Estatistica Descritiva na Avaliacão de Tempo de Tarefas

Hmisc - Pastecs - Psych - Assimetría - Curtose - Histograma - Boxplot - Densidade - Distribuição acumulada empírica - Correlação

Aula 2P1 Analise Descritiva usando o Projeto 1

1 Descrição dos dados

Estamos interessados em testar o tempo computacional de uma função no python quando se executa três tarefas diferentes de analise de dados. Dada uma base de dados, as tarefas são:

- Tarefa A: Executar uma analise descritiva incluindo elaboração de gráficos;
- Tarefa B: Estimar um modelo de regressão;
- Tarefa C: Estimar um modelo de classificação binária.

Para avaliar o desempenho foi desenvolvido um experimento onde foram testadas 50 base de dados de diferente dimensões (diferentes tamanhos de amostras) com cada tarefa. Assim, podemos assumir independência entre as tarefas executadas. As variáveis registradas são:

- $tarefa$: Número de réplicas do experimento;
- $versao$: recebe valor 1 com a versão anterior e 2 com a nova versão disponível;
- t_A: tempo computacional em micro segundos com a tarefa A;
- t_B: tempo computacional em micro segundos com a tarefa B;
- t_C: tempo computacional em micro segundos com a tarefa C.

2 Leitura dos dados

```
# Para fixar o diretório de trabalho como diretório fonte:
setwd(dirname(rstudioapi::getActiveDocumentContext()$path))

# Leitura de dados em arquivo externo
projeto1 = read.csv(file = 'projeto1.csv')

# Outra opção para entrar com os dados:
#projeto1 = read.csv(file.choose(), header=TRUE)

head(projeto1)
```

```
##   tarefa versao       tA       tB      tC
## 1      1      1 401.4400 186.3505 145.330
## 2      2      1 437.7825 195.8830 149.837
## 3      3      1 418.1495 186.3685 143.043
## 4      4      1 414.6125 186.0220 142.776
## 5      5      1 470.5465 186.2660 141.575
## 6      6      1 472.7260 188.0160 134.419
```

```
dim(projeto1)
```

```
## [1] 50  5
```

O arquivo projeto1.csv é uma base de dados com 5 variáveis e 50 observações correspondentes as réplicas do experimento.

3 Estatísticas descritivas

A seguir apresentamos diferentes medidas descritivas de um conjunto de dados quantitativos usando diferentes pacotes.

```r
# Resumo básico
summary(projeto1)
```

```
##      tarefa          versao          tA              tB              tC
##  Min.   : 1.00   Min.   :1.0   Min.   :331.6   Min.   :173.7   Min.   :125.3
##  1st Qu.:13.25   1st Qu.:1.0   1st Qu.:337.3   1st Qu.:175.7   1st Qu.:126.8
##  Median :25.50   Median :1.5   Median :406.4   Median :186.2   Median :127.3
##  Mean   :25.50   Mean   :1.5   Mean   :397.9   Mean   :208.3   Mean   :130.2
##  3rd Qu.:37.75   3rd Qu.:2.0   3rd Qu.:441.8   3rd Qu.:233.3   3rd Qu.:133.9
##  Max.   :50.00   Max.   :2.0   Max.   :492.0   Max.   :326.4   Max.   :149.8
```

```r
# Quantitativas
summary(projeto1[, 3:5])
```

```
##        tA              tB              tC
##  Min.   :331.6   Min.   :173.7   Min.   :125.3
##  1st Qu.:337.3   1st Qu.:175.7   1st Qu.:126.8
##  Median :406.4   Median :186.2   Median :127.3
##  Mean   :397.9   Mean   :208.3   Mean   :130.2
##  3rd Qu.:441.8   3rd Qu.:233.3   3rd Qu.:133.9
##  Max.   :492.0   Max.   :326.4   Max.   :149.8
```

```r
# Qualitativas
table(projeto1[, 2])
```

```
##
##  1  2
## 25 25
```

```r
# Cinco números de Tukey.
five = apply(projeto1[, 3:5], 2, fivenum)
rownames(five) = c("Min", "Q1", "Med", "Q3", "Max")
five
```

```
##           tA       tB       tC
## Min 331.6190 173.7100 125.3480
## Q1  337.1540 175.7240 126.7620
## Med 406.4035 186.1770 127.3415
## Q3  441.9630 244.1275 133.9935
## Max 491.9915 326.4390 149.8370
```

```r
# Análise descritiva usando Hmisc.
library(Hmisc) # Pacote deve ser instalado antes
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```r
describe(projeto1)
```

```
## projeto1
##
```

```
##  5  Variables      50  Observations
## --------------------------------------------------------------------------------
## tarefa
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##       50        0       50        1     25.5       17     3.45     5.90
##      .25      .50      .75      .90      .95
##    13.25    25.50    37.75    45.10    47.55
##
## lowest :  1  2  3  4  5, highest: 46 47 48 49 50
## --------------------------------------------------------------------------------
## versao
##        n  missing distinct     Info     Mean      Gmd
##       50        0        2     0.75      1.5   0.5102
##
## Value        1   2
## Frequency   25  25
## Proportion 0.5 0.5
## --------------------------------------------------------------------------------
## tA
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##       50        0       50        1    397.9    58.54    332.6    333.1
##      .25      .50      .75      .90      .95
##    337.3    406.4    441.8    458.6    468.0
##
## lowest : 331.619 332.491 332.587 332.62  333.035
## highest: 462.692 464.9   470.546 472.726 491.992
## --------------------------------------------------------------------------------
## tB
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##       50        0       50        1    208.3    47.23    174.3    174.6
##      .25      .50      .75      .90      .95
##    175.7    186.2    233.3    296.3    309.5
##
## lowest : 173.71  174.053 174.346 174.349 174.47
## highest: 296.3   301.636 316.008 318.473 326.439
## --------------------------------------------------------------------------------
## tC
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##       50        0       50        1    130.2    5.496    125.8    126.3
##      .25      .50      .75      .90      .95
##    126.8    127.3    133.9    136.4    142.9
##
## lowest : 125.348 125.513 125.62  125.987 126.053
## highest: 141.575 142.776 143.043 145.33  149.837
## --------------------------------------------------------------------------------
```

```r
# Análise descritiva usando pastecs.
library(pastecs) # Pacote deve ser instalado antes
stat.desc(projeto1[, 3:5])
```

```
##                        tA           tB           tC
```

```
## nbr.val      5.000000e+01 5.000000e+01   50.0000000
## nbr.null     0.000000e+00 0.000000e+00    0.0000000
## nbr.na       0.000000e+00 0.000000e+00    0.0000000
## min          3.316190e+02 1.737100e+02  125.3480000
## max          4.919915e+02 3.264390e+02  149.8370000
## range        1.603725e+02 1.527290e+02   24.4890000
## sum          1.989590e+04 1.041348e+04 6511.6880000
## median       4.064035e+02 1.861770e+02  127.3415000
## mean         3.979180e+02 2.082695e+02  130.2337600
## SE.mean      7.224892e+00 6.862670e+00    0.8096535
## CI.mean.0.95 1.451896e+01 1.379105e+01    1.6270597
## var          2.609953e+03 2.354812e+03   32.7769406
## std.dev      5.108770e+01 4.852640e+01    5.7251149
## coef.var     1.283875e-01 2.329981e-01    0.0439603
```

```
# Análise descritiva usando psych.
library(psych) # Pacote deve ser instalado antes
```

```
##
## Attaching package: 'psych'
```

```
## The following object is masked from 'package:Hmisc':
##
##     describe
```

```
psych::describe(projeto1[, 3:5])
```

```
##    vars  n   mean    sd median trimmed   mad    min    max  range  skew
## tA    1 50 397.92 51.09 406.40  396.77 67.78 331.62 491.99 160.37 -0.03
## tB    2 50 208.27 48.53 186.18  199.59 16.59 173.71 326.44 152.73  1.27
## tC    3 50 130.23  5.73 127.34  129.02  1.40 125.35 149.84  24.49  1.69
##    kurtosis   se
## tA    -1.53 7.22
## tB    -0.02 6.86
## tC     2.14 0.81
```

As diferentes medidas estatísticas foram apresentados nos slides e são amplamente conhecidas. Sugerimos usar a função `describe` do pacote `psych`.

3.1 Assimetria e Curtoses

Faremos um maior detalhe destas medidas que não foram suficientemente explicadas nos slides.

```
# Assimetría para as variáveis de tempo das tarefas
tempos = projeto1[, 3:5]
library(e1071)
```

```
##
## Attaching package: 'e1071'
```

```
## The following object is masked from 'package:Hmisc':
##
##     impute
```

```
apply(tempos, 2, skewness)
```

```
##          tA          tB          tC
## -0.03417906  1.26653675  1.69335833
```

```
# Análise de uma variável (tempo da tarefa A)
x = projeto1$tA
skewness(x,type=1)
```

```
## [1] -0.03523068
```

```
skewness(x,type=2)
```

```
## [1] -0.03632978
```

```
skewness(x,type=3)
```

```
## [1] -0.03417906
```

A função `skewness` do pacote `e1071` mede, intuitivamente, a assimetria. Como regra, a assimetria negativa indica que a média dos valores dos dados é menor que a mediana e a distribuição dos dados é inclinada para a esquerda. Por outro lado, a assimetria positiva indica que a média dos valores dos dados é maior do que a mediana e que a distribuição dos dados está inclinada para a direita.

Na função correspondente há 3 métodos (ou tipos) para calcular a assimetria. Eles são descritos em https://www.rdocumentation.org/packages/e1071/versions/1.7-12/topics/skewness:

- Tipo 1: $g_1 = m_3/m_2^{3/2}$ onde $m_r = \sum_i (x_i - \mu)^r / n$
- Tipo 2: $G_1 = g_1\sqrt{n(n-1)}/(n-2)$
- Tipo 3: $b_1 = m_3/s^3 = g_1((n-1)/n)^{3/2}$

O tipo 3 é o default no R. O tipo 1 é usado nos antigos livros de texto e o tipo 2 é usado em SAS o SPSS.

```
# Curtose para as variáveis de tempo das tarefas
tempos = projeto1[, 3:5]
library(e1071)
apply(tempos, 2, kurtosis)
```

```
##          tA          tB          tC
## -1.52558777 -0.02048074  2.14198484
```

```
# Análise de uma variável (tempo da tarefa A)
x = projeto1$tA
kurtosis(x,type=1)
```

```
## [1] -1.464794
```

```
kurtosis(x,type=2)
```

```
## [1] -1.492251
```

```
kurtosis(x,type=3)
```

```
## [1] -1.525588
```

A função `kurtosis` do pacote `e1071` mede, intuitivamente, o excesso de curtose em relação a distribuição normal que tem curtose de excesso zero e, portanto, o formato padrão da cauda ou caso mesocúrtico. O excesso de curtose negativo indicaria uma distribuição de dados de cauda fina e é considerado platicúrtico. O excesso de curtose positivo indicaria uma distribuição de cauda gorda e é considerado leptocúrtico. Na função correspondente há 3 métodos (ou tipos) para calcular a curtose. Eles são descritos em https://www.rdocumentation.org/packages/e1071/versions/1.7-12/topics/kurtosis:

- Tipo 1: $g_2 = m_4/m_2^2 - 3$ onde $m_r = \sum_i (x_i - \mu)^r/n$
- Tipo 2: $G_2 = ((n+1)g_2 + 6) \times (n-1)/((n-2)(n-3))$
- Tipo 3: $b_2 = m_4/s^4 - 3 = (g_2 + 3)(1 - 1/n)^2 - 3$

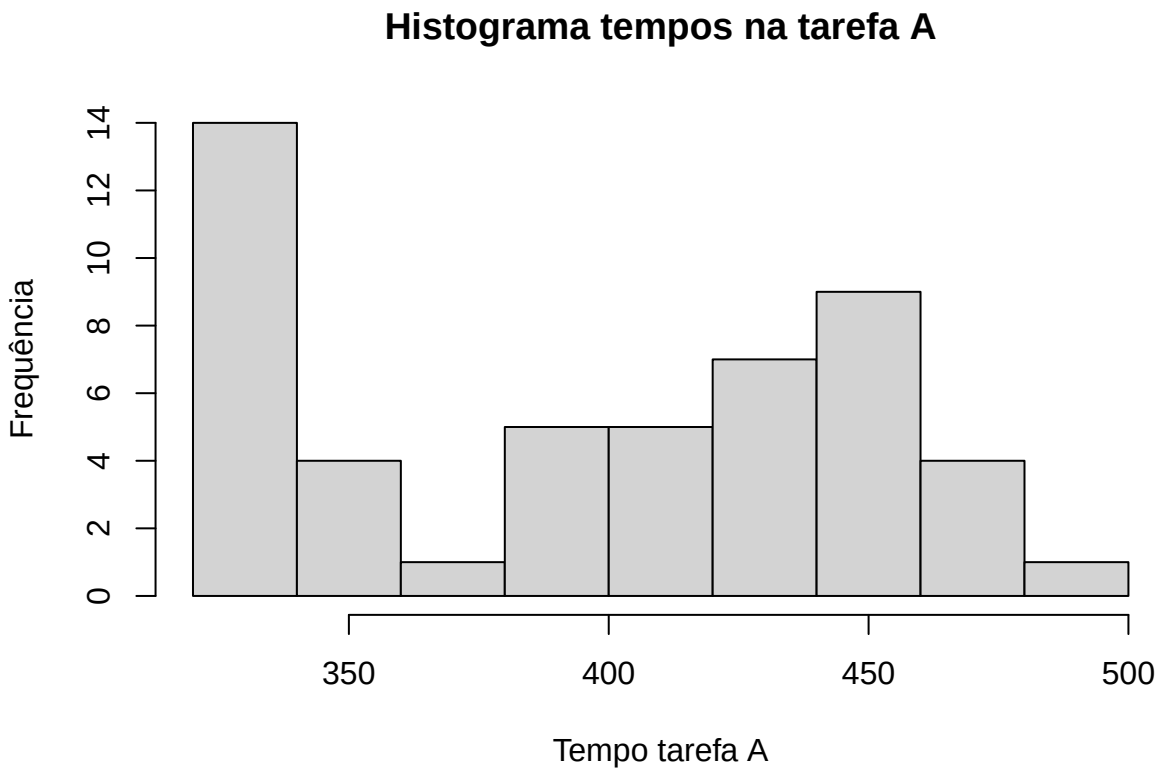
O tipo 3 é o default no R. O tipo 1 é usado nos antigos livros de texto e o tipo 2 é usado em SAS o SPSS.

Nos dados, encontramos maior assimetria a direta (positiva) nos tempos usando as tarefas B e C e assimetria a esquerda (negativa) usando a tarefa A. Note que no caso das tarefas B e C (média $>$ mediana) e no caso da tarefa A (média $<$ mediana). Também encontramos maior curtose (forma leptocúrtica) usanda tarefa C, e menor curtose usanda tarefa A (platicúrtica) e curtose em torno de zero para o método B (mesocúrtica).

3.2 Graficos

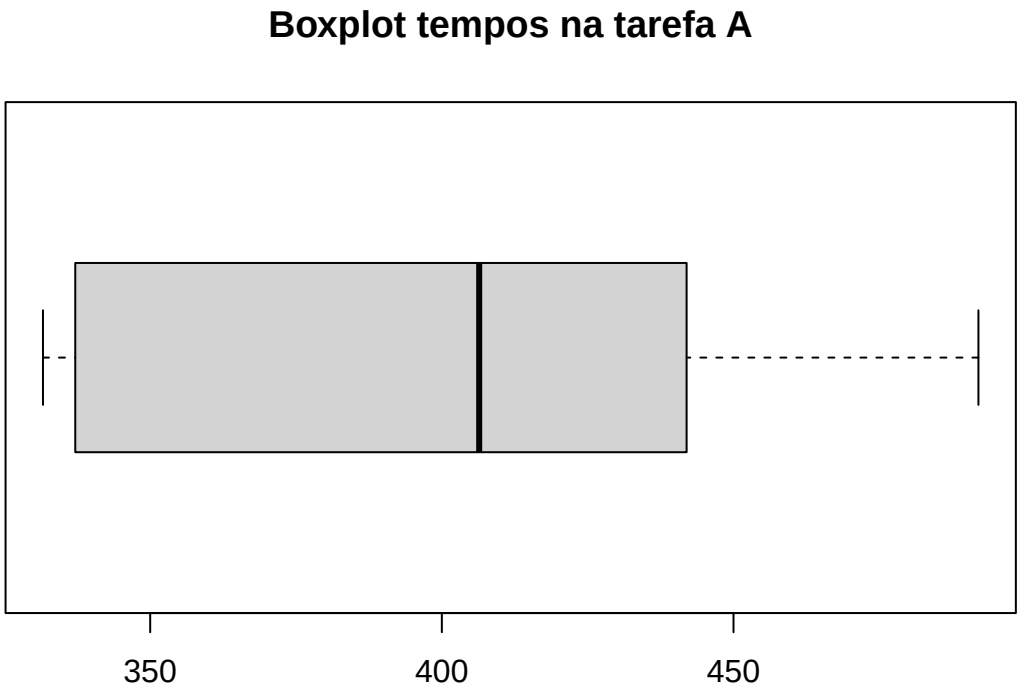
A modo de ilustração, apresentamos os seguintes gráficos para a variável tempo na tarefa A.

```
# Histograma básico
hist(x, main = "Histograma tempos na tarefa A",ylab="Frequência",
     xlab="Tempo tarefa A")
```



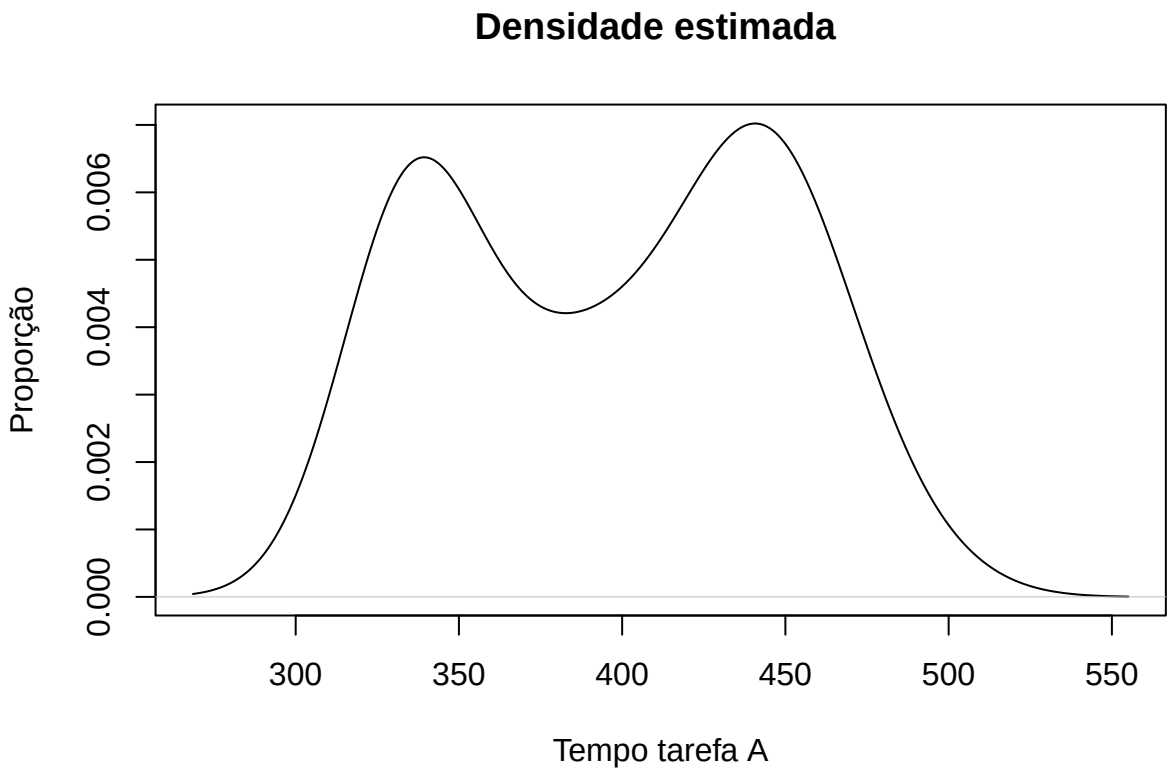
Usando o histograma percebemos que a forma dos dados apresenta bimodalidade.

```r
# Boxplot
boxplot(x, horizontal = TRUE, main = "Boxplot tempos na tarefa A")
```



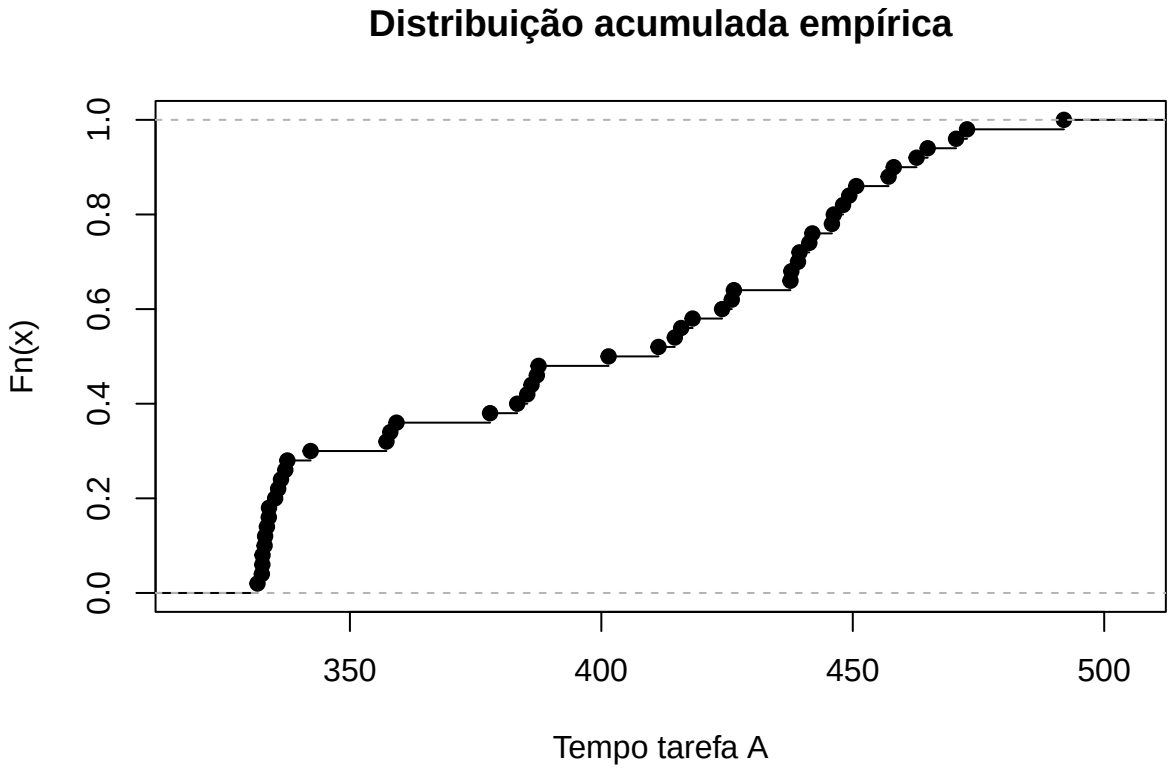
A figura do Box plot confirma a assimetría a esquerda da tarefa A e a forma platicúrtica dos dados.

```r
# Densidade
plot(density(x), main = "Densidade estimada ", ylab="Proporção",
     xlab="Tempo tarefa A")
```



A densidade mostrada acima confirma a bimodalidade dos dados de tempo usando a tarefa A.

```
# Função de distribuição acumulada empírica
plot(ecdf(x), main = "Distribuição acumulada empírica", xlab="Tempo tarefa A")
```



Também percebemos, que não há assimetria na função de distribuição acumulada empírica.

4 Análise descritiva bidimensional

A seguir apresentamos uma análise correlacional entre todas as variáveis de tempo de execução de tarefas

```
#Analise correlacional
cor(projeto1[, 3:5])
```

```
##            tA         tB         tC
## tA  1.0000000 -0.7180469  0.3934356
## tB -0.7180469  1.0000000 -0.2392414
## tC  0.3934356 -0.2392414  1.0000000
```

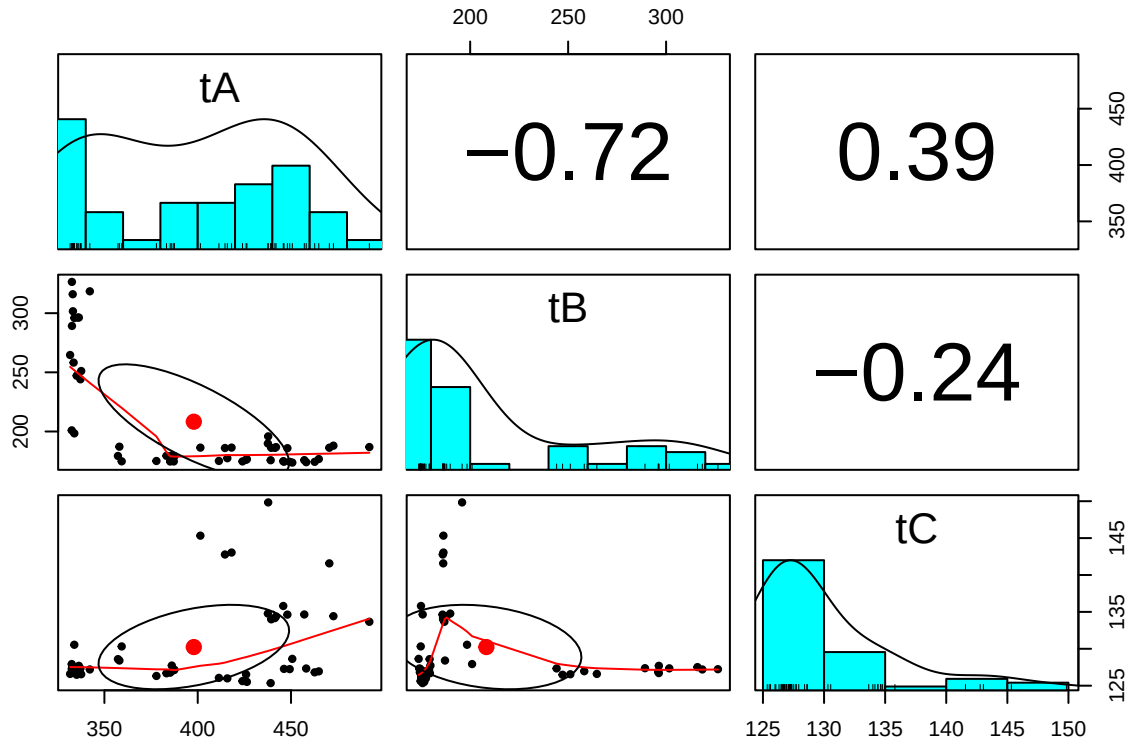
```
#teste de correlação entre variáveis
cor.test(projeto1[, 3],projeto1[, 4])
```

```
##
##  Pearson's product-moment correlation
##
## data:  projeto1[, 3] and projeto1[, 4]
## t = -7.1477, df = 48, p-value = 4.367e-09
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.8304211 -0.5495324
## sample estimates:
##        cor
## -0.7180469
```

Inicialmente apresentamos o índice de correlação de pearson e posteriormente desenvolvemos um teste onde a hipótese nula é que não existe correlação entre o tempo A e o tempo B. Neste caso como o valor-p$= 4.367e-09$ podemos concluir que ambos tempos estão correlacionados. Note que o sinal é negativo indicando que a correlação é inversa aquí.

A seguinte figura é usada para reportar os resultados da análise correlacional e análise descriva simultáneamente

```
#Analise correlacional com mais informação
pairs.panels(projeto1[, 3:5])
```



Na diagonal mostramos o histograma e densidade dos dados, no panel entre cada variável mostramos o diagrama de dispersão dos dados assim o valor da correlação de pearson.