

L24 LAB 13 (MD)

AnaGSanjuanM

2023-02-22

Hecho con gusto por Carla Carolina Pérez Hernández (UAEH)

LABORATORIO - Gráficos en R con ggplot2

LABORATORIO 13

Alumna: Ana Grisel Sanjuan Merida

Instalar primero las paqueterías necesarias

```
install.packages("tidyverse")
```

```
install.packages("readr")
```

```
install.packages("ggplot2")
```

```
install.packages("psych")
```

```
install.packages("ggcorrplot")
```

```
install.packages("ggpubr")
```

```
install.packages("gridExtra")
```

Cargar librería ggplot2

```
library(ggplot2)
```

Cargar los datos. Cuadrante inferior derecho/Upload allí se carga. El de REGESIONES FINALES.csv y el de correl.csv

Leer data frame. Está en el Environment

```
green_data <- read.csv("REGESIONES FINALES.csv")
```

Echando un ojo a los datos

```
names(green_data)
```

```
## [1] "STATE"      "GCI_rank"   "ICE_rank"   "GCI_index"  "ICE_index"  "PIBE"
## [7] "LPIBE"
```

Se muestran varios indicadores

Graficar la relación que existe entre los diversos indicadores parados

Relación entre los índices ICE y GCI. Con la función de ggplot, cargando los datos que provienen de green_data y que los vaya dibujando con mapping, siendo los elementos estéticos aes en el eje de las x, el índice de complejidad económica ICE_index, en el eje de las y, el índice de complejidad verde GCI_index. Será un gráfico de puntos con geom_point e incluirá texto con geom_text, ambos provenientes de la base de datos green_data y las etiquetas estarán representadas por Estados STATE, siendo elementos adicionales el color y el tamaño. Incluirá una línea de regresión con lineal method lm omitiendo el error estándar se=FALSE.

```
p1 <- ggplot(data = green_data,
             mapping = aes (x = ICE_index,
                           y = GCI_index)) +
  geom_point() +
  geom_text(label = green_data$STATE,
            color = "black" ,
            size = 4) +
  geom_point() +
  geom_smooth(method = lm,
             se = FALSE,
             fullrange = TRUE)
```

Para visualizar el objeto p1

```
p1
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

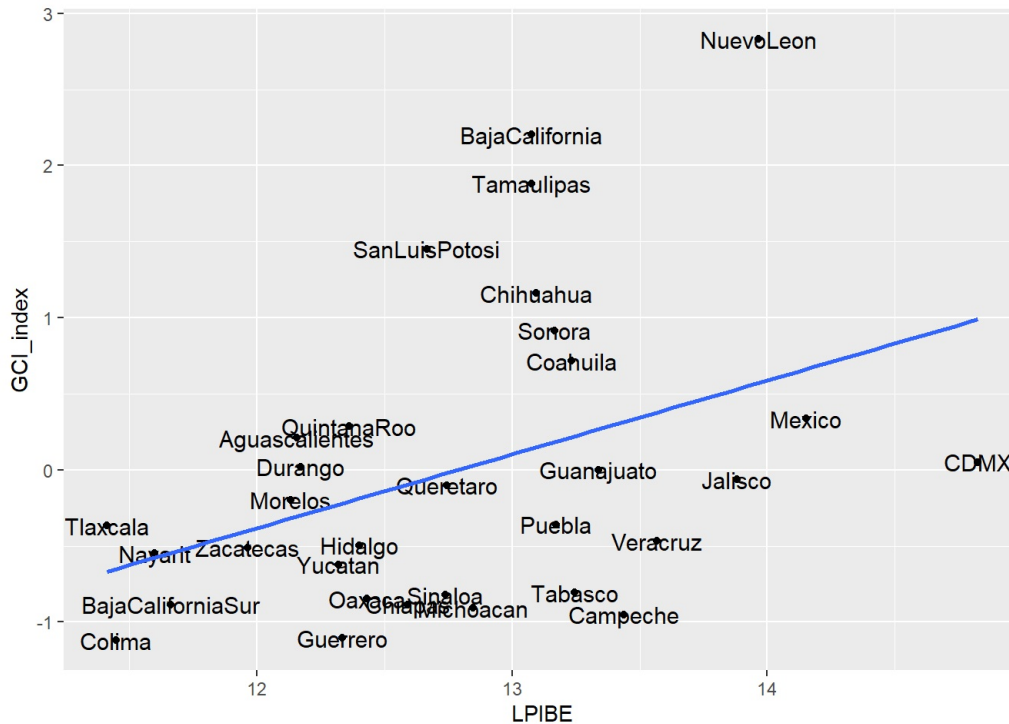


```
p3 <- ggplot(data = green_data,
             mapping = aes (x = LPIBE,
                           y = GCI_index)) +
  geom_point() +
  geom_text(label = green_data$STATE,
            color = "black" ,
            size = 4) +
  geom_point() +
  geom_smooth(method = lm,
             se = FALSE,
             fullrange = TRUE)
```

Para visualizar p3

p3

```
## `geom_smooth()` using formula = 'y ~ x'
```



CONOCER LAS CORRELACIONES

——correl1 install.packages("psych")

```
library(psych)
```

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

```
## The following objects are masked from 'package:ggplot2':
##
## %+%, alpha
```

Abriremos un dataframe llamado correl. indicar que lea la base de datos correl.csv

```
library(readr)
correl <- read_csv("correl.csv")
```

```
## Rows: 32 Columns: 5
## — Column specification —————
## Delimiter: ","
## dbl (5): GCI_rank, ICE_rank, GCI_index, ICE_index, LPIBE
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Las variables que se pueden apreciar son GCI_rank, ICE_rank, GCI_index y LPIBE. Se visualiza en el Environment dando clic en correl.

Cada una de las variables se van a correlacionar. Indicar que las variables se correlacionen de forma paríada

```
attach(correl)
names(correl)
```

```
## [1] "GCI_rank" "ICE_rank" "GCI_index" "ICE_index" "LPIBE"
```

Gráfico de correlación

```
pairs(correl)
```

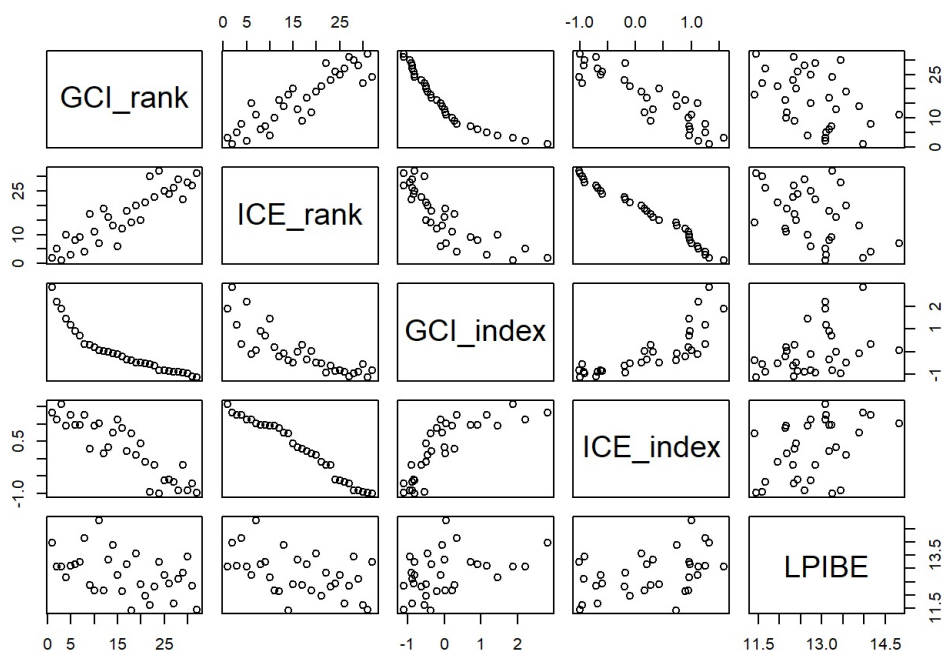
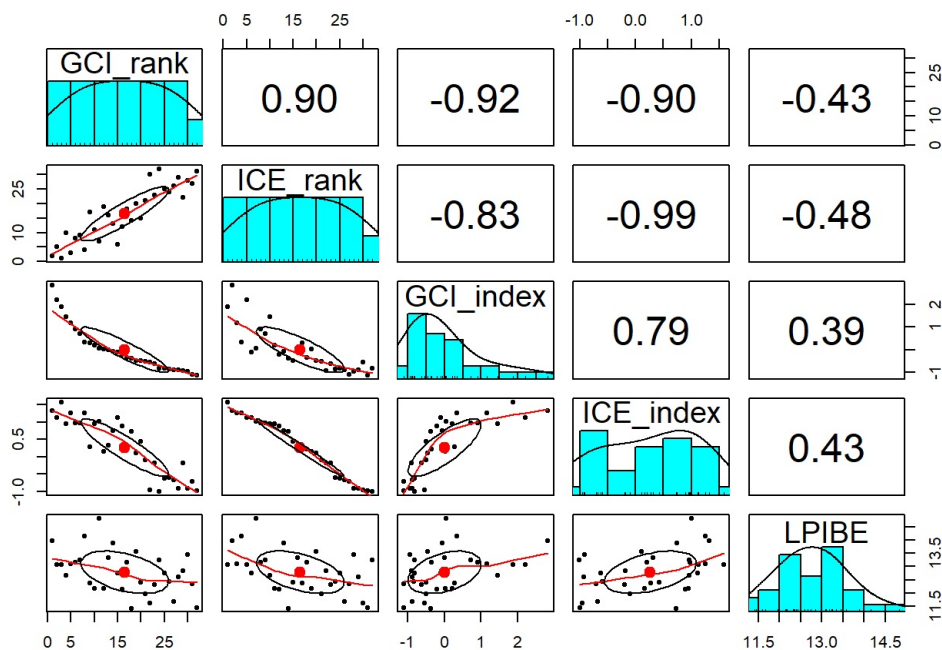


Gráfico de correlación en panel

```
pairs.panels(correl)
```



——correl2

Calculamos la correlación de complejidad. Generamos dataframe llamado `complex_corr`, el signo de asignación al presionar Alt- y aparece así: `<-`. Para obtener correlación del archivo `correl` mediante el método de Pearson.

```
complex_corr <- cor(correl, method = "pearson")
```

Para visualizar `complex_corr`

```
complex_corr
```

```
##          GCI_rank  ICE_rank  GCI_index  ICE_index  LPIBE
## GCI_rank    1.0000000  0.8969941 -0.9166564 -0.8957551 -0.4288321
## ICE_rank    0.8969941  1.0000000 -0.8274473 -0.9875750 -0.4750548
## GCI_index   -0.9166564 -0.8274473  1.0000000  0.7946666  0.3905008
## ICE_index   -0.8957551 -0.9875750  0.7946666  1.0000000  0.4272882
## LPIBE       -0.4288321 -0.4750548  0.3905008  0.4272882  1.0000000
```

En la consola se ven los resultados en forma de matriz (o sea la matriz de correlación)

Redondeamos el coeficiente de correlación. Nombramos la base de datos, redondeamos con round a dos dígitos.

```
complex_corr = round(complex_corr, digits = 2)
```

Para visualizar

```
complex_corr
```

```
##          GCI_rank  ICE_rank  GCI_index  ICE_index  LPIBE
## GCI_rank    1.00    0.90    -0.92    -0.90 -0.43
## ICE_rank    0.90    1.00    -0.83    -0.99 -0.48
## GCI_index   -0.92   -0.83    1.00    0.79  0.39
## ICE_index   -0.90   -0.99    0.79    1.00  0.43
## LPIBE       -0.43   -0.48    0.39    0.43  1.00
```

En la consola se aprecia la matriz redondeada

Mapa de calor de las correlaciones

Matriz de correlación: `install.packages("ggcorrplot")`

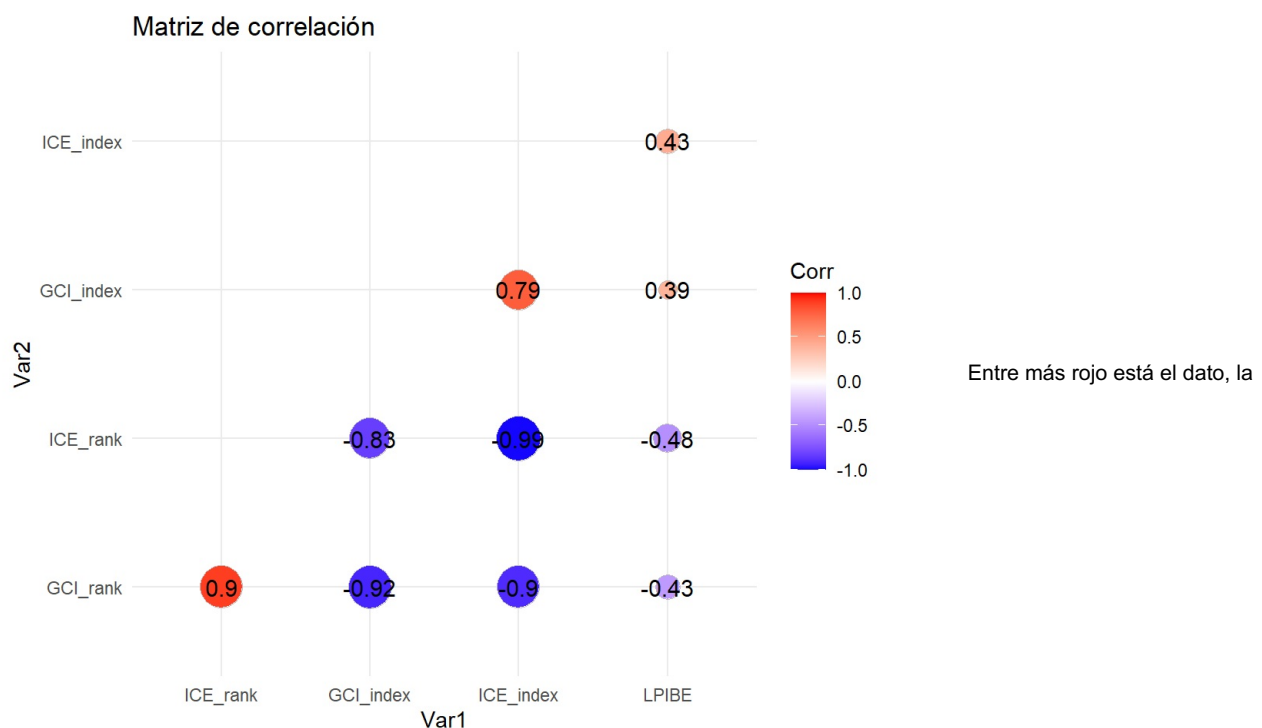
```
library(ggcorrplot)
library(ggplot2)
```

Generamos objeto llamado p4, ggcorrplot los dibujará. Los input serán el complex_corr (la matriz recién generada). El método (method) será en círculos. El type será lower porque solo queremos ver la matriz superior izquierda. El título será Matriz de correlación. El tema será minimalista (theme minimal).

```
p4 <- ggcorrplot(complex_corr, method = "circle", type = "lower", lab = TRUE) +
  ggtitle("Matriz de correlación") +
  theme_minimal()
```

Para visualizar p4

```
p4
```



correlación es lineal apreciable. Entre más azul está el dato, la correlación es inversa apreciable.

Conjunto de gráficos dentro de una sola cuadrícula `install.packages("ggpubr")`

Llamamos a la librería

```
require(ggpubr)
```

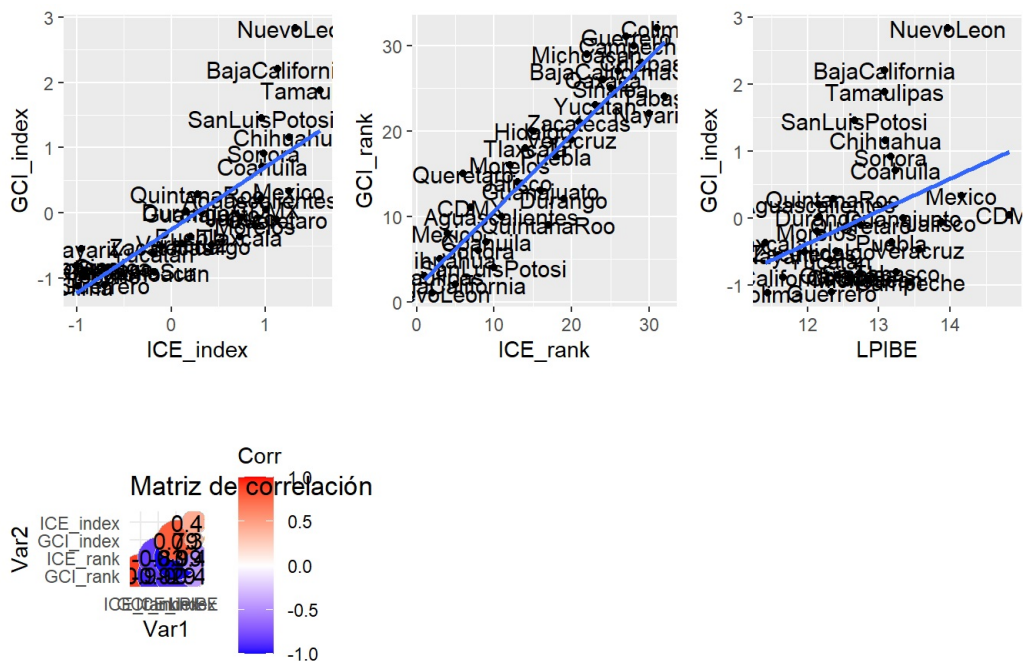
```
## Loading required package: ggpubr
```

Con `ggpubr` se hará la unión (`ggarrange`) entre los objetos obtenidos (`p1`, `p2`, `p3` y `p4`)

```
ggpubr :: ggarrange (p1, p2, p3, p4, etiquetas = c ("A", "B", "C" , "D"))
```

```
## `geom_smooth()` using formula = 'y ~ x'  
## `geom_smooth()` using formula = 'y ~ x'  
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning in as_grob.default(plot): Cannot convert object of class character into  
## a grob.
```



Conjunto de gráficos dentro de una sola cuadrícula en dos renglones para una mayor estética `install.packages("gridExtra")`

```
library(ggplot2)
```

Llamamos a la librería de `gridExtra`

```
require(gridExtra)
```

```
## Loading required package: gridExtra
```

Hacemos un objeto gráfico llamado `F1`. Será un grid (es decir, una rejilla de unión: `grid.arrange`). Los gráficos anteriores (`p1`, `p2`, `p3` y `p4`). Estarán acomodados en dos renglones `nrow = 2`

```
F1 <- grid.arrange (p1, p2, p3, p4, nrow = 2)
```

```
## `geom_smooth()` using formula = 'y ~ x'  
## `geom_smooth()` using formula = 'y ~ x'  
## `geom_smooth()` using formula = 'y ~ x'
```

