intercept), para se obter a estimativa do parâmetro  $\alpha$ , e o inverso (item Scale), para se obter a estimativa do parâmetro  $\delta$ .

```
mod.wei<-survreg(Surv(tempo,tipo) ~ x, dist= "weibull")
```

summary(mod.wei)

```
Call:
survreg(formula = Surv(tempo, tipo) ~ x, dist = "weibull")

              Value Std. Error      z      p
(Intercept)  4.142         0.168 24.7 2.25e-134
x              0.000         0.000  NaN    NaN
Log(scale)   -0.703         0.281 -2.5  1.25e-02

Scale= 0.495

Weibull distribution
Loglik(model)= -43.5   Loglik(intercept only)= -43.5
      Chisq= 0 on 1 degrees of freedom, p= 1
Number of Newton-Raphson Iterations: 6
n= 10
```

exp(4.142) # Aplicando o exponencial para obter  $\hat{\alpha}$

```
[1] 62.92855
```

1/0,495 # Aplicando a divisão para obter  $\hat{\delta}$

```
[1] 2.020202
```

Na estimação por mínimos quadrados, obteve-se  $\hat{\alpha} = 67,111049$  e  $\hat{\delta} = 1,53197$ .

## 4. Distribuições

### 4.1. Distribuição Normal

Se a variável T (tempo de falha) tem distribuição normal com os parâmetros média  $\mu_T$  e desvio padrão  $\sigma_T$ , a sua função densidade de probabilidade é dada por:

$$f(t) = \frac{1}{\sigma_T \sqrt{2\pi}} e^{-\frac{1}{2} \left( \frac{t - \mu_T}{\sigma_T} \right)^2}, \text{ para } \mu_T > 0, t > 0 \text{ e } \sigma_T > 0.$$

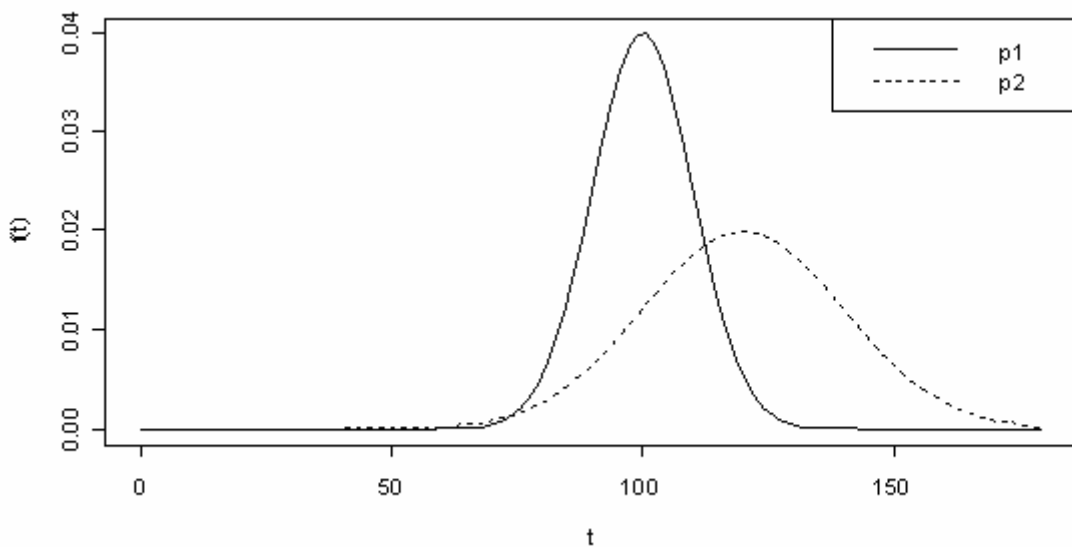
Deste modo, uma representação para este caso é dada por:  $T \sim N(\mu_T, \sigma_T^2)$ . Na figura 2, são apresentadas as formas de duas funções densidades de probabilidades, para os parâmetros de  $\mu_{T1} = 100$  e  $\sigma_{T1} = 10$  referentes ao produto 1 e de  $\mu_{T2} = 120$  e  $\sigma_{T2} = 20$  referentes ao produto 2.

Como pode-se observar na figura 4.2, o produto 2 é melhor que o produto 1 em termos de durabilidade. Para os produtos 1 e 2, existe uma probabilidade de 0,5 para os itens falharem antes dos tempos  $t = 100$  e  $t = 120$ , respectivamente.

No R, o gráfico será construído da seguinte forma:

```
plot(function(t){dnorm(t,mean=100,sd=10)},xlim=c(0,180),ylab="f(t)",xlab="t",lty=1)
plot(function(t){dnorm(t, mean=120,sd=20)},xlim=c(0,180),add=T, lty=2)
legend("topright", c("p1","p2"), lty=c(1,2)) # Legenda
```

Figura 2. Funções densidades de probabilidades da distribuição normal



A função de distribuição acumulada ou de desconfiabilidade da variável  $T$  é dada por:

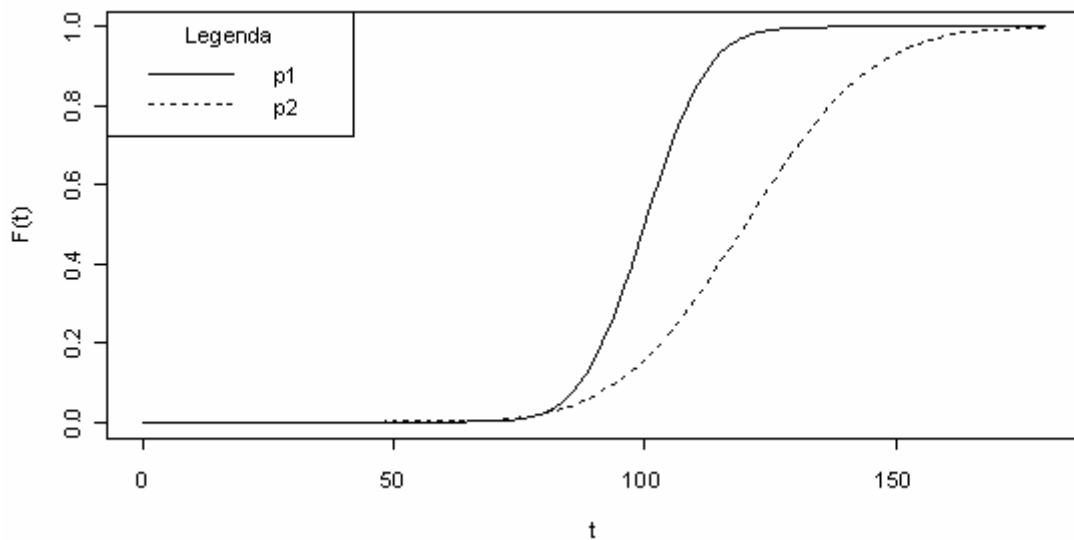
$$F(t) = \Phi\left(\frac{t - \mu_T}{\sigma_T}\right)$$

Na figura 3, são apresentadas as formas das duas funções de desconfiabilidades, referentes aos produtos 1 e 2.

No R, o gráfico será construído da seguinte forma:

```
plot(function(t) {pnorm(t,mean=100,sd=10)}, xlim=c(0,180), ylab="F(t)", xlab="t", lty=1)
plot(function(t){pnorm(t, mean=120,sd=20)},xlim=c(0,180),add=T, lty=2)
legend("topleft", c("p1","p2"), lty=c(1,2), title="Legenda") # Legenda
```

Figura 3. Funções de desconfiabilidades da distribuição normal



Deste modo, a função de confiabilidade da variável T, que é a probabilidade de um item continuar funcionando além do tempo t, é dada por:

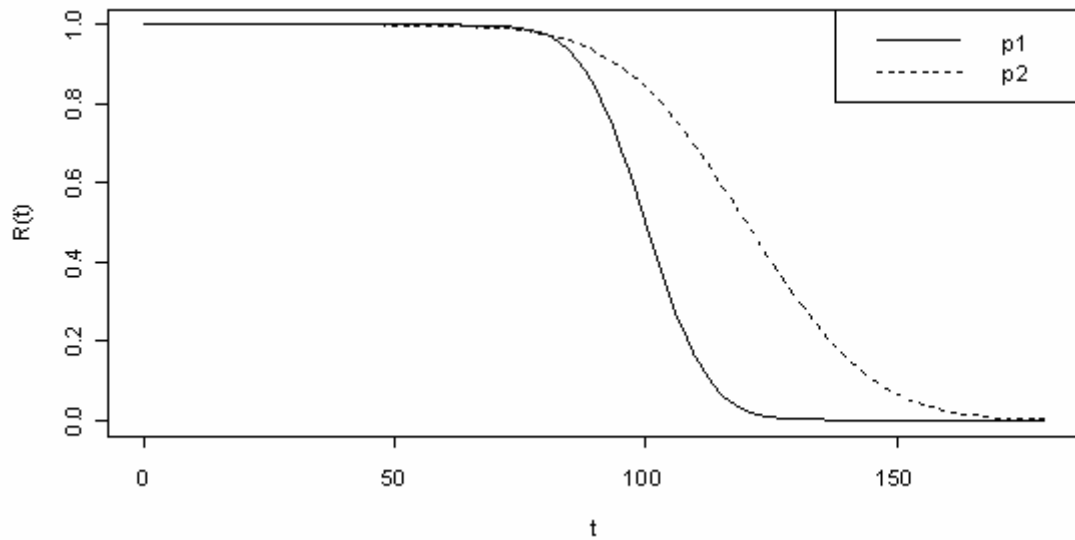
$$R(t) = 1 - \Phi\left(\frac{t - \mu_T}{\sigma_T}\right).$$

Na figura 4, são apresentadas as formas das duas funções de confiabilidades referentes aos produtos 1 e 2. Como pode-se observar, o produto 2 possui uma função de confiabilidade com um decréscimo menos acentuado que a do produto 1, o que caracteriza uma melhor confiabilidade para o primeiro produto.

No R, o gráfico será construído da seguinte forma:

```
plot(function(t){1-pnorm(t, mean=100,sd=10)},xlim=c(0,180),ylab= "R(t)", xlab= "t",
lty=1)
plot(function(t){1-pnorm(t, mean=120,sd=20)},xlim=c(0,180),add=T, lty=2)
legend("topright", c("p1","p2"), lty=c(1,2)) # Legenda
```

Figura 4. Funções de confiabilidades da distribuição normal



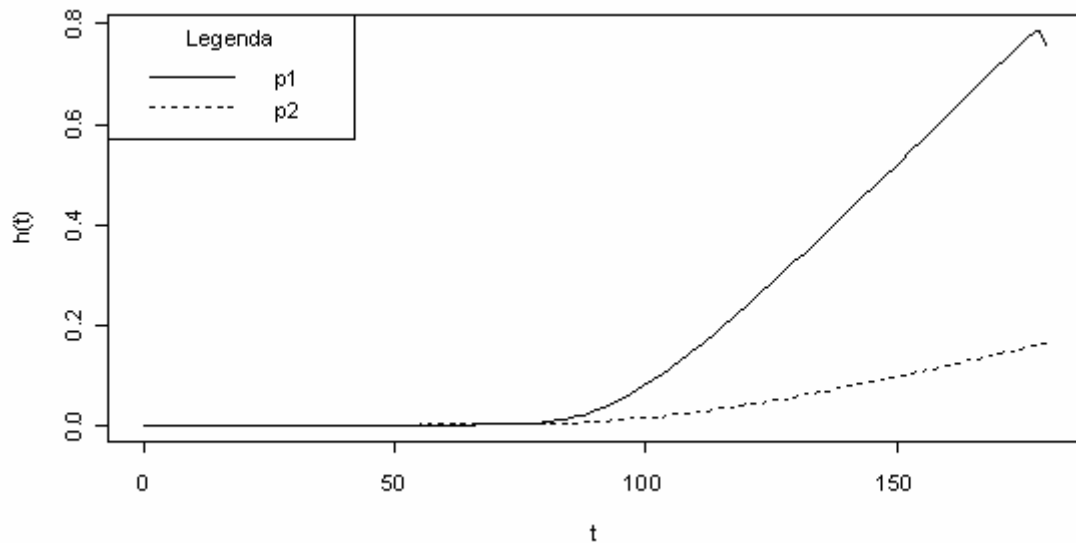
A função de taxa de falha associada à variável T é igual a:  $h(t) = \frac{f(t)}{R(t)}$ .

Na figura 5, são apresentadas as formas das duas funções das taxas de falhas, referentes aos produtos 1 e 2.

No R, o gráfico é construído da seguinte forma:

```
plot(function(t){dnorm(t, mean=100, sd=10) / (1-pnorm(y, mean=100, sd=10) )},
xlim=c(0,180),ylab= "h(t)", xlab= "t", lty=1)
plot(function(t){dnorm(t, mean=120,sd=20) / (1-pnorm(y, mean=120, sd=20) )}, xlim=
c(0,180), add=T, lty=2)
legend("topleft", c("p1","p2"), lty=c(1,2), title="Legenda") # Legenda
```

Figura 5. Funções de taxas de falhas da distribuição normal



#### 4.2. Distribuição Log-normal

Neste caso, a variável T tem distribuição log-normal com os parâmetros  $\mu_T$  e  $\sigma_T$ , sendo que a variável  $Y = \ln(T)$ , com valores positivos de T transformados para a escala logarítmica natural, segue distribuição normal com os parâmetros  $\mu_Y$  e  $\sigma_Y$  da variável Y logaritmicada, quando a função densidade de probabilidade da variável T for dada por:

$$f(t) = \frac{1}{t\sigma_Y\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{y-\mu_Y}{\sigma_Y}\right)^2}, \text{ para } \mu_Y > 0, t > 0 \text{ e } \sigma_Y > 0, \text{ em que:}$$

$y = \ln(t)$  = valor da variável T na escala logarítmica natural;

$\mu_Y$  = média da variável T na escala logarítmica natural;

$\sigma_Y$  = desvio padrão da variável T na escala logarítmica natural.

Deste modo, uma representação para este caso é dada por:  $T \sim \text{LN}(\mu_Y, \sigma_Y^2)$ .

Como exemplo, são apresentadas duas funções densidades de probabilidades, para os parâmetros de  $\mu_{Y1} = 4,6$  e  $\sigma_{Y1} = 0,1$  referentes ao produto 1 e de  $\mu_{Y2} = 4,8$  e  $\sigma_{Y2} = 0,2$  referentes ao produto 2.

Os parâmetros  $\mu_T$  e  $\sigma_T$  associados à população dos dados são iguais a:

$$\mu_T = e^{\left(\mu_Y + \frac{\sigma_Y^2}{2}\right)};$$

$$\sigma_T = \sqrt{e^{(2\mu_Y + \sigma_Y^2)} \times (e^{\sigma_Y^2} - 1)}.$$

Para calculá-los no R, têm-se:

```
exp(4.6+(0.1^2/2)) #  $\mu_{T1}$ 
```

```
[1] 99.98298
```

```
sqrt(exp(2*4.6+0.1^2)*(exp(0.1^2)-1)) #  $\sigma_{T1}$ 
```

```
[1] 10.02335
```

```
exp(4.8+(0.2^2/2)) #  $\mu_{T2}$ 
```

```
[1] 123.9651
```

```
sqrt(exp(2*4.8+0.2^2)*(exp(0.2^2)-1)) #  $\sigma_{T2}$ 
```

```
[1] 25.04303
```

No R, o gráfico será construído da seguinte forma:

```
plot(function(t){dlnorm(t, mean=4.6, sd=0.1)}, xlim=c(0,180), ylab="f(t)", xlab="t", lty=1)
plot(function(t){dlnorm(t, mean=4.8, sd=0.2)}, xlim=c(0,180), add=T, lty=2)
legend("topright", c("p1", "p2"), lty=c(1,2)) # Legenda
```

A função de desconfiabilidade da variável T é dada por:

$$F(t) = \Phi\left(\frac{y - \mu_Y}{\sigma_Y}\right).$$

No R, o gráfico será gerado da seguinte forma:

```
plot(function(t) {plnorm(t, mean=4.6, sd=0.1)}, xlim=c(0,180), ylab="F(t)", xlab="t",
lty=1)
plot(function(t){plnorm(t, mean=4.8, sd=0.2)}, xlim=c(0,180), add=T, lty=2)
legend("topleft", c("p1", "p2"), lty=c(1,2), title="Legenda") # Legenda
```

Deste modo, a função de confiabilidade da variável T, que é a probabilidade de um item continuar funcionando além do tempo t, é dada por:

$$R(t) = 1 - \Phi\left(\frac{y - \mu_Y}{\sigma_Y}\right).$$

No R, o gráfico será construído da seguinte forma:

```
plot(function(t){1-plnorm(t, mean=4.6, sd=0.1)}, xlim=c(0,180), ylab= "R(t)", xlab= "t",
lty=1)
plot(function(t){1-plnorm(t, mean=4.8, sd=0.2)}, xlim=c(0,180), add=T, lty=2)
legend("topright", c("p1","p2"), lty=c(1,2)) # Legenda
```

A função de taxa de falha associada à variável T é igual a:  $h(t) = \frac{f(t)}{R(t)}$ .

No R, o gráfico é construído da seguinte forma:

```
plot(function(t){dlnorm(t, mean=4.6, sd=0.1) / (1-plnorm(t, mean=4.6, sd=0.1) )},
xlim=c(0,180), ylab= "h(t)", xlab= "t", lty=1)
plot(function(t){dlnorm(t, mean=4.8, sd=0.2) / (1-plnorm(t, mean=4.8, sd=0.2) )}, xlim=
c(0,180), add=T, lty=2)
legend("topleft", c("p1","p2"), lty=c(1,2)) # Legenda
```

#### 4.3. Distribuição Exponencial

A variável T poderá ter distribuição exponencial com parâmetro  $\alpha$ , se sua função densidade de probabilidade for dada por:

$$f(t) = \frac{1}{\alpha} e^{-\frac{t}{\alpha}} = \lambda e^{-t\lambda}, \text{ para } t \geq 0.$$

Deste modo, uma representação para este caso é dada por:  $T \sim E(\alpha)$ .

Como exemplo, considere as duas funções densidades de probabilidades, para os parâmetros de  $\alpha_1 = 100$  referente ao produto 1 e de  $\alpha_2 = 120$  referente ao produto 2.

Os parâmetros  $\mu_T$  e  $\sigma_T$  associados à população dos dados são iguais a:

$$\mu_T = \sigma_T = \alpha = \frac{1}{\lambda}.$$

Logo:  $\lambda_{T1} = 1/100$  e  $\lambda_{T2} = 1/120$

No R, o gráfico é construído da seguinte forma:

```
plot(function(t){dexp(t, rate=1/100)}, xlim=c(0,700), ylab="f(t)", xlab="t", lty=1)
plot(function(t){dexp(t, rate=1/120)},xlim=c(0,700),add=T, lty=2)
legend("topright", c("p1","p2"), lty=c(1,2)) # Legenda
```

A função de desconfiabilidade da variável T é dada por:

$$F(t) = 1 - e^{-\frac{t}{\alpha}}.$$

No R, o gráfico será gerado da seguinte forma:

```
plot(function(t) {pexp(t,rate=1/100)}, xlim=c(0,700), ylab="F(t)", xlab="t", lty=1)
plot(function(t){pexp(t, rate=1/120)},xlim=c(0,700),add=T, lty=2)
legend("bottomright", c("p1","p2"), lty=c(1,2), title="Legenda") # Legenda
```

Deste modo, a função de confiabilidade da variável T, que é a probabilidade de um item continuar funcionando além do tempo t, é dada por:

$$R(t) = e^{-\frac{t}{\alpha}}.$$

No R, o gráfico será construído da seguinte forma:

```
plot(function(t){1-pexp(t, rate=1/100)},xlim=c(0,700),ylab= "R(t)", xlab= "t", lty=1)
plot(function(t){1-pexp(t, rate=1/120)},xlim=c(0,700),add=T, lty=2)
legend("topright", c("p1","p2"), lty=c(1,2)) # Legenda
```

A função de taxa de falha associada à variável T é constante e igual a:

$$h(t) = \frac{1}{\alpha} = \lambda.$$

No R, o gráfico é construído da seguinte forma:

```
plot(y = 1, x=1, ylim=c(0.004, 0.014), xlim=c(0, 700), ylab= "h(t)", xlab= "t", type= "n") #
Gerar o gráfico em branco para incluir os limites
abline(a=c(1/100,1/100), b=c(0,700), lty=1) # Gerar h(t)1
abline(a=c(1/120,1/120), b=c(0,700), lty=2) # Gerar h(t)2
legend("topleft", c("p1","p2"), lty=c(1,2), title="Legenda") # Legenda
```

#### 4.4. Distribuição de Weibull

A distribuição de Weibull é a mais utilizada para modelar o tempo de falha de um item. Os parâmetros dessa distribuição possibilitam uma grande flexibilidade para modelar sistemas em que o número de falhas aumenta ou diminui ou permanece constante em função do tempo. Assim, a distribuição de Weibull pode ser usada para representar um grande número de distribuições, incluindo as transformações ou as aproximações para as distribuições normal, log-normal e exponencial.

A função densidade de probabilidade é dada por:

$$f(t) = \frac{\delta}{\alpha^\delta} t^{\delta-1} e^{-\left(\frac{t}{\alpha}\right)^\delta}, \text{ para } t \geq 0.$$

Deste modo, uma representação para este caso é dada por:  $T \sim W(\alpha, \delta)$ . O parâmetro de forma  $\delta$  e o de escala  $\alpha$  são positivos. O parâmetro  $\alpha$  tem a mesma unidade de  $t$  e o  $\delta$  não tem unidade. No caso especial em que  $\delta=1$ , obtém-se a distribuição exponencial.

Como exemplo, considere as duas funções densidades de probabilidades, para os parâmetros de  $\alpha_1 = 100$  e  $\delta_1 = 10$  referentes ao produto 1 e de  $\alpha_2 = 120$  e  $\delta_2 = 20$  referentes ao produto 2.

Os parâmetros  $\mu_T$  e  $\sigma_T$  associados à população dos dados são iguais a:

$$\mu_T = \alpha \Gamma\left(1 + \frac{1}{\delta}\right);$$
$$\sigma_T = \alpha \sqrt{\Gamma\left(1 + \frac{2}{\delta}\right) - \Gamma^2\left(1 + \frac{1}{\delta}\right)}.$$

Para calculá-los no R, têm-se:

$$\mu_{T1} = 95,1351 \text{ e } \sigma_{T1} = 11,4457;$$

$$\mu_{T2} = 116,8205 \text{ e } \sigma_{T2} = 7,2401.$$

No R, o gráfico é construído da seguinte forma:

```
plot(function(t){dweibull(t, shape=10, scale=100)}, xlim=c(0,140), ylim= c(0, 0.07),  
ylab="f(t)", xlab="t", lty=1)  
plot(function(t){dweibull(t, shape=20, scale=120)}, xlim=c(0,140), add=T, lty=2)  
legend("topleft", c("p1", "p2"), lty=c(1,2)) # Legenda
```

A função de desconfiabilidade da variável T é dada por:

$$F(t) = 1 - e^{-\left(\frac{t}{\alpha}\right)^\delta}.$$

No R, o gráfico será construído da seguinte forma:

```
plot(function(t) {pweibull(t, shape=10, scale=100)}, xlim=c(0,140), ylab="F(t)", xlab="t",
lty=1)
plot(function(t){pweibull(t, shape=20, scale=120)},xlim=c(0,140),add=T, lty=2)
legend("topleft", c("p1","p2"), lty=c(1,2), title="Legenda") # Legenda
```

Deste modo, a função de confiabilidade da variável T, que é a probabilidade de um item continuar funcionando além do tempo t, é dada por:

$$R(t) = e^{-\left(\frac{t}{\alpha}\right)^\delta}.$$

No R, o gráfico será construído da seguinte forma:

```
plot(function(t){1-pweibull(t, shape=10, scale=100)},xlim=c(0,140),ylab= "R(t)", xlab=
"t", lty=1)
plot(function(t){1-pweibull(t, shape=20, scale=120)},xlim=c(0,140),add=T, lty=2)
legend("bottomleft", c("p1","p2"), lty=c(1,2)) # Legenda
```

A função de taxa de falha associada à variável T é igual a:

$$h(t) = \left(\frac{\delta}{\alpha}\right)\left(\frac{t}{\alpha}\right)^{\delta-1}.$$

No R, o gráfico é construído da seguinte forma:

```
plot(function(t){dweibull(t, shape=10, scale=100) / (1-pweibull(y, shape=10, scale=100)
)}, xlim=c(0,140),ylab= "h(t)", ylim=c(0,1), xlab= "t", lty=1)
plot(function(t){dweibull(t, shape=20, scale=120) / (1-pweibull(y, shape=20, scale=120)
)}, xlim= c(0,140), add=T, lty=2)
legend("topleft", c("p1","p2"), lty=c(1,2), title="Legenda") # Legenda
```