Aula 2 - Distribuição Normal

Prof. André Luiz Cunha

21/05/2021

1 Transformação de escala

Um passo importante de qualquer análise de dados é a uniformização do intervalo de dados, de modo que todas as variáveis do banco de dados tenham o mesmo intervalo de variação.

Seja o exemplo do conjunto de dados aleatório abaixo, são apresentados dois tipos de transformação encontrados na literatura.

```
# Conjunto de valores aleatórios com média 50 e desvio 15.
(x <- rnorm(100, 50, 15))
```

```
##   [1] 48.96150 42.08793 14.87746 61.29713 45.21530 40.53326 43.39847 41.19372
##   [9] 34.68871 36.56362 60.57486 52.22316 49.40785 48.20739 22.16459 53.82503
##  [17] 63.87418 81.42490 65.94090 72.00511 35.72219 61.99073 50.27387 67.38222
##  [25] 57.20551 36.82166 47.75704 58.63766 19.64442 23.14516 51.51743 51.06058
##  [33] 51.58182 53.21274 54.62572 67.86043 63.32040 80.04896 50.16730 64.68180
##  [41] 41.55601 93.28334 48.23941 61.86263 61.27955 56.61706 58.50996 47.20143
##  [49] 77.17752 46.90741 40.13411 73.86520 43.88775 52.12285 51.91100 44.50457
##  [57] 42.20427 72.72976 83.29199 41.66018 57.31564 48.66457 49.28456 48.59427
##  [65] 50.09632 87.14619 84.28030 69.34535 39.50994 76.02277 53.65351 67.54090
##  [73] 65.45243 61.16378 22.87605 34.52954 40.61786 37.10351 55.50717 63.27704
##  [81] 54.55009 53.85726 47.69825 40.68990 48.92756 59.52475 87.71889 28.67098
##  [89] 41.69141 55.49836 67.89542 41.39418 34.26151 38.31990 54.25358 74.73494
##  [97] 77.58358 51.93507 56.99045 23.87862
```

```
summary(x)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.88   41.99   51.92   53.16   62.31   93.28
```

1.1 Normalização [0,1]

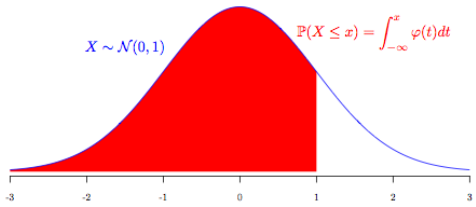
A normalização transforma os dados no intervalo entre 0 e 1.

```
x_norm <- (x - min(x)) / (max(x) - min(x))
summary(x_norm)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.3458  0.4725  0.4883  0.6050  1.0000
```

1.2 Padronizar [-3,3]

A padronização dos dados nada mais é do que a transformação para a escala da curva normal padrão (z-padrão). A Figura 1 ilustra um exemplo de tabela com os valores z-padrão.



	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359
0.1	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753
0.2	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141
0.3	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517
0.4	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879
0.5	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224
0.6	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549
0.7	0.7580	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852
0.8	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133
0.9	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389
1.0	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621
1.1	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830
1.2	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015
1.3	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177
1.4	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9279	0.9292	0.9306	0.9319
1.5	0.9332	0.9345	0.9357	0.9370	0.9382	0.9394	0.9406	0.9418	0.9429	0.9441
1.6	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545
1.7	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633
1.8	0.9641	0.9649	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9699	0.9706
1.9	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.9750	0.9756	0.9761	0.9767
2.0	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817
2.1	0.9821	0.9826	0.9830	0.9834	0.9838	0.9842	0.9846	0.9850	0.9854	0.9857
2.2	0.9861	0.9864	0.9868	0.9871	0.9875	0.9878	0.9881	0.9884	0.9887	0.9890
2.3	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916
2.4	0.9918	0.9920	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936
2.5	0.9938	0.9940	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952
2.6	0.9953	0.9955	0.9956	0.9957	0.9959	0.9960	0.9961	0.9962	0.9963	0.9964
2.7	0.9965	0.9966	0.9967	0.9968	0.9969	0.9970	0.9971	0.9972	0.9973	0.9974
2.8	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.9980	0.9981
2.9	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986
3.0	0.9987	0.9987	0.9987	0.9988	0.9988	0.9989	0.9989	0.9989	0.9990	0.9990

Figura 1: Tabela z-padrão

```
x_pad <- (x - mean(x)) / sd(x)
summary(x_pad)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -2.42208 -0.70684 -0.07834  0.00000  0.57895  2.53838
```

```
## OLHANDO A TABELA
#z = 1,0 ----> p(z) = 0,1587
1 - 2 * 0.1587
```

```
## [1] 0.6826
```

```
#z = 2,0 ----> p(z) = 0,0228
1 - 2 * 0.0228
```

```
## [1] 0.9544
```

```
# p(z) = 95% -----> z(p = 0,025) = ?
1.96
```

```
## [1] 1.96
```

```
# p(z) = 99% -----> z =??
2.575
```

```
## [1] 2.575
```

2 Funções do R

2.1 Números aleatórios

2.1.1 Uniformemente distribuídos

Função: `runif(n, min, max)`

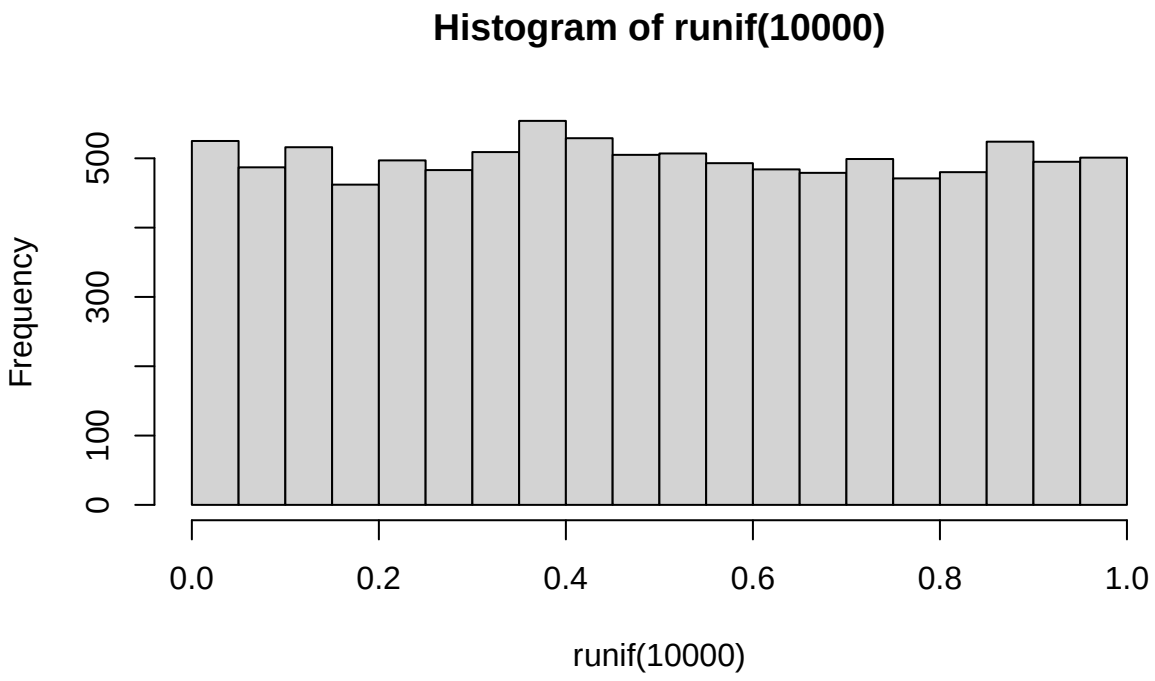
```
runif(10)
```

```
##  [1] 0.5035837 0.5588150 0.9896398 0.7691878 0.2467629 0.6823045 0.1500374
##  [8] 0.7417236 0.2696437 0.6821701
```

```
runif(10, 100, 150)
```

```
##  [1] 100.9030 137.7407 122.0844 131.4070 131.9114 108.5084 127.6959 122.3781
##  [9] 103.9019 121.2205
```

```r
hist(runif(10000))
```



2.1.2 Normalmente distribuídos

Função: `rnorm(n, mean, sd)`

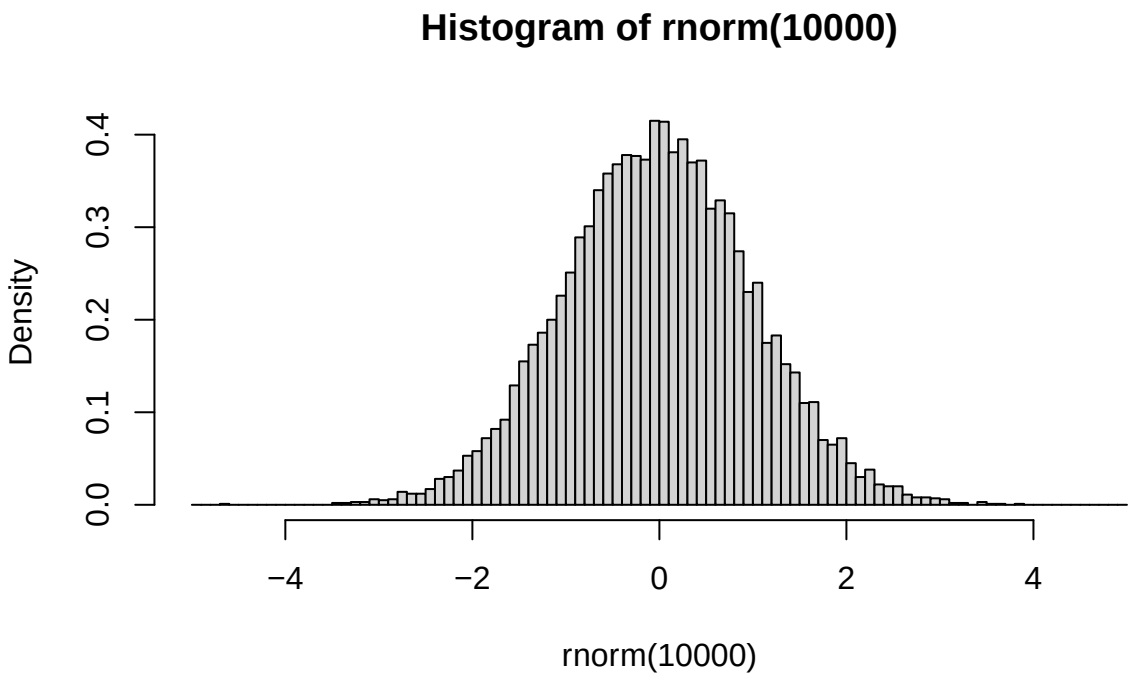
```r
rnorm(10)
```

```
##  [1] -1.0612682  1.4190564 -0.2047450  0.7975572 -0.9381686  1.2214736
##  [7]  1.2555026  0.9488377 -0.1397677 -0.7570233
```

```r
rnorm(10, 100, 15)
```

```
##  [1] 118.69242 102.50845  93.50971  98.95480  95.62000 122.95129 111.17021
##  [8] 102.29481 142.98150 113.36089
```

```r
hist(rnorm(10000), breaks = seq(-5,5,.1),
     freq = FALSE)
```



2.2 Distribuição Normal

Encontrando o valor z-padrão com a função `qnorm(area da curva, mean=0, sd=1)`.

- Unicaudal a esquerda: $z_\alpha = qnorm(\alpha)$;
- Unicaudal a direita: $z_\alpha = qnorm(1-\alpha)$;
- Bicaudal: $z_{\frac{\alpha}{2}} = qnorm(1-\frac{\alpha}{2})$.

```
qnorm(.90)
```

```
## [1] 1.281552
```

```
qnorm(.5)
```

```
## [1] 0
```

Encontrando o o p-valor com a função `pnorm(valor z, mean=0, sd=1)`.

- Unicaudal a esquerda: $p-value = pnorm(z, lower.tail = TRUE)$;
- Unicaudal a direita: $p-value = pnorm(z, lower.tail = FALSE)$;
- Bicaudal: $p-value = 2 \cdot pnorm(abs(z), lower.tail = FALSE)$.

```r
pnorm(1.96)
```

```
## [1] 0.9750021
```

```r
pnorm(1.96, lower.tail = FALSE)
```

```
## [1] 0.0249979
```

```r
pnorm(0)
```

```
## [1] 0.5
```

Encontrando a densidade do valor com a função `dnorm(valor z, mean=0, sd=1)`

```r
dnorm(1.96)
```

```
## [1] 0.05844094
```

```r
dnorm(-1.96)
```

```
## [1] 0.05844094
```

```r
dnorm(0)
```

```
## [1] 0.3989423
```

EXEMPLO 1

```r
## P(z > 1,65)
pnorm(1.65, lower.tail = FALSE)
```

```
## [1] 0.04947147
```

```r
## P(z < 1,65)
pnorm(1.65)
```

```
## [1] 0.9505285
```

```r
## P(1,40 < z < 1,70)
pnorm(1.7) - pnorm(1.4)
```

```
## [1] 0.0361912
```

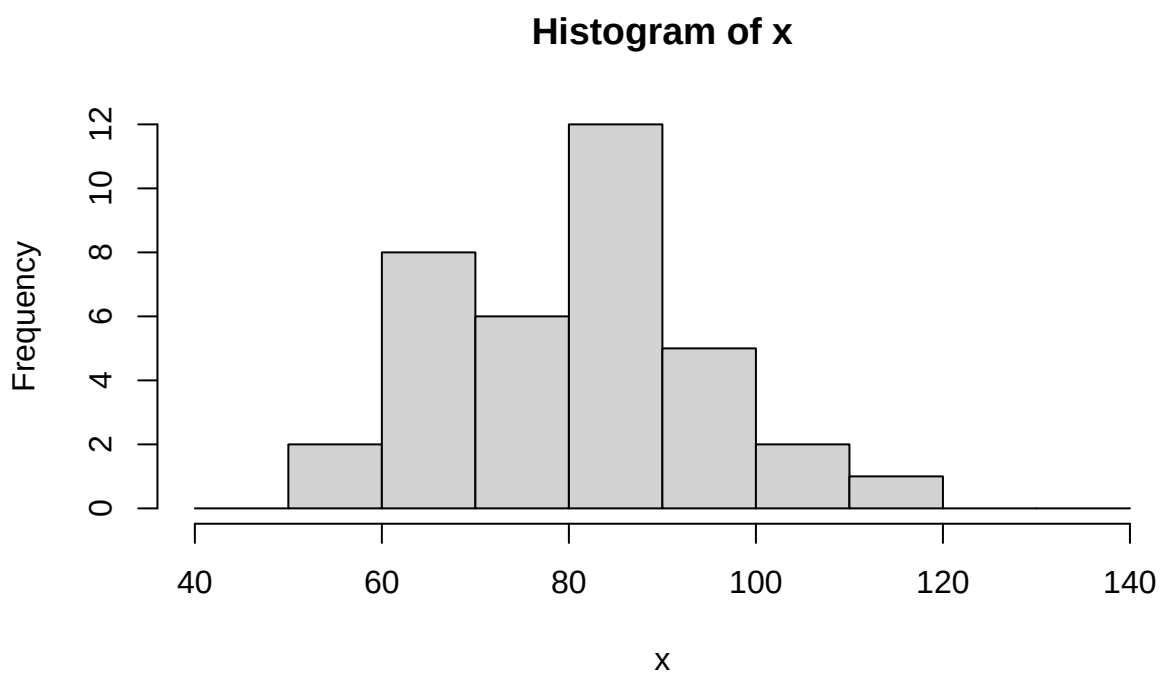
EXEMPLO 2

```r
x = c(58,78,84,90,97,70,
      90,86,82,59,90,70,
      74,83,90,75,88,84,
      68,96,70,94,70,110,
      67,68,75,80,68,82,
      104,92,112,84,98,80)


## Análise descritiva
summary(x)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   58.00   70.00   82.50   82.39   90.00  112.00
```

```r
hdados <- hist(x,
               breaks = seq(40,140,10))
```




```r
hdados$breaks
```

```
##  [1]  40  50  60  70  80  90 100 110 120 130 140
```

```r
hdados$counts
```

```
##  [1]  0  2  8  6 12  5  2  1  0  0
```

```
hdados$density
```

```
##  [1] 0.000000000 0.005555556 0.022222222 0.016666667 0.033333333 0.013888889
##  [7] 0.005555556 0.002777778 0.000000000 0.000000000
```

O parâmetro `density` traz a razão entre a porcentagem de elementos e o intervalo de bins, tanto que a soma das porcentagens `density` é igual a 10%. Ao multiplicar cada densidade pelo intervalo do bin, a porcentagem total será de 100%.

```
sum(hdados$density)
```

```
## [1] 0.1
```

```
sum(hdados$density) * 10
```

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

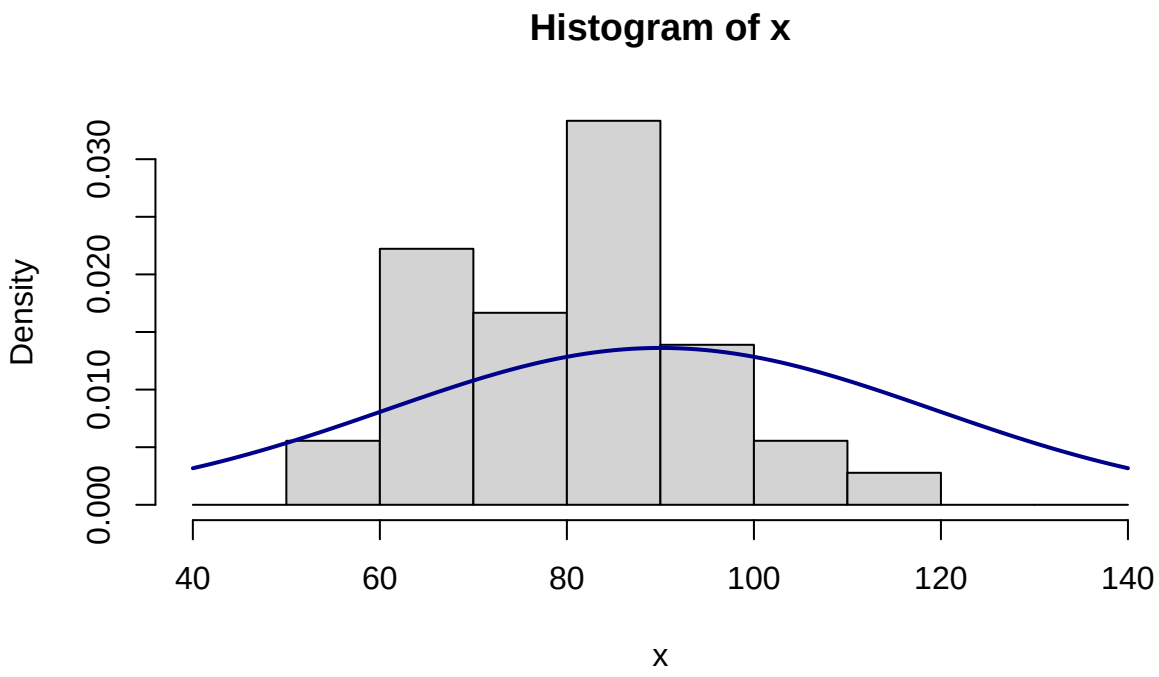
3 Testes do R

3.1 Qui-quadrado (chisq)

Teste de aderência de Qui-quadrado é usado para comparar distribuições observadas com distribuições esperadas em dados discretos (histograma de frequências).

```
#x_pad <- (x - mean(x))/sd(x)

## Densidade dos valores X com a curva normal teórica
bin <- 10
hist.real <- hist(x, breaks = seq(40,140,bin), freq=FALSE)
curve(dnorm(x, mean(x), sd(x)), col='darkblue', lw=2, add=TRUE)
```



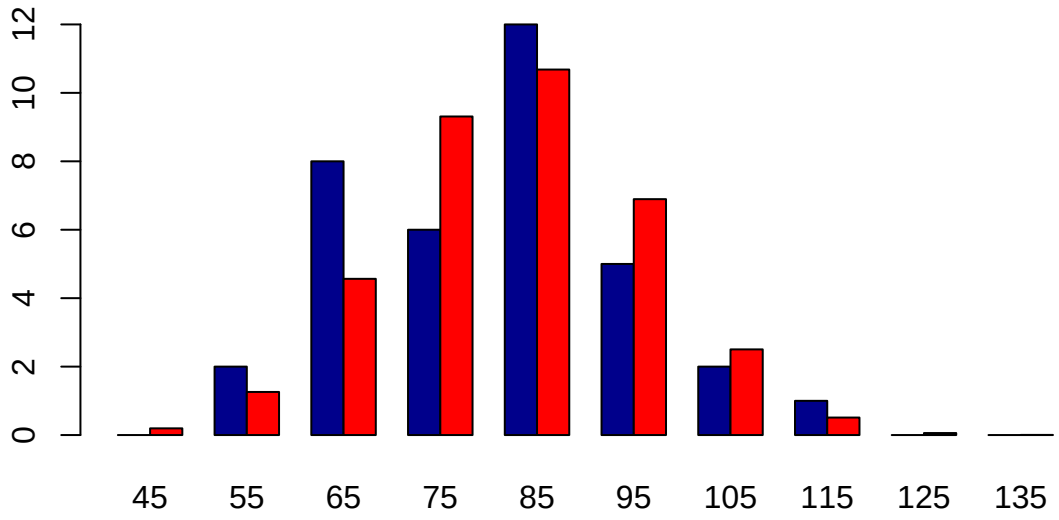

```
hist.real$density
```

```
##  [1] 0.000000000 0.005555556 0.022222222 0.016666667 0.033333333 0.013888889
##  [7] 0.005555556 0.002777778 0.000000000 0.000000000
```

```
hist.real$counts
```

```
##  [1]  0  2  8  6 12  5  2  1  0  0
```

```
## Frequências das distribuições observadas e esperadas
barplot(rbind(hist.real$counts, dnorm(hist.real$mids, mean(x), sd(x))*bin*length(x) ),
        names.arg = hist.real$mids,
        col = c("darkblue", "red"),
        beside = TRUE
)
```




```
(res <- chisq.test(hist.real$counts, # Frequência observada - dados originais
           p = dnorm(hist.real$mids, mean(x), sd(x))*bin, # Frequência teórica - distribuição normal
           rescale.p = TRUE ))
```

```
## Warning in chisq.test(hist.real$counts, p = dnorm(hist.real$mids, mean(x), :
## Chi-squared approximation may be incorrect
```

```
##
##  Chi-squared test for given probabilities
##
## data:  hist.real$counts
## X-squared = 5.7035, df = 9, p-value = 0.7692
```

Como visto no código, o resultado do *Chi-squared test for given probabilities* apresentou o p-valor de 0.7691877 o que **REJEITA a hipótese nula** de normalidade dos dados (diferença entre as distribuições das amostras observadas e esperadas).

3.2 Kolmogorv-Smirnov (KS)

```
(res <- ks.test(x, "pnorm", mean(x), sd(x)))
```

```
## Warning in ks.test(x, "pnorm", mean(x), sd(x)): ties should not be present for
## the Kolmogorov-Smirnov test
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  x
## D = 0.10407, p-value = 0.8304
## alternative hypothesis: two-sided
```

```
ks.test(x_pad, "pnorm")
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  x_pad
## D = 0.06309, p-value = 0.8209
## alternative hypothesis: two-sided
```

O resultado do *One-sample Kolmogorov-Smirnov test* apresentou o p-valor de 0.8303815 o que **REJEITA a hipótese nula** de normalidade dos dados.

3.3 Shapiro-Wilk

```
(res <- shapiro.test(x))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  x
## W = 0.97612, p-value = 0.6139
```

O resultado do *Shapiro-Wilk normality test* apresentou o p-valor de 0.6138666 o que **REJEITA a hipótese nula** de normalidade dos dados.