

# Introdução à Análise de Dados com Linguagem R

## Aula 2

### Operadores Aritméticos e de Atribuição em R

Operador	Função
+	Soma
-	Subtração
/	Divisão
*	Multiplicação
%%	Resto da divisão
%/%	Parte inteira divisão
^	Potenciação
**	Potenciação
<-	Atribuição
=	Atribuição

In [ ]:

### Exercício Prático - Operadores Aritméticos e de Atribuição

#### Instrução 1/3

- Obter o resto da divisão entre os números inteiros 10 e 3.

In [ ]:

#### Instrução 2/3

- Obter a parte inteira da divisão de 10 por 3.

In [ ]:

#### Instrução 3/3

- Obter o quadrado de um número inteiro qualquer.

In [ ]:

# Operadores de Comparação em R

Operador	Significado
==	igual a
!=	diferente de
>	maior que
<	menor que
>=	maior ou igual a
<=	menor ou igual a

In [5]:

TRUE

In [6]:

```
# Leitura de vetores de um inventário florestal
load('./data/dados_modulo_1.rda')

# Mostrar os objetos atualmente disponíveis no ambiente R
ls()
```

1. 'altura'
2. 'categoria'
3. 'dap'
4. 'nomes\_cientificos'

In [9]:

## Função `mapply()`

Versão multivariada das funções `lapply` e `sapply`, utilizada para iterar entre elementos de vetores ou listas.

In [2]:

```
# Definição dos Vetores a e b
a <- c(7, 12, 5, 2, 1)
b <- c(4, 2, 3, 5, 1)

# Nomes para os vetores
dias_semana <- c('Segunda', 'Terça', 'Quarta', 'Quinta', 'Sexta')

# Atribuir nome aos vetores
names(a) <- dias_semana
names(b) <- dias_semana

# Uso da função mapply() para retornar a soma
# entre os elementos dos vetores a e b
mapply(max, a, b)
```

Segunda: 7 Terça: 12 Quarta: 5 Quinta: 5 Sexta: 1

## Função `tapply()`

Aplica uma função sobre um vetor com agrupamento em outro vetor categórico. Recebe como parâmetros: um vetor numérico, um vetor categórico e uma função. O código a seguir aplica a função média sobre o vetor `volume` agrupado ao vetor `ut` (unidades de trabalho)

```
In [11]: # Calcular o volume médio por unidade de trabalho
tapply(dap, categoria, mean)
```

**Explorar:** 76.8872521591047 **Remanescente:** 67.9580670429674 **Substituta:** 71.9543683510638

```
In [ ]:
```

## Testes Lógicos com Vetores

- `any()` Testa se algum elemento do vetor atende a uma condição específica

**Exemplo:** Dado o vetor de nome `dap`, o qual armaneza dados de mensuração de diâmetro de milhares de árvores na Floresta Nacional de Altamira, teste se algum elemento é menor ou igual a 40.

```
In [12]: any(dap >= 40)
```

TRUE

- `all()` Testa se todos os elementos de um vetor atendem a uma condição.

**Exemplo:** Dado o vetor de nome `dap`, testar se algum elemento é menor do que 0:

```
In [13]: all(dap < 40)
```

FALSE

- `is.na()` Testa se o vetor contém valores ausentes (*Not Availables*)

## Exercício Prático - Índice de Vetor

Considerando o vetor de nome `dap`, o qual armaneza dados de mensuração de diâmetro de milhares de árvores na Floresta Nacional de Altamira, mostre:

### Instrução 1/5

- Quantas árvores foram inventariadas.

In [ ]:

#### Instrução 2/5

- **Apenas** o penúltimo elemento desse vetor.

#### Instrução 3/5

- O diâmetro mínimo de medição

In [ ]:

#### Instrução 4/5

- O diâmetro máximo mensurado

In [ ]:

#### Instrução 5/5

- O diâmetro médio mensurado

In [ ]:

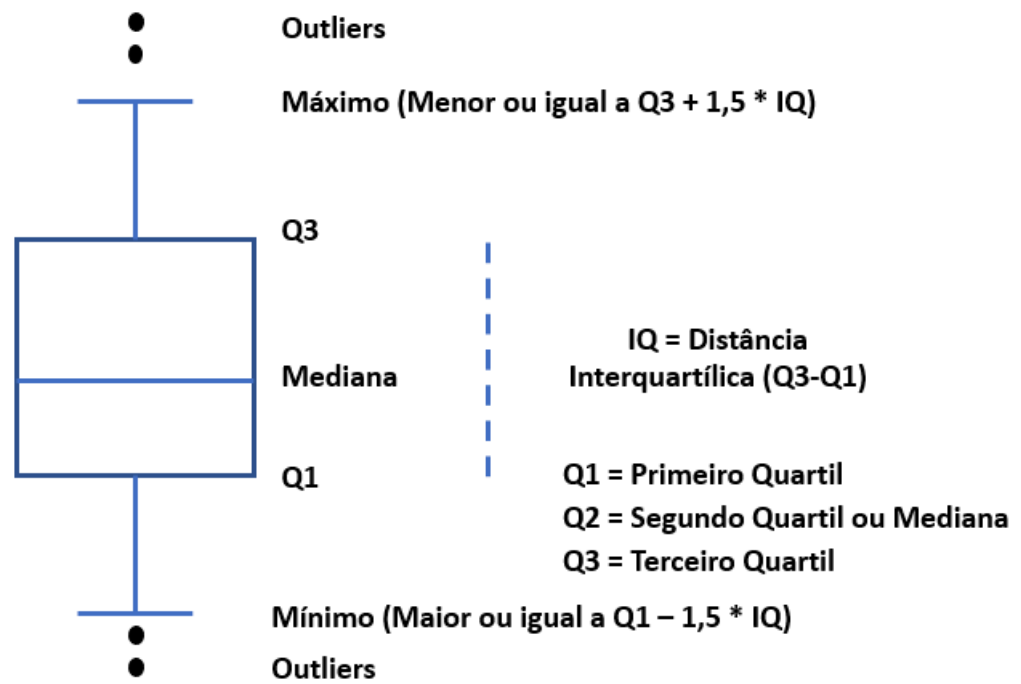
## Visualização Gráfica de Vetores

Hora de tentar algo um pouco diferente. Até agora, você programou *script* e observou seus dados imprimindo-os. Para uma visualização mais informativa de dados, experimente uma saída gráfica.

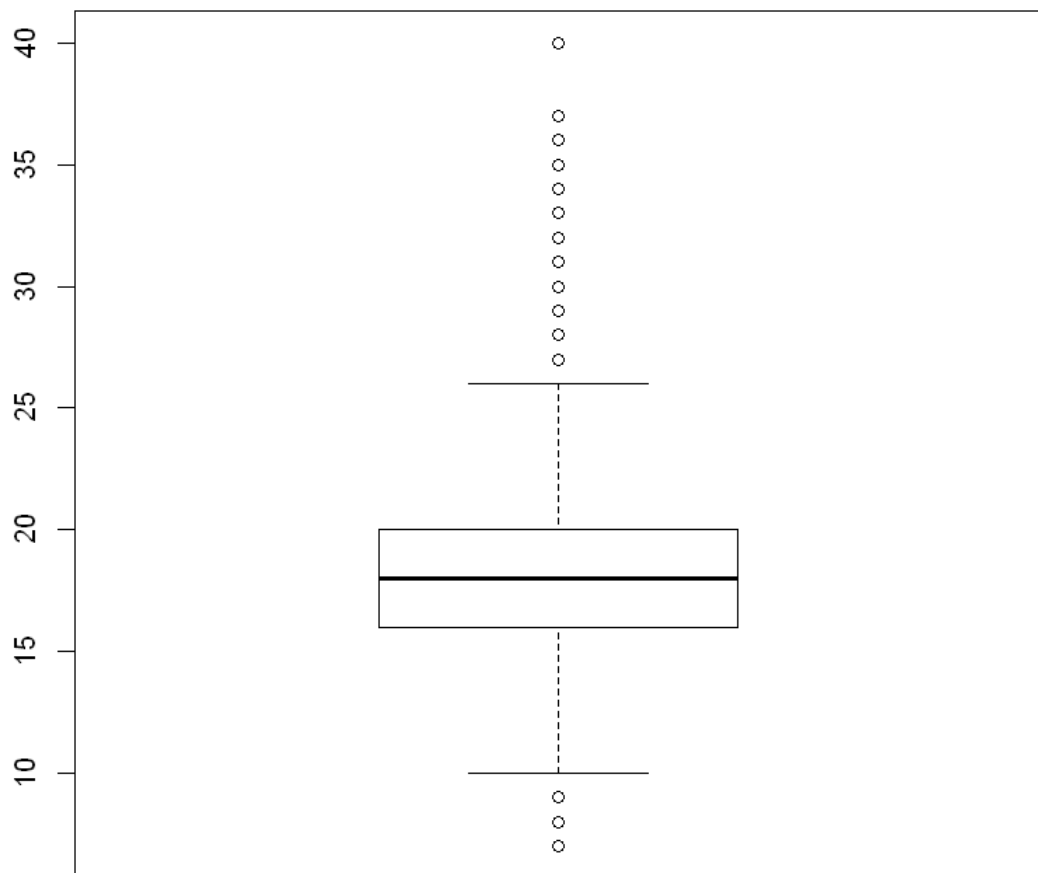
Para este exercício, você irá trabalhar com dados de inventário florestal realizado em uma unidade de produção anual da Floresta Nacional de Altamira. Para tal utilizaremos apenas duas variáveis, a saber: diâmetro a altura do peito (DAP) e altura comercial.

## Boxplot

O *boxplot* ou diagrama de caixa é uma ferramenta gráfica da estatística que nos permite visualizar a distribuição e valores discrepantes (outliers) de dados.



In [7]: `boxplot(altura)`



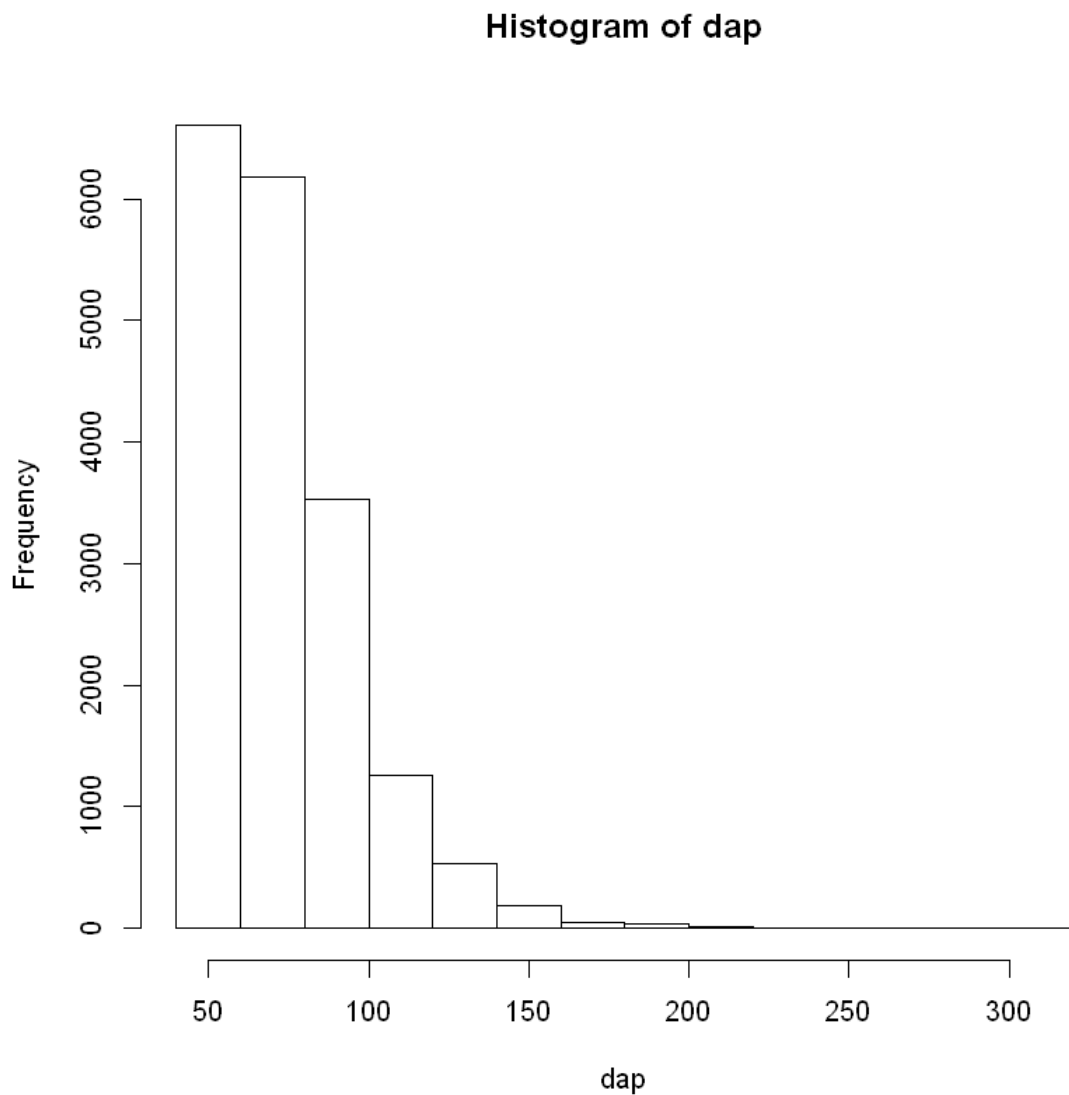
# Histograma

Utilizamos histogramas para visualizar a distribuição de uma variável contínua. Em R o pacote "base" nos fornece a função `hist( )`.

## Exercício Prático

- Mostrar o histograma para os dados da variável DAP disponível no vetor `dap`

```
In [8]: hist(dap)
```



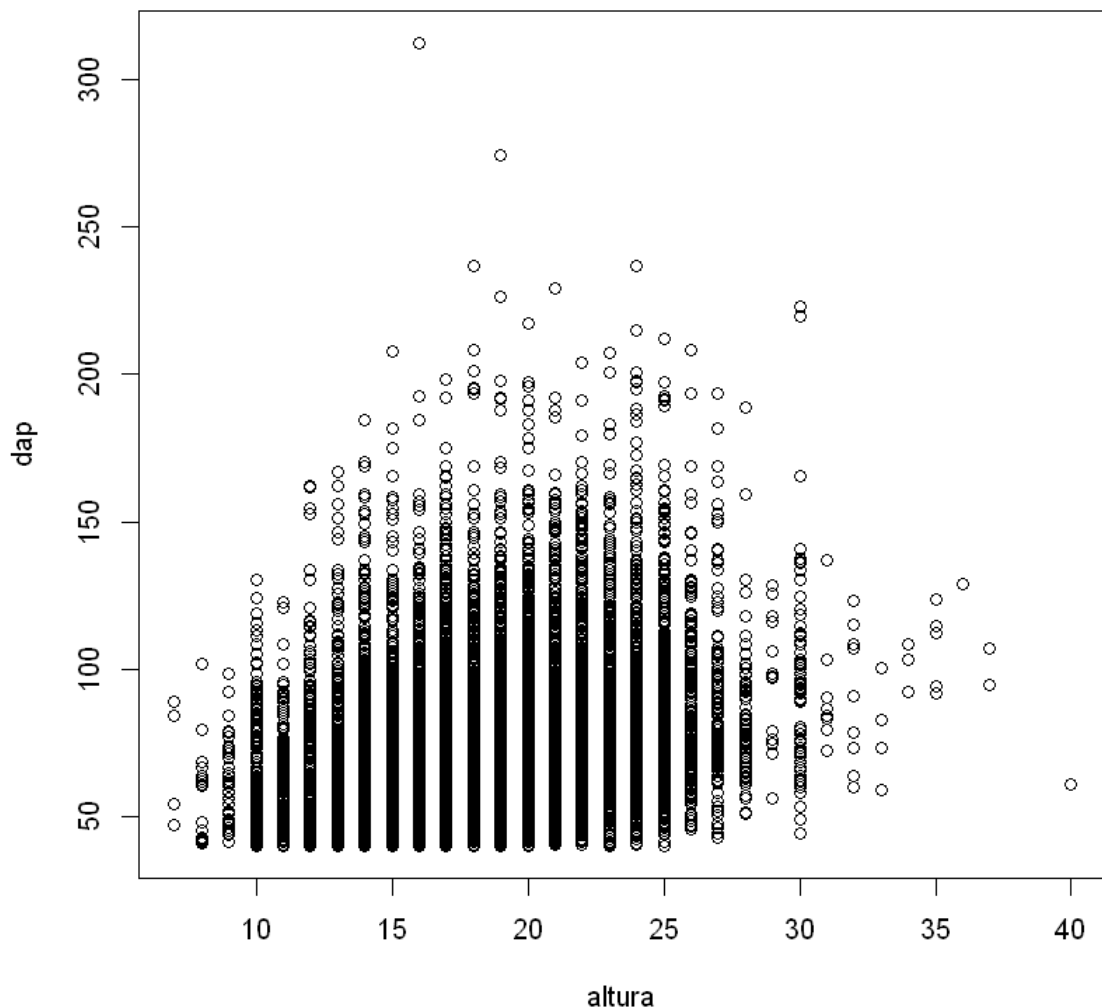
# Gráfico Dispersão

O gráfico de dispersão é utilizado para visualizar a relação entre duas variáveis contínuas. Para gerar um gráfico de dispersão em R devemos utilizar a função `plot` do pacote "base".

## Exercício Prático

Visualizar a relação entre a variável altura e diâmetro.

```
In [9]: plot(altura, dap)
```



## Modificar a Aparência dos Gráficos

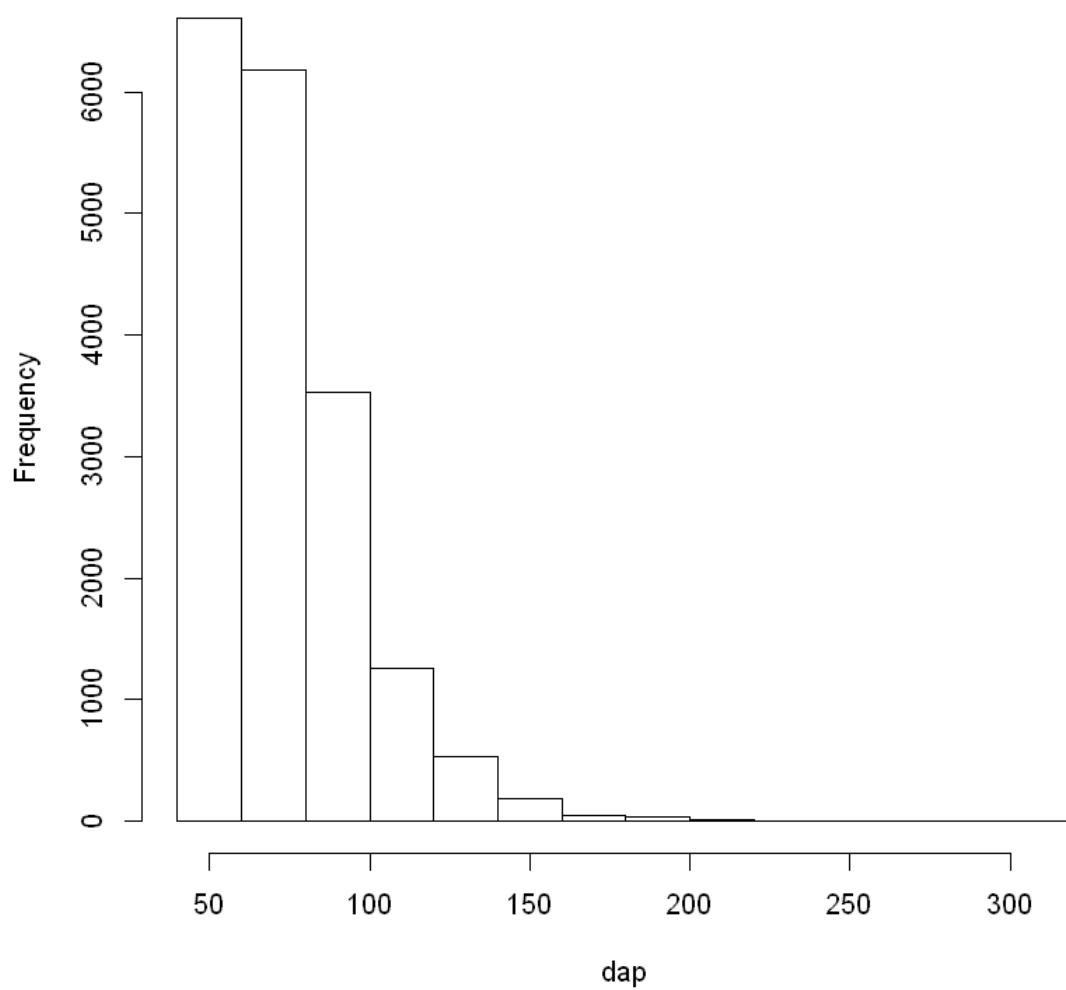
A configuração dos parâmetros de estilo, tamanho e agrupamento dos gráficos pode ser obtida digitando o comando `? par`. Para este curso introdutório utilizaremos apenas o básico da configuração da aparência de gráficos no pacote "base" do R.

## Personalizando Histogramas

- `main` Utilizado para atribuir ou modificar um título do gráfico

```
In [12]: # Modificar o título do gráfico
hist(dap, main = 'Distribuição da variável DAP (cm)')
```

### Distribuição da variável DAP (cm)

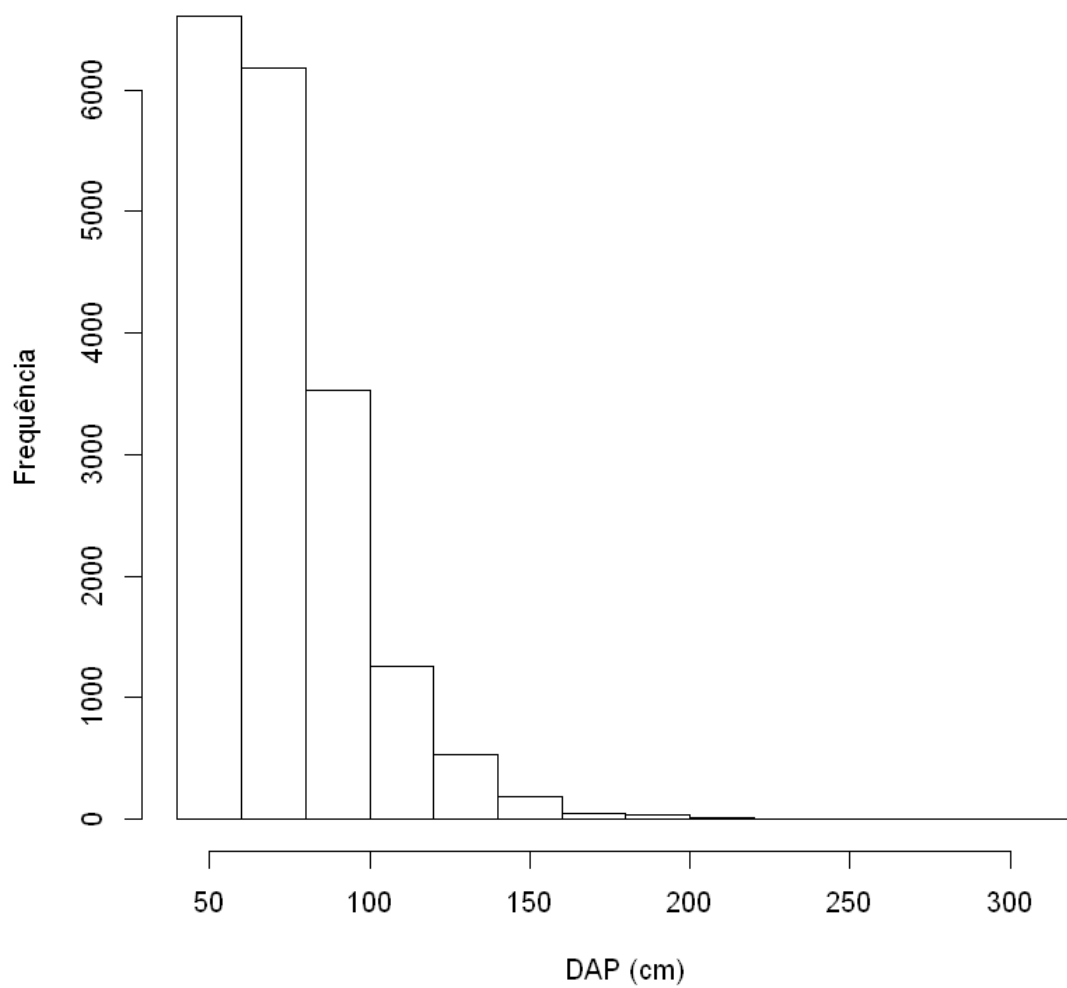


- `xlab` e `ylab` - Modificar os nomes dos eixos x e y.

```
In [13]: # Modificar os rótulos dos eixos x e y
hist(dap,
      main = 'Distribuição da variável DAP', # título do gráfico
      xlab = 'DAP (cm)', # Rótulo do eixo x
      ylab = 'Frequência') # Rótulo do eixo y
```



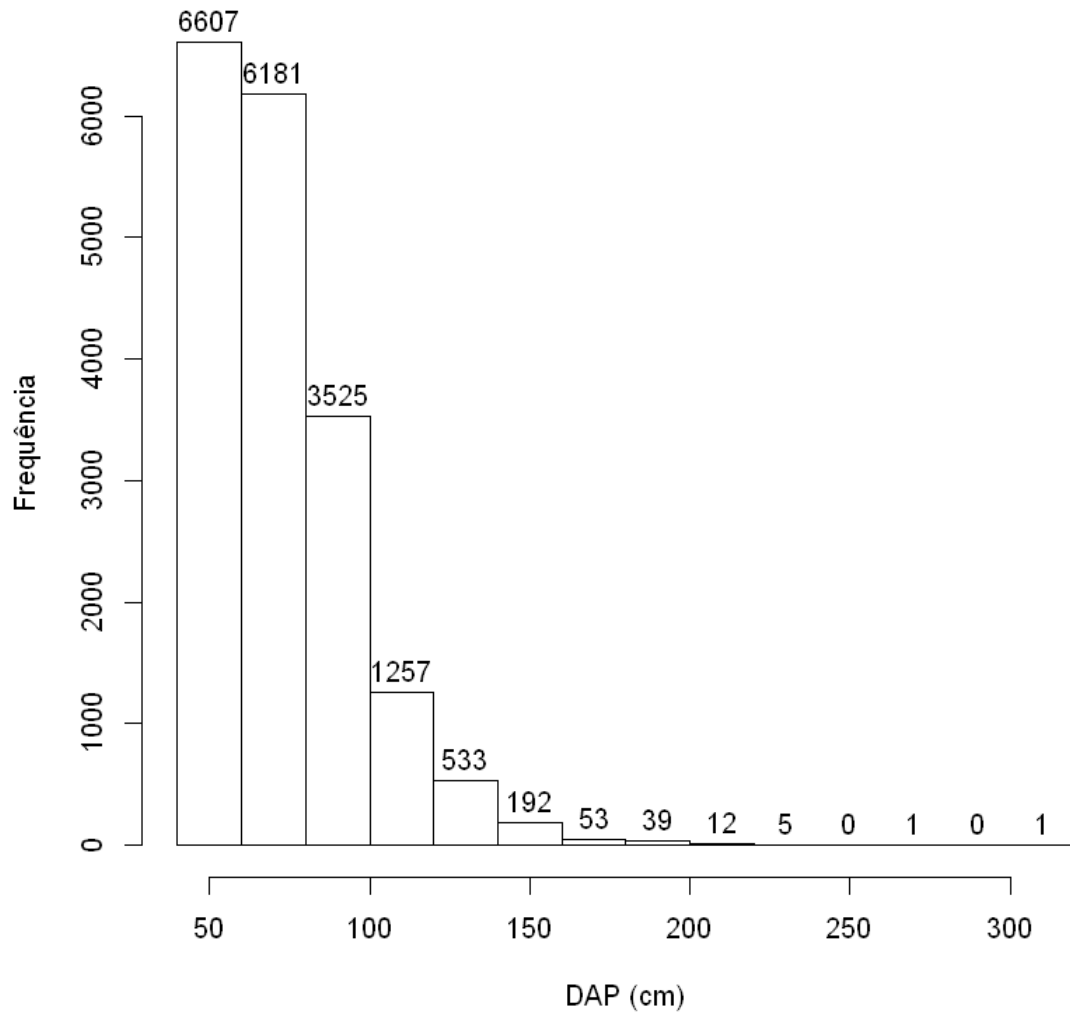
## Distribuição da variável DAP



- `labels` - Mostra os valores de cada barra do histograma.

```
In [14]: hist(dap,  
             main = 'Distribuição da variável DAP',  
             xlab = 'DAP (cm)',  
             ylab = 'Frequência',  
             labels = TRUE)
```

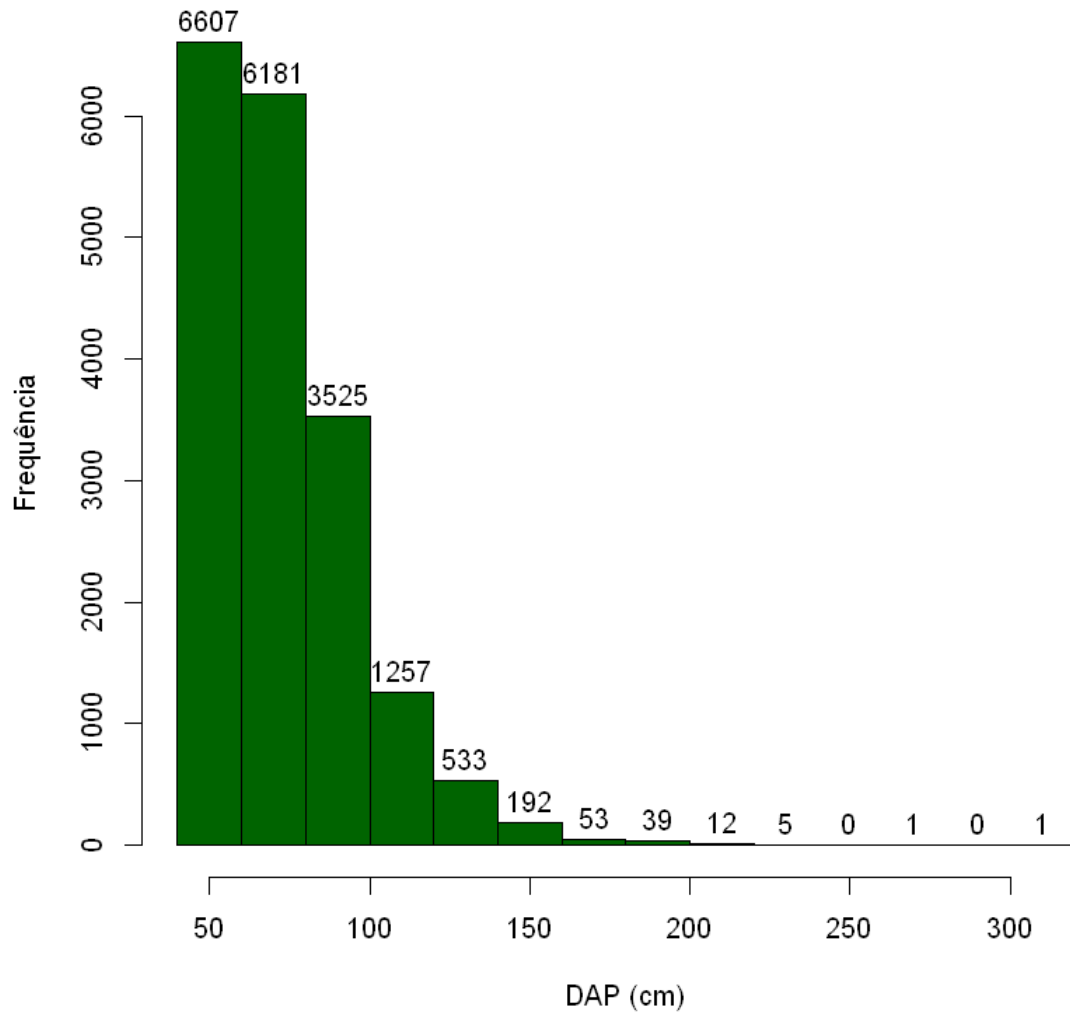
## Distribuição da variável DAP



- `col` - Muda a cor das barras do histograma.

```
In [16]: hist(dap,
              main = 'Distribuição da variável DAP',
              xlab = 'DAP (cm)',
              ylab = 'Frequência',
              labels = TRUE,
              col = 'darkgreen')
```

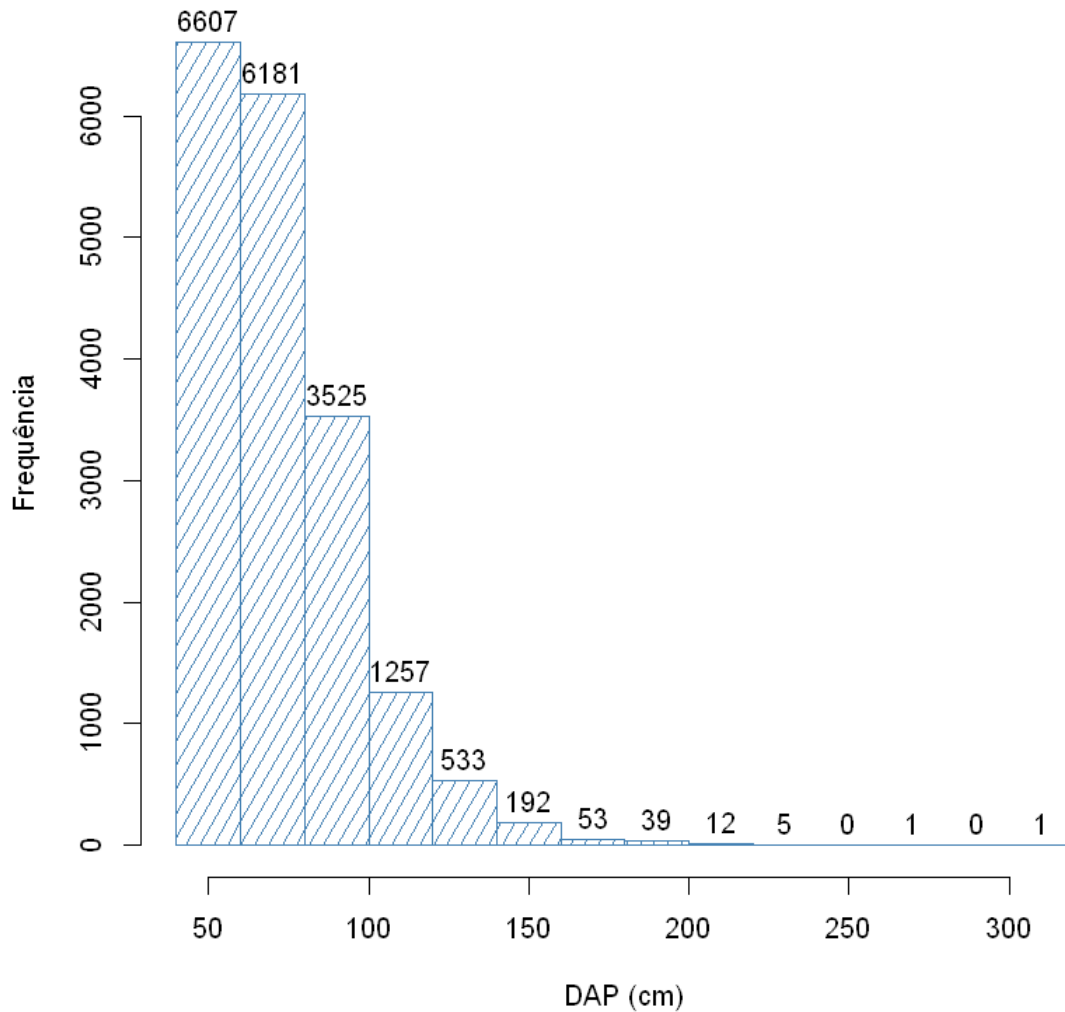
## Distribuição da variável DAP



- `density` e `angle` - Mostram as barras do histograma hachuradas

```
In [17]: hist(dap,
             main = 'Distribuição da variável DAP',
             xlab = 'DAP (cm)',
             ylab = 'Frequência',
             labels = TRUE,
             col = 'steelblue',
             density = 15,
             angle = 60)
```

## Distribuição da variável DAP



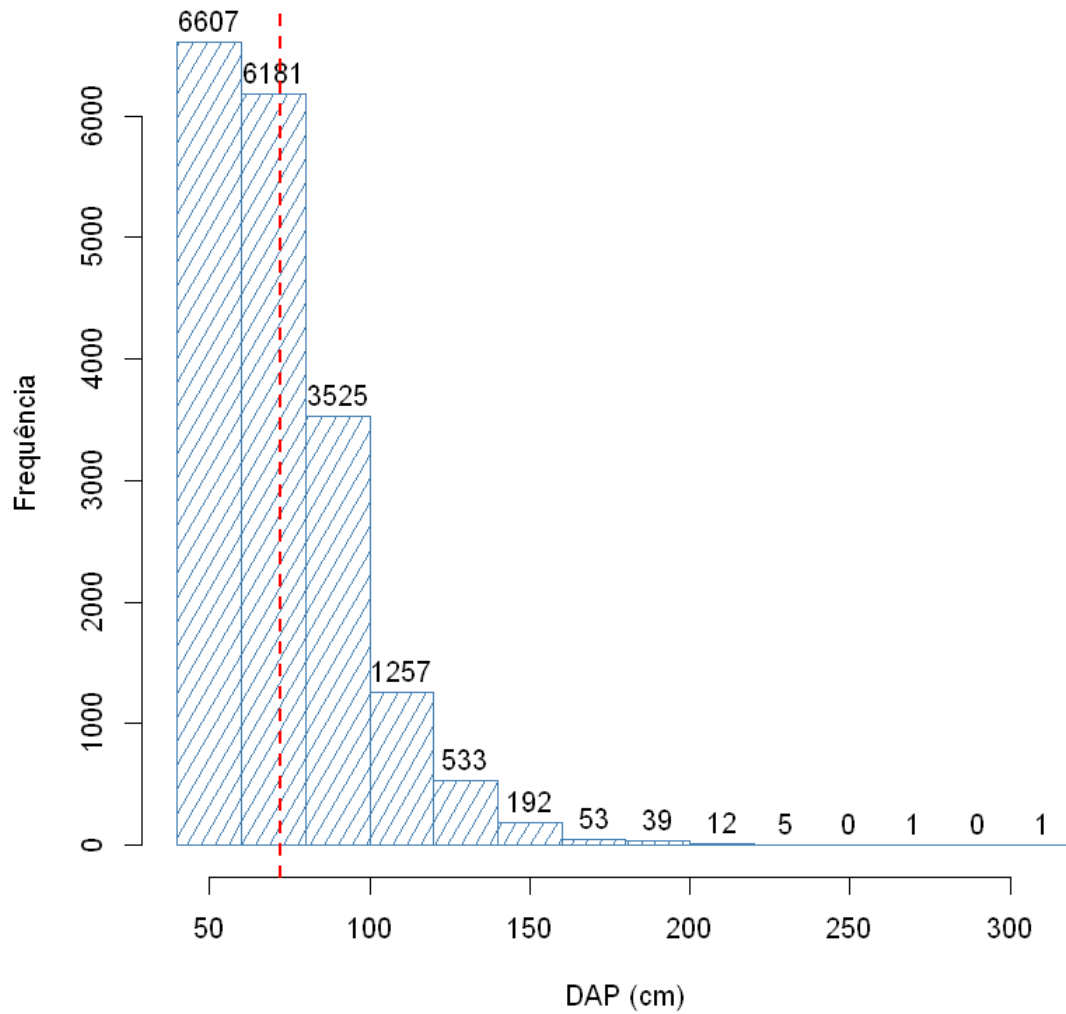
- `Abline( )` - Função para adicionar uma linha reta ao histograma. Para adicionar uma linha vertical deve-se utilizar o argumento `v` e para linha horizontal `h`. O tipo de linha é modificado através do argumento `lty` (*line type*) e a espessura da linha através do argumento `lwd` (*line width*).

O exemplo a seguir mostra como adicionar uma linha ao histograma para representar a média dos diâmetros.

```
In [18]: hist(dap,
  main = 'Distribuição da variável DAP',
  xlab = 'DAP (cm)',
  ylab = 'Frequência',
  labels = TRUE,
  col = 'steelblue',
  density = 15,
  angle = 60)

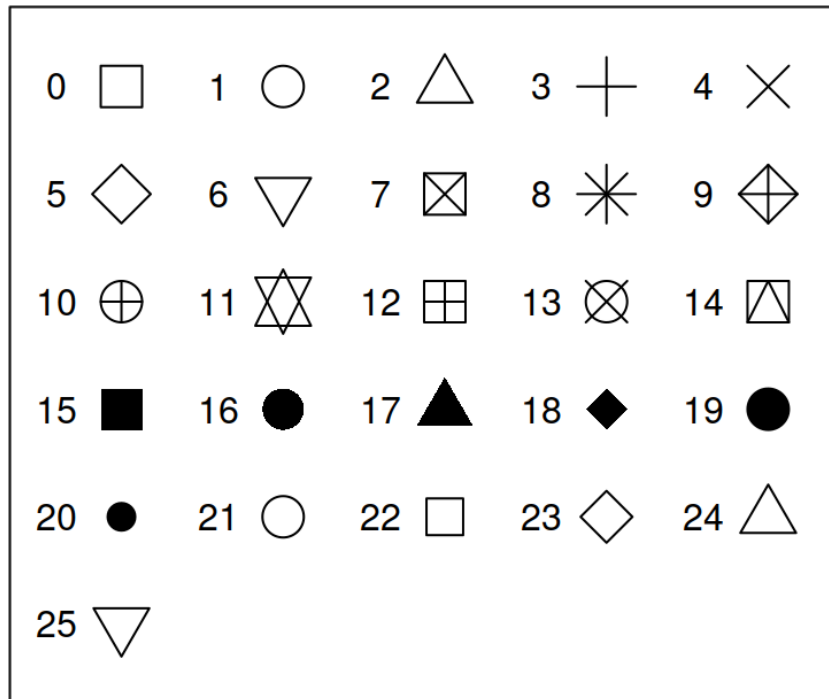
abline(v = mean(dap),
  col = 'red',
  lty = 2,
  lwd = 2)
```

## Distribuição da variável DAP



## Personalizando Gráficos de Dispersão

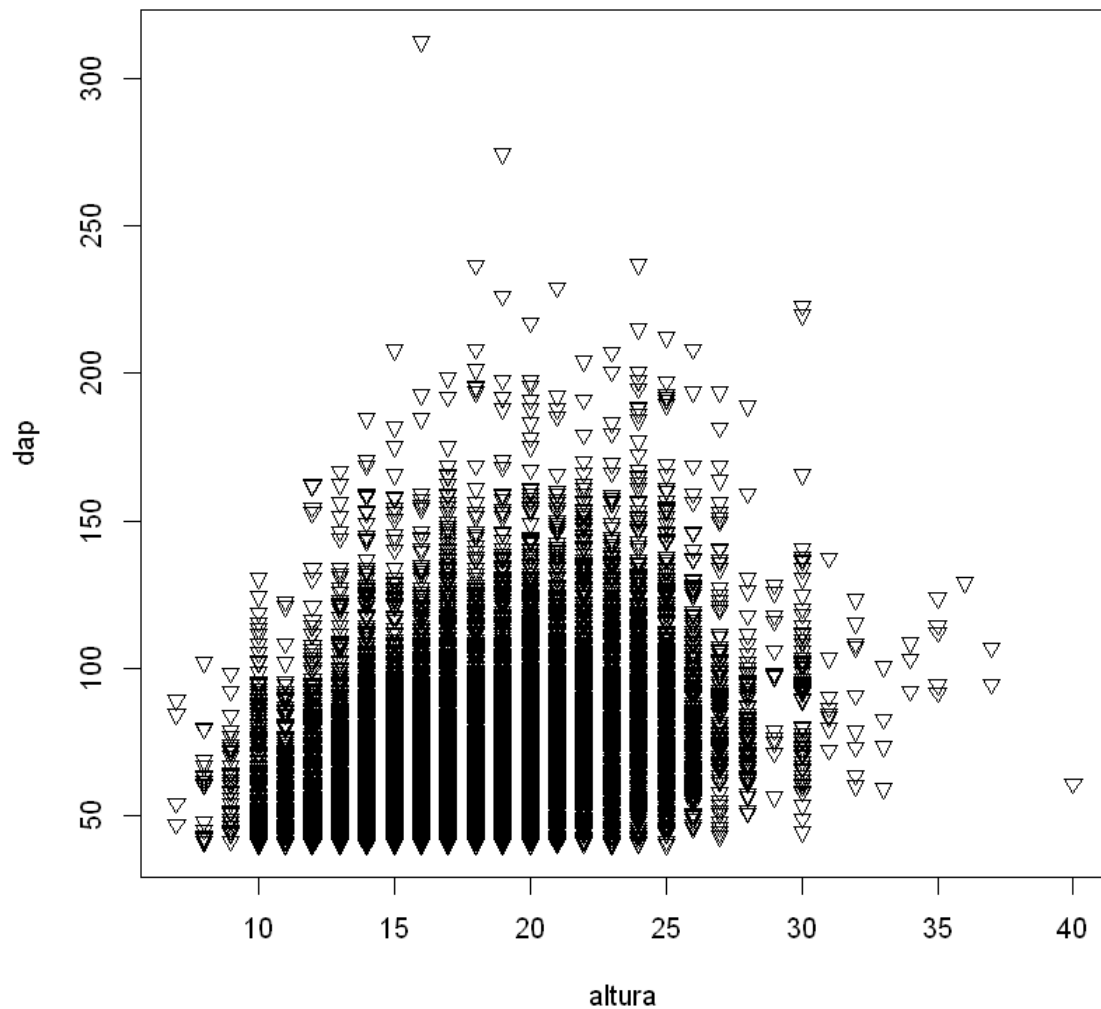
- `pch` - Altera o tipo de caractere dos pontos



- Forma, Tamanho e Cor dos Pontos

Argumento	Saída
<code>col</code>	Cor da borda do ponto.
<code>bg</code>	Cor do Fundo do ponto.
<code>cex</code>	Tamanho do ponto.
<code>lwd</code>	Espessura da Borda do Ponto.

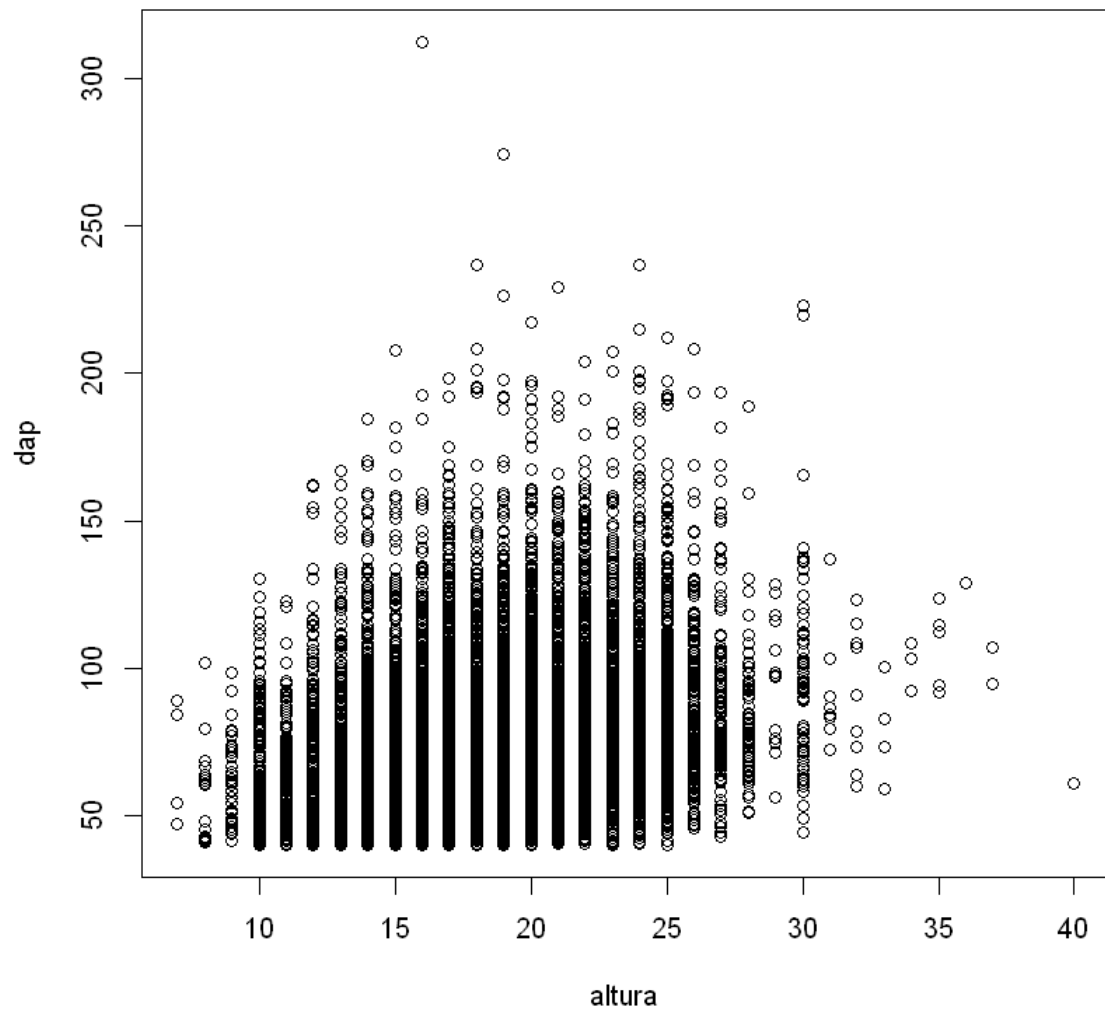
In [23]: `# Alterar o tipo de caractere dos pontos`  
`plot(altura, dap, pch = 25)`



- `las` (*label style*) - Rotação dos rótulos dos eixos x e y.

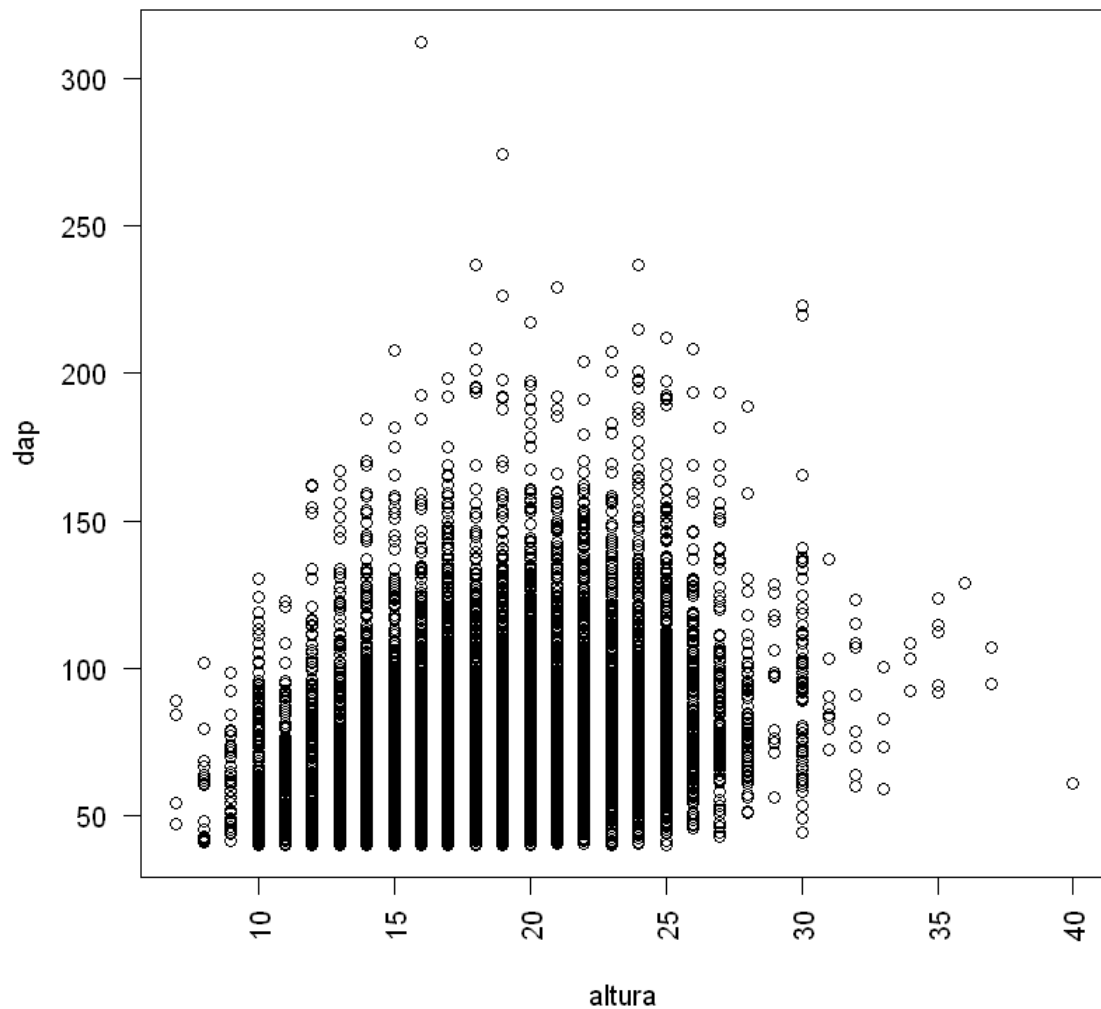
<code>las</code>	Rótulo
0	Paralelo aos eixos
1	Sempre na Horizontal
2	Sempre na Perpendicular
3	Sempre na Vertical

In [24]: `plot(altura, dap, pch = 21, las = 0)`



```
In [25]: plot(altura, dap, pch = 21, las = 2)
```



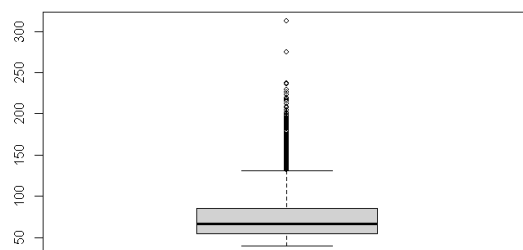
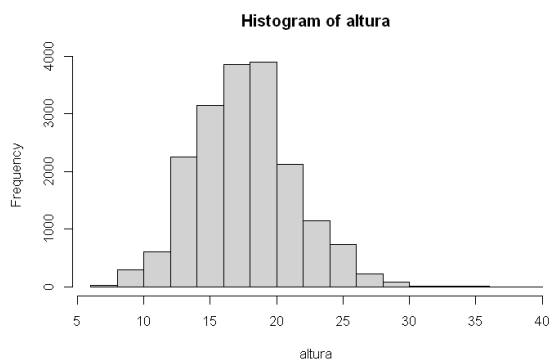
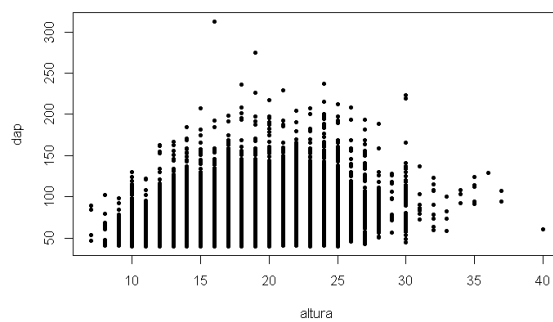
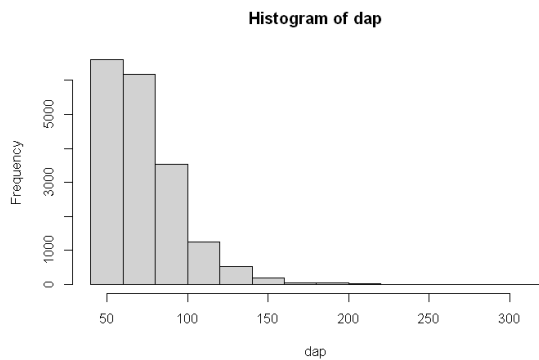


## Agrupar Gráficos em uma Única Figura

```
In [91]: # Parâmetros gráficos
par(mfcol = c(2, 2))

hist(dap, pin = c(12, 8))
hist(altura)
plot(altura, dap, pch = 20)
boxplot(dap)

# Encerrar a função par()
dev.off()
```



In [ ]:

In [ ]:

In [90]: `# Change plot size to 6 x 4`  
`options(repr.plot.width=12, repr.plot.height=8)`

In [ ]: