2. Programación y Data Science:

Data science y programación van de la mano, un programador debe´ria conocer de ciencia de datos pues estos se encargan de recolectar los datos a través de las interfaces y los datos son la materia prima para la data science. Se hace un recuento de la revolución industrial (las revoluciones industriales). Hoy se vive la 4° revolución industrial, IoT, AI, los datos se producen constantemente, las empresas nos conocen por los datos que producimos.
Big Data es muchísimo mayor a las 16 mil columnas.
Volumen. Velocidad. Variedad son los requisitos, manejamos muchos datos no estructurados. Es decir, no en columnas.
Se requieren conocimientos de matemáticas estadística, programación, de negocio y contexto, visualización y comunicación. Responder preguntas a través de los datos.

3. R y proyecto economía naranja:

Para la ciencia de datos es común utilizar dos lenguajes: R y Python.

En este curso veremos R, un lenguaje especializado en manejar datos de manera estadística creado en 1993 en la universidad de Auckland Nueva Zelanda. A lo largo del curso veremos:

- Estructuras tipos de datos
- EDA(Exploratory data análisis)
- Estadistica descriptiva
- Ajuste de datos: Subsetting filtrado, selección
- Visualizacion de datos
- Organización de información en R marckdown
- Proyecto final

Rstudio es la interfaz amigable

¿Qué es la economía naranja?

Es donde se mezclan las industrias culturales con las áreas de soporte como el desarrollo de aplicaciones o software.

Buscaremos responder a la pregunta: *Si tienes un startup que hace software, ¿en qué país abrirías una oficina?*

El dataset de economía naranja fue creado por la profesora con las siguientes variables:

- Aporte de servicios a PIB.
- Aporte de economía naranja a PIB.

- Penetración de internet.
- Inflación.
- Tasa de desempleo.
- Población debajo de la línea de pobreza.
- Edad mediana de la población.
- Porcentaje de la población entre 25-54 años.
- Inversión en educación %PIB.

5. Los primeros cálculos con R y variables

En esta clase vamos a hacer unos cuantos cálculos dentro de R Studio para ir acostumbrándonos a su sintaxis y comandos útiles.

Dos comandos que utilizaras muy seguidos son:

(Ctrl + L): Se encarga de borrar la consola. (Ctrl + Enter): Realiza la operación que selecciones.

Asignar un valor a una variable dentro de R se hace mediante el par de signos <- \

La función View nos muestra nuestro dataset en forma de tabla.

Si eres experimentado y has utilizado Matlab, R Studio es muy similar a ese también, GNU Octave o Spyder de python.

Podemos cargar la vista de un csv con la funcion **view(nombre_dataset)** ojo te pedirá cargar un paquete previo a poder utilizar la funcion de esta manera.

6. Tipos de datos:

Además de trabajar con el dataset de Orange Economy vamos a necesitar el dataset de mtcars.

Dentro de la consola de R Studio, escribirmos la función

```
install.packages mtcars
```

Esto nos va a ayudar a instalar paquetes, como su nombre lo indica, en este caso intentaremos instalar mtcars.

En caso de no estar disponible para tu versión de R, puedes ir al Github de la profesora y descargarlo.

La función **str()** muestra la estructura que tiene el dataset que le pasemos.

Dentro de la consola escribimos **?mtcars** información sobre nuestro de nuestro dataset

En el dataset mtcars podemos ver que hay datos de tipo int y num, la diferencia es que **num** son números con **decimal o con punto flotante** mientras que **int** son **enteros**, en mi caso todas me aparecieron como tipo num.

Podemos ver que las variables **vs** y **am** dentro de mtcars aunque están marcadas con **int** su función es de **tipo boolean**, para convertir estos datos utilizaremos la función **as.logical**

```
# significa estructura(dataset)
> str(mtcars)
'data.frame':   32 obs. of  12 variables:
 $ model: chr  "Mazda RX4" "Mazda RX4 Wag" "Datsun 710" "Hornet 4 Drive"...
 $ mpg  : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl  : int  6 6 4 6 8 6 8 4 4 6 ...
 $ disp : num  160 160 108 258 360 ...
 $ hp   : int  110 110 93 110 175 105 245 62 95 123 ...
 $ drat : num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt   : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec : num  16.5 17 18.6 19.4 17 ...
 $ vs   : int  0 0 1 1 0 1 0 1 1 1 ...
 $ am   : int  1 1 1 0 0 0 0 0 0 0 ...
 $ gear : int  4 4 4 3 3 3 3 4 4 4 ...
 $ carb : int  4 4 1 1 2 1 4 2 2 4 ...
```

?mtcars

mtcars {datasets} R Documentation

Motor Trend Car Road Tests

Descripcion:

The data was extracted from the 1974 *Motor Trend* US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

Usage:

```
mtcars
```

Format:

A data frame with 32 observations on 11 (numeric) variables.

[, 1]	mpg	Miles/(US) gallon
[, 2]	cyl	Number of cylinders
[, 3]	disp	Displacement (cu.in.)
[, 4]	hp	Gross horsepower
[, 5]	drat	Rear axle ratio
[, 6]	wt	Weight (1000 lbs)
[, 7]	qsec	1/4 mile time
[, 8]	vs	Engine (0 = V-shaped, 1 = straight)
[, 9]	am	Transmission (0 = automatic, 1 = manual)
[,10]	gear	Number of forward gears
[,11]	carb	Number of carburetors

Note:

Henderson and Velleman (1981) comment in a footnote to Table 1: 'Hocking [original transcriber]'s noncrucial coding of the Mazda's rotary engine as a straight six-cylinder engine and the Porsche's flat engine as a V engine, as well as the inclusion of the diesel Mercedes 240D, have been retained to enable direct comparisons to be made with previous analyses.'

Source:

Henderson and Velleman (1981), Building multiple regression models interactively. *Biometrics*, **37**, 391–411.

Example:

```
require(graphics)
pairs(mtcars, main = "mtcars data", gap = 1/4)
coplot(mpg ~ disp | as.factor(cyl), data = mtcars,
       panel = panel.smooth, rows = 1)
## possibly more meaningful, e.g., for summary() or bivariate plots:
mtcars2 <- within(mtcars, {
   vs <- factor(vs, labels = c("V", "S"))
   am <- factor(am, labels = c("automatic", "manual"))
   cyl  <- ordered(cyl)
   gear <- ordered(gear)
   carb <- ordered(carb)
})
summary(mtcars2)
```

Cambiando las variables a lógicas

```
> mtcars$vs = as.logical(mtcars$vs)
> mtcars$am = as.logical(mtcars$am)
> str(mtcars)
'data.frame':   32 obs. of  12 variables:
$ model: chr  "Mazda RX4" "Mazda RX4 Wag" "Datsun 710" "Hornet 4 Dri
ve"...
$ mpg  : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
$ cyl  : int  6 6 4 6 8 6 8 4 4 6 ...
$ disp : num  160 160 108 258 360 ...
$ hp   : int  110 110 93 110 175 105 245 62 95 123 ...
$ drat : num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
$ wt   : num  2.62 2.88 2.32 3.21 3.44 ...
$ qsec : num  16.5 17 18.6 19.4 17 ...
$ vs   : logi  FALSE FALSE TRUE TRUE FALSE TRUE ...
$ am   : logi  TRUE TRUE TRUE FALSE FALSE FALSE ...
$ gear : int  4 4 4 3 3 3 3 4 4 4 ...
$ carb : int  4 4 1 1 2 1 4 2 2 4 ...
```

7. Estructura del dataset del proyecto

summary(): Muestra un resumen del dataset que le mandemos (similar a Describe de Pandas). **transform():** Modifica los valores de un dataset.

Summary para mtcars:

```
> summary(mtcars)
    model                mpg              cyl              disp
```

```
 Length:32          Min.   :10.40   Min.   :4.000   Min.   : 71.1
 Class :character   1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8
 Mode  :character   Median :19.20   Median :6.000   Median :196.3
                    Mean   :20.09   Mean   :6.188   Mean   :230.7
                    3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0
                    Max.   :33.90   Max.   :8.000   Max.   :472.0
       hp             drat            wt             qsec
 Min.   : 52.0   Min.   :2.760   Min.   :1.513   Min.   :14.50
 1st Qu.: 96.5   1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89
 Median :123.0   Median :3.695   Median :3.325   Median :17.71
 Mean   :146.7   Mean   :3.597   Mean   :3.217   Mean   :17.85
 3rd Qu.:180.0   3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90
 Max.   :335.0   Max.   :4.930   Max.   :5.424   Max.   :22.90
     vs              am              gear            carb
 Mode :logical   Mode :logical   Min.   :3.000   Min.   :1.000
 FALSE:18        FALSE:19        1st Qu.:3.000   1st Qu.:2.000
 TRUE :14        TRUE :13        Median :4.000   Median :2.000
                                 Mean   :3.688   Mean   :2.812
                                 3rd Qu.:4.000   3rd Qu.:4.000
                                 Max.   :5.000   Max.   :8.000
```

Cambiando a kilos el peso wt

```
> wt <- (mtcars$wt*1000)/2
> wt
 [1] 1310.0 1437.5 1160.0 1607.5 1720.0 1730.0 1785.0 1595.0
 [9] 1575.0 1720.0 1720.0 2035.0 1865.0 1890.0 2625.0 2712.0
[17] 2672.5 1100.0  807.5  917.5 1232.5 1760.0 1717.5 1920.0
[25] 1922.5  967.5 1070.0  756.5 1585.0 1385.0 1785.0 1390.0
> mtcars.new <- transform(mtcars,wt=wt*1000/2)
> summary(mtcars.new)
    model                mpg             cyl             disp
 Length:32          Min.   :10.40   Min.   :4.000   Min.   : 71.1
 Class :character   1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8
 Mode  :character   Median :19.20   Median :6.000   Median :196.3
                    Mean   :20.09   Mean   :6.188   Mean   :230.7
                    3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0
                    Max.   :33.90   Max.   :8.000   Max.   :472.0
       hp             drat            wt              qsec
 Min.   : 52.0   Min.   :2.760   Min.   : 756.5   Min.   :14.50
 1st Qu.: 96.5   1st Qu.:3.080   1st Qu.:1290.6   1st Qu.:16.89
 Median :123.0   Median :3.695   Median :1662.5   Median :17.71
 Mean   :146.7   Mean   :3.597   Mean   :1608.6   Mean   :17.85
 3rd Qu.:180.0   3rd Qu.:3.920   3rd Qu.:1805.0   3rd Qu.:18.90
 Max.   :335.0   Max.   :4.930   Max.   :2712.0   Max.   :22.90
     vs              am              gear            carb
 Mode :logical   Mode :logical   Min.   :3.000   Min.   :1.000
 FALSE:18        FALSE:19        1st Qu.:3.000   1st Qu.:2.000
 TRUE :14        TRUE :13        Median :4.000   Median :2.000
                                 Mean   :3.688   Mean   :2.812
                                 3rd Qu.:4.000   3rd Qu.:4.000
                                 Max.   :5.000   Max.   :8.000
```

Para Economia naranja

```
> str(orangeec)
'data.frame':   18 obs. of  13 variables:
 $ V1 : chr  "Country" "Argentina" "Belize" "Bolivia" ...
 $ V2 : chr  "GDP PC" "20900" "8300" "7500" ...
 $ V3 : chr  "GDP US bill" "637.7" "1854" "37.1" ...
 $ V4 : chr  "GDP Growth %" "2.9" "0.8" "4.2" ...
 $ V5 : chr  "Services % GDP" "60.9" "62.2" "50" ...
 $ V6 : chr  "Creat Ind % GDP" "3.8" "" "" ...
 $ V7 : chr  "Inflation" "25.7" "1.1" "2.8" ...
 $ V8 : chr  "Unemployment" "8.1" "10.1" "4" ...
 $ V9 : chr  "% pop below poverty line" "25.7" "41" "38.6" ...
 $ V10: chr  "Internet penetration % population" "93.1" "52.3" "78.6" ...
 $ V11: chr  "Median age" "31.7" "22.7" "24.3" ...
 $ V12: chr  "% pop 25-54" "39.38" "36.62" "37.48" ...
 $ V13: chr  "Education invest % GDP" "5.9" "7.4" "7.3" ...
```

Resumen para el data set Orangeec

```
> summary(orangeec)
      V1                 V2                 V3                 V4           
 Length:18          Length:18          Length:18          Length:18          
 Class :character   Class :character   Class :character   Class :character   
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   
      V5                 V6                 V7                 V8           
 Length:18          Length:18          Length:18          Length:18          
 Class :character   Class :character   Class :character   Class :character   
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   
      V9                V10                V11                V12           
 Length:18          Length:18          Length:18          Length:18          
 Class :character   Class :character   Class :character   Class :character   
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   
    V13           
 Length:18          
 Class :character   
 Mode  :character   
```

8. Vectores

Un **vector** es un ente matemático que se usa para **guardar información de un mismo tipo**, dentro de R se crean los vectores con la función **c**.

Los vectores son colecciones ordenadas de elementos del mismo tipo

Creación de vectores con operaciones:

```
> tiempo_platzi <- c(25,5,10,15,10)
> tiempo_lecturas <- c(30,10,5,10,15)
> tiempo_aprendizaje  <- tiempo_platzi + tiempo_lecturas
> tiempo_aprendizaje
[1] 55 15 15 25 25
```

```
> dias_aprendizaje <- c("Lunes", "Martes","Miercoles", "Jueves", "Viernes")
> dias_aprendizaje
[1] "Lunes"     "Martes"    "Miercoles"   "Jueves"    "Viernes"
> dias_mas_de_20min <- c(TRUE, FALSE, FALSE, TRUE, TRUE)
> dias_mas_de_20min
[1]  TRUE FALSE FALSE  TRUE  TRUE
> total_tiempo_platzi <- sum(tiempo_platzi)
> total_tiempo_platzi
[1] 65
> total_tiempo_lecturas <- sum(tiempo_lecturas)
> total_tiempo_lecturas
[1] 70
> total_tiempo_adicional <- total_tiempo_platzi + total_tiempo_lecturas
> total_tiempo_adicional
[1] 135
```

Kvkh

9. Matrices

Las matrices o variables indexadas (Arrays) son generalizaciones multidimensionales de vectores. De hecho, son vectores indexados por dos o mas indices y que se imprimen de modo especial. Para crearlas utilizamos la funci´on “matrix”. Los parametros principales de esta funci´on son:

- data (vector que contiene los valores que formar´an la matriz)
- nrow (n´umero de filas),
- ncol (n´umero de columnas).

Byrow es el booleano para indicar si se llena la matriz por filas

```
> tiempo_matriz <- matrix(c(tiempo_platzi, tiempo_lecturas), nrow=2, byrow
= "TRUE")
> dias <- c("Lunes", "Martes", "Miercoles", "Jueves", "Viernes")
> tiempo <- c("Tiempo platzi", "Tiempo lecturas")
> colnames(tiempo_matriz) <- dias
> rownames(tiempo_matriz) <-tiempo
> tiempo_matriz
                Lunes Martes Miercoles Jueves Viernes
Tiempo platzi      25      5        10     15      10
Tiempo lecturas    30     10         5     10      15
# retorna la suma de los elementos de la columna
> colSums(tiempo_matriz)
    Lunes    Martes Miercoles    Jueves   Viernes
       55        15        15        25        25
```

10. Ejercicios con matrices

Usamos Rbin para agregar una fila

```
> final_matriz <- rbind(tiempo_matriz, c(10,15,30,5,0))
> final_matriz
                Lunes Martes Miercoles Jueves Viernes
Tiempo platzi      25      5        10     15      10
Tiempo lecturas    30     10         5     10      15
```

```
                  10     15        30      5       0
> colSums(final_matriz)
    Lunes    Martes Miercoles    Jueves   Viernes
       65        30        45        30        25
# elemento de la fila 1 columna 5
> final_matriz[1,5]
[1] 10
```

RETO: Agregar una columna

Agregamos una columna con cbind

```
> final_matriz2 <- cbind(final_matriz, c(8,14,20))
> final_matriz2
                Lunes Martes Miercoles Jueves Viernes
Tiempo platzi      25      5        10     15      10     8
Tiempo lecturas    30     10         5     10      15    14
                   10     15        30      5       0    20
```

11. Operaciones para comparar y ubicar datos

En R cuentas con los operadores de comparación comunes como == o |, pero además cuentas con el operador:

%in% Que sirve para checar si un elemento se encuentra en el dataset Para hacer una selección de elementos de un vector, matriz o data frame podemos usar la función subset.

Podemos **renombrar una variable de nuestro dataset** orangeec, para ello debemos tener instalado el **paquete plyr**. En caso de no tener el paquete instalado solamente corremos en la consola el código install.packages("plyr"), después lo activas manual o con la consola.

Muestra los carros que tienen menos de 6 cilindros

```
> mtcars[mtcars$cyl<6,]
            model  mpg cyl  disp  hp drat    wt  qsec    vs    am gear carb
3      Datsun 710 22.8   4 108.0  93 3.85 2.320 18.61  TRUE  TRUE    4    1
8       Merc 240D 24.4   4 146.7  62 3.69 3.190 20.00  TRUE FALSE    4    2
9        Merc 230 22.8   4 140.8  95 3.92 3.150 22.90  TRUE FALSE    4    2
18       Fiat 128 32.4   4  78.7  66 4.08 2.200 19.47  TRUE  TRUE    4    1
19    Honda Civic 30.4   4  75.7  52 4.93 1.615 18.52  TRUE  TRUE    4    2
20 Toyota Corolla 33.9   4  71.1  65 4.22 1.835 19.90  TRUE  TRUE    4    1
21  Toyota Corona 21.5   4 120.1  97 3.70 2.465 20.01  TRUE FALSE    3    1
26      Fiat X1-9 27.3   4  79.0  66 4.08 1.935 18.90  TRUE  TRUE    4    1
27  Porsche 914-2 26.0   4 120.3  91 4.43 2.140 16.70 FALSE  TRUE    5    2
28   Lotus Europa 30.4   4  95.1 113 3.77 1.513 16.90  TRUE  TRUE    5    2
32     Volvo 142E 21.4   4 121.0 109 4.11 2.780 18.60  TRUE  TRUE    4    2
```

Filtro para la data Orangeec:

```
> orangeec[orangeec$V2>=15000,]
           V1     V2          V3           V4             V5             V6
1     Country GDP PC GDP US bill GDP Growth % Services % GDP Creat Ind % GDP
2   Argentina  20900       637.7          2.9           60.9            3.8
```

```
3       Belize   8300        1854          0.8           62.2                 
4      Bolivia   7500        37.1          4.2             50                 
5       Brazil  15600     2055000            1           72.8              2.6
6        Chile  24500         277          1.5           64.3              2.2
8   Costa Rica  16900        58.1          3.2           73.5                2
10 El Salvador   8900          28          2.4           64.9                 
11   Guatemala   8100        75.7          2.8           63.2                 
12    Honduras   5600        22.9          4.8           57.8                 
13      Mexico  19900     1149000            2             64              7.4
14   Nicaragua   5800        13.7          4.9           50.8                 
15      Panama  25400        61.8          5.4             82              6.3
16    Paraguay   9800        29.6          4.3           54.5              4.1
18     Uruguay  22400        58.4          3.1           68.8                1
```

Aporacion de menos de 2% a la economía naranja

```
> orangeec[orangeec$V6<=2,]
            V1    V2    V3  V4   V5  V6  V7   V8   V9  V10  V11   V12 V13
3       Belize  8300  1854 0.8 62.2     1.1 10.1   41 52.3 22.7 36.62 7.4
4      Bolivia  7500  37.1 4.2   50     2.8    4 38.6 78.6 24.3 37.48 7.3
8   Costa Rica 16900  58.1 3.2 73.5   2 1.6  8.1 21.7 86.7 31.3 44.03 7.1
9      Ecuador 11500 102.3 2.7 56.9   2 0.4  4.6 21.5 79.9 27.7 39.59   5
10 El Salvador  8900    28 2.4 64.9       1    7 32.7 57.7 27.1 39.23 3.5
11   Guatemala  8100  75.7 2.8 63.2     4.4  2.3 59.3 42.1 22.1 34.12 2.8
12    Honduras  5600  22.9 4.8 57.8     3.9  5.9 29.6 38.2   23 36.63 5.9
14   Nicaragua  5800  13.7 4.9 50.8     3.9  6.5 29.6   43 25.7 40.24 4.5
17        Peru 13300 215.2 2.5 56.8 1.5 2.8  6.7 22.7 67.6   28 40.19 3.8
18     Uruguay 22400  58.4 3.1 68.8   1 6.2  7.3  9.7 88.2   35 39.34 4.4
```

Luego de los filtros:

```
> newoeangeec
          V1     V2          V3           V4             V5              V6
1    Country GDP PC GDP US bill GDP Growth % Services % GDP Creat Ind % GDP
2  Argentina  20900       637.7          2.9           60.9             3.8
8  Costa Rica 16900        58.1          3.2           73.5               2
16   Paraguay  9800        29.6          4.3           54.5             4.1
         V7           V8                    V9                            V10
1 Inflation Unemployment%pop below poverty line Internet penetration%population
2      25.7          8.1                  25.7                           93.1
8       1.6          8.1                  21.7                           86.7
16      3.6          6.5                  22.2                           89.6
           V11         V12                  V13
1   Median age % pop 25-54 Education invest % GDP
2         31.7       39.38                  5.9
8         31.3       44.03                  7.1
16        28.2       41.08                    5
```

Subconjunto en function de V6:

```
> newoeangeec <- subset(orangeec,V10 >80 & V13>=4.5, select = V6)
> newoeangeec
```

```
                V6
1  Creat Ind % GDP
2              3.8
8                2
16             4.1
```

Renombrando columnas del dataset (pero antes instalamos la librería **plyr**)

Cambio de nombre de columnas:

```
> rename(orangeec, c("V6"="AporteEcNja"))
            V1     V2          V3           V4             V5     AporteEcNja
1Country GDP PC GDP US bill GDP Growth % Services % GDP Creat Ind % GDP
2    Argentina  20900       637.7          2.9           60.9             3.8
3       Belize   8300        1854          0.8           62.2
4      Bolivia   7500        37.1          4.2             50
5       Brazil  15600     2055000            1           72.8             2.6
6        Chile  24500         277          1.5           64.3             2.2
7     Colombia  14500       309.2          1.8           61.4             3.3
8   Costa Rica  16900        58.1          3.2           73.5               2
9      Ecuador  11500       102.3          2.7           56.9               2
10 El Salvador   8900          28          2.4           64.9
11   Guatemala   8100        75.7          2.8           63.2
12    Honduras   5600        22.9          4.8           57.8
13      Mexico  19900     1149000            2             64             7.4
14   Nicaragua   5800        13.7          4.9           50.8
15      Panama  25400        61.8          5.4             82             6.3
16    Paraguay   9800        29.6          4.3           54.5             4.1
17        Peru  13300       215.2          2.5           56.8             1.5
18     Uruguay  22400        58.4          3.1           68.8               1
```

12. Factores, listas y echar un vistazo al dataset:

Factores: Tipo de dato con variables categóricas.

```
> Nivel_curso <- c("Basico", "Intermedio", "Avanzado")
> Nivel_curso
[1] "Basico"     "Intermedio" "Avanzado"
```

Echando un vistazo al dataframe:

head: es una función que nos retorna los primeros elementos de un dataset, por defecto nos retorna los primeros 6.

```
> head(mtcars)
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
> head(orangeec)
         V1     V2          V3           V4              V5               V6
1   Country GDP PC GDP US bill GDP Growth % Services % GDP Creat Ind % GDP
2 Argentina  20900       637.7          2.9            60.9              3.8
3    Belize   8300        1854          0.8            62.2
4   Bolivia   7500        37.1          4.2              50
5    Brazil  15600     2055000            1            72.8              2.6
6     Chile  24500         277          1.5            64.3              2.2
         V7           V8                        V9
1 Inflation Unemployment % pop below poverty line
2      25.7          8.1                      25.7
3       1.1         10.1                        41
4       2.8            4                      38.6
5       3.4         11.8                       4.2
6       2.2            7                      14.4
                            V10        V11       V12                  V13
1 Internet penetration%population Median age%pop 25-54 Education invest%GDP
2                            93.1       31.7     39.38                  5.9
3                            52.3       22.7     36.62                  7.4
4                            78.6       24.3     37.48                  7.3
5                            70.7         32     43.86                  5.9
6                            77.5       34.4     43.08                  4.9
```

tail: función similar a head solamente que esta función nos retorna los últimos elementos.

```
> tail(mtcars)
                mpg cyl  disp  hp drat    wt qsec vs am gear carb
Porsche 914-2  26.0   4 120.3  91 4.43 2.140 16.7  0  1    5    2
Lotus Europa   30.4   4  95.1 113 3.77 1.513 16.9  1  1    5    2
Ford Pantera L 15.8   8 351.0 264 4.22 3.170 14.5  0  1    5    4
Ferrari Dino   19.7   6 145.0 175 3.62 2.770 15.5  0  1    5    6
Maserati Bora  15.0   8 301.0 335 3.54 3.570 14.6  0  1    5    8
Volvo 142E     21.4   4 121.0 109 4.11 2.780 18.6  1  1    4    2
> tail(orangeec)
          V1    V2      V3  V4   V5  V6  V7  V8   V9  V10  V11   V12 V13
13    Mexico 19900 1149000   2   64 7.4   6 3.6 46.2   65 28.3 40.81 5.3
14 Nicaragua  5800    13.7 4.9 50.8     3.9 6.5 29.6   43 25.7 40.24 4.5
15    Panama 25400    61.8 5.4   82 6.3 0.9 5.5   23 69.7 29.2 40.35 3.2
16  Paraguay  9800    29.6 4.3 54.5 4.1 3.6 6.5 22.2 89.6 28.2 41.08   5
17      Peru 13300   215.2 2.5 56.8 1.5 2.8 6.7 22.7 67.6   28 40.19 3.8
18   Uruguay 22400    58.4 3.1 68.8   1 6.2 7.3  9.7 88.2   35 39.34 4.4
```

Además de poder visualizar un dataset con str podemos instalar el paquete dplyr:

Una vez instalado usamos la función **glimpse()**.

correr glimpse(orangeec)

Listas: superobjetos permiten almacenar vectores matrices dataframes

Una lista se construye con la funcion list que devuelve un objeto de tipo lista con tantos componentes como argumentos se le suministren y es utilizado para devolver el resultado de una funcion.

```
> my_vector <- 1:8
> my_matrix <- matrix(1:9, ncol=3)
> my_matrix
     [,1] [,2] [,3]
```

```
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
> my_df <- mtcars[1:4,]
> my_df
                mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710     22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive 21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
> my_list <- list(my_vector, my_matrix, my_df)
> my_list
[[1]]
[1] 1 2 3 4 5 6 7 8

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

[[3]]
                mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4      21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag  21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710     22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive 21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
```

13. Que es EDA: Explotatory Data Analisis:

Nhn

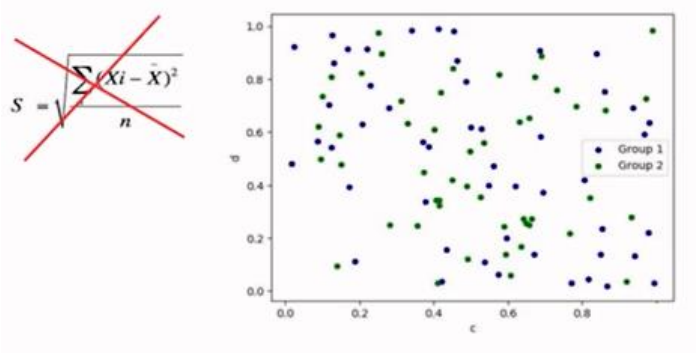
El *EDA* es un ciclo iterativo en el que:

1. Generas preguntas acerca de tus datos.
2. Buscas respuestas visualizando, transformando y modelando tus datos.
3. Usas lo que has aprendido para refinar tus preguntas y/o generar nuevas interrogantes

El análisis exploratorio de datos no es un proceso formal regido por un conjunto estricto de reglas. El *EDA* es, más que nada, un estado mental. Durante las fases iniciales del *EDA* deberías ser libre de investigar todas las ideas que se te ocurran. Algunas de estas ideas prosperarán, mientras que otras serán como callejones sin salida. A medida que tu exploración continúa, te concentrarás en ciertas áreas particularmente productivas sobre las que eventualmente escribirás y comunicarás a otras personas.

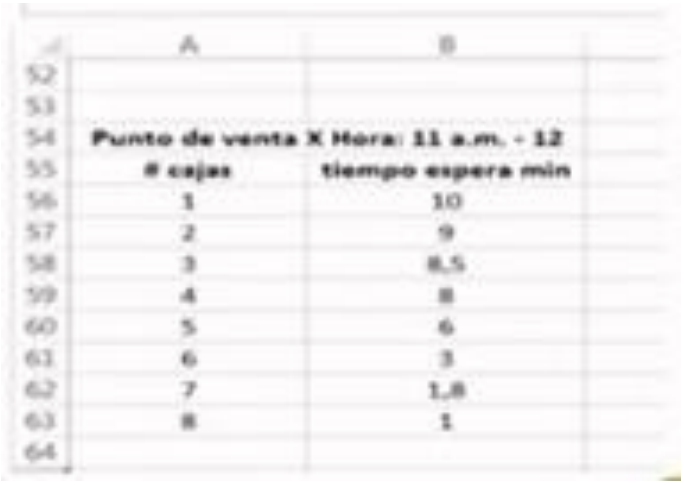
El *EDA* es una parte importante de cualquier análisis, aun si las preguntas están servidas en bandeja, pues siempre tendrás que examinar la calidad de tus datos. La limpieza de datos es una aplicación del *EDA*: haces preguntas acerca de si tus datos cumplen con tus expectativas o no. Para limpiar tus datos tendrás que desplegar todas las herramientas del *EDA*: visualización, transformación y modelado.

Consiste en la importancia de visualizar los datos antes de enfocarnos en las formulas estadísticas



Imaginemos que tenemos a clientes enojados por la demora en la atención en un supermercado con 4 puntos de venta

Tomamos muestras de cada caja y en conjunto para ver las cajas atendiendo



Punto de venta X Hora: 11 a.m. - 12

# cajas	tiempo espera min
1	10
2	9
3	8.5
4	8
5	6
6	3
7	1.8
8	1

Obtenemos los 4 datasets de los supermercados

Propiedad	Valor
Media de cada una de las variables x	9.0
Varianza de cada una de las variables x	11.0
Media de cada una de las variables y	7.5
Varianza de cada una de las variables y	4.12
Correlación entre cada una de las variables x e y	0.816
Recta de regresión	$y = 3 + 0.5x$

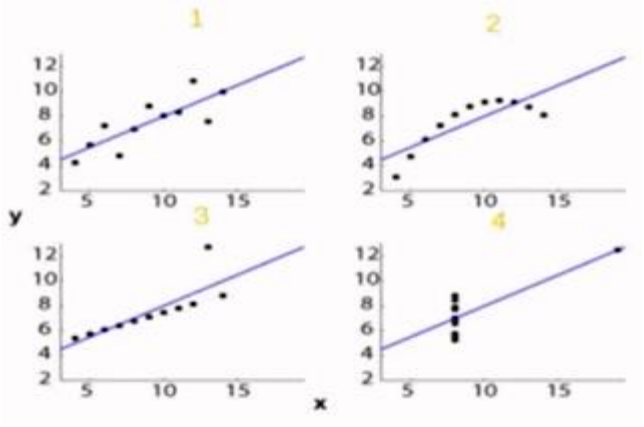
Recibimos las estadísticas descriptivas y vemos que es demasiado raro tener 4 comportamientos iguales

	Set 1	Set 2	Set 3	Set 4
Mean X	9.00	9.00	9.00	9.00
Mean Y	7.50	7.50	7.50	7.50
Std Dev X	3.32	3.32	3.32	3.32
Std Dev Y	2.03	2.03	2.03	2.03
Correlation	0.82	0.82	0.82	0.82
Regression	Y = 3.0 + 0.5X	Y = 3.0 + 0.5X	Y = 3.0 + 0.5X	Y = 3.0 + 0.5X

Pedimos datos crudos y envían lo siguiente:

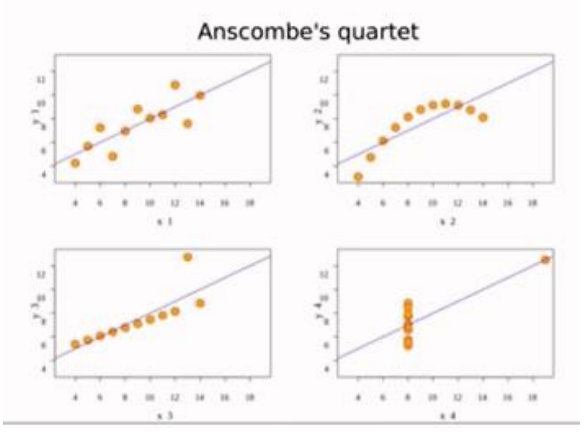
1		2		3		4	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

Veos que forma tienen los datos, todos tienen la misma recta de regresión pero con comportamiento distinto



Y vemos que nos topamos con el fenómeno del Cuarteto de Anscombe.

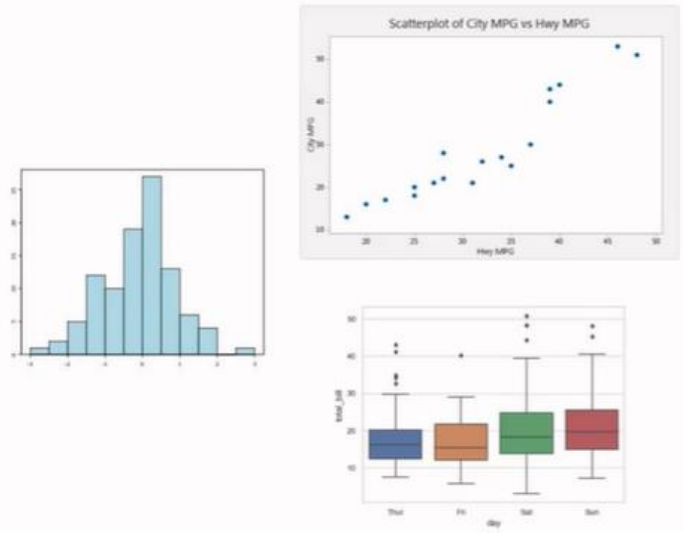
Este cuarteto nos dice la importancia de ver los datos antes de enfocarnos en las formulas estadisticas



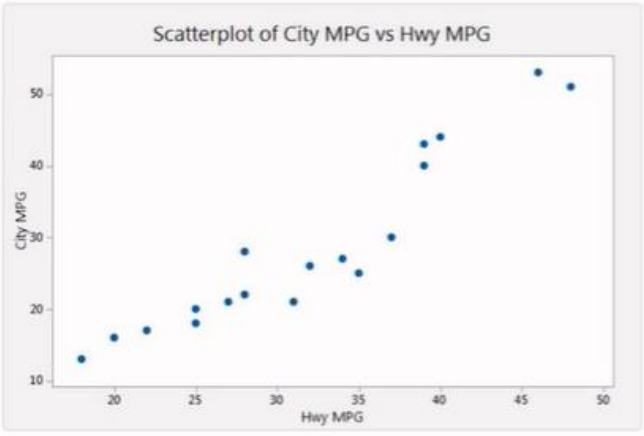
El **cuarteto de Anscombe** comprende cuatro conjuntos de datos que tienen las **mismas propiedades estadísticas**, pero que evidentemente son **distintas al inspeccionar sus gráficos** respectivos

14. Graficas de dispersión e histogrmas:

Existen varios tipos de gráficas para visualizar la información al momento de hacer EDA:

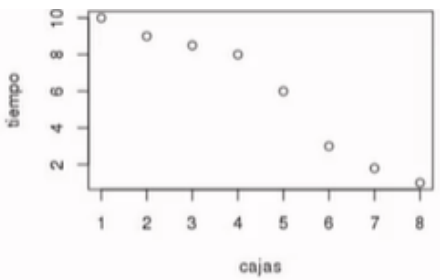


Scatterplot o graficas de dispersion: mezclamos o cruzamos variables continuas o datos numéricos, en ninguno de los ejes tenemos etiquetas, variables categorías o palabras, no podemos unir puntos con scatterlplot pero si con grafico de lineas



Ejemplo:

Número cajas	Tiempo espera fila
1	10
2	9
3	8.5
4	8
5	6
6	3
7	1.8
8	1

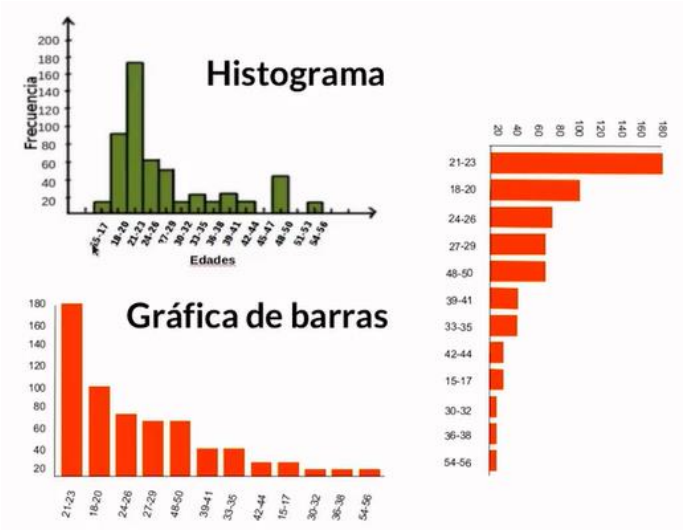


tg

Relaciona tiempo de espera en fila frente al numero de cajas que esta funcionando

Ubicamos la variable independiente en en el eje x y los dependientes en el eje y

Histogramas: sirve para ver la distribución de las frecuencias de una variable, nos muestra lo que hay o lo que no hay en uns distribución.



Caracteristicas:

Las barras van pegadas

El orden en el eje x van en orden ascendente de menor a mayor nos presenta lo que hay y lo que no hay, en lo que no hay, hay un hueco, en barras no hay tal hueco

No podemos covertir los números en etiquetas o palabras, siempre tendremos variabes continuas y numéricas

Boxplot:

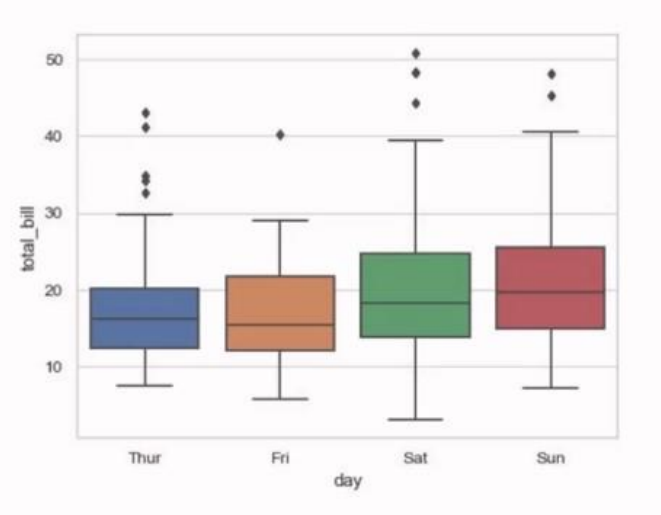
Nos muestra 5 elementos claves en estadística descriptiva

- Mínimo
- Máximo
- Primero cuartir
- 2do cuartil o Mediana
- 3er cuartil

15. Box plot y su interpretación:

Los 5 puntos clave en estadística descriptiva se pueden visualizar en el box plot:

- Primer cuartil: es el piso de la caja o línea inferior.
- Mediana: es la línea que se encuentra dentro de la caja.
- Tercer cuartil: es el techo de la caja o línea superior.
- Mínimo: la extensión inferior de la caja.
- Máximo: la extensión superior de la caja.



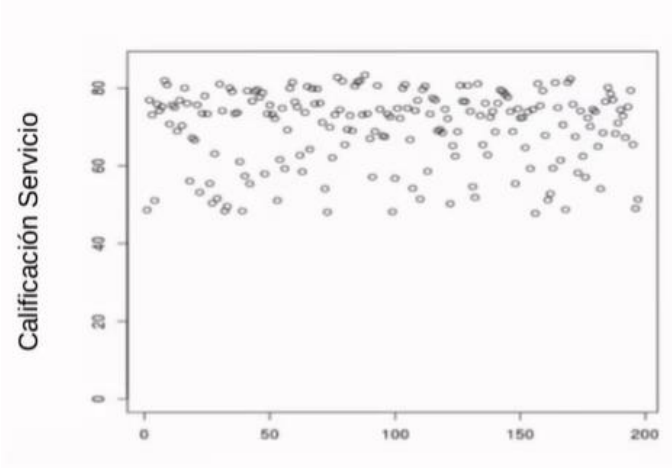
Imaginemos que tenemos somos los directores de servicio al cliente de una cadena de ropa en 4 paises y ordenamos una evaluación de servicio para ver como vamos

Los 197 datos que corresponden a las 197 tiendas evaluadas

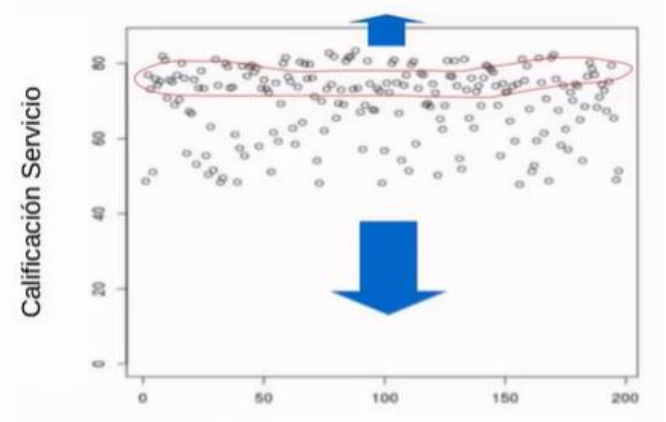
Nos llega:

Tienda	CALIFICACION	Tienda	CALIFICACION
Armenia 1-COL	48.6	Medellín 2-COL	81.8
Arequipa 2-PER	80.4	Callao 2-PER	83.3
Guayaquil 2-ECU	76.7	Bogotá 1-COL	48.1
Bogotá 5-COL	75.9	Manizales 1-COL	76.9
Caracas 3-VEN	81.9	Lima 4-PER	80.7
Bogotá 4-COL	80	Santa Marta 1-COL	76.1
Barrancabermeja-COL	66	Cuenca 1-ECU	73.9
Lima 3-PER	73.4	Quito 3-ECU	68.8
Cusco 1-PER	81	Guayaquil 1-ECU	73.9
Cuenca 3-ECU	79	Bogotá 2-COL	81.1
Cali 1-COL	73.4	Bello 1-COL	51.2
Santo Domingo 1-ECU	73.7	Monteria 1-COL	81.4
Bogotá 3-COL	80.6	Barranquilla 2-COL	64.7
Cúcuta 1-COL	68.8	Trujillo 1-PER	47.8
Quito 1-ECU	75.6	Medelin 3-COL	81.4
Lima 2-PER	59.2	Pereira 2-COL	73.9
Caracas 2-VEN	73.2	Maracaibo 3-VEN	80.1
Cartagena 1-COL	81.5	Quito 2-ECU	78.5
Maracaibo 1-VEN	62	Callao 1-PER	74.4
Valencia 1-VEN	82.7	Bucaramanga 2-COL	51.3
Bogotá 6-COL	72.9	Barquisimeto 1-VEN	69
		Lima 1-PER	81.6

No es tan sencillo ver la comportacion en tabla, por lo que botamos los datos al plano

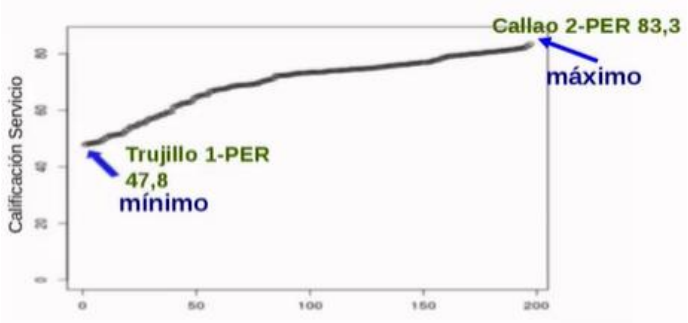


- 0 es pesimo servicio
- 100 es excelente servicio



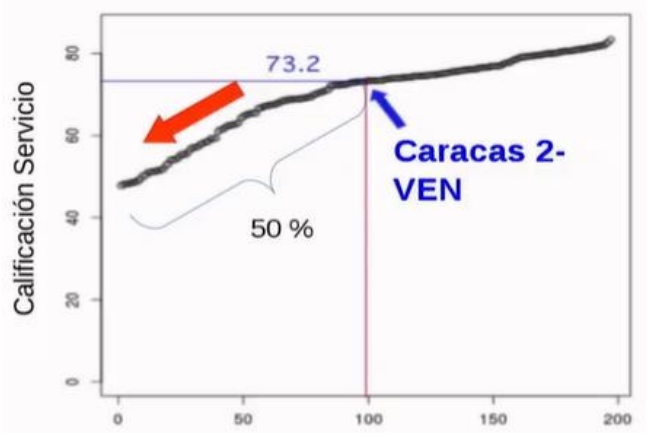
Hay una concentración de puntos en 75 y 80, No hay tiendas por encima de 85 y tampoco por debo de 40 La información es limitada.pero aun no podemos presentar el informe

Decidimos ordenar los puntos de menor a mayor



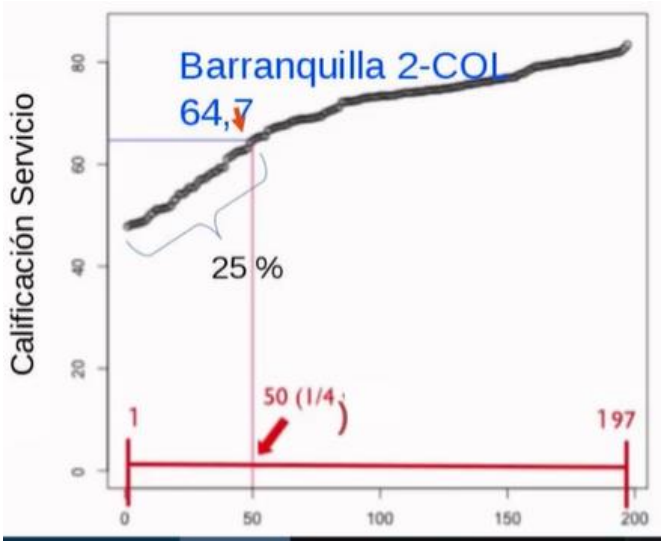
Obtenemos el menor y mayor

Que pasa cuando vemos el punto en la mitad del camino



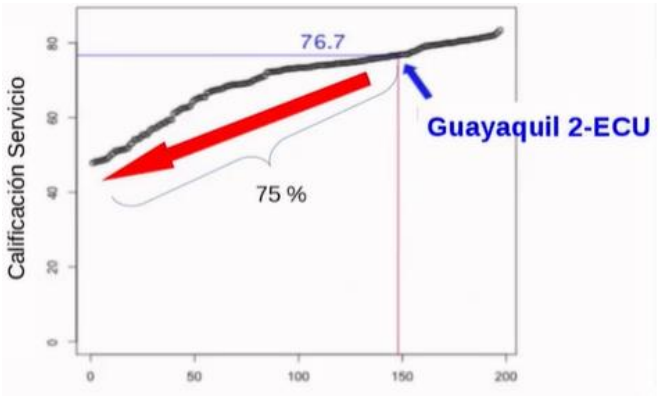
Se puede decir que el 50% de las tiendas lograron una calificación menor o mayor a 73.2

Y si nos vamos al cuarto del camino



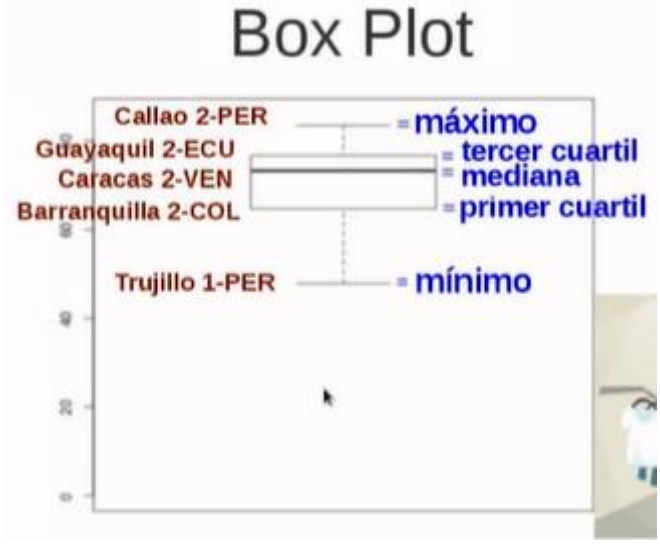
Se puede decir que el 25% logro una calificación menor a 64,7

Si nos vamos a los ¾ del camino

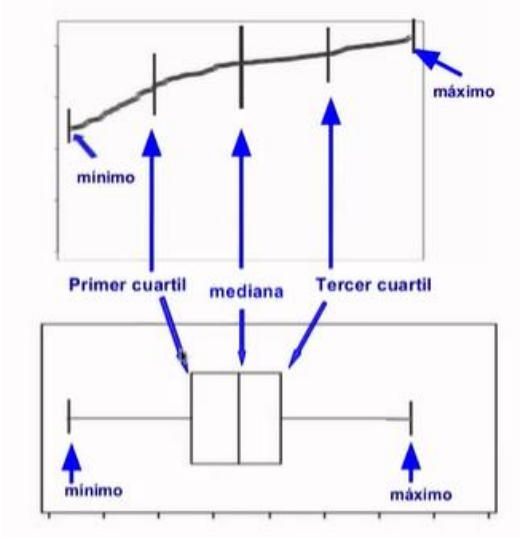


Se puede decir que el 75% logro una calificación menor a 76,7

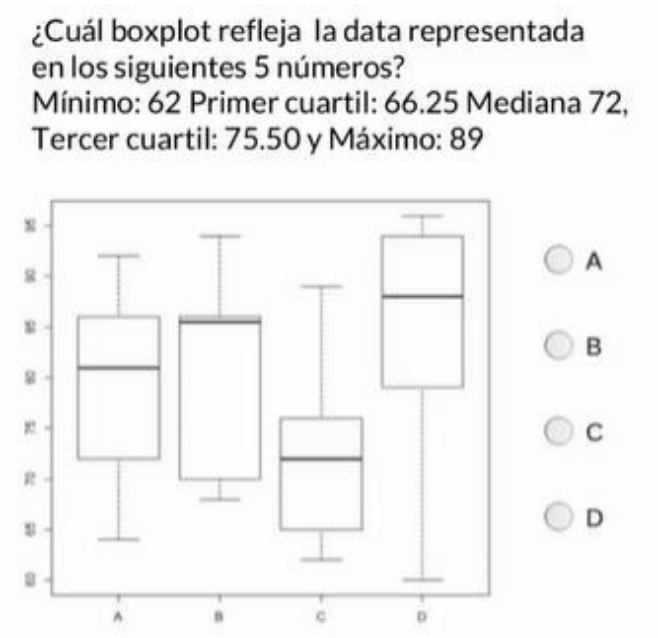
Esto que acabamos de revisar son los 5 componente clave de la estadistica descriptiva.



Todo eso es lo que vemos en las graficas boxplot

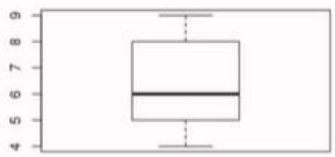


Reto:



Alternatica C

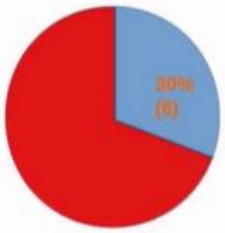
Calificación Servicio	Puntos de venta
A	9
B	6
C	8
D	7
E	5
F	4
G	8
H	5
I	6
J	6



Mediana 6: la mitad logro una calificación menor (inferior) a 6 o mayor (superior) a 6

Calificación Servicio	Puntos de venta
A	9
B	6
C	8
D	7
E	5
F	4
G	8
H	5
I	6
J	6

Qué porcentaje de las tiendas Lograron calificación de 6?



No se dice que el 50% de las tiendas logro una calificación de 6, para esotndriamos que tener 5 numeros 6 y no es cierto

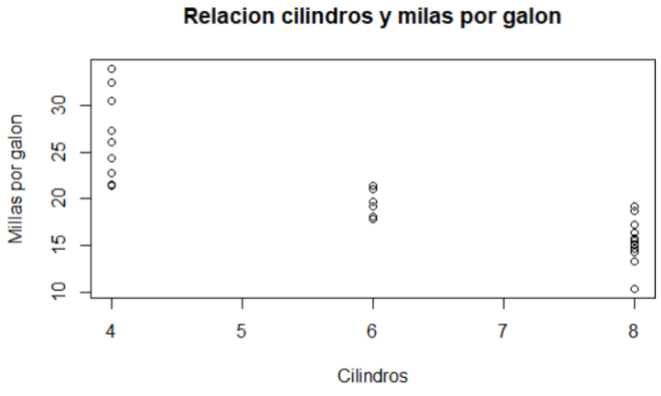
16. EDA con dataset proyecto – Graficas de dispersión

Para realizar EDA con una gráfica de dispersion dentro de R debemos utilizar la funcion plot, los argumentos que debemos pasarle son:

la información en el eje X y Y. **xlab:** título para el eje x. **ylab:** título para el eje y. **main:** título de la gráfica.

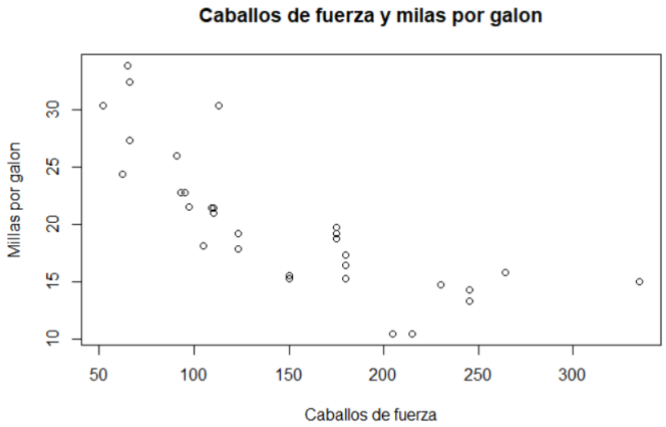
Creando scatterplots con el dataset mtcars:

```
> plot(mtcars$mpg ~ mtcars$cyl,
+      xlab="Cilindros", ylab = "Millas
 por galon",
+      main = "Relacion cilindros y mil
as por galon")
```



Gnfgn

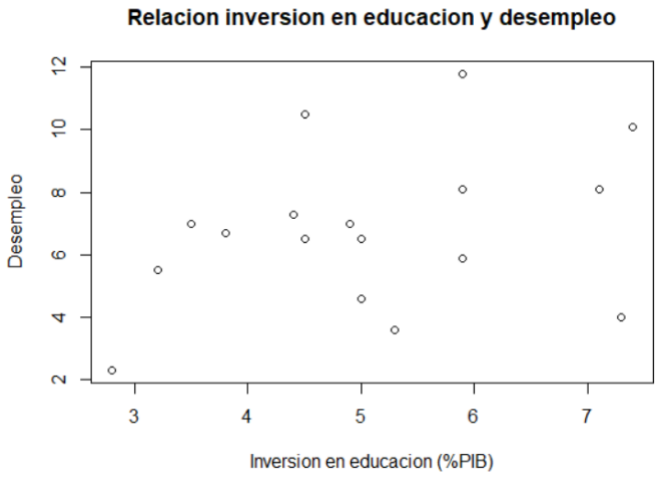
```
> plot(mtcars$mpg ~ mtcars$hp,
+      xlab="Caballos de fuerza", ylab
= "Millas por galon",
+      main = "Caballos de fuerza y mil
as por galon")
```



Tine una correlación negativa,

Para orangeec:

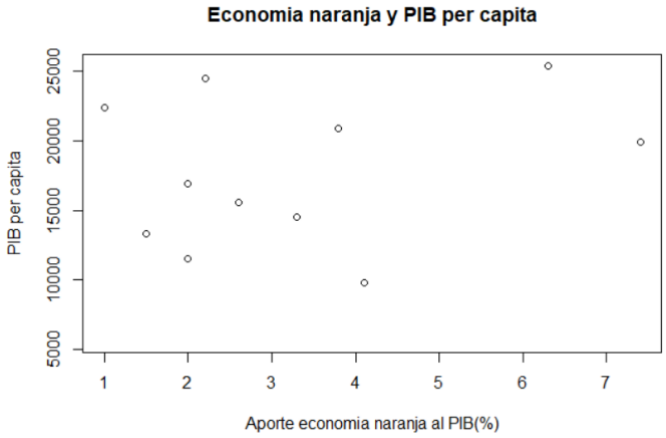
```
> plot(orangeec$V8 ~ orangeec$V13,
+      xlab = "Inversion en educacion (
%PIB)",
+      ylab = "Desempleo",
+      main = "Relacion inversion en ed
ucacion y desempleo")
```



Hay países que invierten en educacion hasta el 7% de su PIB y tiene alto desempleo, y otros pero la muestra es mixta.

```
> plot(orangeec$V2 ~ orangeec$V6,
```

```
+       xlab = "Aporte economia naranja
al PIB(%)",
+       ylab = "PIB per capita",
+       main = "Economia naranja y PIB p
er capita")
```



17. EDA con histogramas

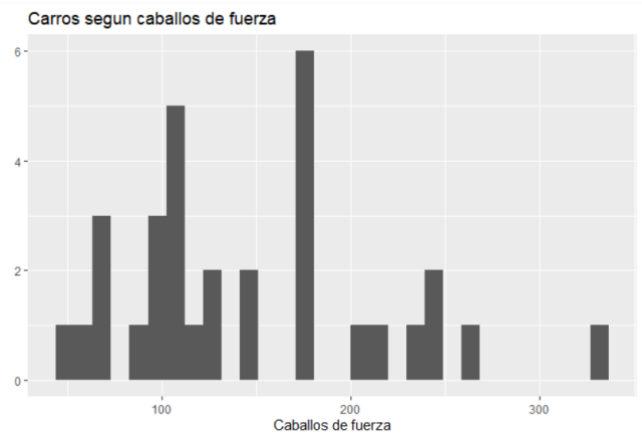
Para realizar EDA con un histograma dentro de R debemos utilizar la función hist(), los argumentos que debemos pasarle son:

- la información en el eje X.
- geom: describir el tipo de gráfica que se va a imprimir.
- xlab: título para el eje x.
- main: título de la gráfica.

histogramas sin instalar ningun paquete con qplot

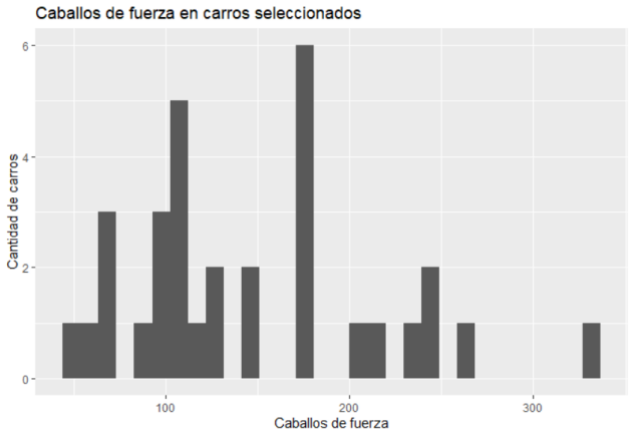
```
> qplot(mtcars$hp, geom= "histogram",
+       xlab = "Caballos de fuerza",
+       main = "Carros segun caballos de fuerza")
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Usa 30 como el ancho de cada una de las barritas



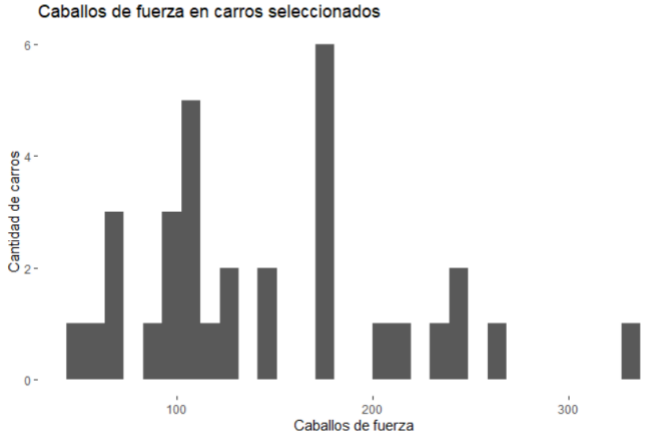
Instalar ggplot2

```
> ggplot(mtcars, aes(x = hp)) +
+   geom_histogram() + labs(x = “Caballos de fuerza”,
+   y = “Cantidad de carros”, title = “Caballos de fuerza en carros selecci
onados”)
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



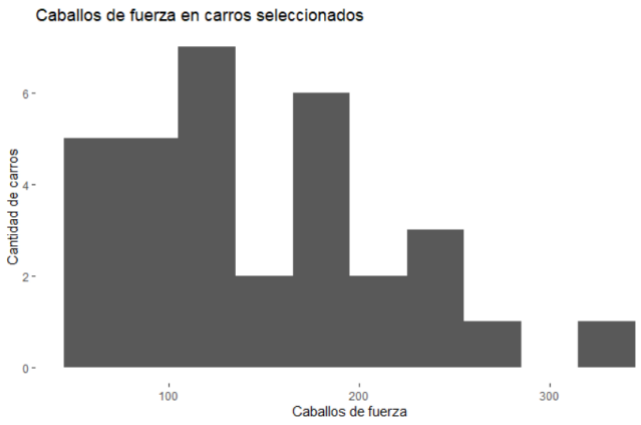
Mejorando la grafica:

```
> ggplot(mtcars, aes(x = hp)) +
+   geom_histogram() + labs(x = “Caballos de fuerza”,
+   y = “Cantidad de carros”, title = “Caballos de fuerza en carros selecciona
dos”) + theme(legend.position = “none”) +
+   theme(panel.background = element_blank() ,
+   panel.grid.major = element_blank(), panel.grid.minor = element_blank())
```

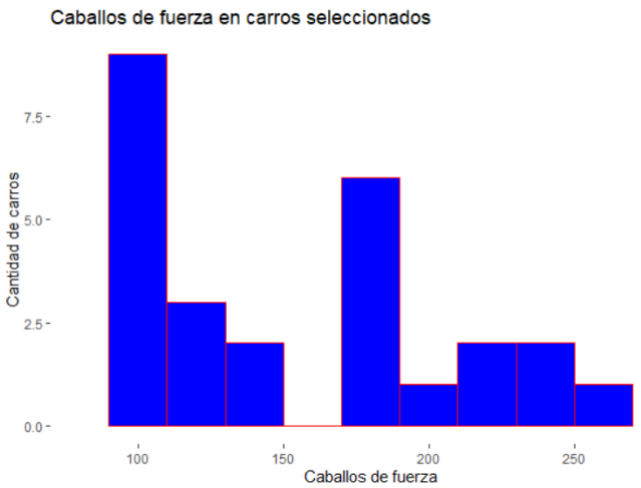


Ajustando el ancho de la barra:

```
> ggplot(mtcars, aes(x = hp)) +
+   geom_histogram(binwidth = 30) + labs(x = “Caballos de fuerza”,
+                           y = “Cantidad de carros”, title = “Caballos de
fuerza en carros seleccionados”) +
+   theme(legend.position = “none”) +
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
```




```
> ggplot() + geom_histogram(data = mtcars,
+                           aes(x=hp), fill = “blue”, color= “red”,
+                           binwidth = 20) +
+   labs(x = “Caballos de fuerza”,
+        y = “Cantidad de carros”, title = “Caballos de fuerza en carros se
leccionados”) +
+   xlim(c(80,280)) +
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
Warning messages:
1: Removed 6 rows containing non-finite values (stat_bin).
 # r quitó 6 filas que estaban ausentes, o sea solo graica los valores
existentes
2: Removed 2 rows containing missing values (geom_bar).
```



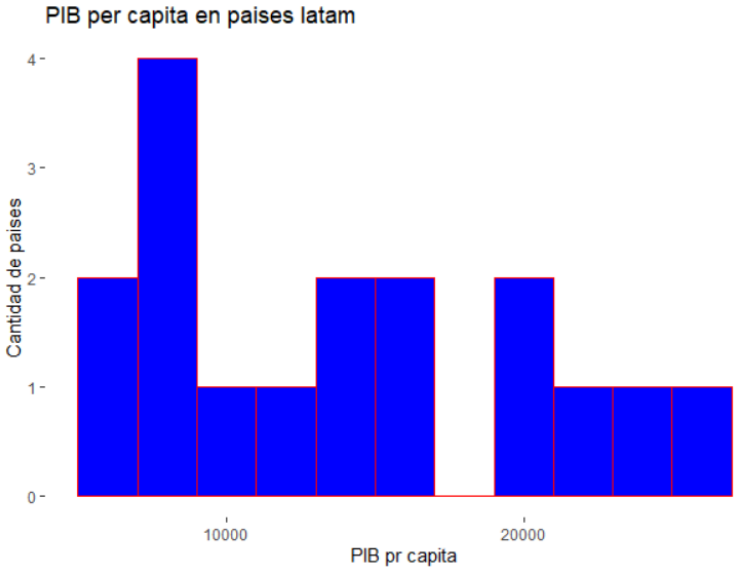
Detalles de la grafica:

- geom: describir el tipo de gráfica que se va a imprimir.
- **aes():** contenido estético del gráfico. Es decir, la función le dará indicios a ggplot2 sobre cómo dibujar las formas y tamaños
- **fill** color de barra
- **color** contorno de barra
- **binwidth** ancho de barra
- **labs()** etiquetas del eje (x,y)
- **title** nombre del histograma
- **xlim** escalas en el eje x
- **theme():** color de fondo

18. EDA con dataset proyecto – histogramas – ggplot2

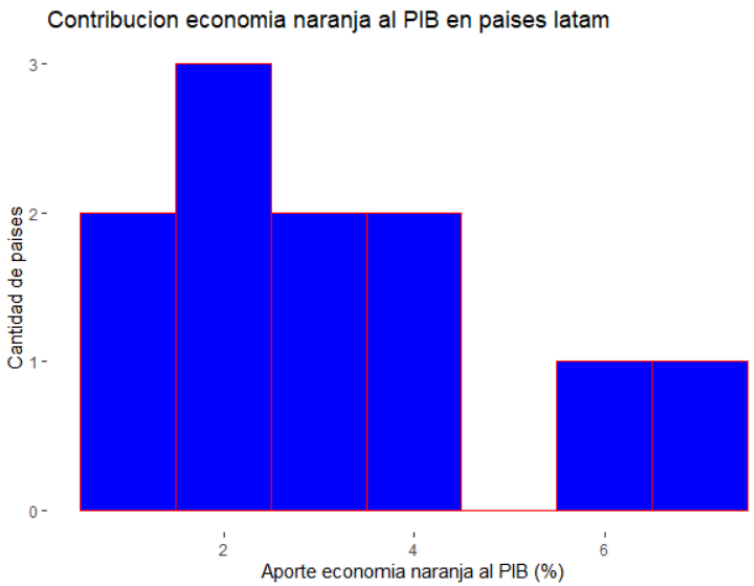
Vhtdnvhn

```
> ggplot() + geom_histogram(data = orangeec,
+                           aes(x=GDP.PC), fill = "blue", color= "red",
+                          binwidth = 2000) +
+   labs(x = "PIB per capita",
+        y ="Cantidad de paises",title = "PIB per capita en paises latam")+
+   theme(legend.position = "none") +
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
```




```
> ggplot() + geom_histogram(data = orangeec,
+                           aes(x=Creat.Ind...GDP), fill = "blue", color= "red",
+                           binwidth = 1) +
+   labs(x = "Aporte economia naranja al PIB (%)",
```

```
+        y = "Cantidad de paises", title = "Contribucion economia naranja al PIB en
paises latam") +
+   theme(legend.position = "none") +
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
Warning message:
Removed 6 rows containing non-finite values (stat_bin).
>                           aes(x=V2), fill = "blue", color= "red",
```

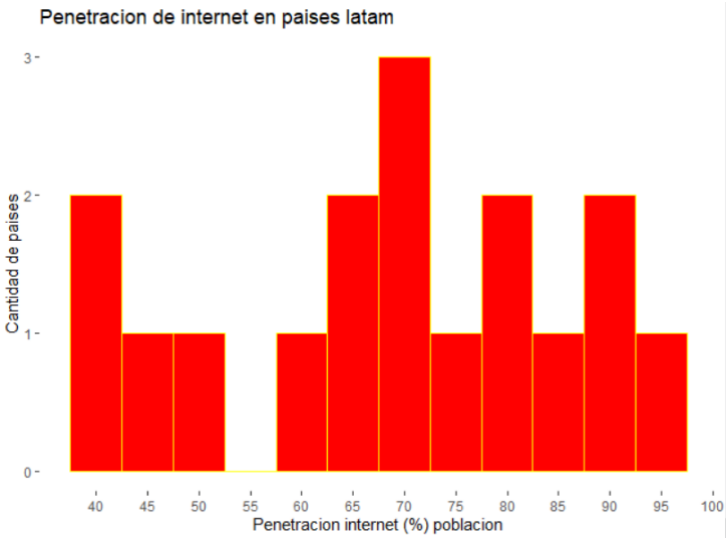


2% de aporte hay 3 paises

Hay 1 pais que aporta al 6% y otro que aporta la 7%

Veamos la distribucion de frecuencia en cuanto a la penetración de internet

```
> ggplot() + geom_histogram(data = orangeec,
+                           aes(x=Internet.penetration...population), fill
= "red", color= "yellow",
+                           binwidth = 5) +
+   # para poner los números en las etiquetas que faltan
+   scale_x_continuous(breaks = seq(40, max(100), 5))+
+   labs(x = "Penetracion internet (%) poblacion",
+        y = "Cantidad de paises", title = "Penetracion de internet en pais
es latam") +
+   theme(legend.position = "none") +
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
```



No hay países que tengan el 55% de penetracion de internet en la pobacion

El 95% de la poblacion de 1 pais tiene acceso a internet

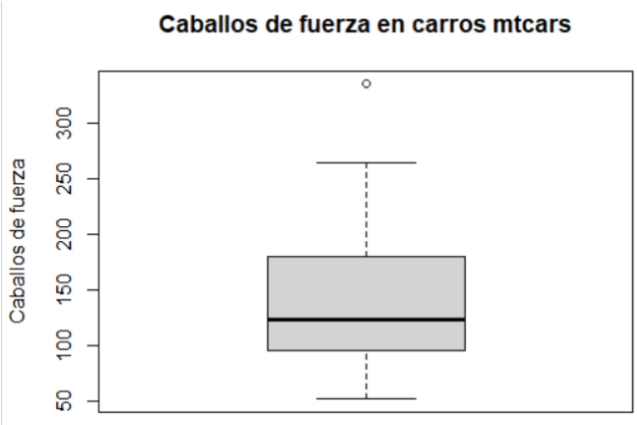
19. EDA con box plot – ggplot2

Para realizar EDA con un box plot dentro de R debemos utilizar la función boxplot, los argumentos que debemos pasarle son:

- la información que vamos a explorar.
- ylab: título para el eje y.
- main: título de la gráfica. También podemos usar ggplot2 para crear un Box Plot.

Veamos la forma de mtcars con boxplot sin cargar ningún paquete

```
boxplot(mtcars$hp, ylab = "Caballos de fuerza",
        main = "Caballos de fuerza en carros mtcars")
```



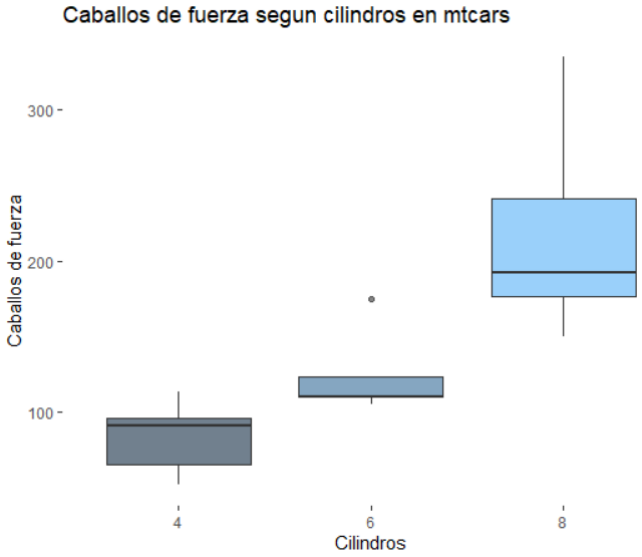
Hay una maximo como a 260, hay un outlier de 335 caballos de fuerza, un maserati (muy util usar los boxplots para encontrar los outliers)

Ahora graficamos usando ggplot2, recordemos que **para los boxplots siempre debemos cruzar una variable numérica en X y una categórica en Y**

Ggplot para boxplot mas bonitos

```
> ggplot(mtcars, aes(x=as.factor(cyl),y=hp,fill=cyl))+
+   geom_boxplot(alpha=0.6) +
```

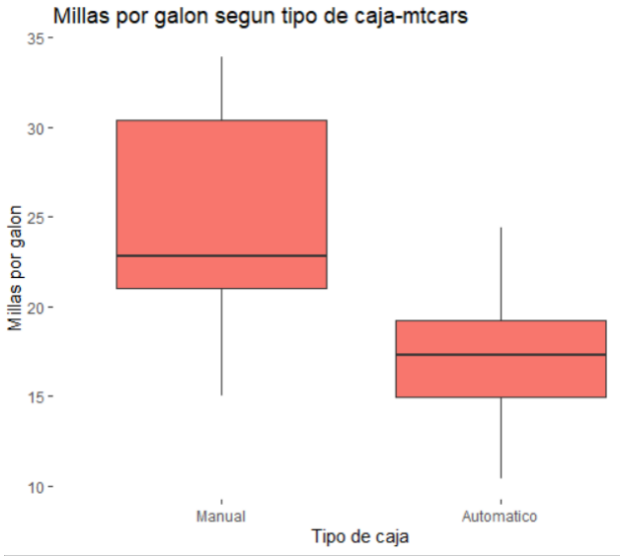
```
+   labs(x="Cilindros", y="Caballos de fuerza",
+        title = "Caballos de fuerza segun cilindros en mtcars") +
+   theme(legend.position = "none") +
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
```



Alpha para cambiar la transparencia de os colores

Nota: cuando tenemos cajas "achatadas" tenemos datos uniformes, para caja "alargadas" tenemos datos inconsistentes

```
> ggplot(mtcars, aes(x=am, y=mpg, fill="red")) +
+   geom_boxplot()+
+   labs(x="Tipo de caja", y="Millas por galon",
+        title = "Millas por galon segun tipo de caja-mtcars") +
+   theme(legend.position = "none") +
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
> mtcars$am <- factor(mtcars$am, levels = c(1,0),
+                     labels = c("Manual", "Automatico"))
```



Los carros de caja automática están menos dispersos

20. EDA con dataset proyecto – oxplot - dplyr

El boxplot cruza variables categoricas con numerios y en el dataset orangeec todas son variables numéricas

Entonces clasificamos los países según el pib per capita de cada país, usando promedios

```
> economy = mean(orangeec$GDP.PC)
> economy
[1] 14052.94
```

El promedio pir per capta de los países de latimoamerica es 14052.94 para cada habitante al año

```
> orangeec <- orangeec %>%  # va a pasar a mutate
+   mutate(Strong_economy = ifelse(GDP.PC < economy,  # creamos una columna
+                                  "Por debajo promedio pib per capita",
+                                  "Sobre-arriba promedio pib per capita"))
```

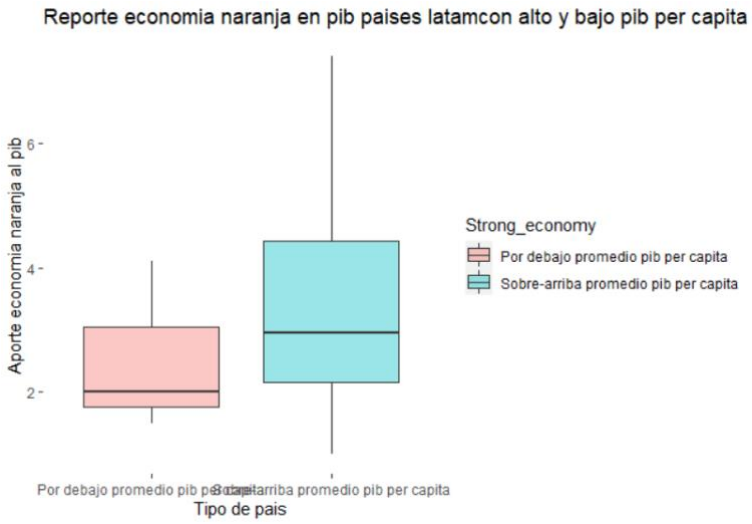
Creat.Ind...GDP	Inflation	Unemployment	X..pop.below.poverty.line	Internet.penetration...population	Median.age	X..pop.25.54	Education.invest...GDP	Strong_economy
NA	4.4	2.3	59.3	42.1	22.1	34.12	2.8	Por debajo promedio pib per capita
6.3	0.9	5.5	23.0	69.7	29.2	40.35	3.2	Sobre-arriba promedio pib per capi
NA	1.0	7.0	32.7	57.7	27.1	39.23	3.5	Por debajo promedio pib per capita
1.5	2.8	6.7	22.7	67.6	28.0	40.19	3.8	Por debajo promedio pib per capita
1.0	6.2	7.3	9.7	88.2	35.0	39.34	4.4	Sobre-arriba promedio pib per capi
3.3	4.3	10.5	28.0	63.2	30.0	41.91	4.5	Sobre-arriba promedio pib per capi
NA	3.9	6.5	29.6	43.0	25.7	40.24	4.5	Por debajo promedio pib per capita
2.2	2.2	7.0	14.4	77.5	34.4	43.08	4.9	Sobre-arriba promedio pib per capi
2.0	0.4	4.6	21.5	79.9	27.7	39.59	5.0	Por debajo promedio pib per capita
4.1	3.6	6.5	22.2	89.6	28.2	41.08	5.0	Por debajo promedio pib per capita
7.4	6.0	3.6	46.2	65.0	28.3	40.81	5.3	Sobre-arriba promedio pib per capi
3.8	25.7	8.1	25.7	93.1	31.7	39.38	5.9	Sobre-arriba promedio pib per capi
2.6	3.4	11.8	4.2	70.7	32.0	43.86	5.9	Sobre-arriba promedio pib per capi
NA	3.9	5.9	29.6	38.2	23.0	36.63	5.9	Por debajo promedio pib per capita

Haremos los boxplots

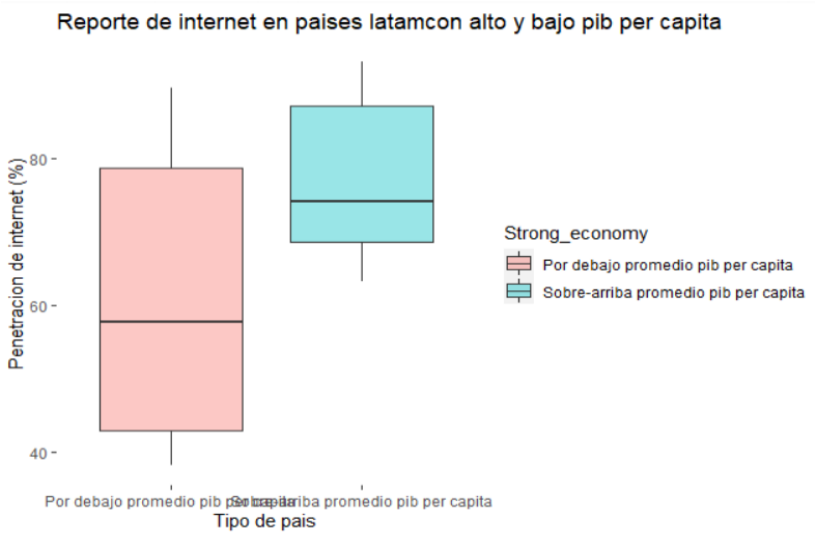
Una será los países que están por arriba del promedio del pib per capita y el otro por debajo

```
FALTA
+   geom_boxplot(alpha = 0.4) +
+   labs(x= "Tipo de pais", y="Aporte economia naranja al pib",
```

```
+        title="Reporte economia naranja en pib paises latamcon alto y bajo
 pib per capita") +
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
Warning message:
Removed 6 rows containing non-finite values (stat_boxplot).
```




```
>  ggplot(orangeec, aes(x= Strong_economy,
+   y= Internet.penetration...population, fill = Strong_economy))+
+   geom_boxplot(alpha = 0.4) +
+   labs(x= "Tipo de pais", y="Penetracion de internet (%)",
+        title="Reporte de internet en paises latamcon alto y bajo pib per
capita")+
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
```



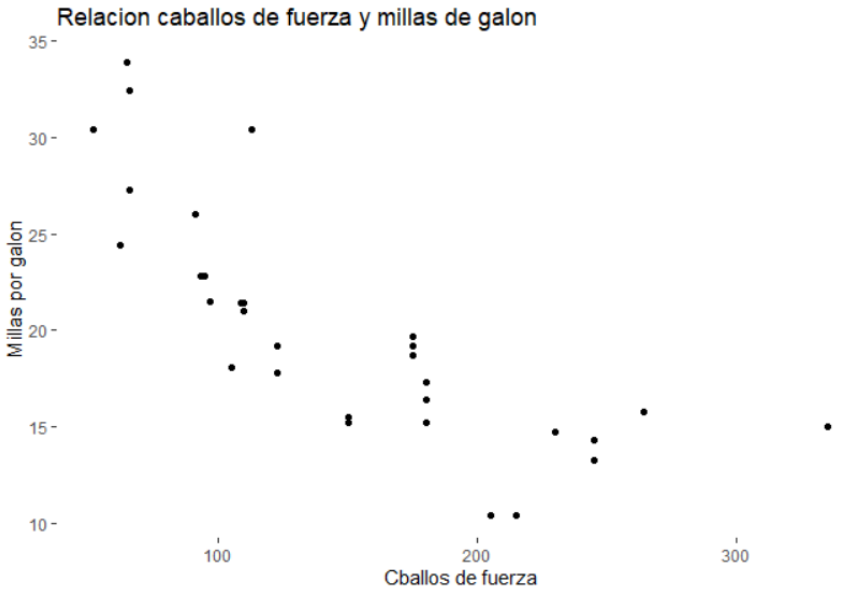
Los datos que tienen mayor al promedio son mas homogéneos y tienen mayor penetraciond e internet

Para el celeste: El 25% de los países que están sobre tiene una epenetracion menor al 70% o el 75% tienen una penetración por encima del 70%

21. EDA con graficas de dispersión con mas de dos variables – ggplot2

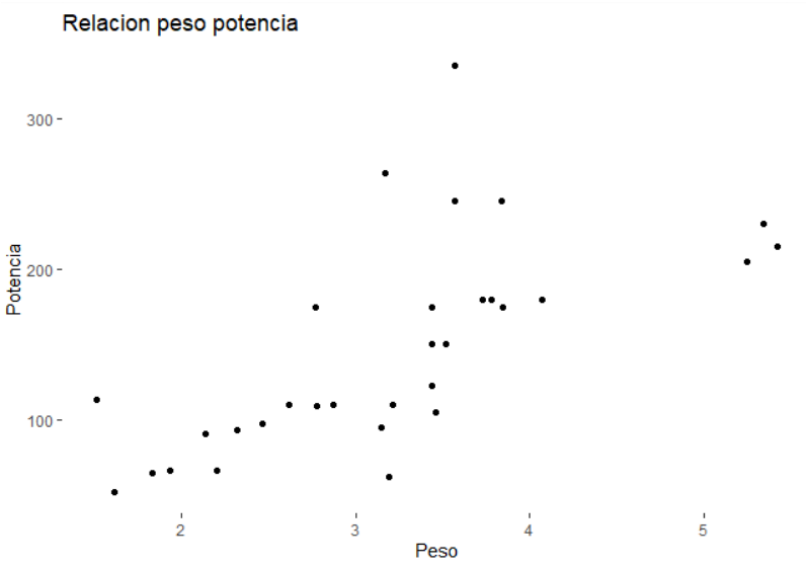
N hgn ghnhg

```
> ggplot(mtcars, aes(hp,mpg)) +
+   geom_point() +
+   labs(x = "Cballos de fuerza", y= "Millas por galon",
+        title = "Relacion caballos de fuerza y millas de galon")+
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
```



Los carros que tieen mas caballos tienen menos eficiencia, menos caballos recorren mas millas por galon

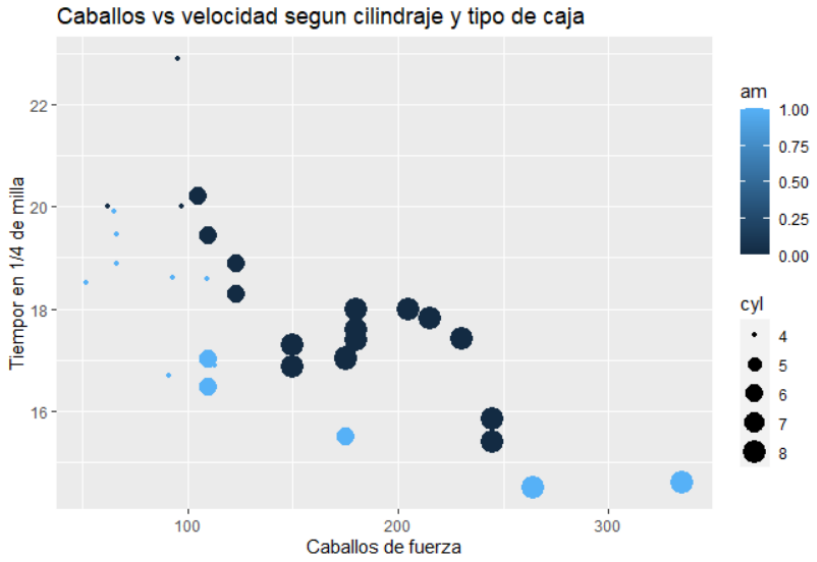
```
> ggplot(mtcars, aes(wt,hp)) +
+   geom_point() +
+   labs(x = "Peso", y= "Potencia",
+        title = "Relacion peso potencia")+
+   theme(panel.background = element_blank(),
+         panel.grid.major = element_blank(),
+         panel.grid.minor = element_blank())
```



Los carros que son mas pesados tienen mas potencia

Qset es la velocidad que se tarda para recorrer un cuarto de mila

```
> ggplot(mtcars, aes(hp,qsec)) +
+   geom_point(aes(color = am, size = cyl)) +
+   labs(x = "Caballos de fuerza", y= "Tiempor en 1/4 de milla",
+        title = "Caballos vs velocidad segun cilindraje y tipo de caja")
```



Caracteriwsticas de la grafica

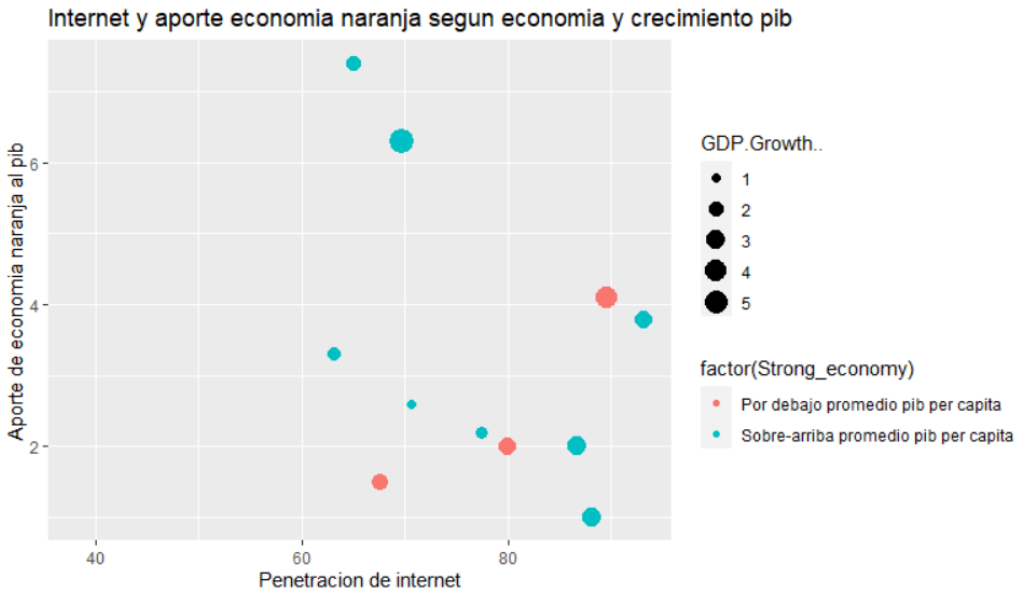
- Los carros que tienen mas caballos se tardan menos tiempo
- los carros celestes son manuales y los azules son automáticos
- El tamanio de la burbuja son los cilindros

22. EDA con dataset proyecto usando graficas de dispersión con mas de 2 variables – ggplot - plotly

```
> ggplot(orangeec,aes(Internet.penetration...population,Creat.Ind...GDP))+
+   geom_point(aes(color = factor(Strong_economy), size = GDP.Growth..)) +
+   labs(x="Penetracion de internet",y="Aporte de economia naranja al pib",
```

```
+        title = "Internet y aporte economia naranja segun economia y creci
miento pib")
Warning message:
Removed 6 rows containing missing values (geom_point).
```

Vythbht



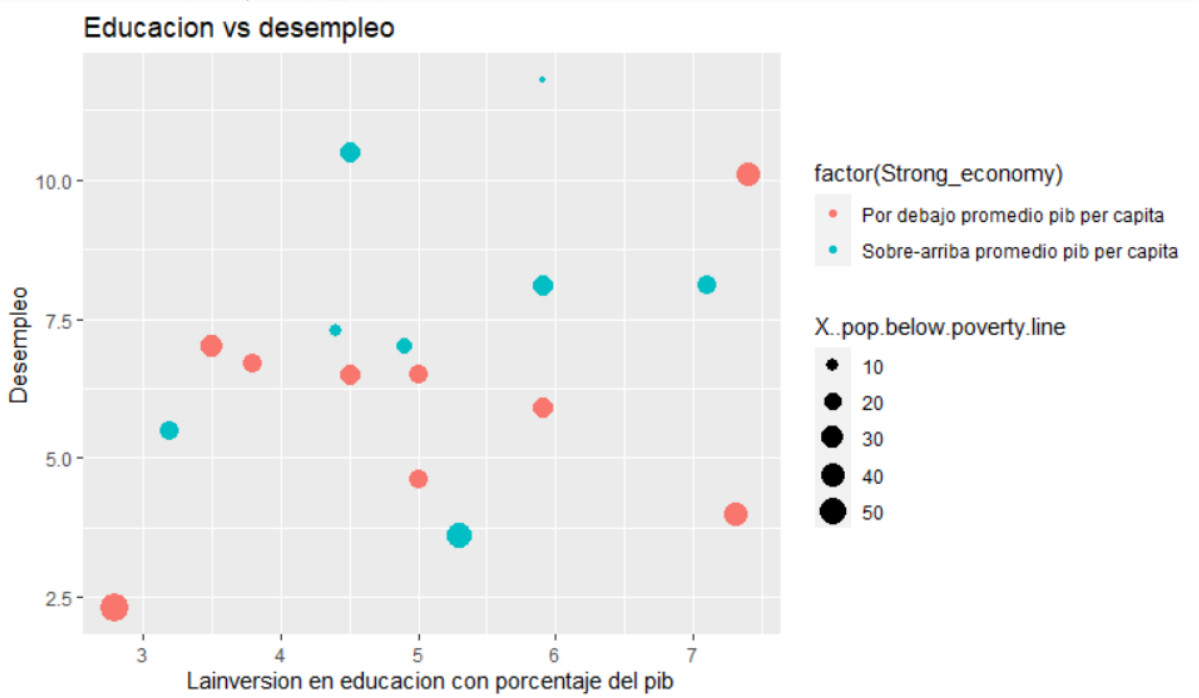
Tanamio de las burbujas es el crecimiento que ha tenido el país en su pib el utimo a;o

Un país con una enetracion de 70% de internet tuvo un crecimiento de 5% del aporte

RETO

- Inversión x
- Y desempleo
- Color strong ecomony
- Tamaño de la burbuja % por debajo de la línea de pobreza

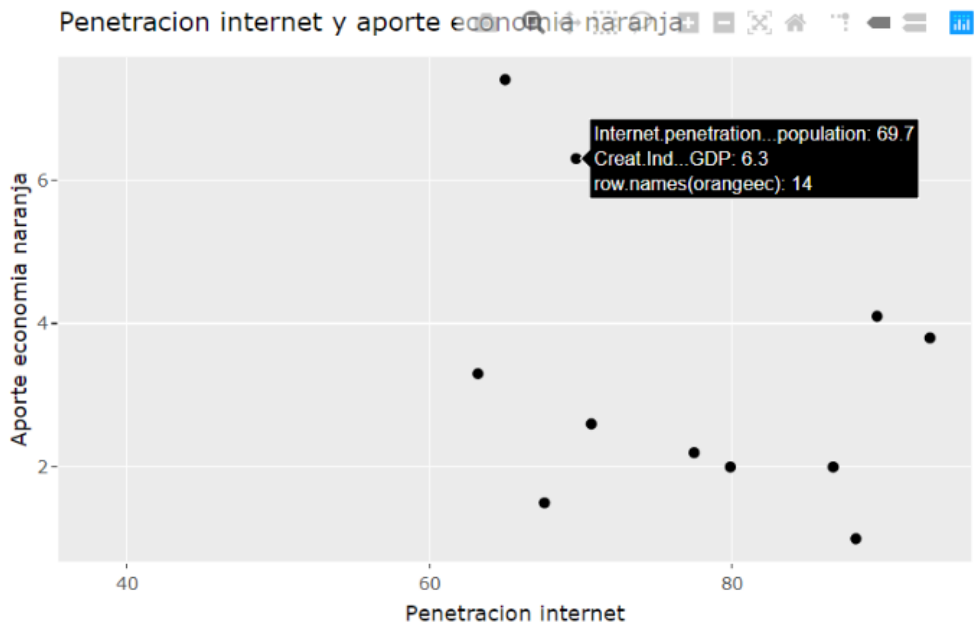
```
> ggplot(orangeec, aes(Education.invest...GDP,Unemployment)) +
+   geom_point(aes(color = factor(Strong_economy), size = X..pop.below.pove
rty.line)) +
+   labs(x="Lainversion en educacion con porcentaje del pib",y="Desempleo",
+        title = "Educacion vs desempleo")
```



Haciendo un scatterplot interactivo:

Instalamos plotly: para mas información

```
> my_graph <- ggplot(orangeec, aes(Internet.penetration...population,
+                                  Creat.Ind...GDP, label = row.names(orang
eec))) +
+   geom_point() +
+   labs(x= "Penetracion internet", y ="Aporte economia naranja",
+        title = "Penetracion internet y aporte economia naranja")
> my_graph
Warning message:
Removed 6 rows containing missing values (geom_point).
> p = ggplotly(my_graph)
> p
```

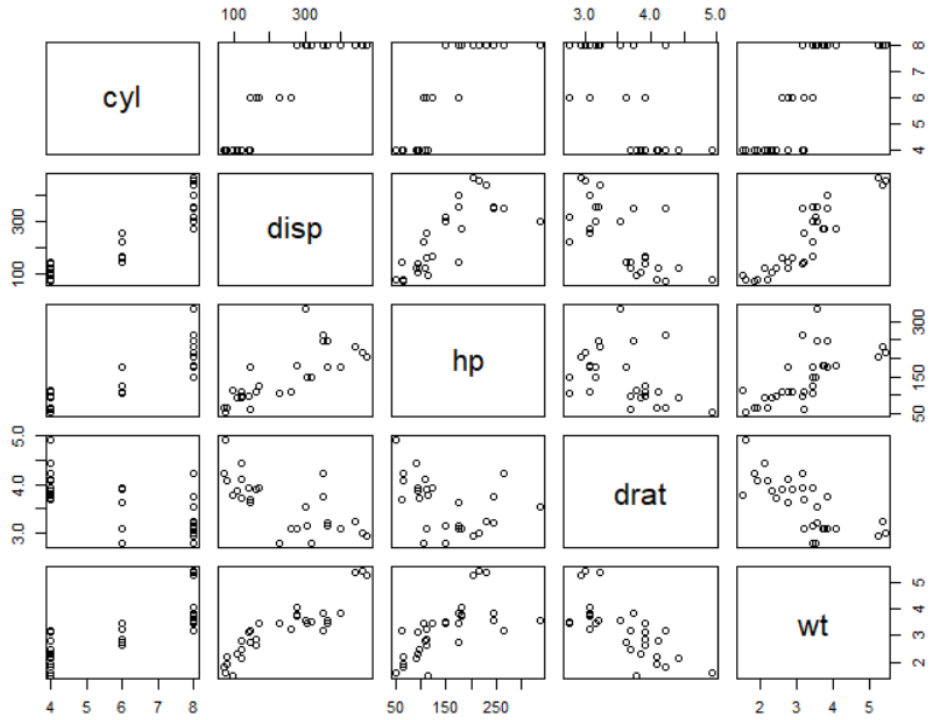


23. Buscando correlaciones con pairs

La función pairs nos permite cruzar todas las variables del dataset a modo de tabla donde el eje x de una gráfica corresponde a la columna donde se encuentra y el eje y a la fila.

- **select():** para seleccionar variables o columnas.
- **filter:** para filtrar datos de un dataset, retorna las filas que pasen el filtro.

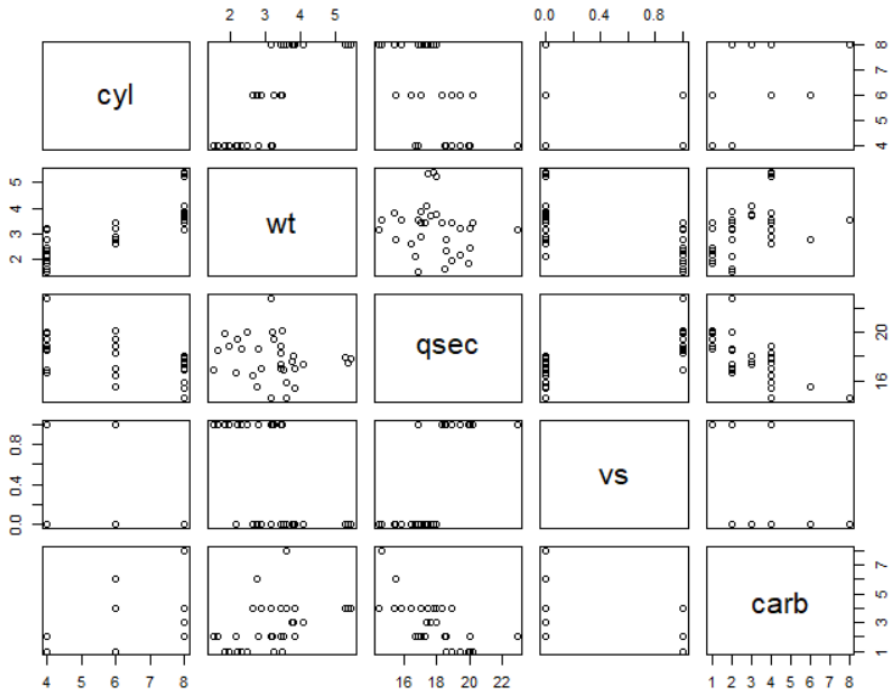
```
> pairs(mtcars[,2:6])
```



En el cuadrado 2*1 vemos que en el eje x están los cilindros y en el eje y están las disposiciones y si sucesivamente

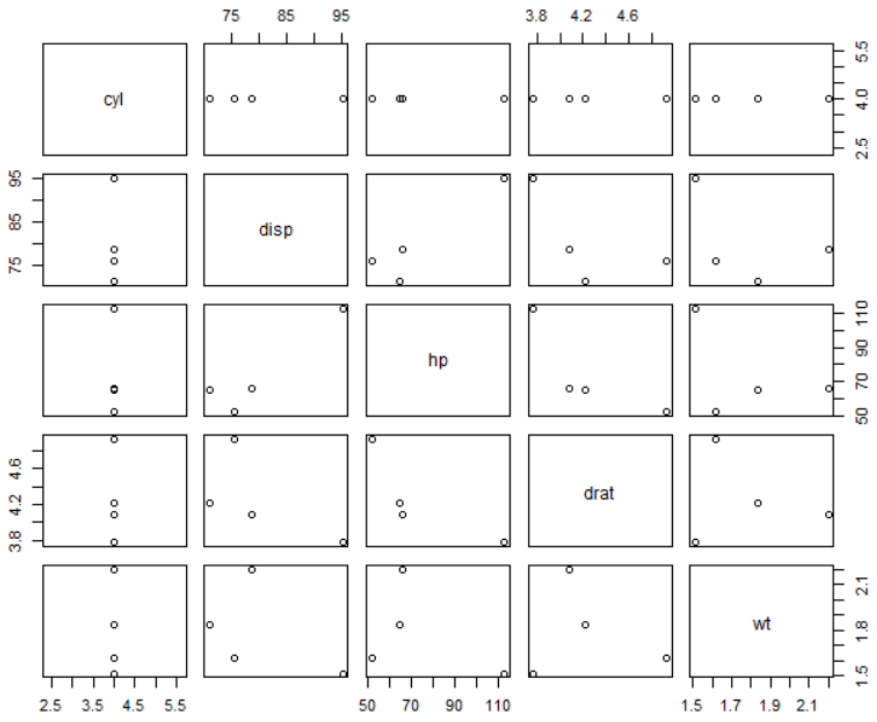
Mas caballos(hp) mas disp. Tiene correlcion positiva

```
# Seleccionando las columnas de interés y creando nuevo subset
> newdata <- subset(mtcars, select = c(2,6,7,8,11))
> pairs(newdata)
```




```
> eficientes <- filter(mtcars, mpg >=30)
> eficientes
                mpg cyl disp  hp drat    wt  qsec vs am gear carb
Fiat 128       32.4   4 78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic    30.4   4 75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla 33.9   4 71.1  65 4.22 1.835 19.90  1  1    4    1
Lotus Europa   30.4   4 95.1 113 3.77 1.513 16.90  1  1    5    2
```

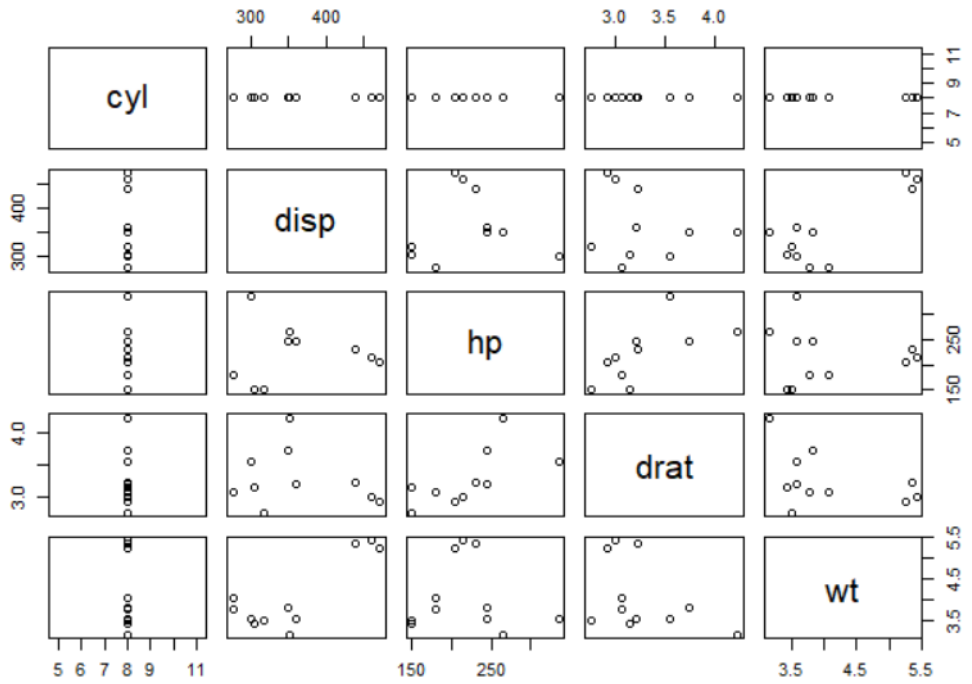
```
> pairs(eficientes[,2:6])
```




```
> merc <- mtcars %>%
+   filter(mtcars$mpg <17)
> merc
                  mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Duster 360       14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 450SE       16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
```

```
Merc 450SLC          15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
Cadillac Fleetwood   10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental  10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Chrysler Imperial    14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Dodge Challenger     15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
AMC Javelin          15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Camaro Z28           13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Ford Pantera L       15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Maserati Bora        15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
```

```
> pairs(merc[,2:6])
```



La correlación se usa con la función cor()

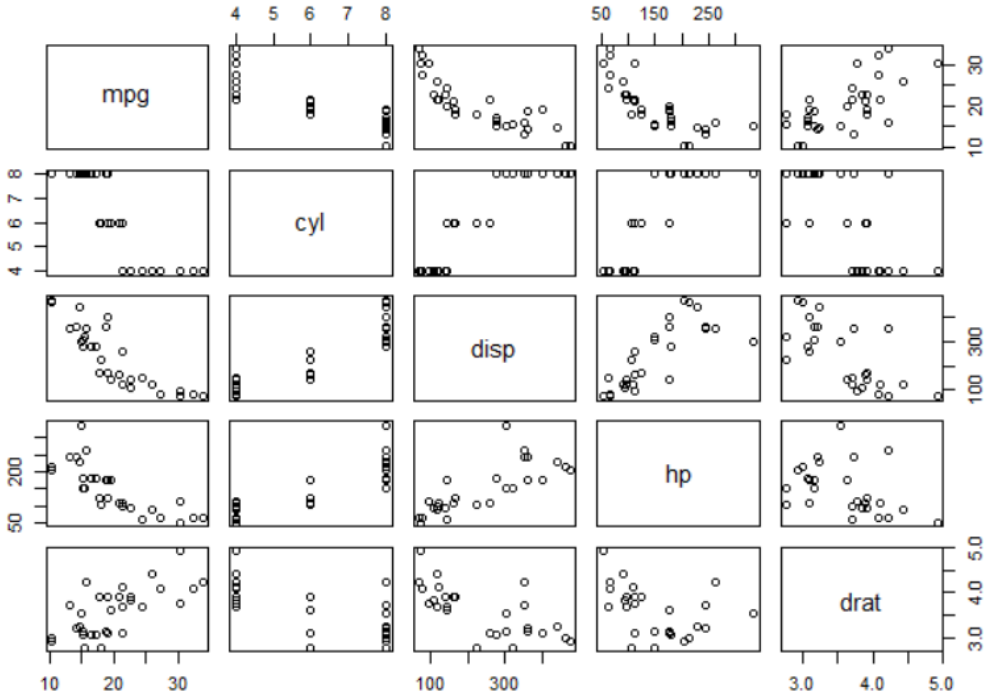
La correlación se mueve desde -1 a 1

Si se acerca a 0 no hay correlación

24. Corfirmando correlaciones con la función cor:

Recordemos que la correlacion se mueve entre -1 (negativa), 0 (no hay correlacion) y 1 (positiva).

```
> pairs(mtcars[,1:5])
```



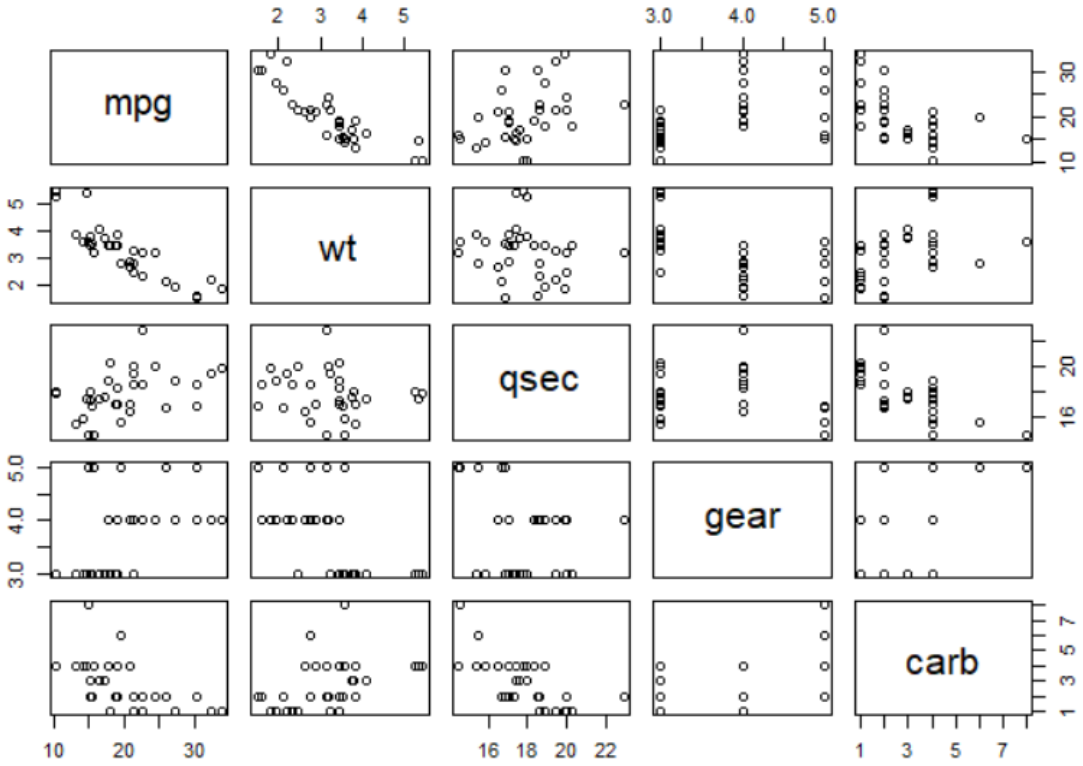

```
> cor(mtcars[,1:5])
            mpg        cyl       disp         hp       drat
mpg   1.0000000 -0.8521620 -0.8475514 -0.7761684  0.6811719
cyl  -0.8521620  1.0000000  0.9020329  0.8324475 -0.6999381
disp -0.8475514  0.9020329  1.0000000  0.7909486 -0.7102139
hp   -0.7761684  0.8324475  0.7909486  1.0000000 -0.4487591
drat  0.6811719 -0.6999381 -0.7102139 -0.4487591  1.0000000
```

disp y mpg -0.8 correlacion negativa: a mayor disp menos millas por galon corren los carros

```
> cor(newdata2)
            mpg         wt       qsec       gear       carb
mpg   1.0000000 -0.8676594  0.4186840  0.4802848 -0.5509251
wt   -0.8676594  1.0000000 -0.1747159 -0.5832870  0.4276059
qsec  0.4186840 -0.1747159  1.0000000 -0.2126822 -0.6562492
gear  0.4802848 -0.5832870 -0.2126822  1.0000000  0.2740728
carb -0.5509251  0.4276059 -0.6562492  0.2740728  1.0000000
```

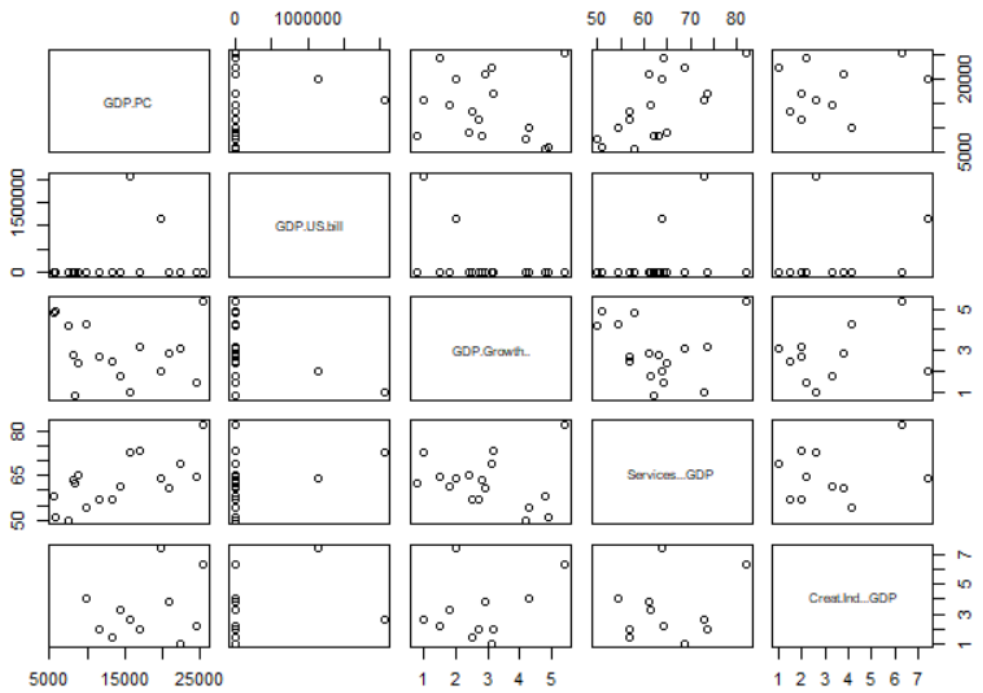
```
# forma gráfica
pairs(newdata)

# forma numérica
cor(newdata)
```



25. Buscando correlaciones con pairs en dataset proyecto

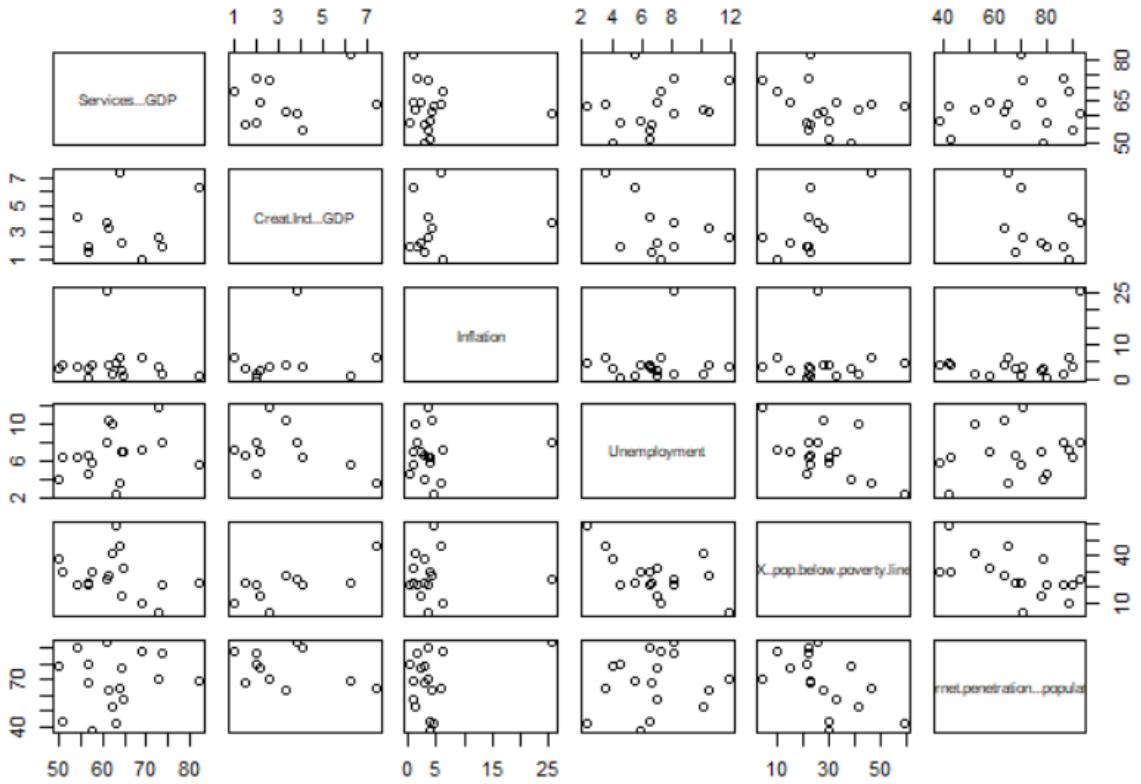
```
> pairs(orangeec[,2:6])
```



4,1 las burbujas van subiendo entre el aporte de servicio al pib y pib per capita

3,5 el comportamiento no es claro El aporte de la economía naranja al pib y el crecimiento del pib

```
> pairs(orangeec[,5:10])
```



Desempleo y el aporte de las industrias creativas esta desenciendo mas de uno menos de otro

Descendente entre ipenetracion nternet y el aporte al pib

6,2 no se sabe la correlación

26. Confirmando correlaciones con la funcion cor en dataset proyecto

```
> cor(orangeec[,2:6])
                  GDP.PC GDP.US.bill GDP.Growth.. Services...GDP Creat.Ind...GDP
GDP.PC         1.0000000   0.1680323   -0.1611431      0.6733943               NA
GDP.US.bill    0.1680323   1.0000000   -0.4213369      0.3039234               NA
GDP.Growth..  -0.1611431  -0.4213369    1.0000000     -0.1430906               NA
Services...GDP 0.6733943   0.3039234   -0.1430906      1.0000000               NA
Creat.Ind...GDP        NA          NA           NA             NA                1
```

Observamos los valores NA en algunas variables, pero necesitamos números en todas las celdas., esto lo solucionamos pasando use="complete.obs" como parámetro adicional a cor

No considerando los NA

```
> cor(orangeec[,2:6], use='complete.obs')
                   GDP.PC GDP.US.bill GDP.Growth.. Services...GDP Creat.Ind
GDP.PC         1.00000000 -0.04987362    0.1186869      0.6437520 0.2606328
GDP.US.bill   -0.04987362  1.00000000   -0.5254890      0.2552986 0.2421848
GDP.Growth..   0.11868685 -0.52548898    1.0000000      0.2552048 0.3124484
Services...GDP 0.64375196  0.25529859    0.2552048      1.0000000 0.2201699
Creat.Ind...GDP 0.26063277 0.24218479    0.3124484      0.2201699 1.0000000
```

Hngbnhgn

```
> cor(orangeec[,5:10], use='complete.obs')
                      Services...GDP Creat.Ind...GDP  Inflation Unemployment
Services...GDP             1.0000000       0.2201699 -0.1985176   0.17703222
Creat.Ind...GDP            0.2201699       1.0000000  0.1189514  -0.41885405
Inflation                  0.1985176       0.1189514  1.0000000   0.14179995
Unemployment               0.1770322      -0.4188541  0.1418000   1.00000000
X..pop.below.poverty.line -0.2534107       0.7072581  0.1702550  -0.56935718
Internet.penetration      -0.1453060      -0.3435164  0.4459355  -0.02534538
                     X..pop.below.poverty.line Internet.penetration
Services...GDP                      -0.2534107          -0.14530602
Creat.Ind...GDP                      0.7072581          -0.34351645
Inflation                            0.1702550           0.44593550
Unemployment                        -0.5693572          -0.02534538
X..pop.below.poverty.line            1.0000000          -0.30802896
Internet.penetration                -0.3080290           1.00000000
```

```
> cor(newdata3, use='complete.obs')
                      Services...GDP Creat.Ind...GDP Internet.penetration
Services...GDP           1.000000000       0.2201699           -0.1453060
Creat.Ind...GDP          0.220169925       1.0000000           -0.3435164
Internet.penetration    -0.145306022      -0.3435164            1.0000000
Median.age               0.356375194      -0.4412121            0.3581014
X..pop.25.54             0.338980318      -0.1306267           -0.1824810
Education.invest...GDP  -0.003218743      -0.1598673            0.4355714
                        Median.age X..pop.25.54 Education.invest...GDP
Services...GDP           0.3563752    0.3389803           -0.003218743
Creat.Ind...GDP         -0.4412121   -0.1306267           -0.159867273
Internet.penetration     0.3581014   -0.1824810            0.435571370
Median.age               1.0000000    0.2667101            0.216822054
X..pop.25.54             0.2667101    1.0000000            0.531343612
Education.invest...GDP   0.2168221    0.5313436            1.000000000
```

27. Protegiendonos de los peligros del promedio

Viendo el resumen de mtcars

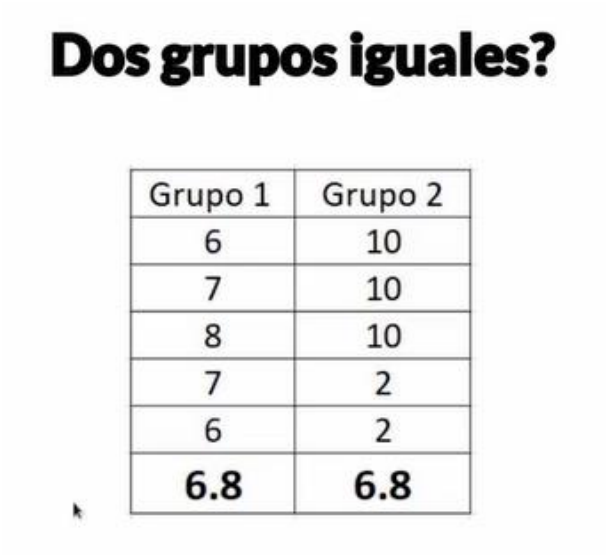
```
> summary(mtcars)
      mpg             cyl             disp             hp       
 Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0  
 1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5  
 Median :19.20   Median :6.000   Median :196.3   Median :123.0  
 Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7  
 3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0  
 Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0  
      drat             wt             qsec             vs        
 Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000  
 1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000  
 Median :3.695   Median :3.325   Median :17.71   Median :0.0000  
 Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375  
 3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000  
```

```
Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000
      am              gear            carb
Min.   :0.0000   Min.   :3.000   Min.   :1.000
1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000
Median :0.0000   Median :4.000   Median :2.000
Mean   :0.4062   Mean   :3.688   Mean   :2.812
3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000
Max.   :1.0000   Max.   :5.000   Max.   :8.000
```

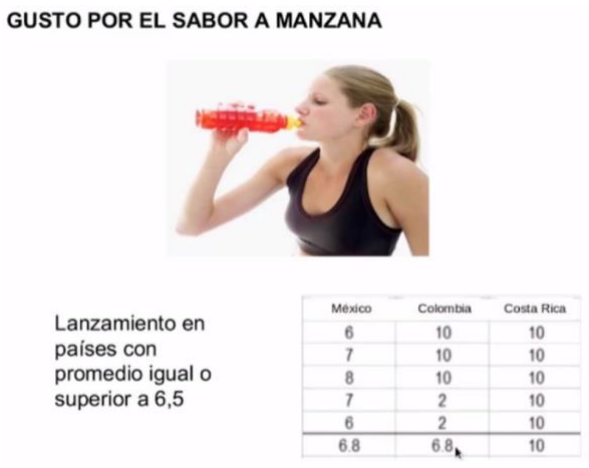
En promedio los carros recorren 20 millas por galon

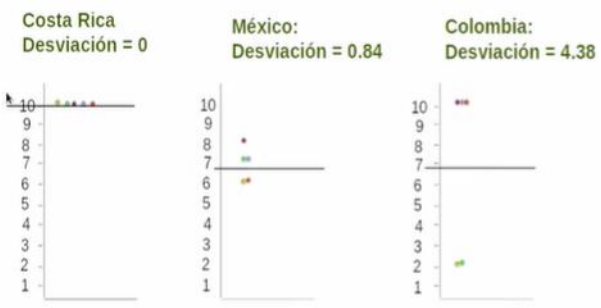
Desviación estándar: es una medida de dispersión, nos indica cuánto pueden alejarse los valores respecto al promedio (media). Es útil para determinar el rango en que puede moverse una determinada variable. (por lo tanto es útil para buscar probabilidades de que un evento ocurra), es adimencional**.**

Podremos encontrar casos donde dos grupos de datos distintos tengan el mismo promedio, pero sus datos son muy diferentes uno del otro. No es lo mismo un grupo de datos donde su desviación es menor a 1, que aquel donde sus datos tienen una desviación de 4 o 6 puntos.



el grupo 1 los numeros están mas cercnos a diferencia del grupo 2





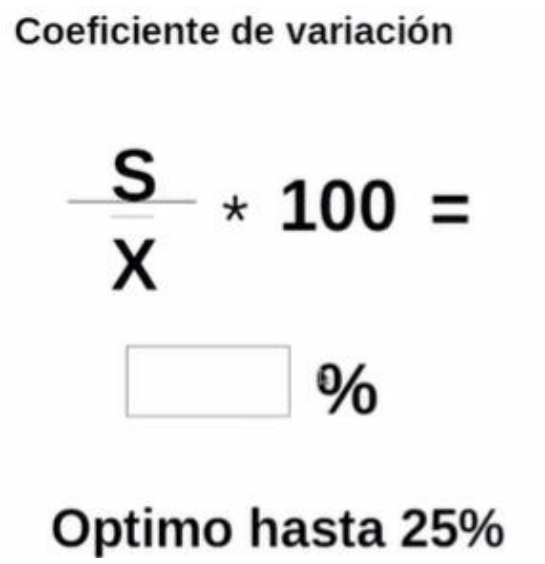
La calificación 10 no esta desviado

En mexico: estan cerca los valores

En colombia: los datos están muy distanciadas, tiene bastante desviación

Coeficiente de variación: este expresa la desviación estándar como porcentaje de la media, mostrando una mejor interpretación porcentual del grado de variabilidad que la desviación estándar (que tanto esta desviado el promedio respecto a los datos)

El coeficiente de variación nos dice que tan desviados están los datos



Si el coeficiente no pasa de 25 los datos están cercamos al promedio , son homogéneos o parecidos

Veamos las desviaciones en mtcars con sd

```
> desviacion <- sd(mtcars$mpg)
> desviacion
[1] 6.026948
> prom <- mean(mtcars$mpg)
> prom
[1] 20.09062
> coefdevariacion = (desviacion)/ prom *100
> coefdevariacion
[1] 29.99881
```

Hay carros que recorren muchas mas millas por galon y otros no, la desviación supera el 25%

28. Eliminacon de NA's para hacer los cálculos

Vamos a ver que tanto están desviados nuestros datos del promedio en el dataset de Economía naranja.

```
> desv <- sd(orangeec$Internet.penetration...population)
> desv
[1] 17.27419
> prom <- mean(orangeec$Internet.penetration...population)
> prom
[1] 68.41765
> coefici <- desv / prom * 100
> coefici
[1] 25.24816
```

Al momento de sacar el promedio de nuestro dataset orangeec encontramos variables que tienen valores NA, para que estos no afecten nuestro cálculo solamente debemos añadir como argumento na.rm=TRUE.

Para comunicar hallazgos en la variable resultaría mucho mejor aporyarnos con la mediana o con los cuartiles que con el promedio

```
> mean(orangeec$Creat.Ind...GDP, na.rm = TRUE)
[1] 3.290909
> prome <- mean(orangeec$Creat.Ind...GDP, na.rm = TRUE)
> sd(orangeec$Creat.Ind...GDP)
[1] NA
> sd(orangeec$Creat.Ind...GDP, na.rm = TRUE)
[1] 2.007712
> desv <- sd(orangeec$Creat.Ind...GDP, na.rm = TRUE)
> coeficiente <- (desv/prome)*100
> coeficiente
[1] 61.00784
```

Hay paises que aportan muchísimo mas que el promedio y tmb otros por debajo del promedio por lo que seria mejor apoyarnos de los cuartiles

29. Estadistica y visualizacion aplicada a analisis de datos de mercadeo

Presentación de resultados hecha a un cliente de una industria automotriz

https://medium.com/@soniaardila1/c%C3%B3mo-escogemos-nuestra-nave-espacial-una-c%C3%B3smica-historia-de-mercadeo-e26f5599263d

análisis de datos de los clientes en el punto de venta

30. Generando tablas , fitrando y seleccionando datos - dplyr parte1

Ajustar los datos en mtcars para tener mejores visualizaciones.

```
> eficientes <- mean(mtcars$mpg)
> eficientes
[1] 20.09062
> mtcars <- mtcars %>%
+   mutate(mas_eficientes=ifelse(mpg<eficientes,
+                                "bajo promedio","en o sobre promedio"))
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb	mas_eficientes
1	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4	en o sobre promedio
2	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4	en o sobre promedio
3	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1	en o sobre promedio
4	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1	en o sobre promedio
5	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2	bajo promedio
6	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1	bajo promedio
7	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4	bajo promedio
8	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2	en o sobre promedio
9	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2	en o sobre promedio
10	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4	bajo promedio
11	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4	bajo promedio
12	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3	bajo promedio
13	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3	bajo promedio
14	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3	bajo promedio
15	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4	bajo promedio

Carros mas veloces que recorren el cuarto de milla en menos de 16 segundos

```
> mas_veloces <- mtcars[mtcars$qsec<16,]
     # velocidad que se toma en el cuarto de milla
> mas_veloces
    mpg cyl disp  hp drat   wt  qsec vs am gear carb mas_eficientes
7  14.3   8  360 245 3.21 3.57 15.84  0  0    3    4  bajo promedio
24 13.3   8  350 245 3.73 3.84 15.41  0  0    3    4  bajo promedio
29 15.8   8  351 264 4.22 3.17 14.50  0  1    5    4  bajo promedio
```

```
30 19.7   6  145 175 3.62 2.77 15.50  0  1    5    6  bajo promedio
31 15.0   8  301 335 3.54 3.57 14.60  0  1    5    8  bajo promedio
```

Clasificando a los mas veloces

```
> mtcars <- mtcars %>%
+   mutate(veloci_cuarto_milla = ifelse(qsec<16,
+                                       "Menos de 16 seg","Mas de 16 seg"))
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb	mas_eficientes	veloci_cuarto_milla
1	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4	en o sobre promedio	Mas de 16 seg
2	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4	en o sobre promedio	Mas de 16 seg
3	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1	en o sobre promedio	Mas de 16 seg
4	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1	en o sobre promedio	Mas de 16 seg
5	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2	bajo promedio	Mas de 16 seg
6	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1	bajo promedio	Mas de 16 seg
7	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4	bajo promedio	Menos de 16 seg
8	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2	en o sobre promedio	Mas de 16 seg
9	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2	en o sobre promedio	Mas de 16 seg
10	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4	bajo promedio	Mas de 16 seg
11	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4	bajo promedio	Mas de 16 seg
12	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3	bajo promedio	Mas de 16 seg
13	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3	bajo promedio	Mas de 16 seg
14	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3	bajo promedio	Mas de 16 seg
15	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4	bajo promedio	Mas de 16 seg

```
> mtcars <- mtcars %>%
+   mutate(peso_kilos = (wt/2)*1000)
> mtcars <- mtcars %>%
+   mutate(Peso = ifelse(peso_kilos <= 1500,
+                        "Livianos", "Pesados"))
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb	mas_eficientes	veloci_cuarto_milla	peso_kilos	Peso
1	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4	en o sobre promedio	Mas de 16 seg	1310.0	Livianos
2	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4	en o sobre promedio	Mas de 16 seg	1437.5	Livianos
3	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1	en o sobre promedio	Mas de 16 seg	1160.0	Livianos
4	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1	en o sobre promedio	Mas de 16 seg	1607.5	Pesados
5	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2	bajo promedio	Mas de 16 seg	1720.0	Pesados
6	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1	bajo promedio	Mas de 16 seg	1730.0	Pesados
7	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4	bajo promedio	Menos de 16 seg	1785.0	Pesados
8	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2	en o sobre promedio	Mas de 16 seg	1595.0	Pesados
9	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2	en o sobre promedio	Mas de 16 seg	1575.0	Pesados
10	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4	bajo promedio	Mas de 16 seg	1720.0	Pesados
11	17.8	6	167.6	123	3.92	3.440	18.90	1	0	4	4	bajo promedio	Mas de 16 seg	1720.0	Pesados
12	16.4	8	275.8	180	3.07	4.070	17.40	0	0	3	3	bajo promedio	Mas de 16 seg	2035.0	Pesados
13	17.3	8	275.8	180	3.07	3.730	17.60	0	0	3	3	bajo promedio	Mas de 16 seg	1865.0	Pesados
14	15.2	8	275.8	180	3.07	3.780	18.00	0	0	3	3	bajo promedio	Mas de 16 seg	1890.0	Pesados
15	10.4	8	472.0	205	2.93	5.250	17.98	0	0	3	4	bajo promedio	Mas de 16 seg	2625.0	Pesados

31. Generando tablas, filtrando y seleccionando datos – dyplyr parte2

Creando nuevas variables

```
> orangeec <- orangeec %>%
  mutate(Crecimiento_GDP = ifelse(GDP.Growth..>=2.5,
                                  "2.5% o más", "Menos de 2.5%"))
> orangeec <- orangeec %>%
  mutate(Anaranjados= ifelse(Creat.Ind...GDP>=2.5,
                             "Mas anaranjados", "Menos anaranjados"))
# Ranking
> orangeec %>%
+   arrange(desc('Create.Ind...GPD'))
       Country GDP.PC GDP.US.bill GDP.Growth.. Services...GDP Creat.Ind...GDP
1    Argentina  20900       637.7          2.9           60.9             3.8
2       Belize   8300      1854.0          0.8           62.2              NA
3      Bolivia   7500        37.1          4.2           50.0              NA
4       Brazil  15600   2055000.0          1.0           72.8             2.6
5        Chile  24500       277.0          1.5           64.3             2.2
6     Colombia  14500       309.2          1.8           61.4             3.3
7   Costa Rica  16900        58.1          3.2           73.5             2.0
8      Ecuador  11500       102.3          2.7           56.9             2.0
9  El Salvador   8900        28.0          2.4           64.9              NA
10   Guatemala   8100        75.7          2.8           63.2              NA
11    Honduras   5600        22.9          4.8           57.8              NA
12      Mexico  19900   1149000.0          2.0           64.0             7.4
13   Nicaragua   5800        13.7          4.9           50.8              NA
14      Panama  25400        61.8          5.4           82.0             6.3
15    Paraguay   9800        29.6          4.3           54.5             4.1
16        Peru  13300       215.2          2.5           56.8             1.5
17     Uruguay  22400        58.4          3.1           68.8             1.0
   Inflation Unemployment X..pop.below.poverty.line Internet.penetration
1       25.7          8.1                     25.7                 93.1
2        1.1         10.1                     41.0                 52.3
3        2.8          4.0                     38.6                 78.6
4        3.4         11.8                      4.2                 70.7
5        2.2          7.0                     14.4                 77.5
6        4.3         10.5                     28.0                 63.2
7        1.6          8.1                     21.7                 86.7
8        0.4          4.6                     21.5                 79.9
9        1.0          7.0                     32.7                 57.7
10       4.4          2.3                     59.3                 42.1
11       3.9          5.9                     29.6                 38.2
12       6.0          3.6                     46.2                 65.0
13       3.9          6.5                     29.6                 43.0
14       0.9          5.5                     23.0                 69.7
15       3.6          6.5                     22.2                 89.6
16       2.8          6.7                     22.7                 67.6
17       6.2          7.3                      9.7                 88.2
   Median.age X..pop.25.54 Education.invest Crecimiento_GDP       Anaranjados
1        31.7        39.38              5.9      2.5% o más   Mas anaranjados
2        22.7        36.62              7.4   Menos de 2.5%              <NA>
3        24.3        37.48              7.3      2.5% o más              <NA>
4        32.0        43.86              5.9   Menos de 2.5%   Mas anaranjados
5        34.4        43.08              4.9   Menos de 2.5% Menos anaranjados
6        30.0        41.91              4.5   Menos de 2.5%   Mas anaranjados
```

```
7        31.3        44.03              7.1      2.5% o más Menos anaranjados
8        27.7        39.59              5.0      2.5% o más Menos anaranjados
9        27.1        39.23              3.5   Menos de 2.5%              <NA>
10       22.1        34.12              2.8      2.5% o más              <NA>
11       23.0        36.63              5.9      2.5% o más              <NA>
12       28.3        40.81              5.3   Menos de 2.5%   Mas anaranjados
13       25.7        40.24              4.5      2.5% o más              <NA>
14       29.2        40.35              3.2      2.5% o más   Mas anaranjados
15       28.2        41.08              5.0      2.5% o más   Mas anaranjados
16       28.0        40.19              3.8      2.5% o más Menos anaranjados
17       35.0        39.34              4.4      2.5% o más Menos anaranjados
> TopNajanja <- orangeec %>%
+   filter(Country %in% c('Mexico', 'Panama', 'Argentina',
+                         'Colombia', 'Brazil'))
> TopNajanja
    Country GDP.PC GDP.US.bill GDP.Growth Services Creat.Ind...GDP Inflation
1 Argentina  20900       637.7        2.9     60.9             3.8      25.7
2    Brazil  15600   2055000.0        1.0     72.8             2.6       3.4
3  Colombia  14500       309.2        1.8     61.4             3.3       4.3
4    Mexico  19900   1149000.0        2.0     64.0             7.4       6.0
5    Panama  25400        61.8        5.4     82.0             6.3       0.9
  Unemployment X..pop.below.poverty.line Internet.penetration Median.age
1          8.1                     25.7                93.1       31.7
2         11.8                      4.2                70.7       32.0
3         10.5                     28.0                63.2       30.0
4          3.6                     46.2                65.0       28.3
5          5.5                     23.0                69.7       29.2
  X..pop.25.54 Education.invest...GDP Crecimiento_GDP     Anaranjados
1        39.38                   5.9      2.5% o más Mas anaranjados
2        43.86                   5.9   Menos de 2.5% Mas anaranjados
3        41.91                   4.5   Menos de 2.5% Mas anaranjados
4        40.81                   5.3   Menos de 2.5% Mas anaranjados
5        40.35                   3.2      2.5% o más Mas anaranjados
```

Dv

```
> TopNajanja %>%
+   arrange(desc('Create.Ind...GDP'))
    Country GDP.PC GDP.US.bill GDP.Growth Services Creat.Ind...GDP Inflation
1 Argentina  20900       637.7        2.9     60.9             3.8      25.7
2    Brazil  15600   2055000.0        1.0     72.8             2.6       3.4
3  Colombia  14500       309.2        1.8     61.4             3.3       4.3
4    Mexico  19900   1149000.0        2.0     64.0             7.4       6.0
5    Panama  25400        61.8        5.4     82.0             6.3       0.9
  Unemployment X..pop.below.poverty.line Internet.penetration Median.age
1          8.1                     25.7                93.1       31.7
2         11.8                      4.2                70.7       32.0
3         10.5                     28.0                63.2       30.0
4          3.6                     46.2                65.0       28.3
5          5.5                     23.0                69.7       29.2
  X..pop.25.54 Education.invest...GDP Crecimiento_GDP     Anaranjados
1        39.38                   5.9      2.5% o más Mas anaranjados
2        43.86                   5.9   Menos de 2.5% Mas anaranjados
3        41.91                   4.5   Menos de 2.5% Mas anaranjados
```

```
4         40.81                    5.3   Menos de 2.5% Mas anaranjados
5         40.35                    3.2      2.5% o más Mas anaranjados
```

Nkanvkv

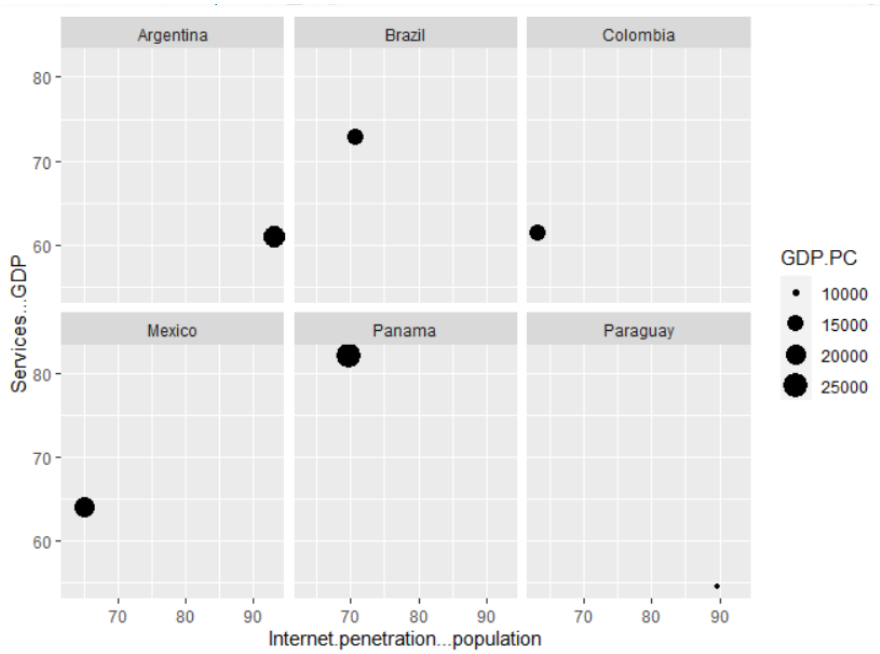
32. Viendo mas información con fase wrap – Parte 1

Bgfbfg

33. Viendo mas información con fase wrap – Parte 2

fnn

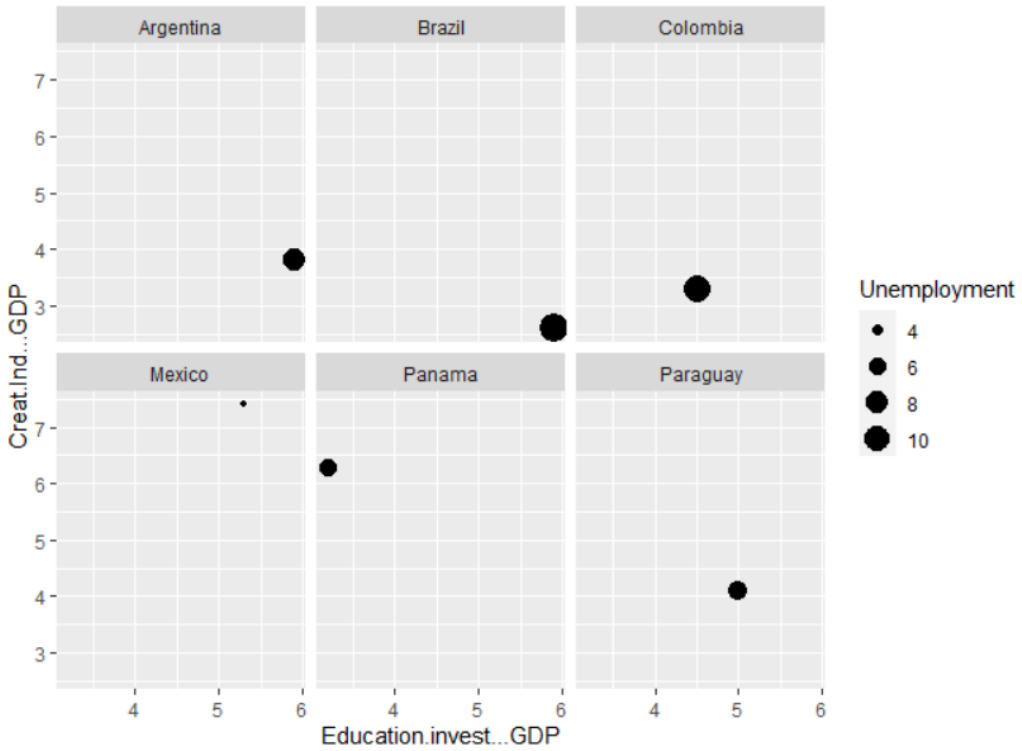
```
> TopNajanja <- orangeec %>%
+   filter(Country %in% c('Mexico', 'Panama', 'Argentina',
+                         'Colombia', 'Brazil', 'Paraguay'))
> library(ggplot2)
> ggplot(TopNajanja, aes(x= Internet.penetration...population,
+                        y=Services...GDP,size=GDP.PC)) +
+   geom_point() + facet_wrap(~Country)
```



Mas esferas mas grande tienen pbi mas grande

Panama tiene un 70 de aporte al pib 80% van a sus servicios argentina tiene 90 de aporte al sus servicios pero el del pib no es tan alto

```
> ggplot(TopNajanja, aes(x= Education.invest...GDP,
+                        y=Creat.Ind...GDP,size=Unemployment)) +
+   geom_point() + facet_wrap(~Country)
```

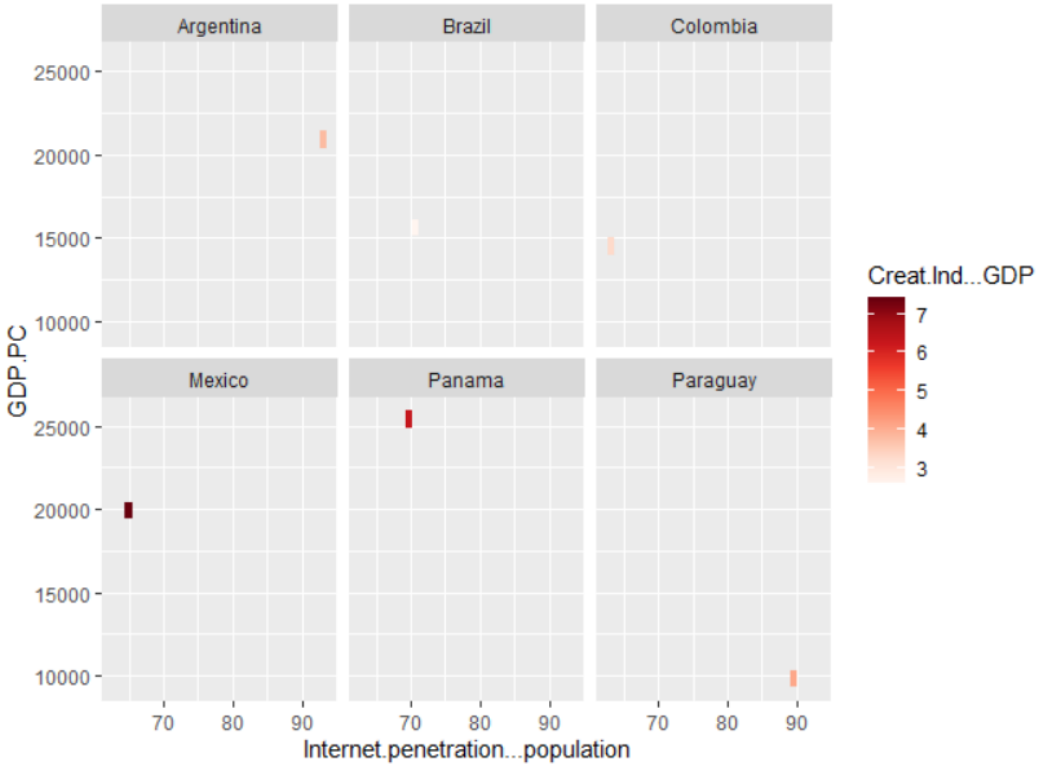


Argentina tien mas aporte al pib que brazil
Mexico tiene por en cima de 5% invierte en educación

Ahora instalamos un paquete para mejorar nuestras visualizaciones:

```
install.packages("RColorBrewer")
```

```
> mycolors <- brewer.pal(9, "Reds") # creando la paleta de colores
> ggplot(TopNajanja, aes(x=Internet.penetration...population,
+                        y=GDP.PC, fill=Creat.Ind...GDP))+
+   geom_tile() +
+   facet_wrap(~Country)+
+   scale_fill_gradientn(colors = mycolors)
```



El de brazil casi ni se ve

34. Conociendo R Markdown y organizando los hallazgos del análisis en un PDF

Conociendo R Markdown y organizando los hallazgos del análisis en un documento PDF. Es momento de generar nuestro documento con todas las gráficas y observaciones que hemos realizado a nuestro dataset, para ello necesitamos instalar el paquete rmarkdown y knitr:

```
install.packages("rmarkdown")
install.packages("knitr")
```

R Markdown nos permite generar archivos en formato HTML, PDF y Word. **La mejor opción es trabajar en un formato HTML para compartirlo por internet** y posteriormente convertirlo ya sea a PDF o Word.

Dentro de nuestro archivo de R Markdown iremos escribiendo con sintaxis de markdown el archivo y cuando escribamos código por si solo se va a ejecutar y añadir las gráficas o cálculos a nuestro archivo.

Conforme agregamos chunks de codigo podemos ir dando click en knit o ctrl + shift + k para ir actualizando el archivo, podemos publicar o exportar este archivo.

file:///C:/Fundamentos%20en%20R-%20Platzi/Economia-Naranja.html

Economia Naranja

Janice Escobedo

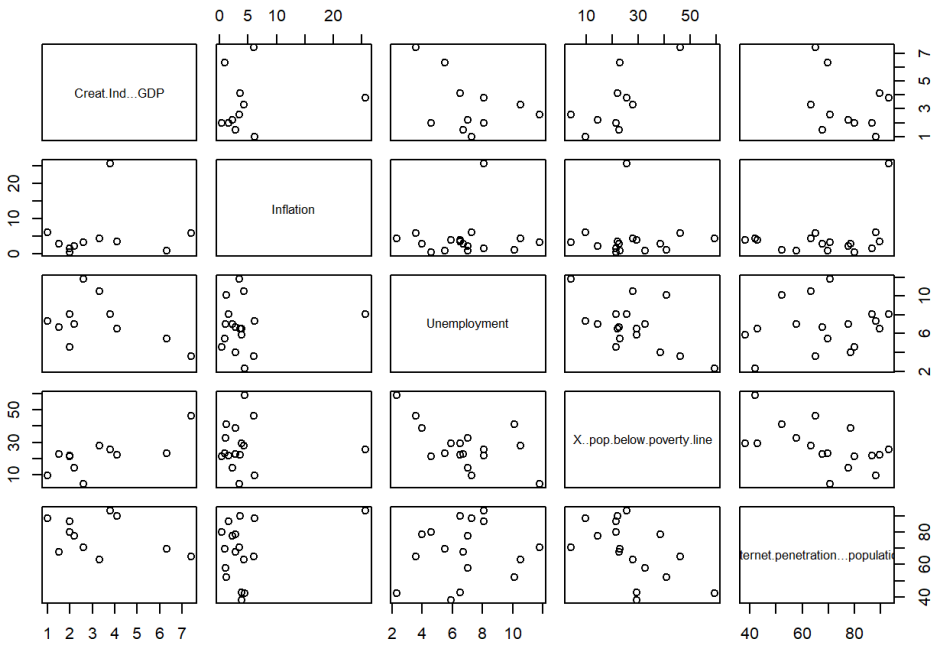
25/10/2020

```
orangeec <- read.csv("/Fundamentos en R- Platzi/orangeec.csv")
data(orangeec)
## Warning in data(orangeec): data set 'orangeec' not found
summary(orangeec)
##    Country              GDP.PC       GDP.US.bill         GDP.Growth..
##  Length:17          Min.   : 5600   Min.   :     13.7   Min.   :0.800
##  Class :character   1st Qu.: 8300   1st Qu.:     37.1   1st Qu.:2.000
##  Mode  :character   Median :13300   Median :     75.7   Median :2.800
##                     Mean   :14053   Mean   : 188693.0   Mean   :2.959
##                     3rd Qu.:19900   3rd Qu.:    309.2   3rd Qu.:4.200
##                     Max.   :25400   Max.   :2055000.0   Max.   :5.400
##
##  Services...GDP  Creat.Ind...GDP   Inflation       Unemployment
##  Min.   :50.00   Min.   :1.000   Min.   : 0.400   Min.   : 2.300
##  1st Qu.:56.90   1st Qu.:2.000   1st Qu.: 1.600   1st Qu.: 5.500
##  Median :62.20   Median :2.600   Median : 3.400   Median : 6.700
##  Mean   :62.64   Mean   :3.291   Mean   : 4.365   Mean   : 6.794
```

```
##  3rd Qu.:64.90   3rd Qu.:3.950   3rd Qu.: 4.300   3rd Qu.: 8.100
##  Max.   :82.00   Max.   :7.400   Max.   :25.700   Max.   :11.800
##                  NA's   :6
##  X..pop.below.poverty.line Internet.penetration...population   Median.age
##  Min.   : 4.20             Min.   :38.20                      Min.   :22.10
##  1st Qu.:21.70             1st Qu.:57.70                      1st Qu.:25.70
##  Median :25.70             Median :69.70                      Median :28.20
##  Mean   :27.65             Mean   :68.42                      Mean   :28.28
##  3rd Qu.:32.70             3rd Qu.:79.90                      3rd Qu.:31.30
##  Max.   :59.30             Max.   :93.10                      Max.   :35.00
##
##   X..pop.25.54   Education.invest...GDP
##  Min.   :34.12   Min.   :2.800
##  1st Qu.:39.23   1st Qu.:4.400
##  Median :40.19   Median :5.000
##  Mean   :39.88   Mean   :5.082
##  3rd Qu.:41.08   3rd Qu.:5.900
##  Max.   :44.03   Max.   :7.400
##
```

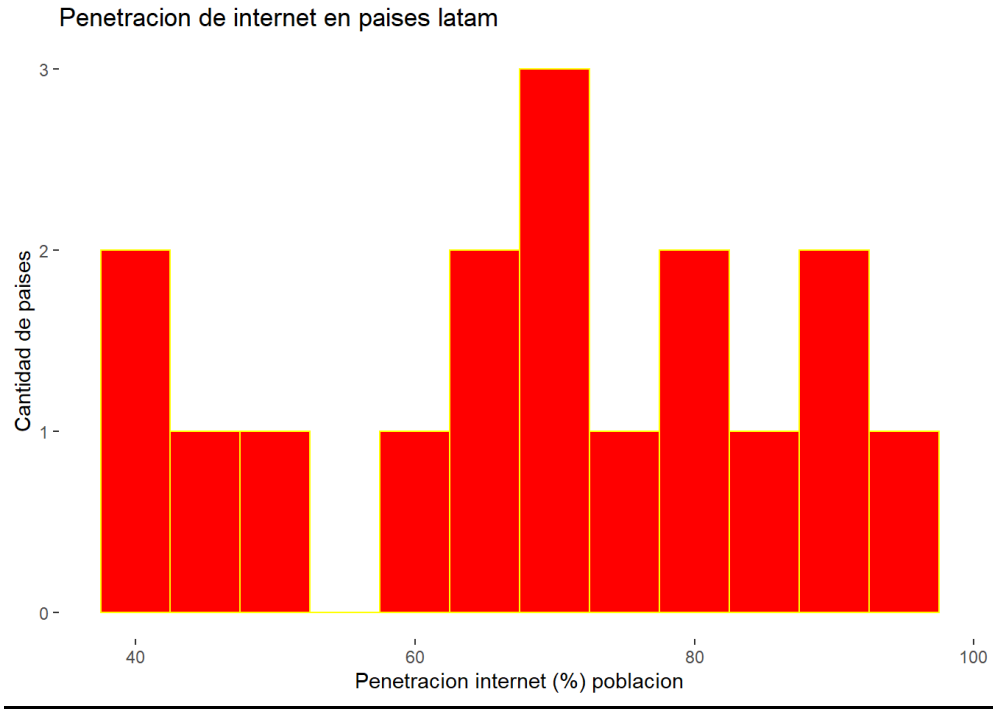
Parece que hay correlacion entre aporte de economia naranja al pib y la tasa de desempleo

```
pairs(orangeec[,6:10])
```



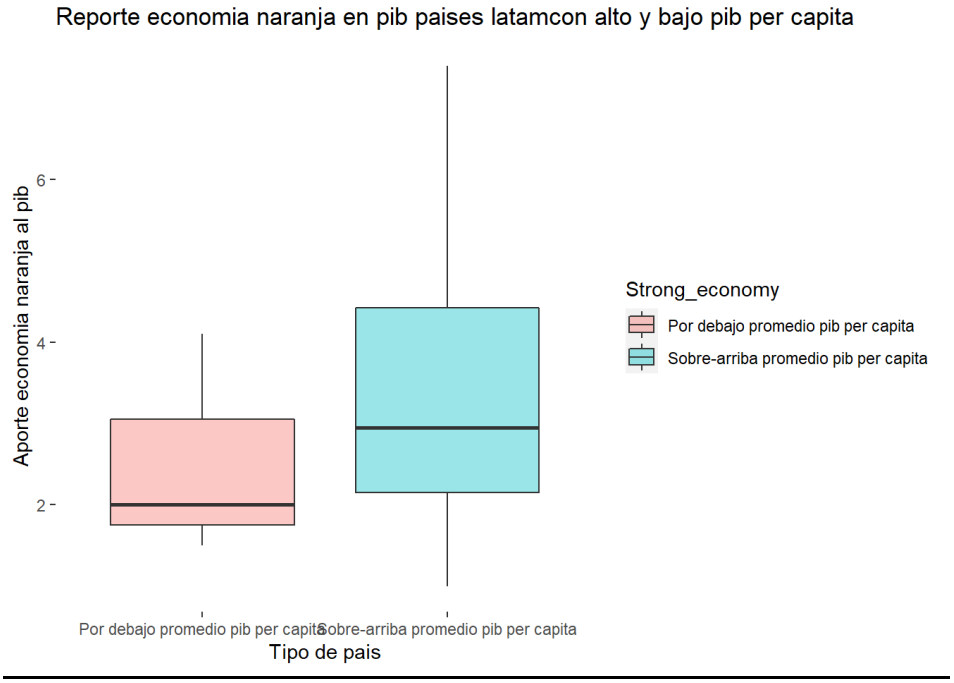
```
library(ggplot2)
ggplot() + geom_histogram(data = orangeec,
                          aes(x=Internet.penetration...population), fill = "red", color= "yellow",
                          binwidth = 5) +
```

```
  labs(x = "Penetracion internet (%) poblacion",
       y = "Cantidad de paises", title = "Penetracion de internet en paises l
atam") +
  theme(legend.position = "none") +
  theme(panel.background = element_blank(),
        panel.grid.major = element_blank(),
        panel.grid.minor = element_blank())
```




```
library(dplyr)
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
economy = mean(orangeec$GDP.PC)
economy
## [1] 14052.94
orangeec <- orangeec %>%
  mutate(Strong_economy = ifelse(GDP.PC < economy,
                                 "Por debajo promedio pib per capita",
                                 "Sobre-arriba promedio pib per capita"))
ggplot(orangeec, aes(x= Strong_economy,y= Creat.Ind...GDP,
                     fill = Strong_economy))+
  geom_boxplot(alpha = 0.4) +
  labs(x= "Tipo de pais", y="Aporte economia naranja al pib",
```

```
       title="Reporte economia naranja en pib paises latamcon alto y bajo pib
per capita")+
  theme(panel.background = element_blank(),
       panel.grid.major = element_blank(),
       panel.grid.minor = element_blank())
## Warning: Removed 6 rows containing non-finite values (stat_boxplot).
```



El boxplot indica que los paises suben el promedio en pib percapita tienen unad dispersion en cuanto a los aporten ==s de economia naranja al pib, OJO confirmar con desviacion estandar

Jm

35. Invitacion a continuar recorriendo el mundo del data science

Cuando no tengamos un DataFrame completo, siempre podemos usar los vectores para construir visualizaciones.

Observa que utilizando plot simplemente existe una representación automática de los datos, es decir cuando pasamos dos variables numéricas obtenemos un scatterplot y cuando pasamos una numérica y otras categóricas obtenemos un boxplot.

```
> cajas <- c(1,2,3,4,5,6,7,8)
> tiempo <- c(10,9,8,5.8,6,3,1.8,1)
> plot(tiempo~cajas)
> plot(orangeec$Services...GDP~orangeec$Education.invest...GDP)
> # Obtenemos un scatteplot porque son valores numericos
> plot(mtcars$mpg~mtcars$am)
> # da un boxplot porque cruza variable numerica y una categorica
```

r es un lenguaje de programacion especializado en estadistica para hacer

data science