

Taller de R básico. Día 4.

Viacheslav Shalisko

20 de octubre de 2016

A. Cargar la tabla de datos

Estructura de datos (tabla `Datos_del_censo.csv`):

1. *Centro* - código del CU
2. *Especie* - nombre científico
3. *Codigo* - identificador único del árbol
4. *AB* - área basal del árbol (dm^2)
5. *DTr* - diametro del tronco equivalente (cm)
6. *Alt* - estatura del árbol (m)
7. *DCop* - diametro promedio de la copa (m)
8. *ExcCop* - excentricidad de la copa

```
arbolado <- read.csv("Materiales/Datos_del_censo.csv")
```

Partes B. y C. fueron tratados en la sesión anterior.

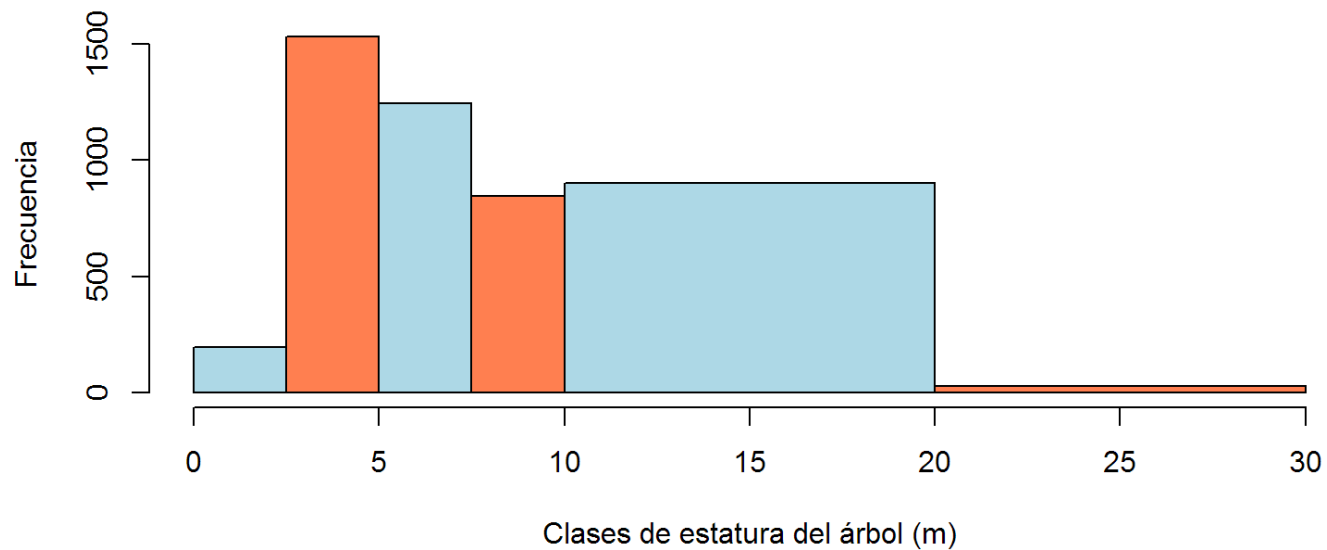
Continuación

D. Gráficas simples

Histogramas

```
H0 <- hist(arbolado$Alt,  
  col = c("lightblue", "coral"),  
  xlab = "Clases de estatura del árbol (m)", ylab = "Frecuencia",  
  main = "Histograma de estaturas",  
  breaks = c(0, 2.5, 5, 7.5, 10, 20, 30), prob = FALSE)
```

Histograma de estaturas

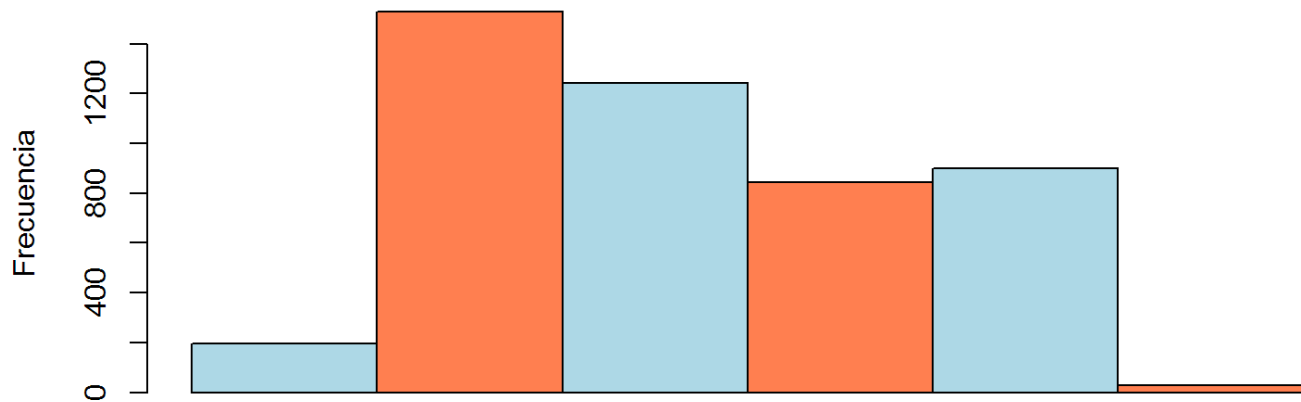


```
str(H0)
```

```
## List of 6
## $ breaks : num [1:7] 0 2.5 5 7.5 10 20 30
## $ counts : int [1:6] 196 1529 1242 844 900 28
## $ density : num [1:6] 0.0165 0.1291 0.1048 0.0712 0.019 ...
## $ mids : num [1:6] 1.25 3.75 6.25 8.75 15 25
## $ xname : chr "arbolado$Alt"
## $ equidist: logi FALSE
## - attr(*, "class")= chr "histogram"
```

```
barplot(H0$counts, space = 0,
  col = c("lightblue","coral"),
  xlab = "Clases de estatura del árbol (m)", ylab = "Frecuencia",
  main = "Histograma de estaturas")
```

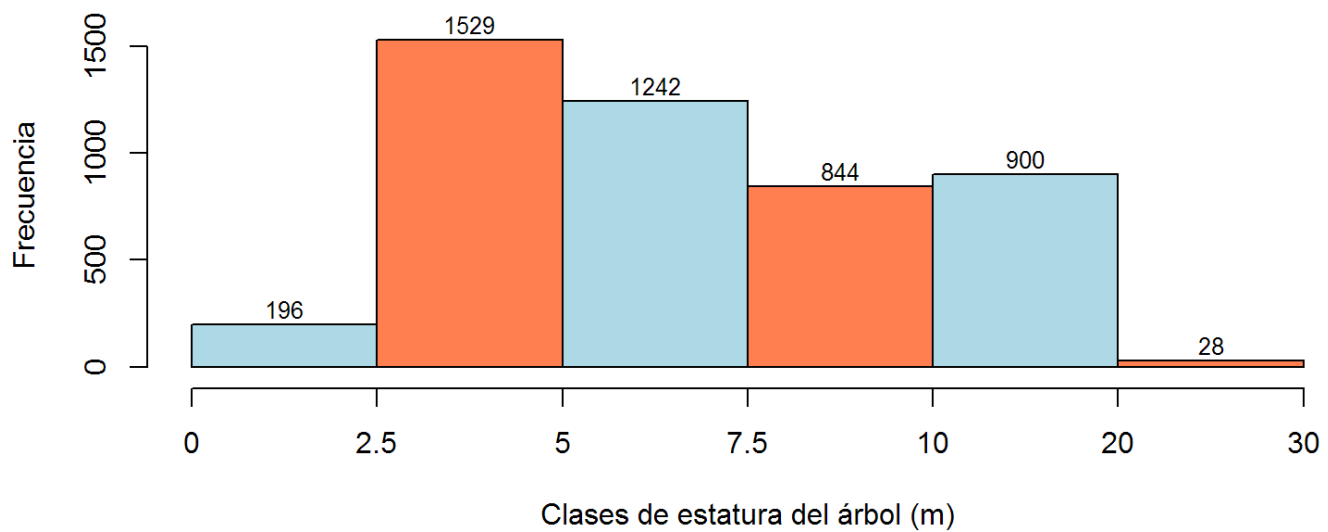
Histogr ma de estaturas



Clases de estatura del  rbol (m)

```
B0 <- barplot(H0$counts, space = 0, ylim = c(-100,1700),
  col = c("lightblue","coral"),
  xlab = "Clases de estatura del  rbol (m)", ylab = "Frecuencia",
  main = "Histogr ma de estaturas")
axis(at = seq(from = 0, to = 6, by = 1),
  labels = c(0,2.5,5,7.5,10,20,30), side = 1)
text(B0, H0$counts + 0.05 * max(H0$counts), labels=round(H0$counts), cex = 0.8)
```

Histogr ma de estaturas



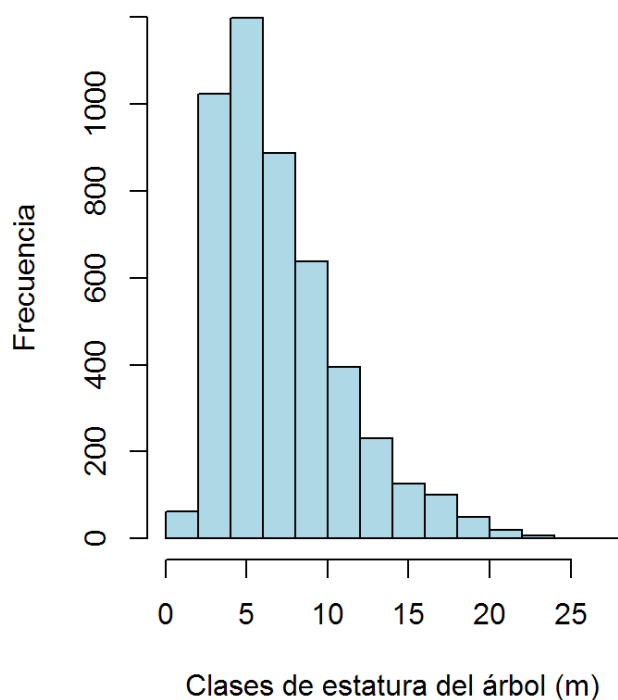
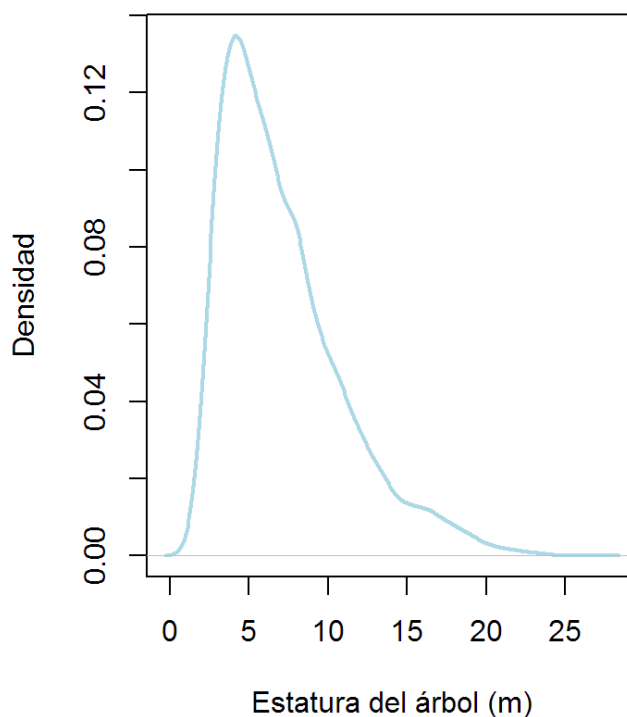
```

par(mfcol = c(1, 2))

hist(arbolado$Alt,
     col = c("lightblue"),
     xlab = "Clases de estatura del árbol (m)", ylab = "Frecuencia",
     main = "Histogr ma de estaturas", prob = FALSE)

plot(density(arbolado$Alt, na.rm = TRUE),
     col = c("lightblue"), lty=1, lwd=2,
     xlab = "Estatura del  rbol (m)", ylab = "Densidad",
     main = "Distribuci n de estaturas")

```

Histogr ma de estaturas**Distribuci n de estaturas**

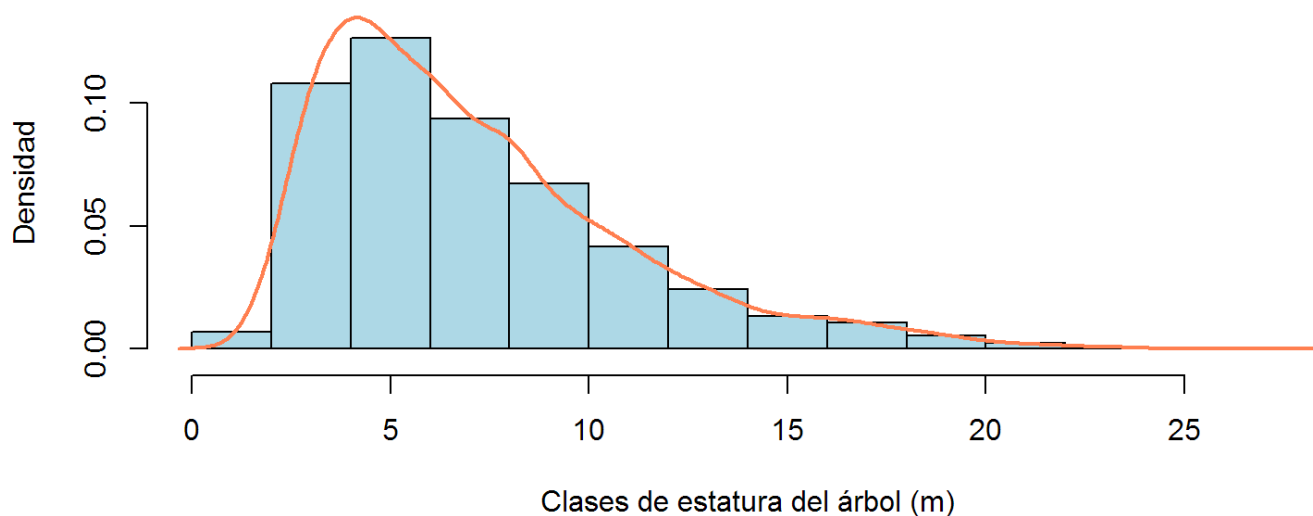
```

H1 <- hist(arbolado$Alt, ylim = c(-0.005,0.14),
          col = c("lightblue"),
          xlab = "Clases de estatura del  rbol (m)", ylab = "Densidad",
          main = "Histogr ma de estaturas con gr fica de densidad de distribuci n", prob = TRUE)

densidad <- density(arbolado$Alt, na.rm = TRUE)
lines(densidad, col = c("coral"), lty=1, lwd=2)

```

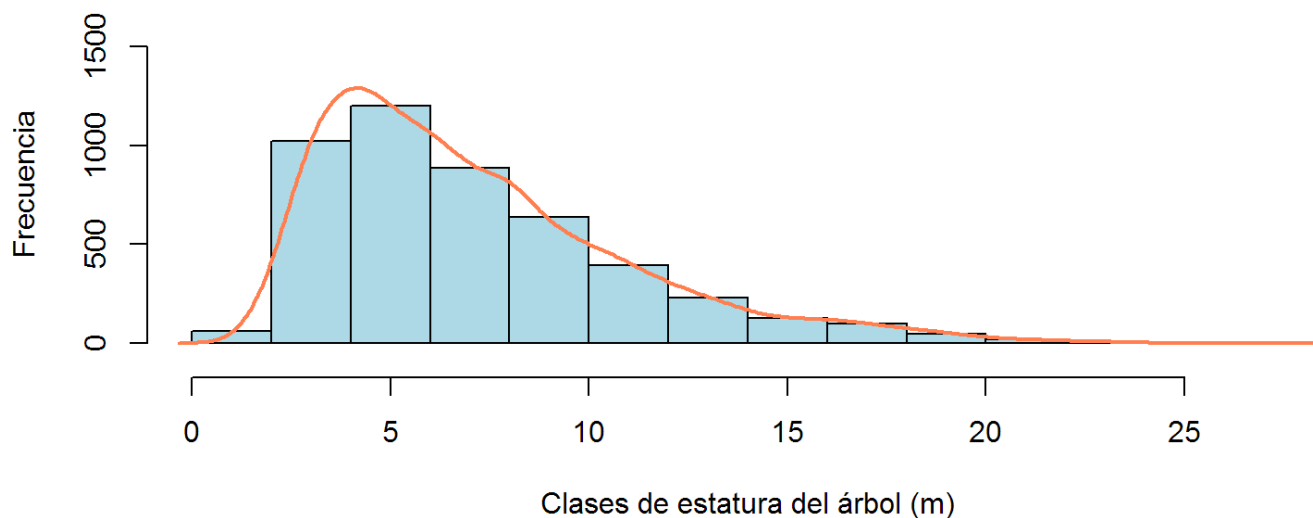
Histogr ma de estaturas con gr fica de densidad de distribuci n



```
H2 <- hist(arbolado$Alt, ylim = c(-100,1700),
  col = c("lightblue"),
  xlab = "Clases de estatura del  rbol (m)", ylab = "Frecuencia",
  main = "Histogr ma de estaturas con gr fica de densidad de distribuci n", prob = FALSE)

densidad <- density(arbolado$Alt, na.rm = TRUE)
densidad$y <- densidad$y*diff(H2$mids[1:2])*length(arbolado$Alt)
lines(densidad, col = c("coral"), lty=1, lwd=2)
```

Histogr ma de estaturas con gr fica de densidad de distribuci n



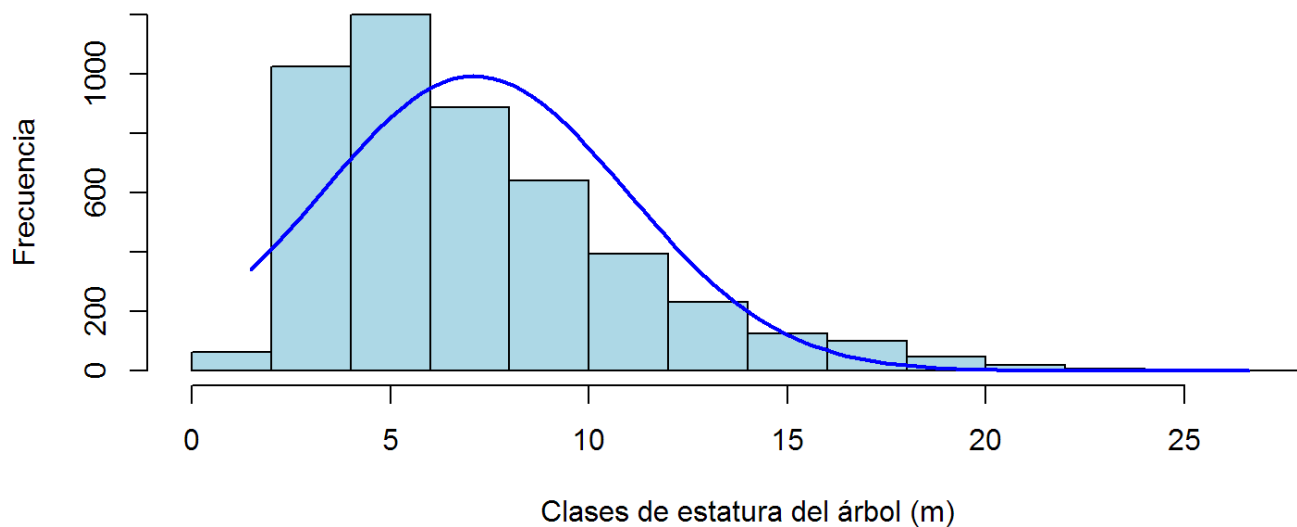
```

plot(H1, col = c("lightblue"),
     xlab = "Clases de estatura del árbol (m)", ylab = "Frecuencia",
     main = "Histograma de estaturas con gráfica de distribución normal")

xfit<-seq(min(arbolado$Alt, na.rm = TRUE),max(arbolado$Alt, na.rm = TRUE),length=100)
yfit<-dnorm(xfit,mean=mean(arbolado$Alt, na.rm = TRUE),sd=sd(arbolado$Alt, na.rm = TRUE))
yfit <- yfit*diff(H1$mids[1:2])*length(arbolado$Alt)
lines(xfit, yfit, col="blue", lwd=2)

```

Histograma de estaturas con gráfica de distribución normal



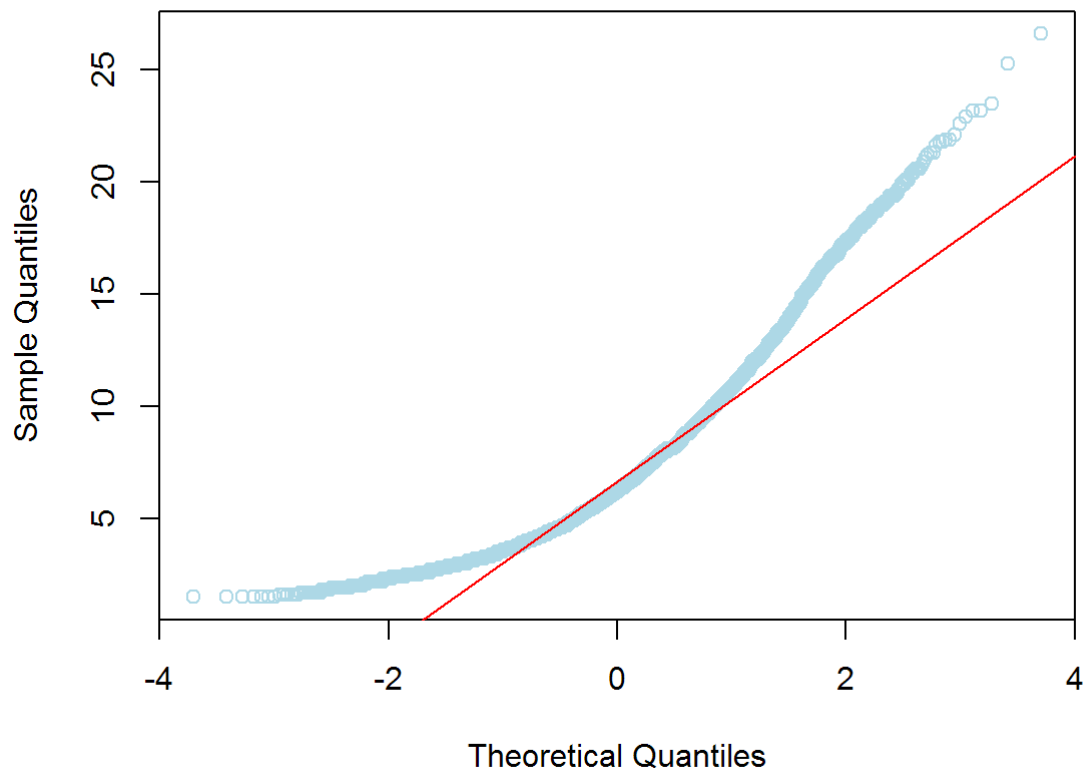
Gráficas QQ

```

qqnorm(arbolado$Alt, col = "lightblue")
qqline(arbolado$Alt, col = "red")

```

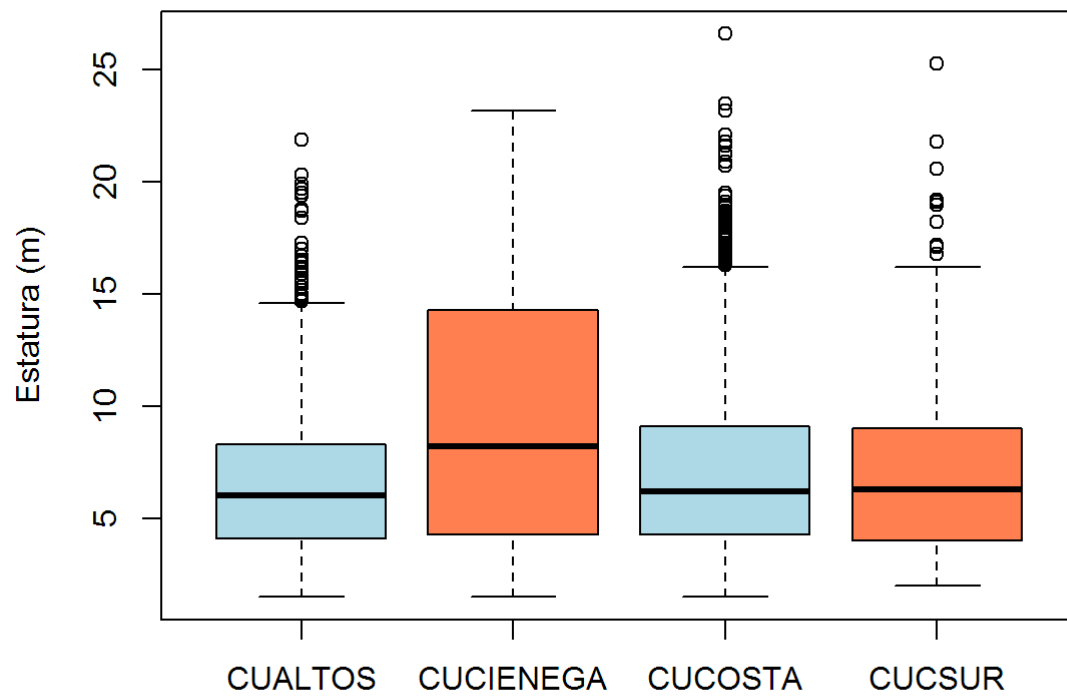
Normal Q-Q Plot



Gráficas de caja

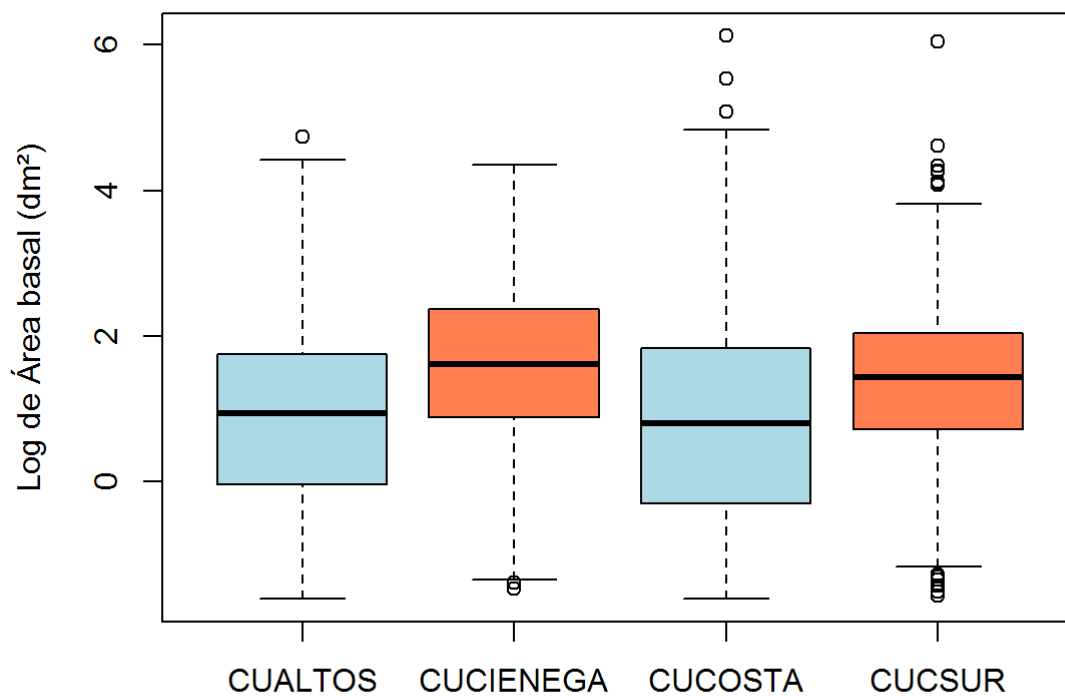
```
boxplot(arbolado$Alt ~ arbolado$Centro,  
        col = c("lightblue","coral"),  
        main = "Estatura de árboles por centro",  
        ylab = "Estatura (m)")
```

Estatura de árboles por centro



```
boxplot(log(arbolado$AB) ~ arbolado$Centro,  
        col = c("lightblue","coral"),  
        main = "Área basal de árboles por centro",  
        ylab = "Log de Área basal (dm²)")
```

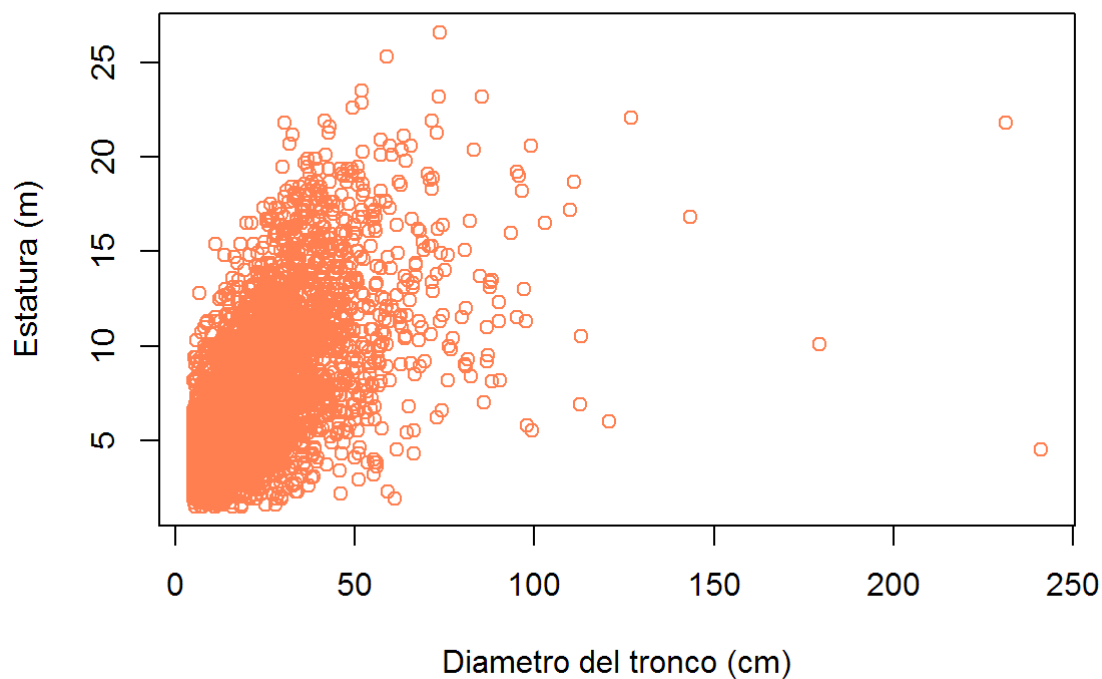

Área basal de árboles por centro



Gráficas de puntos

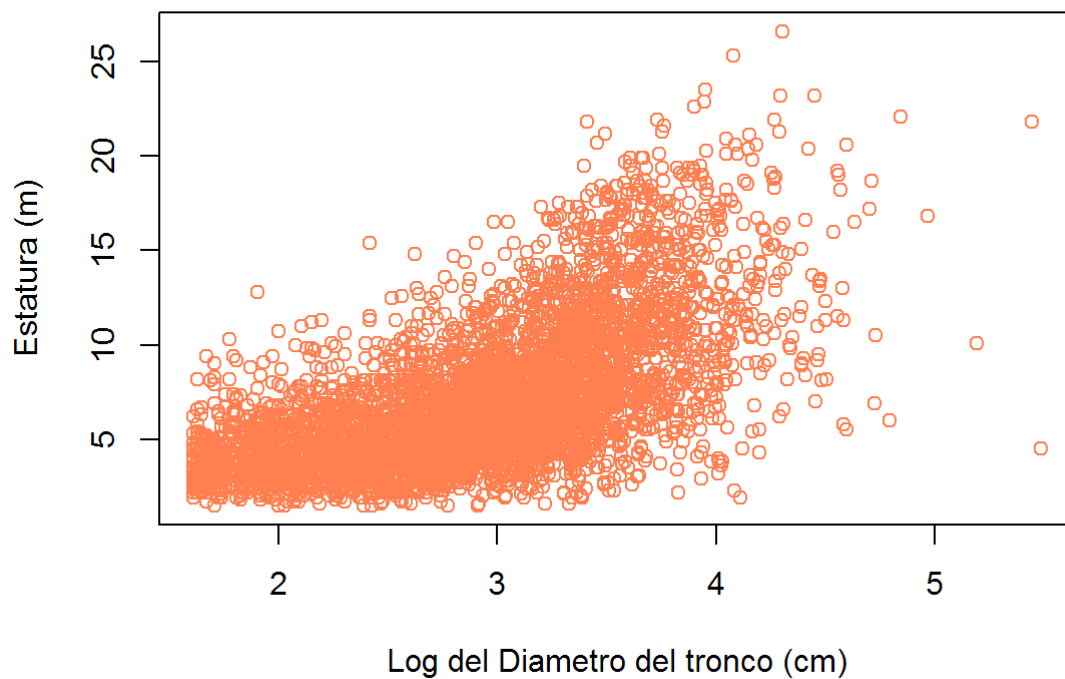
```
plot(arbolado$Alt ~ arbolado$DTr,
     col = "coral",
     main = "Estatura de árboles vs. diametro del tronco",
     ylab = "Estatura (m)", xlab = "Diametro del tronco (cm)")
```

Estatura de árboles vs. diametro del tronco



```
plot(arbolado$Alt ~ log(arbolado$DTr),  
     col = "coral",  
     main = "Estatura de árboles vs. diametro del tronco",  
     ylab = "Estatura (m)", xlab = "Log del Diametro del tronco (cm)")
```

Estatura de árboles vs. diametro del tronco



Graficas de puntos pueden incluir grupos de datos

```

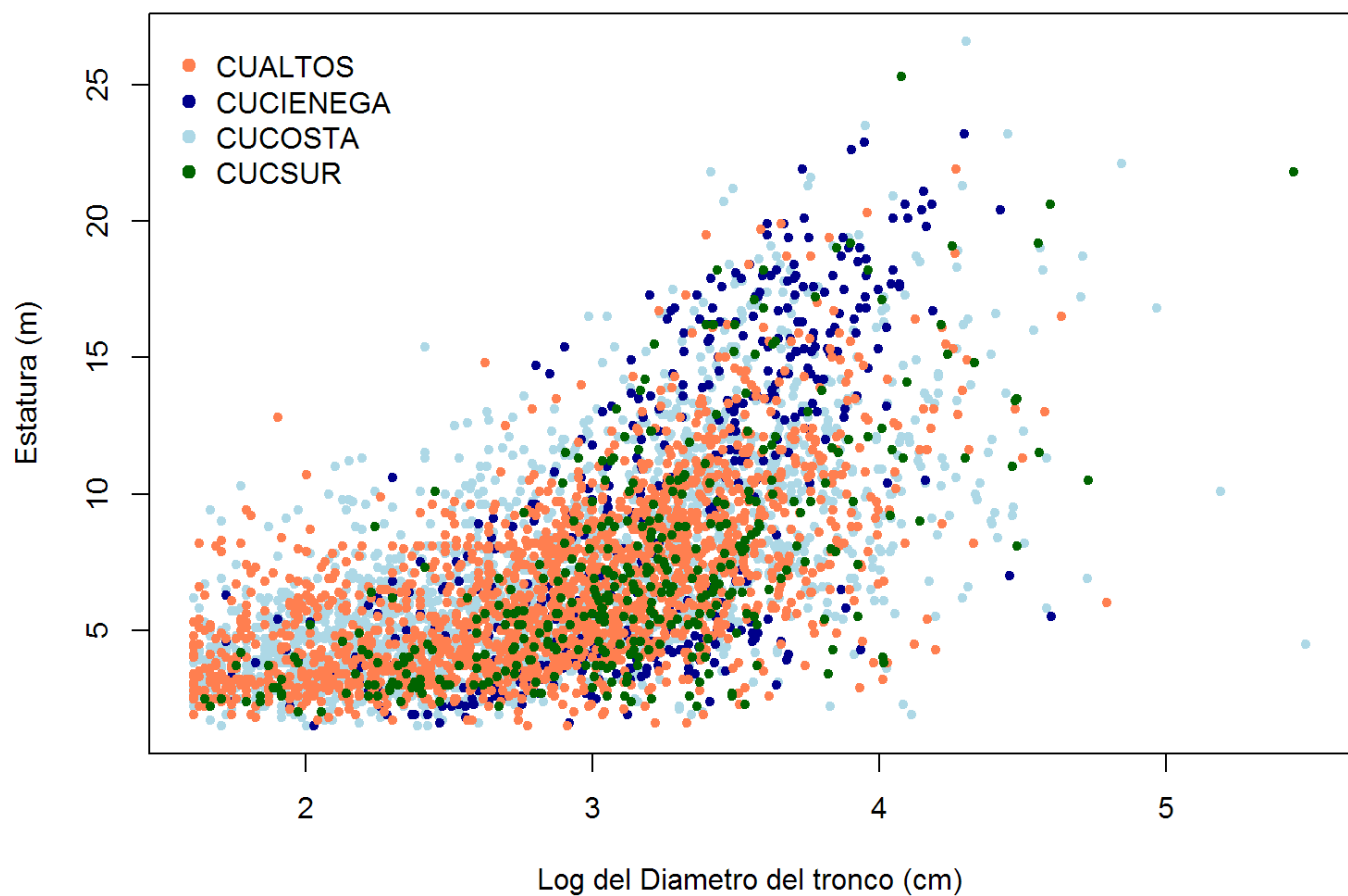
colors <- c("coral","darkblue","lightblue","darkgreen")
color.index <- as.numeric(arbolado$Centro)

plot(arbolado$Alt ~ log(arbolado$DTr),
     col = colors[color.index], pch = 20,
     main = "Estatura de árboles vs. diametro del tronco",
     ylab = "Estatura (m)", xlab = "Log del Diametro del tronco (cm)")

legend( 1.5, 27, levels(arbolado$Centro),
       bty = "n", pch = c(19,19,19,19), col = colors)

```

Estatura de árboles vs. diametro del tronco



E. Estructuras de control

Funciones

Sintáxis general de definición de las funciones

```
nombre_de_función <- function(variables_de_entrada) {
  cuerpo de función donde se usan las variables de entrada y se genera valor de
  salida
}
```

Existen la dos formas de definición de las funciones:

- a) en una sola línea
- b) con el bloque encerrado en los símbolos {}

Las funciones pueden ser anónimas o tener el nombre

```
# función para quitar los espacios en principio y al final de una cadena de texto
trim <- function(x) gsub("^\\s+|\\s+$", "", x)

# función para poner primera letra de una cadena de texto como mayuscula y otras como minusculas
simpleCap <- function(x) {
  paste(toupper(substr(x, 1, 1)), tolower(substr(x, 2, nchar(x))), sep="")
}
```

Aplicacion de las funciones para depurar nombres de las especies

```
lista_especies <- unique(as.character(arbolado$Especie))
length(lista_especies)
```

```
## [1] 150
```

```
arbolado$Especie <- sapply(as.character(arbolado$Especie), simpleCap)
arbolado$Especie <- sapply(arbolado$Especie, trim)
lista_depurada_especies <- unique(arbolado$Especie)
arbolado$Especie <- as.factor(arbolado$Especie)

length(lista_depurada_especies)
```

```
## [1] 144
```

Elementos de control de flujo de ejecución

1. if , else , else if - estructuras condicionales
2. for - ciclos (loops, bucles) con un número predeterminado de iteraciones
3. while - ciclos con un número indefinido de iteraciones
4. repeat - ciclos con un número infinito de iteraciones (se usan rara vez, principalmente en elementos de interface de usuario)
5. next , return - elementos que permiten interrumpir o saltar iteraciones en los ciclos

Sintaxis general de las estructuras condicionales

```

if(condición lógica) {
  código para caso que la condición se cumple
} else {
  código para caso que la condición no se cumple
}

```

Sintáxis general de los ciclos con numero predeterminado de elementos

```

for(iterador in rango_de_posibles_valores_del_iterador) {
  código del cuerpo de ciclo, donde se puede utilizar iterador como una variable
}

```

Ejemplo:

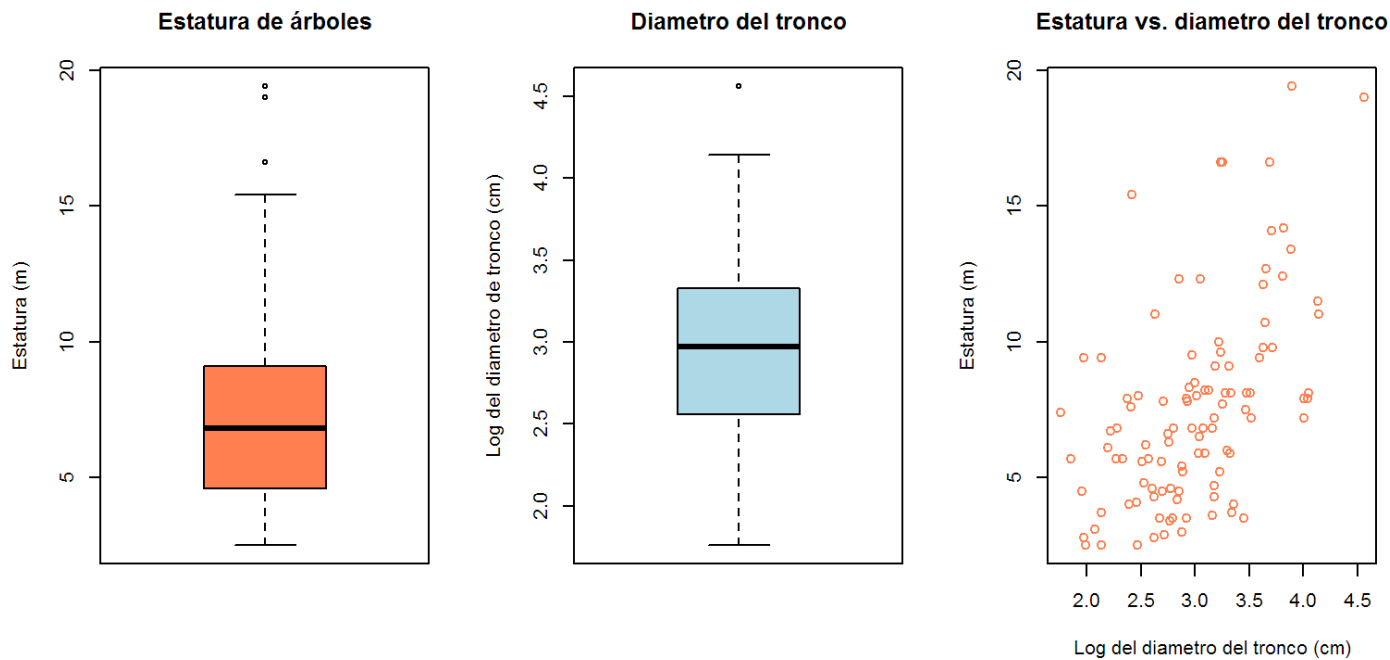
```

for(i in 1:length(lista_depurada_especies)){
  # seleccionar registros que corresponden a una especie determinada
  tabla_especie <- arbolado[arbolado$Especie == lista_depurada_especies[i],]
  # verificar que número de árboles de la especie es mayor que 50
  if(dim(tabla_especie)[1] > 50) {
    # imprimir nombre de especie y número de árboles
    cat(paste("<h4><i>",lista_depurada_especies[i],"</i></h4>"))
    cat(paste('<pre class="r"><code>', "n =",dim(tabla_especie)[1],"</code></pre>'))
    # especificar lienzo para tres graficas
    par(mfcol = c(1, 3))
    # dibujar tres gráficas
    boxplot(tabla_especie$Alt,
            col = "coral", main = "Estatura de árboles", ylab = "Estatura (m)")
    boxplot(log(tabla_especie$DTr),
            col = "lightblue", main = "Diametro del tronco", ylab = "Log del diametro de tronco (c
m)")
    plot(tabla_especie$Alt ~ log(tabla_especie$DTr),
         col = "coral", main = "Estatura vs. diametro del tronco",
         ylab = "Estatura (m)", xlab = "Log del diametro del tronco (cm)")
  }
}

```

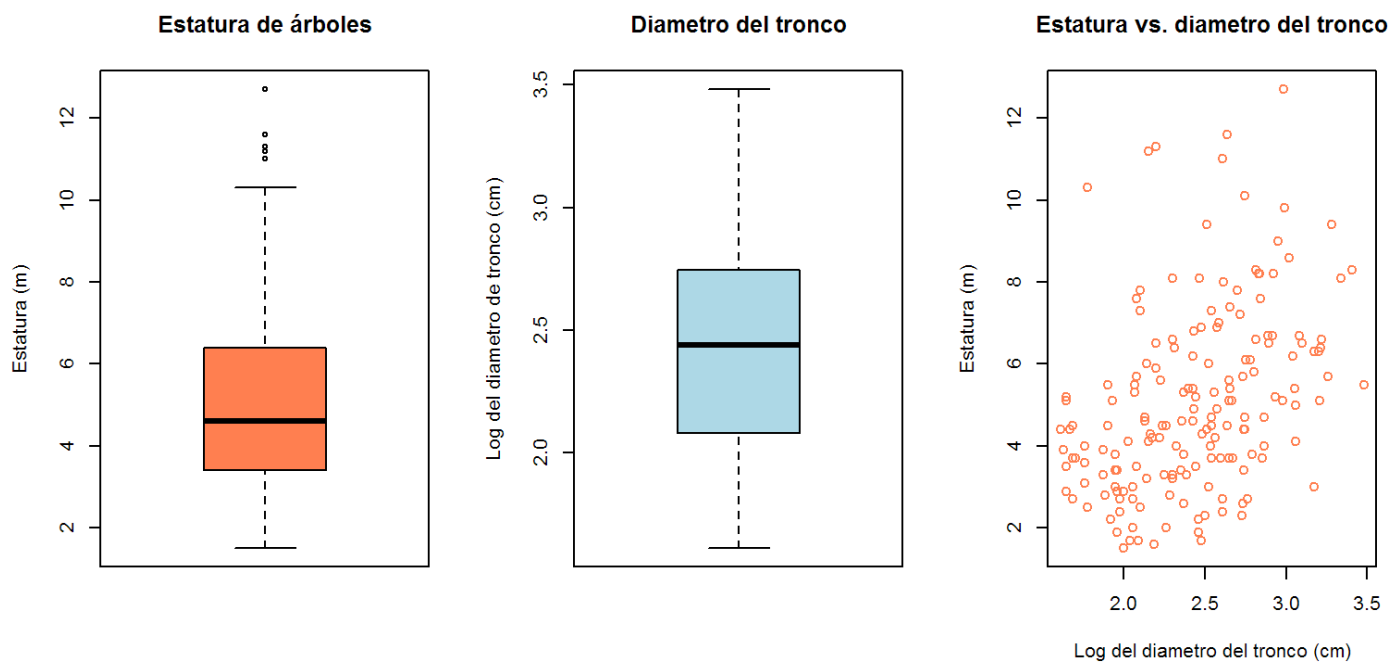
Pithecellobium dulce

n = 107



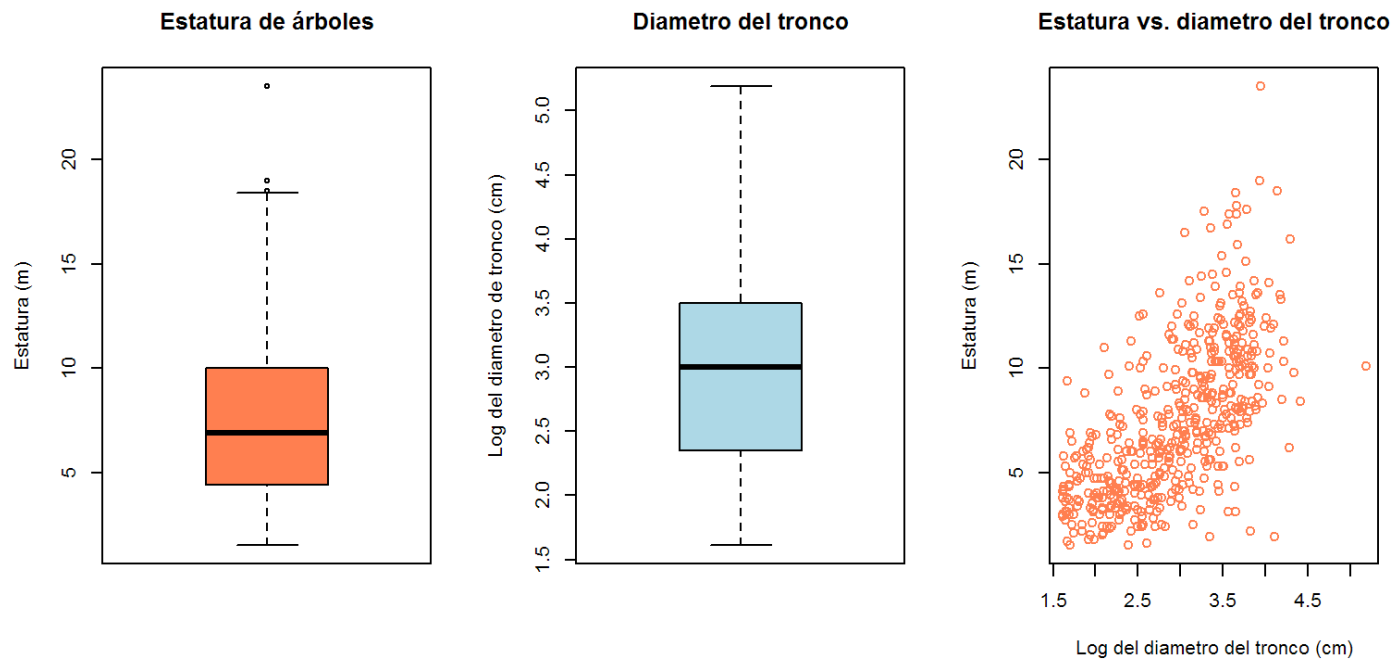
Acacia macracantha

n = 176



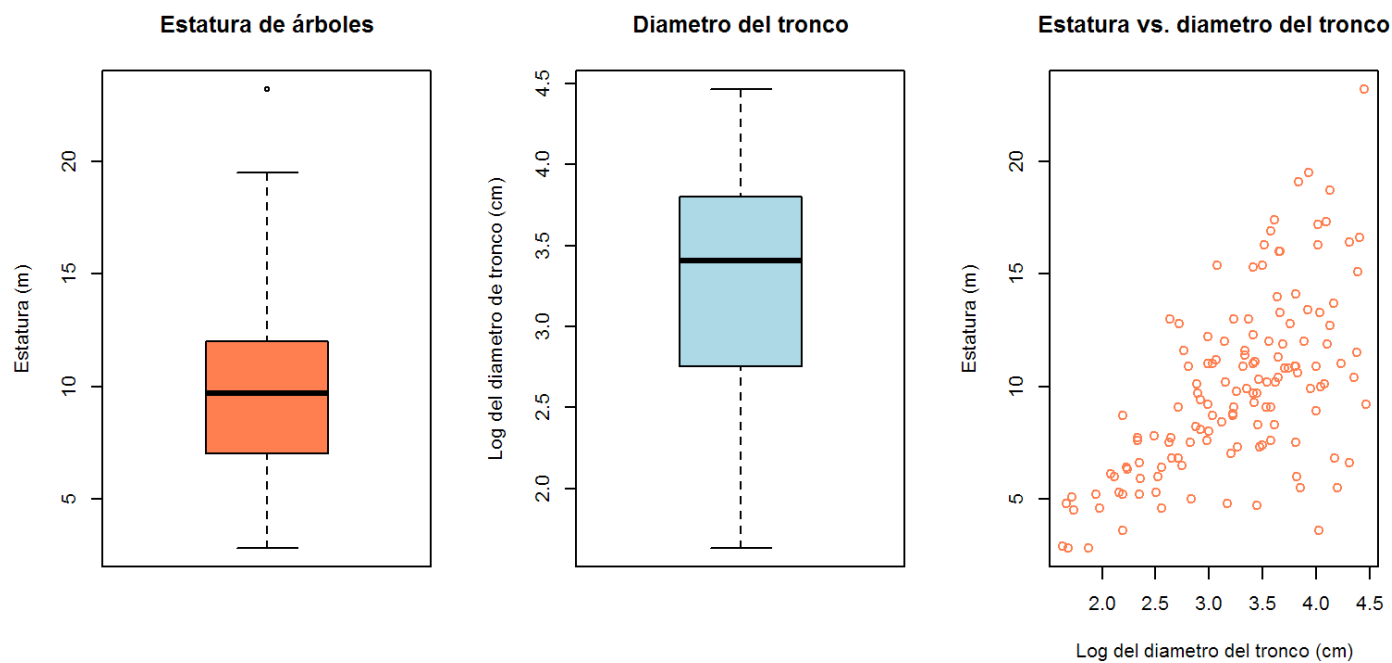
Pithecellobium lanceolatum

n = 501



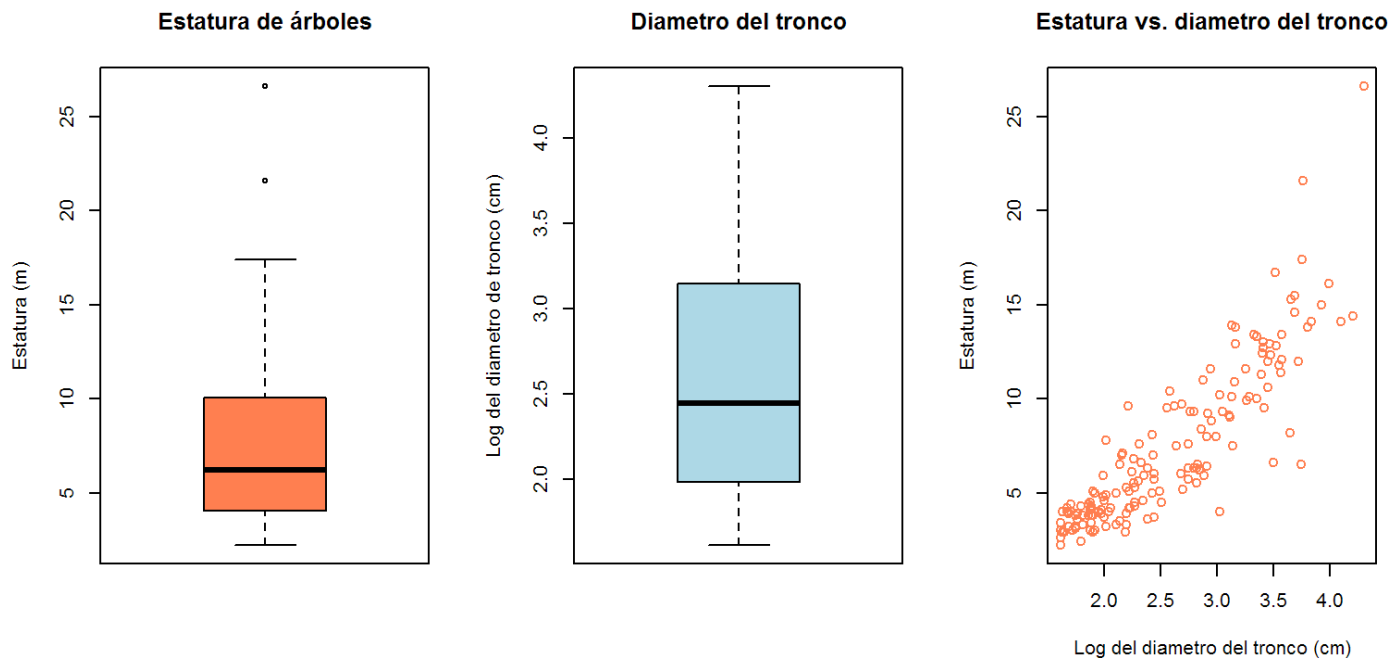
Salix bonplandiana

n = 139



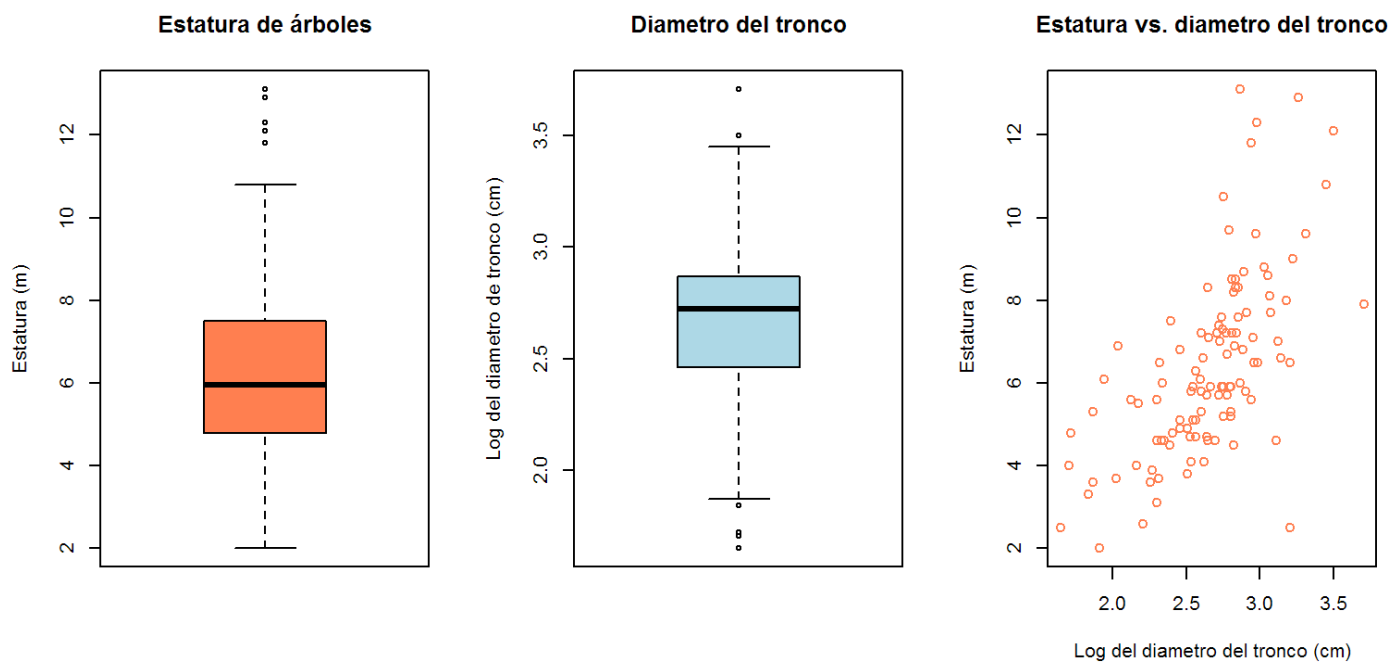
Tabebuia rosea

n = 159



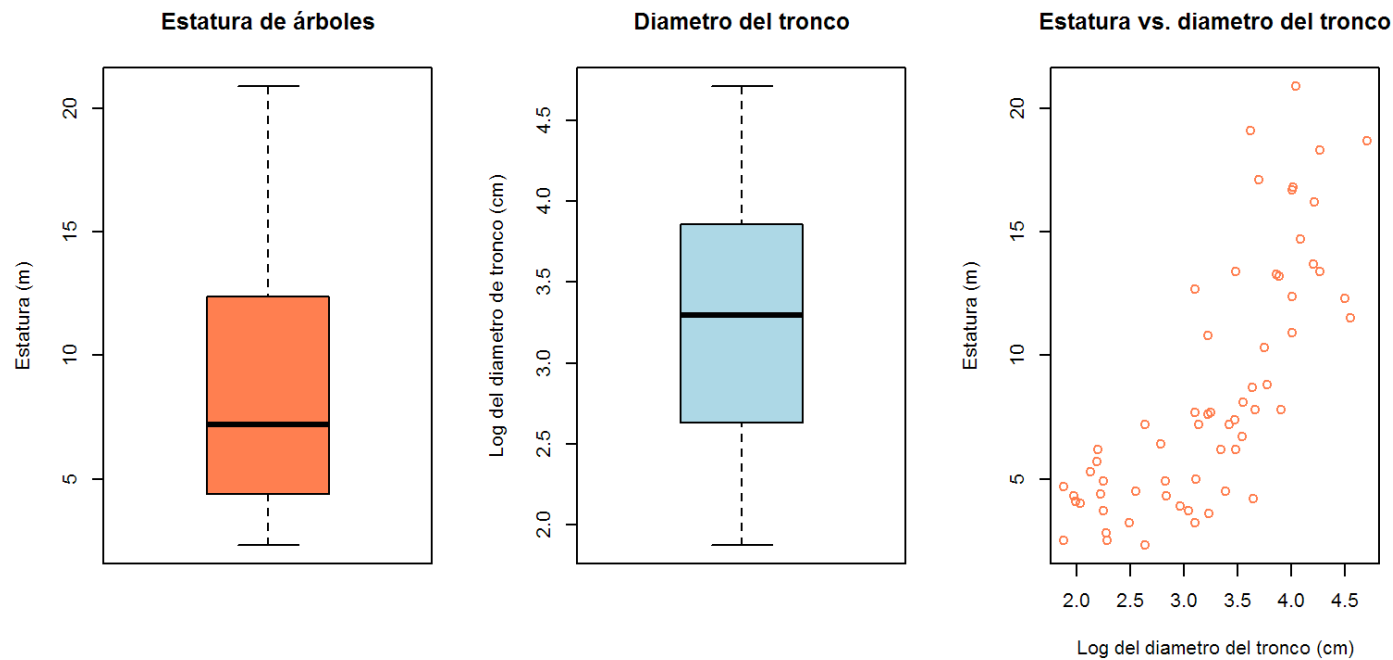
Terminalia catappa

n = 110



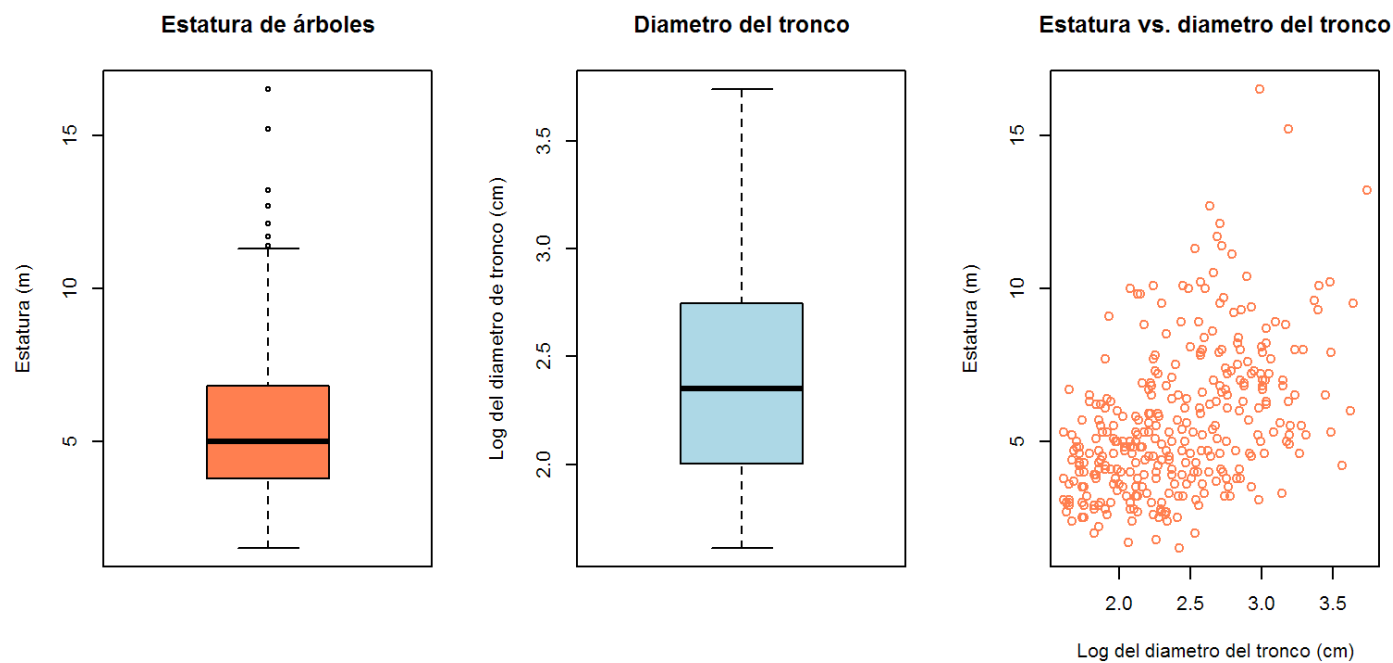
Enterolobium cyclocarpum

n = 62



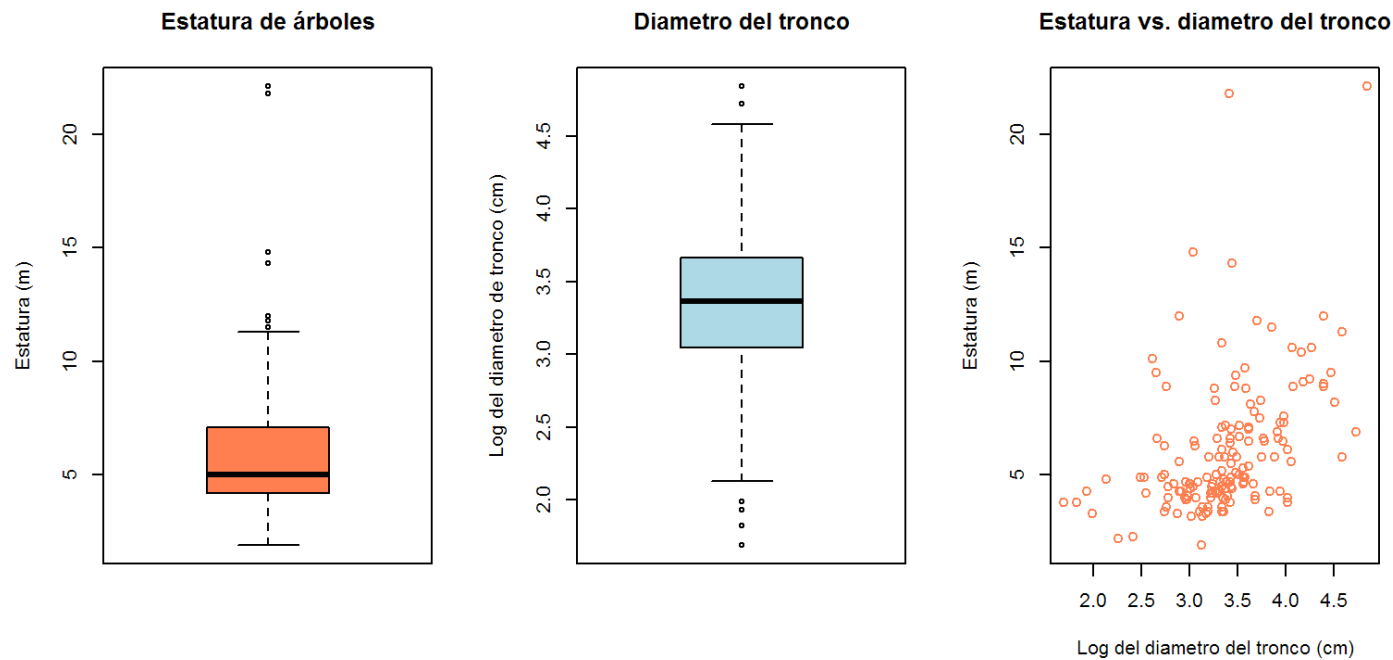
Guazuma ulmifolia

n = 329



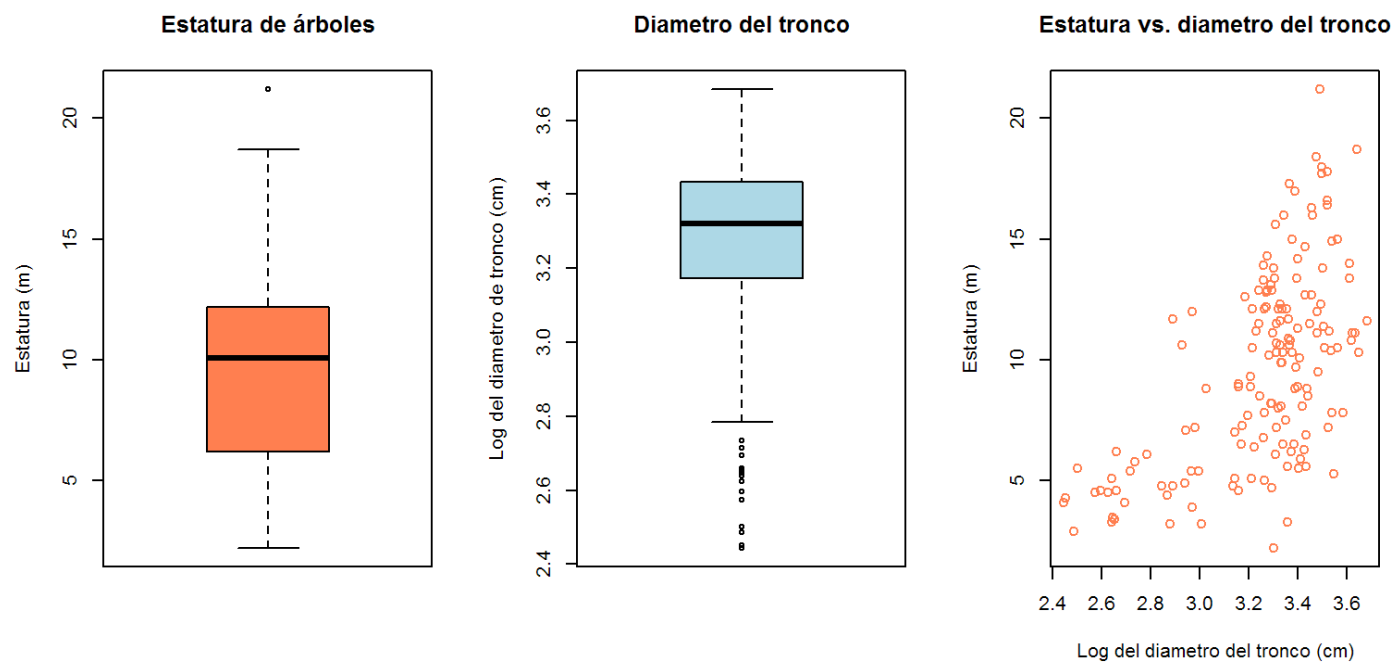
Ficus benjamina

n = 153



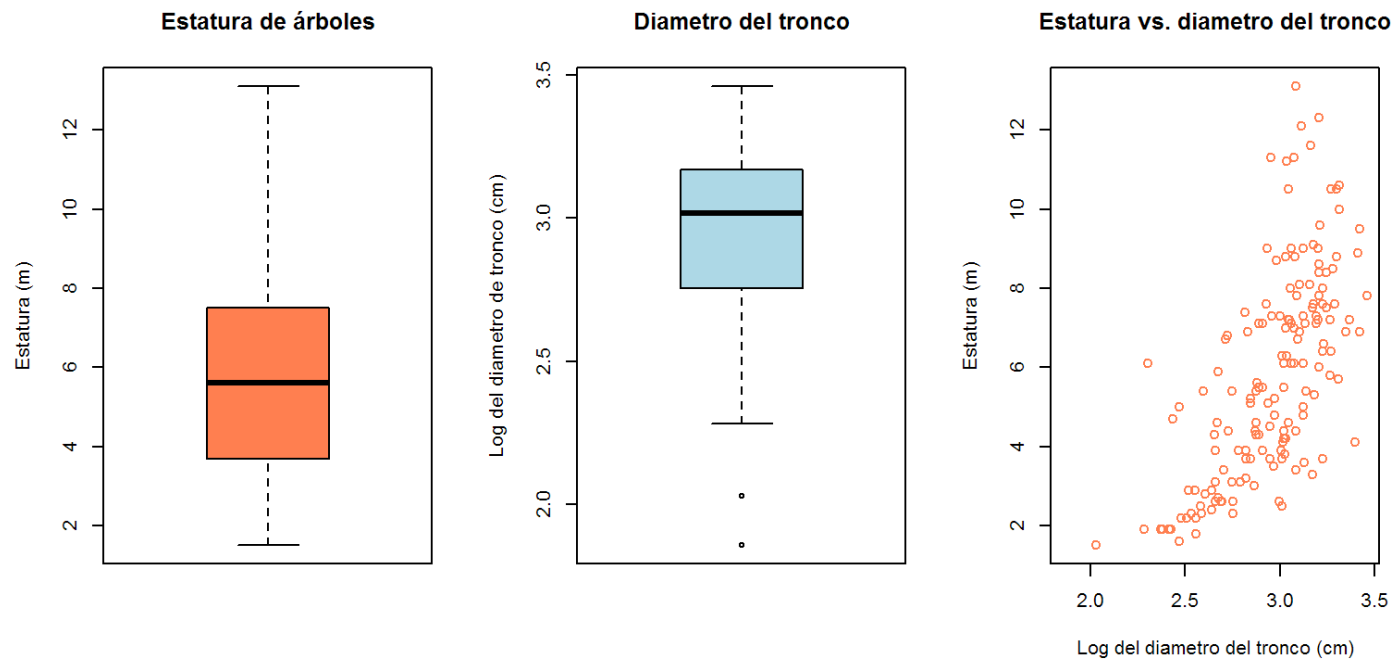
Cocus nucifera

n = 157



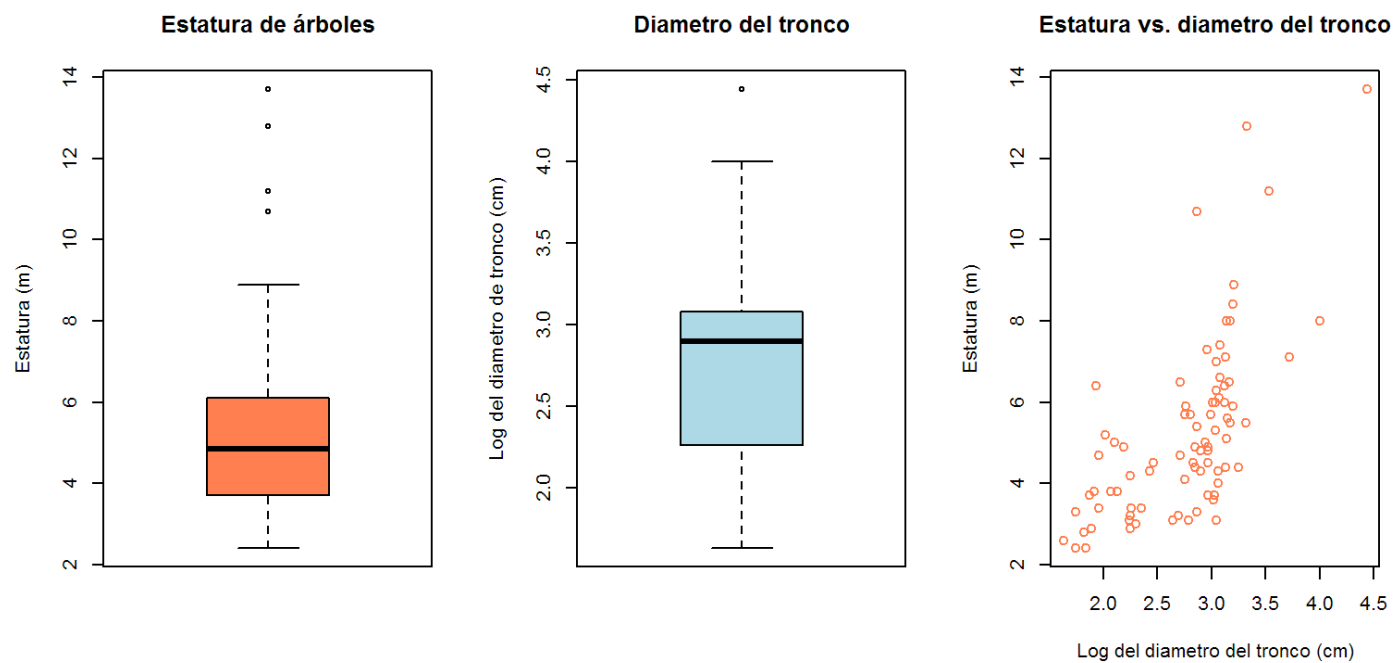
Arecastrum romanzoffianum

n = 162



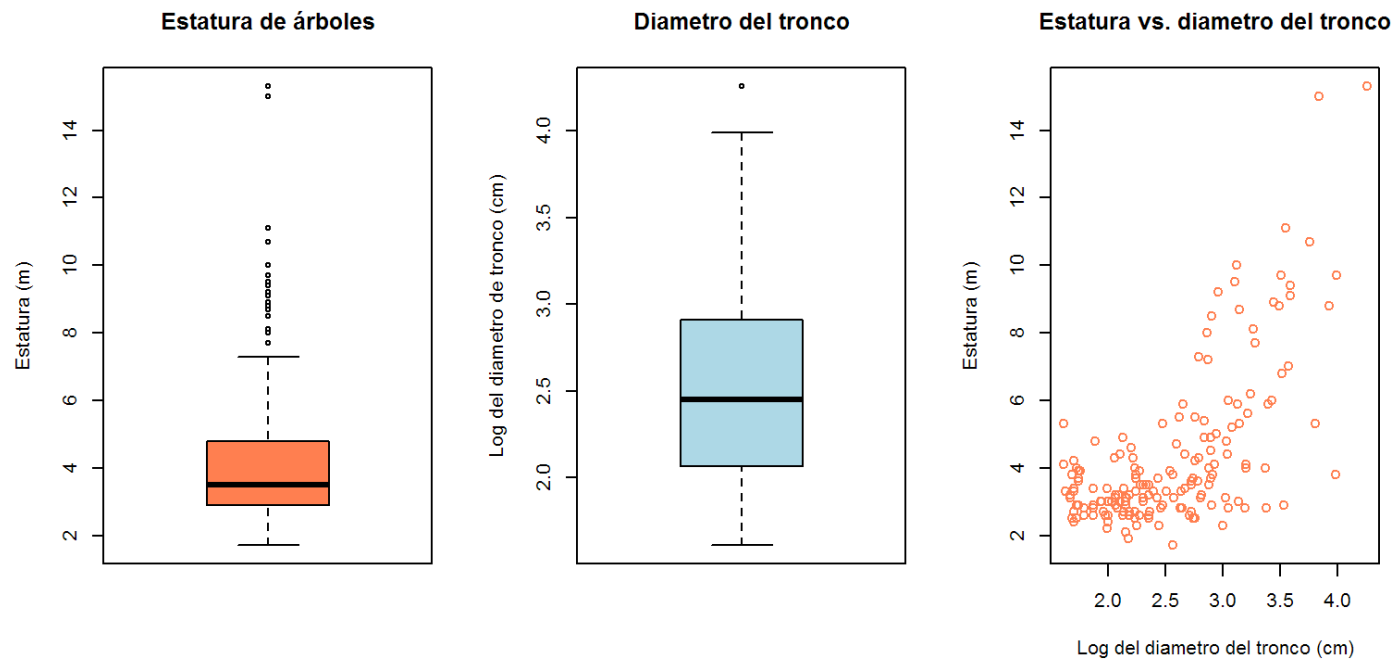
Dypsis lutescens

n = 82



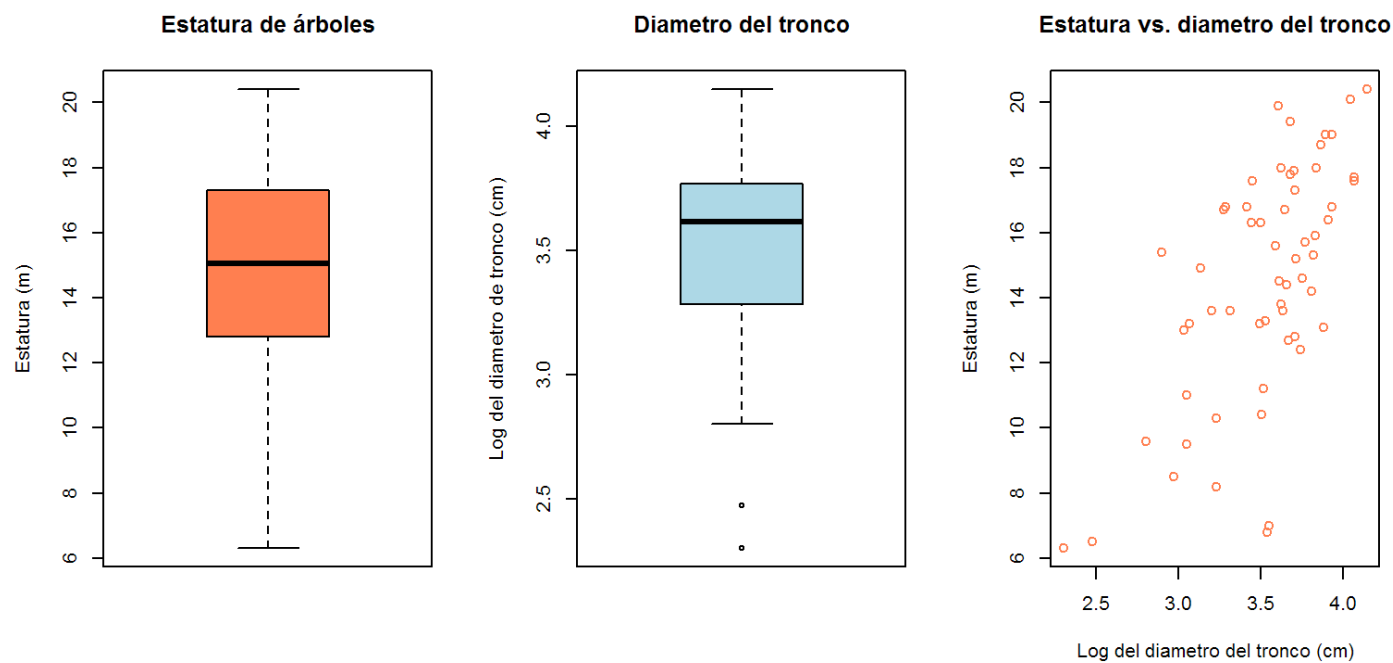
Acacia farnesiana

n = 174



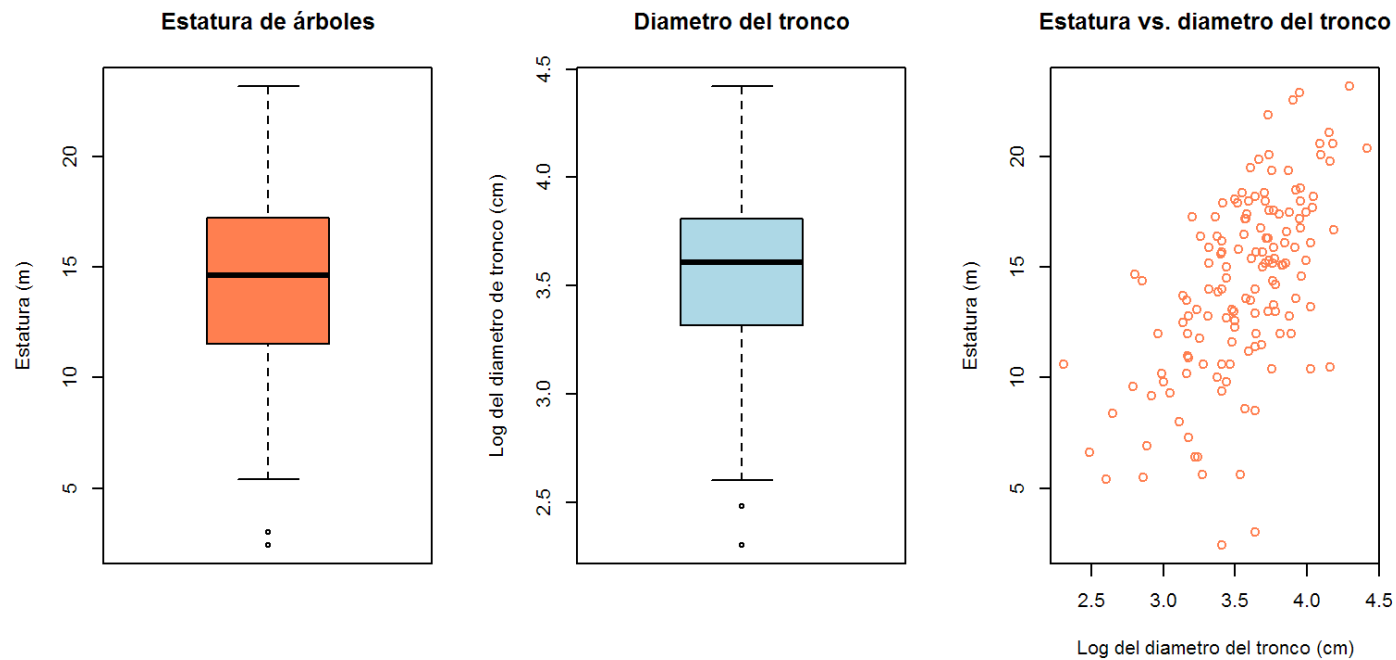
Eucalyptus camaldulensis

n = 58



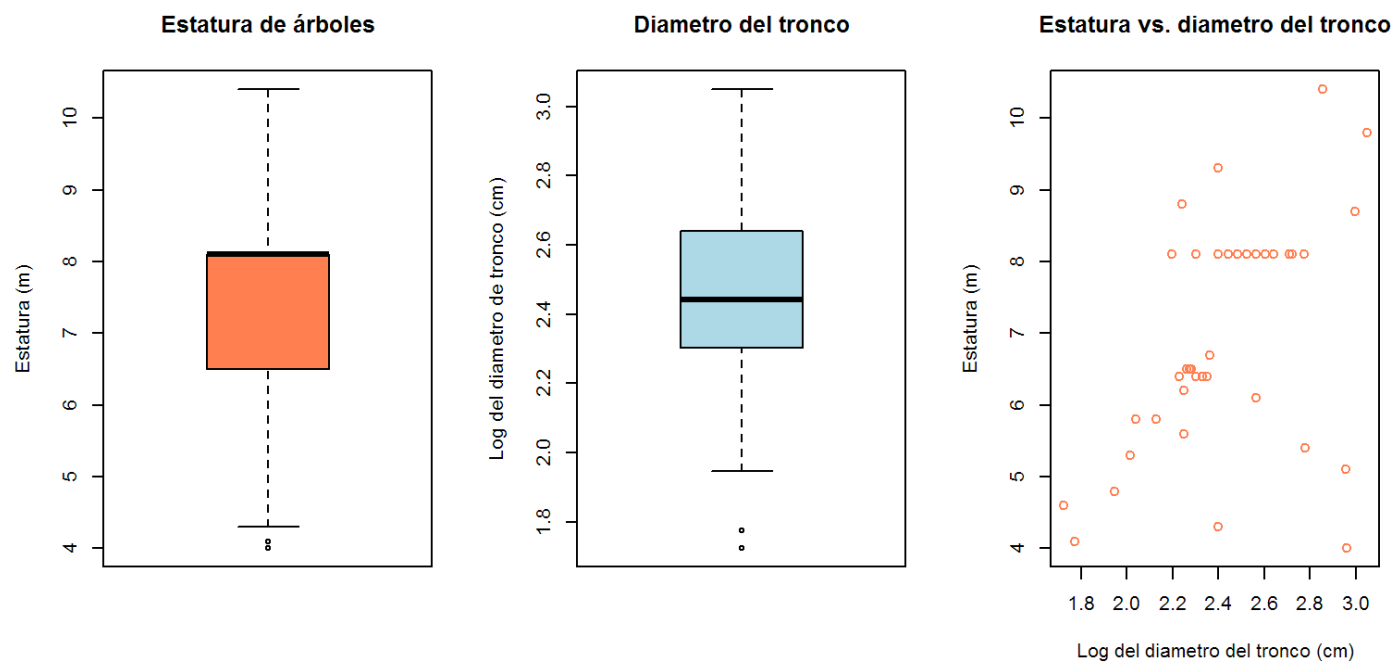
Casuarina equisetifolia

n = 145



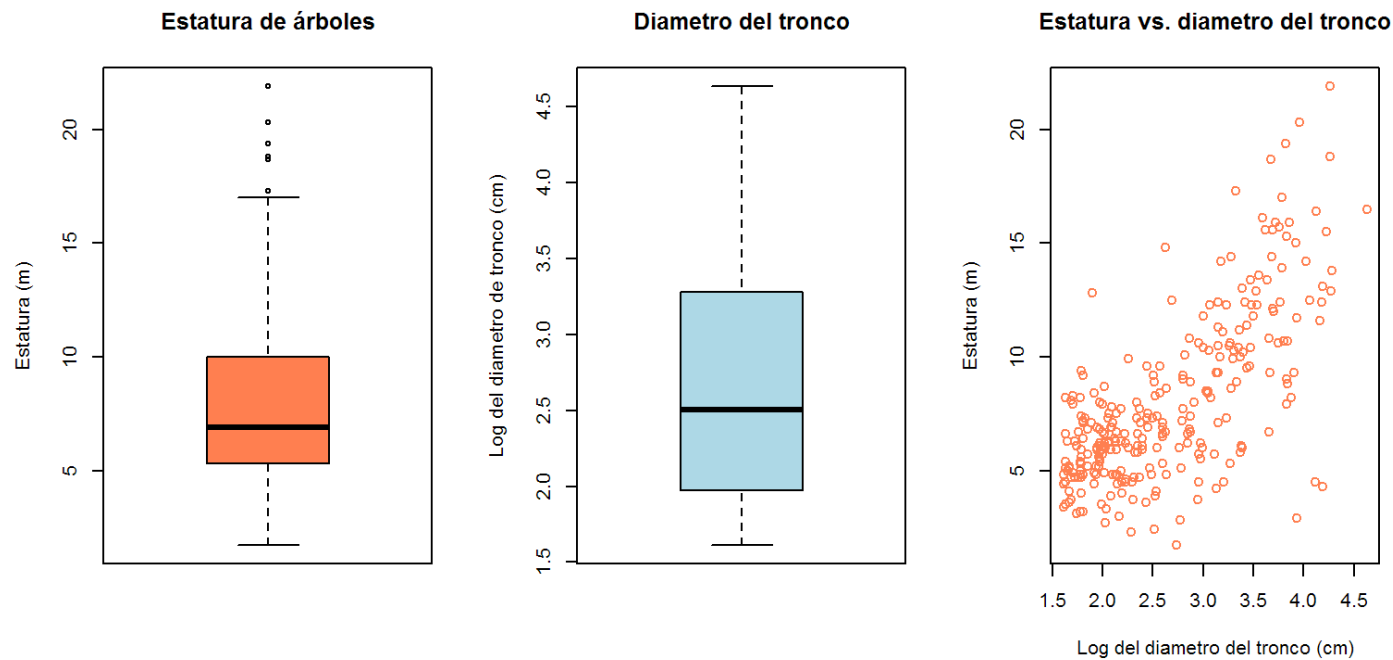
Cupressus sempervirens

n = 77



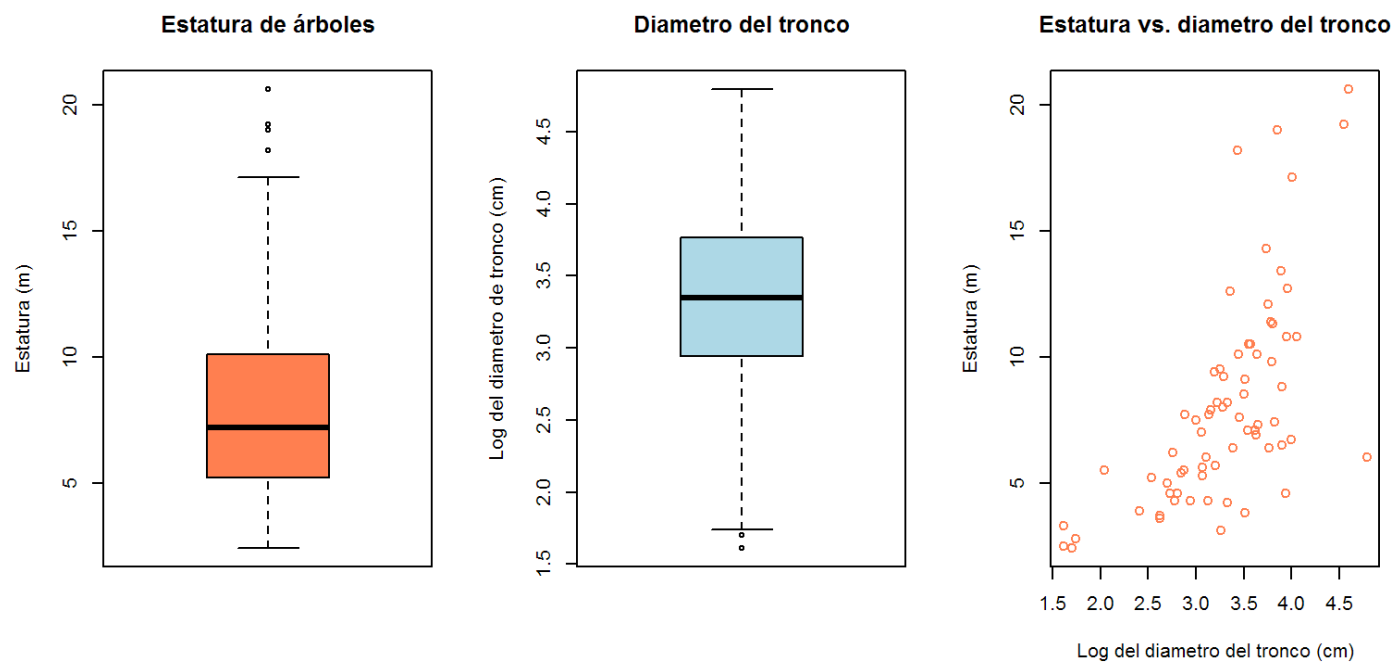
Fraxinus uhdei

n = 278



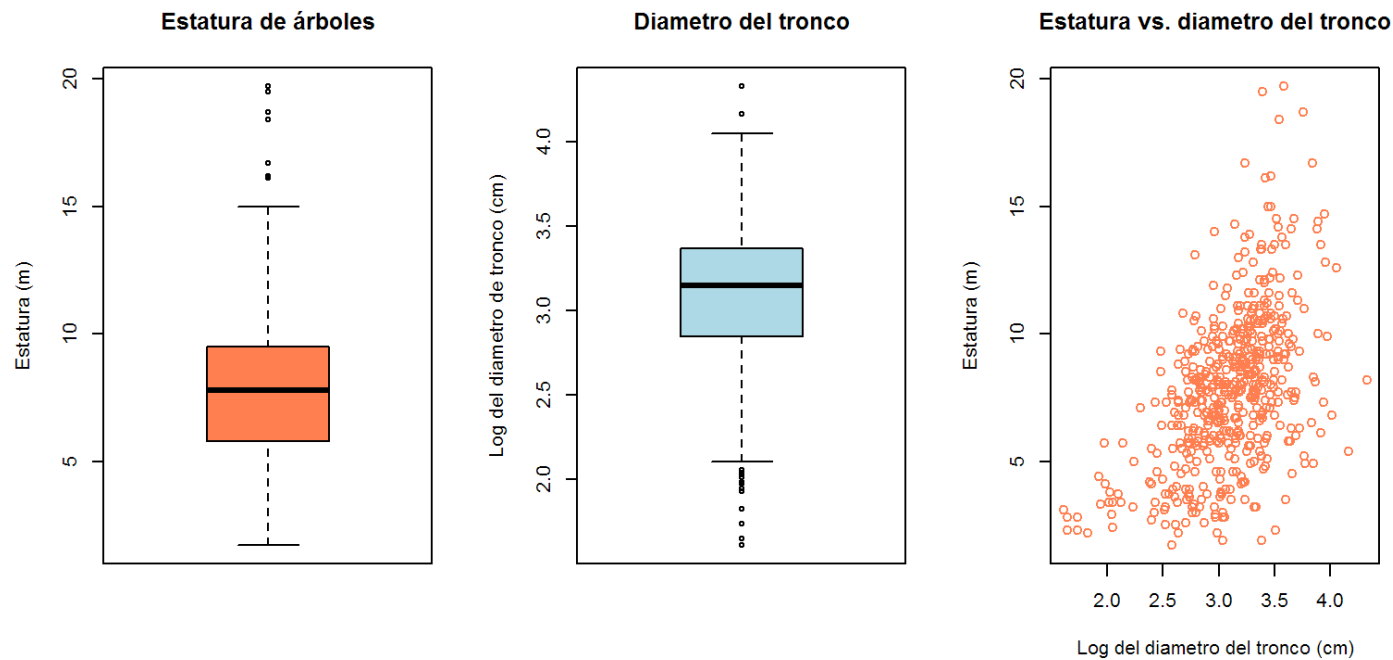
Jacaranda mimosifolia

n = 70



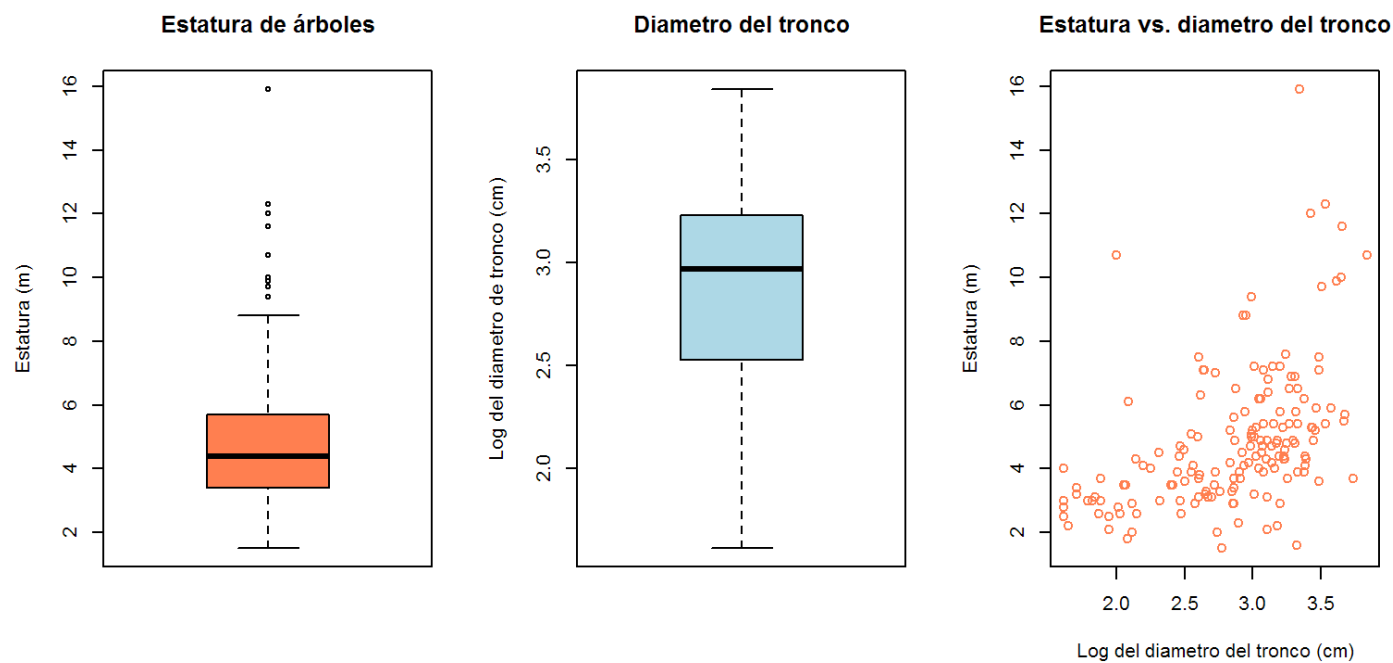
Quercus resinosa

n = 530



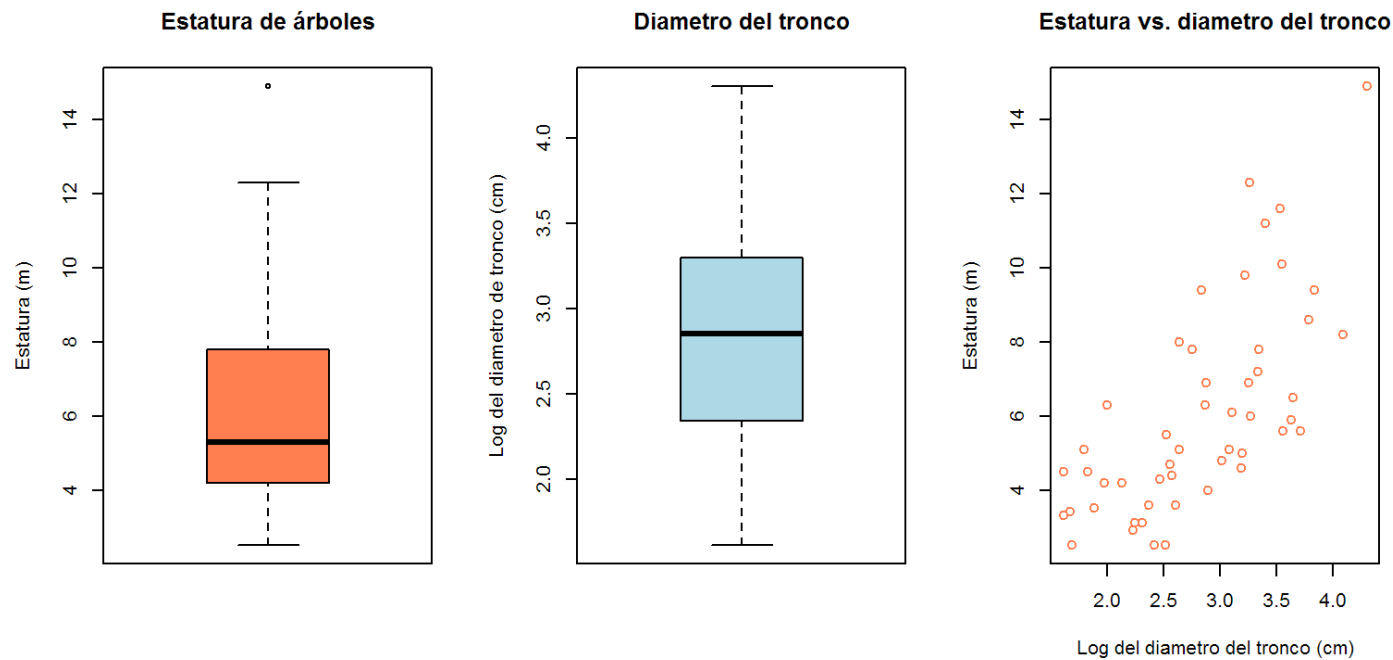
Acacia pennatula

n = 167



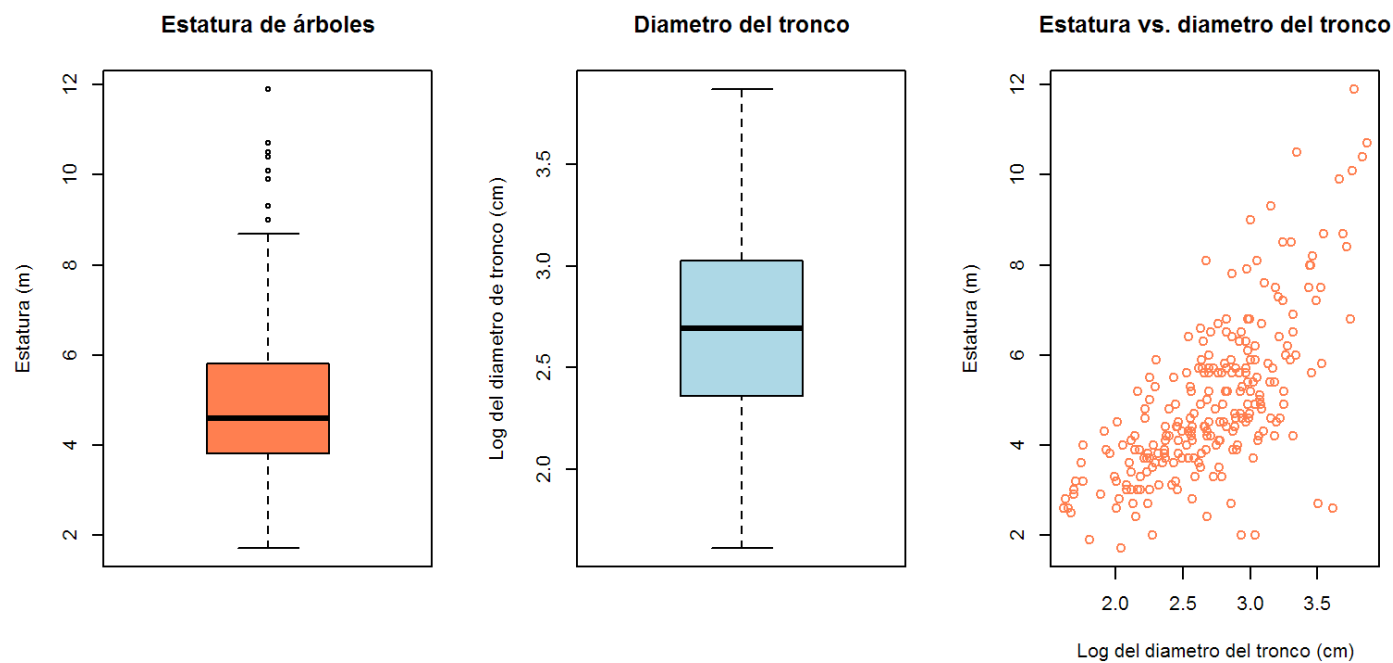
Lippia umbellata

n = 52



Eysenhardtia polystachya

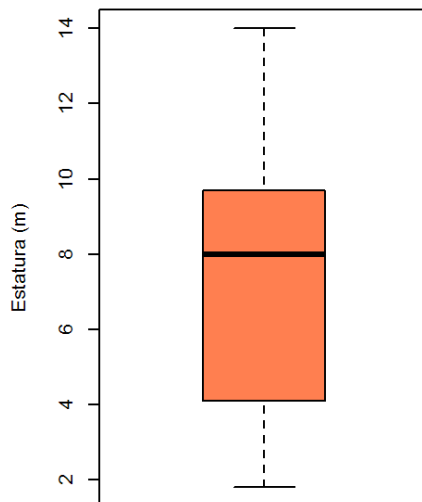
n = 245



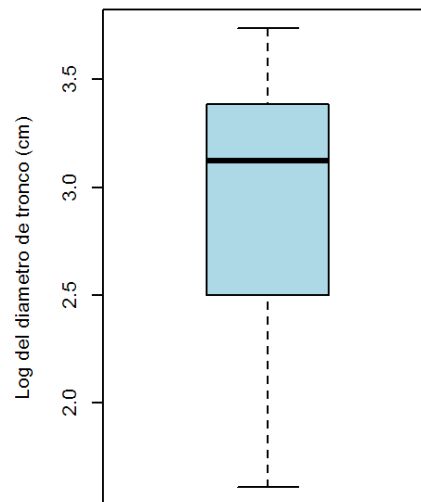
Pinus devoniana

n = 63

Estatura de árboles



Diametro del tronco



Estatura vs. diametro del tronco

