

# Análisis Exploratorio

Mariela Sued, Jemina García y Ana M. Bianco

08 abril 2020

# Aprender de los datos:

- \* La idea es visualizar en forma rápida las principales características del conjunto de datos, analizando posibles relaciones o conexiones entre ciertas variables.
- ▶ La relevancia que tiene hacer un análisis gráfico previo es fundamental, pero no siempre es determinante. Si bien una gráfica facilita la visualización de relaciones entre variables, esto no lo confirma.

# Visualizaciones interesantes

- ▶ visiten: (<https://gisanddata.maps.arcgis.com/apps/opsdashboard/index.html#/bda7594740fd40299423467b48e9ecf6>)
- ▶ y también  
([https://www.gapminder.org/tools/#\\$chart-type=bubbles](https://www.gapminder.org/tools/#$chart-type=bubbles))

# Borramos todo y establecemos el directorio de trabajo

```
rm(list=ls())
```

```
setwd("C:/Users/Ana/Dropbox/Ciencias_de Datos_Fundamentos/optativa_2020/Clase_1")  
#setwd("~/Dropbox/Ciencias_de Datos_Fundamentos/optativa_2020/Clase_1")
```

# Datos

Carguemos los datos del Titanic que están en formato CSV (comma separated values)

```
titanico<- read.csv("datos_titanic.csv",header=T)
head(titanico)
```

```
## PassengerId Survived Pclass
## 1          1         0       3
## 2          2         1       1
## 3          3         1       3
## 4          4         1       1
## 5          5         0       3
## 6          6         0       3

##                               Name    Sex Age SibSp
## 1                        Braund, Mr. Owen Harris   male  22     1
## 2 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female  38     1
## 3                        Heikkinen, Miss. Laina female  26     0
## 4 Futrelle, Mrs. Jacques Heath (Lily May Peel) female  35     1
## 5                        Allen, Mr. William Henry   male  35     0
## 6                        Moran, Mr. James         male  NA     0

## Parch      Ticket    Fare Cabin Embarked
## 1    0      A/5 21171  7.2500      S
## 2    0      PC 17599 71.2833   C85      C
## 3    0 STON/O2. 3101282  7.9250      S
## 4    0      113803 53.1000  C123      S
## 5    0      373450  8.0500      S
## 6    0      330877  8.4583      Q
```

```
attach(titanico)
```

# Tipos de Variables

Primero comenzamos a trabajar con las variables en forma individual.

Debemos identificar el tipo de variable con la que vamos a trabajar ya que dependiendo del tipo de variables que tengamos, se determina que análisis o tratamiento a aplicar.

Los tipos de datos pueden ser: numérico, entero, caracter, factor y lógico

# Variables Categóricas

```
class(Sex)
```

```
## [1] "factor"
```

Es importante que la clase de las variables categóricas sean factor. Sino deberíamos cambiarlo

El comando *table* aplicado a una variable categórica cuenta la frecuencia de cada categoría

```
table(Sex)
```

```
## Sex
```

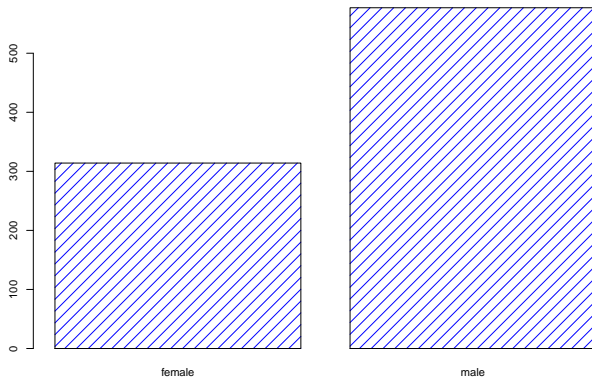
```
## female    male
```

```
##      314      577
```

*barplot* realiza un diagrama de barras de una variable categórica, para la que hemos calculado antes la tabla de frecuencia con *table*

# Variables Categóricas

```
counts.sexo <- table(Sex)  
barplot(counts.sexo,col="blue",density=8)
```





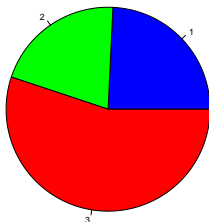
# Variables Categóricas

```
Pclass<-as.factor(Pclass)
counts.clase<- table(Pclass)
counts.clase
```

```
## Pclass
##    1    2    3
## 216 184 491
```

```
pie(counts.clase, col=c("blue","green","red"),
     main="Grafico de Torta de Clases")
```

Grafico de Torta de Clases

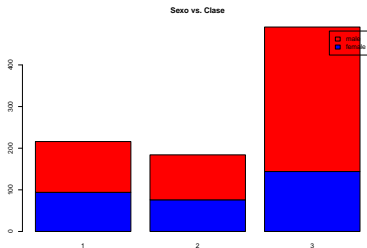


# Más de una Variable Categórica

```
counts<- table(Sex,Pclass)
counts
```

```
##           Pclass
## Sex         1     2     3
##  female    94    76   144
##   male    122   108   347
```

```
barplot(counts,col= c( " blue " , "red " ),main="Sexo vs. Clase",
        legend = rownames( counts ) )
```



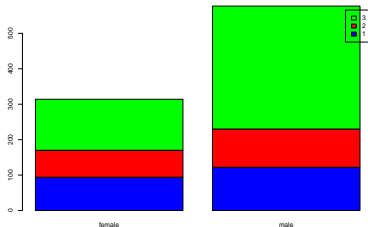
# Más de una Variable Categórica

De otro modo

```
counts<- table(Pclass,Sex)  
counts
```

```
##      Sex  
## Pclass female male  
##      1      94  122  
##      2      76  108  
##      3     144  347
```

```
barplot(counts,col= c( " blue " , "red ", "green"),legend = rownames( counts ) )
```



# Carguemos los datos de iris

```
data(iris)  
#Inspecciono los primeros y los ultimos casos.  
head(iris)
```

##	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
## 1	5.1	3.5	1.4	0.2	setosa
## 2	4.9	3.0	1.4	0.2	setosa
## 3	4.7	3.2	1.3	0.2	setosa
## 4	4.6	3.1	1.5	0.2	setosa
## 5	5.0	3.6	1.4	0.2	setosa
## 6	5.4	3.9	1.7	0.4	setosa

```
attach(iris)
```

## Variables numéricas

Al tratar una variable cuantitativa podríamos calcularle cualquiera de las medidas tales como promedio, desvío standard, etc.

Si la variables tiene observaciones faltantes las funciones no se van a ejecutar hay que decirle a la función que las omita.

```
range(Sepal.Length)
```

```
## [1] 4.3 7.9
```

```
mean(Sepal.Length)
```

```
## [1] 5.843333
```

```
summary(Sepal.Length)
```

##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	4.300	5.100	5.800	5.843	6.400	7.900

```
#promedio omitiendo los valores missings.
```

## Medidas de Resumen Comunes

Nombre del comando	Explicación
summary(data)	Resumen estadístico
min(data)	Mínimo
max(data)	Máximo
range(data)	Rango
mean(data)	Media aritmética
median(data)	Mediana
length(data)	Tamaño
sd(data)	Desviación típica
var(data), cov(data)	Varianza
cor(data)	Correlación
sort(data)	Ordenar
table(data)	Tabla de frecuencias absolutas

# Medidas de Resumen Comunes

## Un poco de matrices en R

Una matriz es una estructura bidimensional con dos dimensiones: filas y columnas. Podríamos crear una matriz de la siguiente forma indicando número de filas y de columnas.

```
A<- matrix(1:10, nrow = 2, ncol = 5)
```

A

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    3    5    7    9
## [2,]    2    4    6    8   10
```

o bien

```
B<- matrix(1:8, nrow = 4, ncol = 2)
```

B

```
##      [,1] [,2]
## [1,]    1    5
## [2,]    2    6
## [3,]    3    7
## [4,]    4    8
```



## Dimensión e indexación

Para conocer la dimension de una matriz tenemos el comando `dim()`

```
dim(A)
```

```
## [1] 2 5
```

Para referirnos a una posición de una matriz, indicamos fila y columna:

```
A[1,5]
```

```
## [1] 9
```

```
A[2,] # da la fila 2 de A
```

```
## [1] 2 4 6 8 10
```

```
A[,3] # da la columna 3 de A
```

```
## [1] 5 6
```

# Comandos cbind y rbind

Muchas veces creamos matrices pegando vectores por fila o por columna.

```
x<- 1:4 ; y<- 11:14
```

```
A<- cbind(x,y) # pego por columnas
```

```
B<-rbind(x,y) #pego por filas
```

```
A; dim(A)
```

```
##      x  y  
## [1,] 1 11  
## [2,] 2 12  
## [3,] 3 13  
## [4,] 4 14
```

```
## [1] 4 2
```

```
B; dim(B)
```

```
##    [,1] [,2] [,3] [,4]  
## x     1     2     3     4  
## y    11    12    13    14
```

```
## [1] 2 4
```

# Medidas de Resumen

Podemos calcular estas funciones también sobre un conjunto de variables

La siguiente matriz contiene las variables Sepal.Length Sepal.Width  
Petal.Length Petal.Width

```
SUB<-cbind( Sepal.Length, Sepal.Width, Petal.Length, Petal.Width )  
# o bien  
SUB<-iris[,-5]
```

Por ejemplo, puedo calcular la media

```
apply(SUB,2,mean,na.rm=TRUE)
```

```
## Sepal.Length Sepal.Width Petal.Length Petal.Width  
##      5.843333      3.057333      3.758000      1.199333
```

```
# 2 indica que debe aplicar mean por columna
```

# Histogramas

El histograma es el más conocido de los gráficos para resumir un conjunto de datos cuantitativos o numéricos.

- ▶ Para construir un histograma es necesario previamente construir una tabla de frecuencias: dividimos el rango de los  $n$  datos en intervalos o clases, que son excluyentes y exhaustivas.
- ▶ Contamos la cantidad de datos en cada intervalo o clase  $i$ , es decir la frecuencia,  $f_i$  y calculamos la frecuencia relativa:  $fr_i = f_i/n$
- ▶ Graficamos el histograma en un par de ejes coordenados representando en las abscisas los intervalos y sobre cada uno de ellos un rectángulo cuya área es la frecuencia relativa de dicho intervalo.

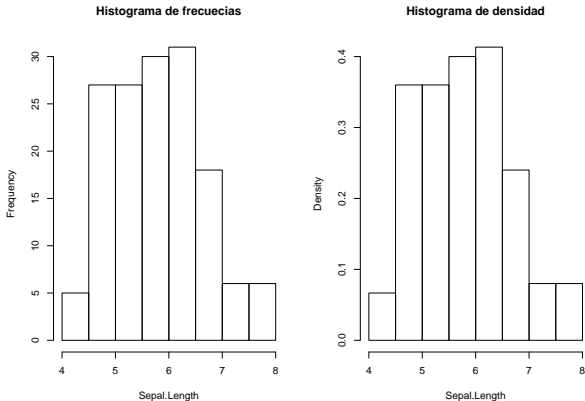
# Histogramas

```
hist(Sepal.Length)
```



# Histogramas

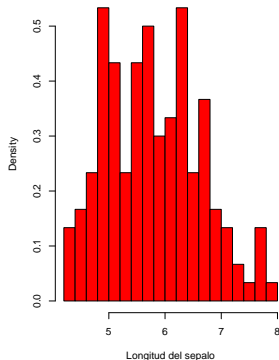
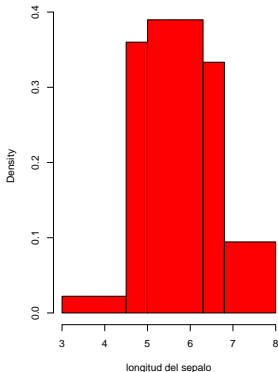
```
par(mfrow=c(1,2))  
hist(Sepal.Length,main="Histograma de frecuencias")  
hist(Sepal.Length,freq=F,main="Histograma de densidad")
```



```
par(mfrow=c(1,1))
```

# Histogramas

```
par(mfrow=c(1,2))  
hist(Sepal.Length,freq=F,col="2",  
      breaks = c(3,4.5,5,6.3,6.8,8),xlab="longitud del sepalo",main="")  
hist(Sepal.Length,freq=F,col="2",nclass=15,  
      xlab="Longitud del sepalo",main="")
```



# Histogramas

Para guardar los gráficos por líneas de comandos

```
pdf ("histogramas.pdf ")
par(mfrow=c(1,2))
hist(Sepal.Length,freq=F,col="2",breaks = c(3,4.5,5,6.3,6.8,8),
     xlab="longitud del sepalo",main="")
hist(Sepal.Length,freq=F,col="2",nclass=15,
     xlab="Longitud del sepalo",main="")
par(mfrow=c(1,1))
graphics.off()
```



# Histogramas

Histograma con la curva de densidad superpuesta

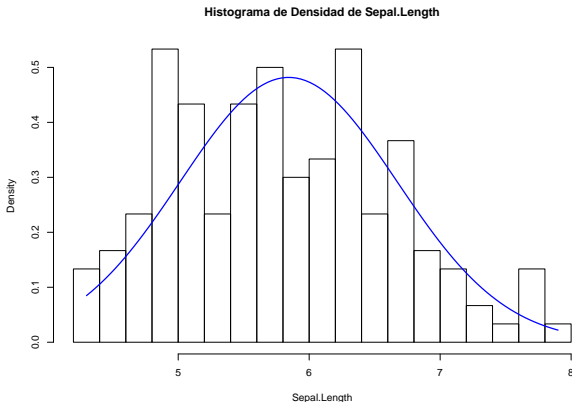
```
media.es<-mean(Sepal.Length)
desvio.es<-sd(Sepal.Length)
grilla<-seq(range(Sepal.Length)[1],
            range(Sepal.Length)[2],length=100)

# Calculamos la densidad normal sobre grilla
funn<-dnorm(grilla,media.es,desvio.es)
```

# Histogramas

Histograma con la curva de densidad superpuesta

```
hist(Sepal.Length, nclass=15, freq=F,  
     main="Histograma de Densidad de Sepal.Length")  
lines(grilla, funn, col="blue", lwd=2)
```



# Boxplot

Los cuartiles y la mediana dividen a la muestra en cuatro partes igualmente pobladas: 25% de la muestra en cada una de ellas.

Entre Q1 y Q2 se halla el 50% central de los datos y el rango de estos es  $IQR=Q3-Q1$ .

Observemos que porcentaje de datos hay a la izquierda de Q1 a la derecha de Q3 entre Q1 y Q3, Q1 y el máximo entre el mínimo y Q3.

Resultan muy útiles para describir la muestra las siguientes medidas

- ▶ Mínimo
- ▶ Q1 cuartil inferior
- ▶ Q2 mediana
- ▶ Q3 cuartil superior
- ▶ Máximo

# Boxplot

Nombre del comando	Explicación
<code>quantile(data, 0.25)</code>	Cuantil Q1
<code>quantile(data, 0.75)</code>	Cuantil Q3
<code>IQR(data)</code>	Rango intercuartil

# Boxplot

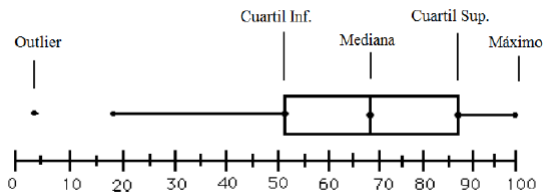


Figure: boxplot

# Boxplot

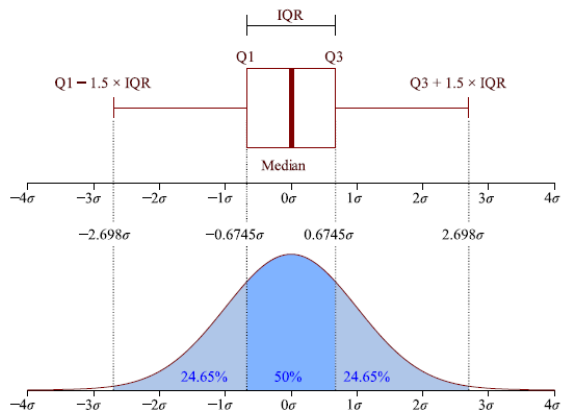
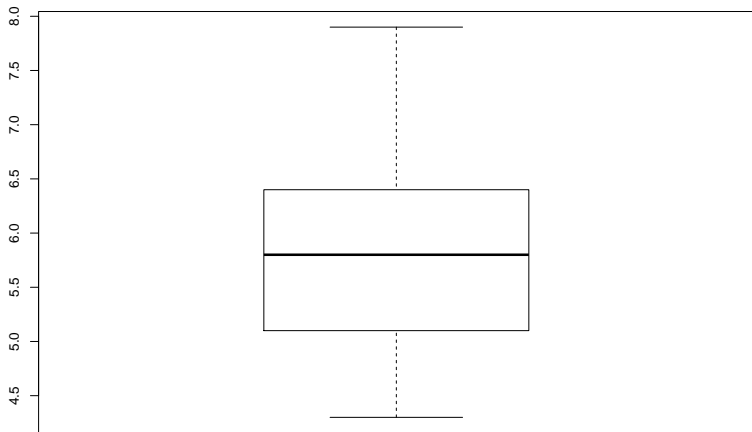


Figure: boxplot

# Boxplot

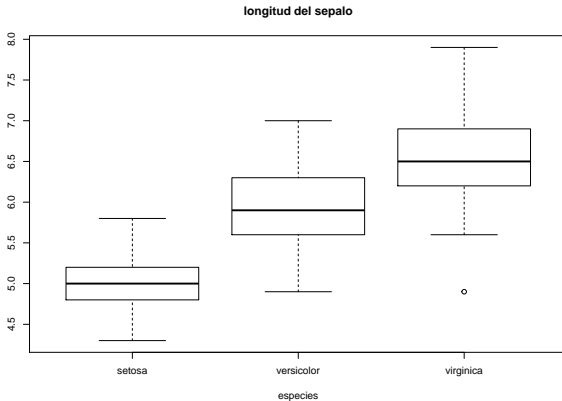
```
boxplot(Sepal.Length)
```



# Boxplot

El ~ indica que variable clasifica

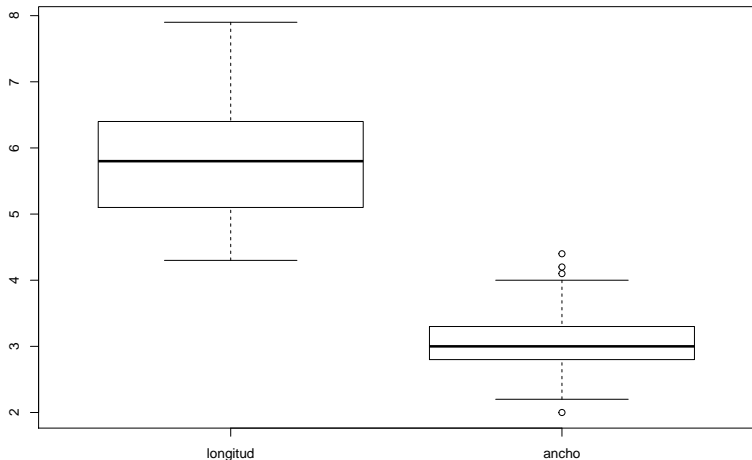
```
boxplot(Sepal.Length~Species, xlab="especies",  
        main="longitud del sepalo")
```





# Boxplot

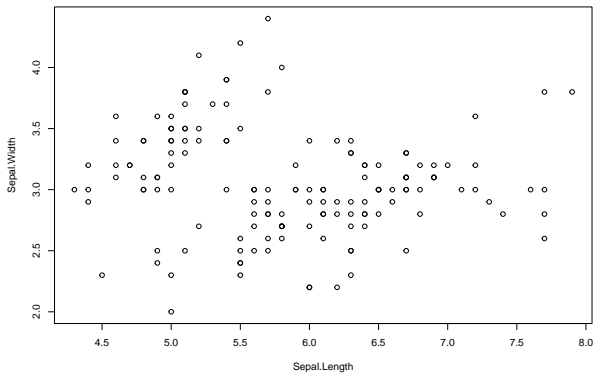
```
boxplot(Sepal.Length, Sepal.Width,  
        names=c("longitud", "ancho"))
```



# Gráficos para pares de variables

Diagrama de dispersión de Sepal.Length vs. Sepal.Width

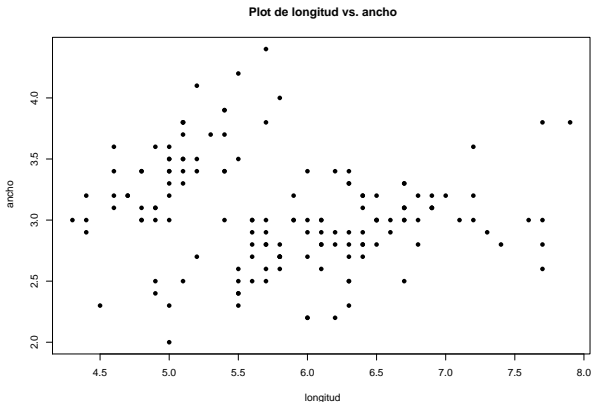
```
plot(Sepal.Length, Sepal.Width)
```



# Gráficos para pares de variables

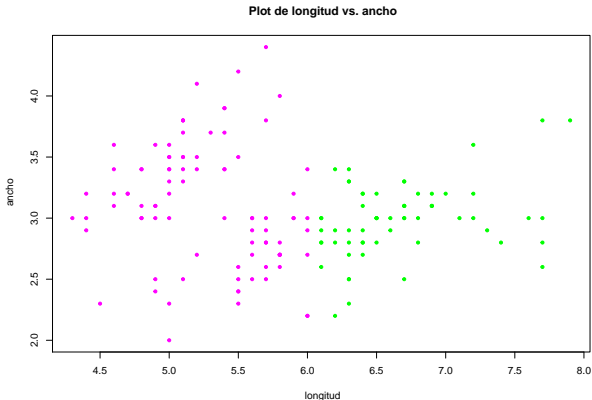
Podemos personalizarlo poniendo labels en cada eje, título y elija un caracter para los puntos.

```
plot(Sepal.Length,Sepal.Width,xlab =" longitud", ylab =" ancho",  
     main =" Plot de longitud vs. ancho",pch=16)
```



# Gráficos para pares de variables

```
plot(Sepal.Length,Sepal.Width,xlab = " longitud", ylab = " ancho",  
     main = " Plot de longitud vs. ancho",pch=16,type="n")  
#solo graficamos la caja  
points(Sepal.Length[Sepal.Length<=6],Sepal.Width[Sepal.Length<=6],  
       pch=20,col="magenta")  
points(Sepal.Length[Sepal.Length>6],Sepal.Width[Sepal.Length>6],  
       pch=20,col="green")
```



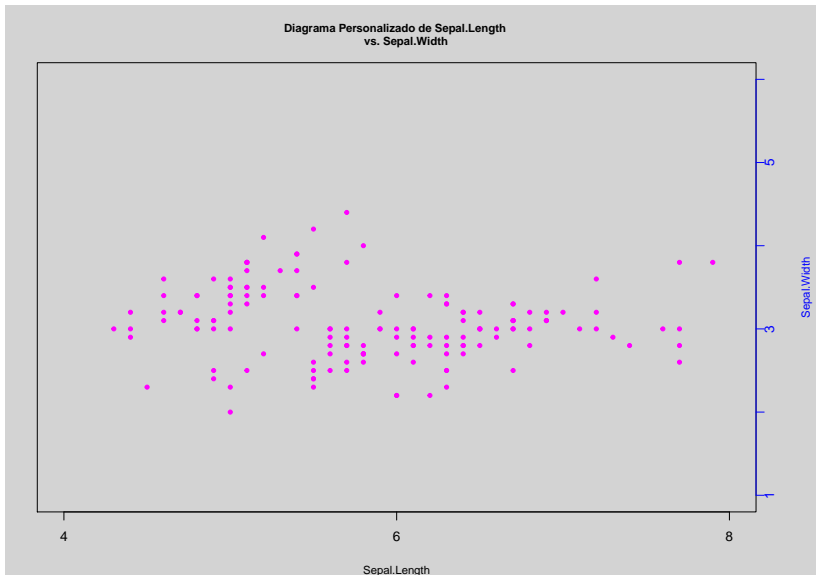
# Gráficos para pares de variables

Ahora personalizando el gráfico

```
par(bg="lightgray",mar=c(4,2,3.5, 4))
#c(bottom, left, top, right) defalut es c(5, 4, 4, 2) + 0.1.
plot(Sepal.Length,Sepal.Width,type="n",xlim=c(4,8),
      ylim=c(1,6),xlab="", ylab="",xaxt="n", yaxt="n")
#solo graficamos la caja
points(Sepal.Length,Sepal.Width,pch=20,col="magenta")

#solo graficamos los puntos con el simbolo deseado
#Ahora nos encargamos de los ejes
axis(1,c(4,6,8),cex=2)
mtext("Sepal.Length",side=1,cex=0.8,line=3)
axis(4,cex=0.8,col="blue",labels=FALSE)
mtext(c(1,3,5),side=4,at=c(1,3,5),col="blue",line=0.3)
mtext("Sepal.Width",side=4,cex=0.8,line=2.5,col="blue")
#titulo
title("Diagrama Personalizado de Sepal.Length
      vs. Sepal.Width",cex.main=0.8)
```

# Gráficos para pares de variables



# Gráficos para pares de variables

Diagrama de dispersión para todas las variables de SUB.

```
pairs(SUB,col="magenta")
```

