**Páginas relevantes**

<https://www.princeton.edu/~otorres/Panel101R.pdf> (“Getting Started in Fixed/Random Effects Models using R”)

<https://www.econometrics-with-r.org/> (Buen “libro” de econometría en R)

**TUTORIAL R - Edx**

**Ls()** (list?)

#Te tira todas las variables que tenes

**rm() (remove)**

#para eliminar variables

**rm(list=ls())**

borrar todos los elementos en el environment

**Class ()**

# ver de qué clase es la variable

**2L**

Hace que el número (cualquiera) sea entero

**is.**

le preguntás (por ej is.numeric si la variable es numérica)

**as.algo(lo que querés coercionar)**

obligàs a la variable a ser de determinado tipo, ej

as.numeric(TRUE) y te sale que cad vez que tocás

**c()**

Crear un vector

**Names(x) <- Y**

Le ponés al vector x los strings en el vector Y

También se puede crear el vector con nombres directamente poniendo entre c() el nombre = el valor

**Length()**

Te muestra la longitud del vector, supongo que cantidad de datos

**Sum()**

Suma todos los elementos del vector

**<**

Te dice si cada elemento de un vector es menor al del otro vector o no

**Nombre del Vector[lugar en el vector o “string” del vector]**

Te tira un vector nuevo con ese elemento que le pediste. O sea es para pedir elementos de un vector

También podés poner un vector adentro de los corchetes para poder pedir más de un elemento de otro vector

Si le ponés un menos antes de lo que querés pedir dentro de los corchetes, te tira un vector con todos los elementos del otro vector menos ese elemento.

También podés pedir que excluya un vector.

También podés ponerle TRUE y FALSE para todos los elementos que hay en el vector y solo te tira los que tengan TRUE en su lugar. (si el vector lógico no tiene la misma cantidad de elementos que el vector original, R repite el patrón que pusiste hasta cubrir toda la longitud del vector(

**TALLER R - UBA**

tinyurl.com/ceceubaR

tinyurl.com/materialR (curso anterior)

programar/algortimo: selección secuencia repetición

**WHILE**: se ejecuta n veces mientras se cumple una condición, itera el proceso mientras se siga cumpliendo

¡ si no ejecutás un WHILE con un BREAK, puede seguir al inifinito

Condición:

-IF / THEN

**FOR**: vos determinás la cantidad de ejecuciones

Algoritmo - Interpretación - Complilación (ejecución) - resultado

R es interpretado y compilado, por lo cual solo con copiar y pegar el código del algoritmo te devuelve un resultado.

(si no lo es, solo funciona en la computadora en la que fue compilado)

Pasos UDIE (understand, design, implement, evaluate)

|  |  |
| --- | --- |
| SCRIPT: donde escribes el codigo | OBJETOS/VARIABLES/DATOS/FUNCIONES |
| CONSOLA: te muestra los códigos que corriste | PLOTS(graficos)  PAQUETES(algoritmos empaquetados)  HELP |

**Comando para intalar paquetes**:

Install.packages(“…”)

Si te bajás un paquete, lo tenes que pedir para que R lo use porque sino no lo incluye, lo deja afuera para no ser tan lento (no como eXcell)

# TUTORIAL 3 – ECONOMETRÍA AVANZADA UdeSA – “Loops e interpretaciones”

Link: <https://www.youtube.com/watch?v=HOo26uszMSE&feature=youtu.be>

**Correlación entre dos variables:**

*Cor(x1,x2)*

**Correlación entre más de dos variables (matriz de correlación):**

*df <- data.frame(x1,x2,x3)*

*cor(df)*

**Cargar una base de datos de Excel>**

*install.packages(“readxl”)*

*library(readxl)*  #lo llamás al paquete

*df <- read\_excel("base.xlsx")* #df podría ser “pirulito”. Ahora cargue una base de datos de un archivo de excel que se llama base.xlsx

**Grafico de dispersion entre muchas variables con este paquete:**

*pairs(df, col="blue")*

**Especificar la cantidad de dígitos que querés ver. En este caso con la matriz de correlaciones:**

*round(cor(df),2) acá serían dos dígitos*

**Pedir información sobre el comando de un paquete:**

*?corrplot*

**Cómo hacer LOOPS (con for), varias alternativas:**

*for (dia in c(1,2,3,4,5,6)){ el nombre dia es puramente arbitrario, pero va a ser el nombre del resultado del loop*

*print(paste("DÃ­as de cuarentena:", dia))*

*} primero el paréntesis la variable que va cambiando, y después la orden para esa variable entre llaves.*

*#Alternativa:*

*for (dia in 1:6){*

*print(paste("DÃ­as de cuarentena:", dia))*

*}*

**En número es DISTINTO de otro, operación diferencia sería:**

*!2 y lo que quieras*

**División de un número por otro:**

*2 / 4*

**División de un número por otro mostrando resultado más resto:**

*5 %% 2*

**Si en número no es múltiplo de otro, no lo consideres y pasa a otro:**

*#Agregando "next":*

*for (i in 1:10) {*

*if (!i %% 2){*

*next*

*}*

*print(i)*

*}*

**Otro tipo de LOOP (con while):**

*#While*

*i <- 1*

*while (i < 6) {*

*print(i)*

*i = i+1*

*}*

**Agregar una variable a una base de datos (la base llamada “df”):**

*df$edad2<-df$edad^2 la nueva variable se va a llamar “edad2” y es el resultado de elevar “edad” al cuadrado*

**Limpiar la consola**

Control+L

**Comando para mostrar resultados de regresiones lindos (necesitas el paquete stargazer):**

*Stargazer*

**Definir el directorio (siempre hacerlo al empezar el codigo):**

*setwd(“acá el directorio“) (*invertir las barras que vienen cuando copiás, se usa esta barra: / )

**Para ver una base de datos:**

*Read\_dta (“df”)*

**Cargar una base de datos de PANEL:**

*Nombrequeleponés <- pdata.frame(df, index = c(“variabledeindividuo”, “variabledelperiodo”)*

**Fijar un objeto:**

*Attach(df)*

**Definir una fórmula (por ej un modelo de regresión) (necesitás el paquete Formula) (así no tenés que repetir la fórmula todo el tiempo)**

*Formula(y ~ x + h)*

**Test de Hausman:**

*phtest(estimacion.fe,estimacion.re)*

*# H0: FE es consistente, RE es consistente y eficiente*

*# HA: FE es consistente, RE es inconsistente*

**Borrar valores faltantes**

*mydata=na.omit(mydata)*

**Convertir variables a forma categórica (tipo dummies)**

*mydata$columna <- factor(mydata$columna)*

**Tabla de doble entrada entre dos variables (algo así como agruparlas)**

*xtabs(~var1 + var2, data = mydata)*

**PAQUETES:**

Para correr modelos de regresión:

*plm*