

Práctica 1

Usando RSNNS

- Descripción de R.
- Abrir y usar RStudio.
- Manejo sencillo del script para la práctica.

Lenguaje R

- Lenguaje de programación abierto para la computación estadística y el análisis de datos.
- Se puede encontrar en: <https://www.r-project.org/about.html>
- Contiene multitud de operaciones para el procesamiento de distintas estructuras de datos, especialmente para matrices.
- Amplio abanico de librerías de herramientas de análisis de datos desarrolladas en este lenguaje.

RStudio

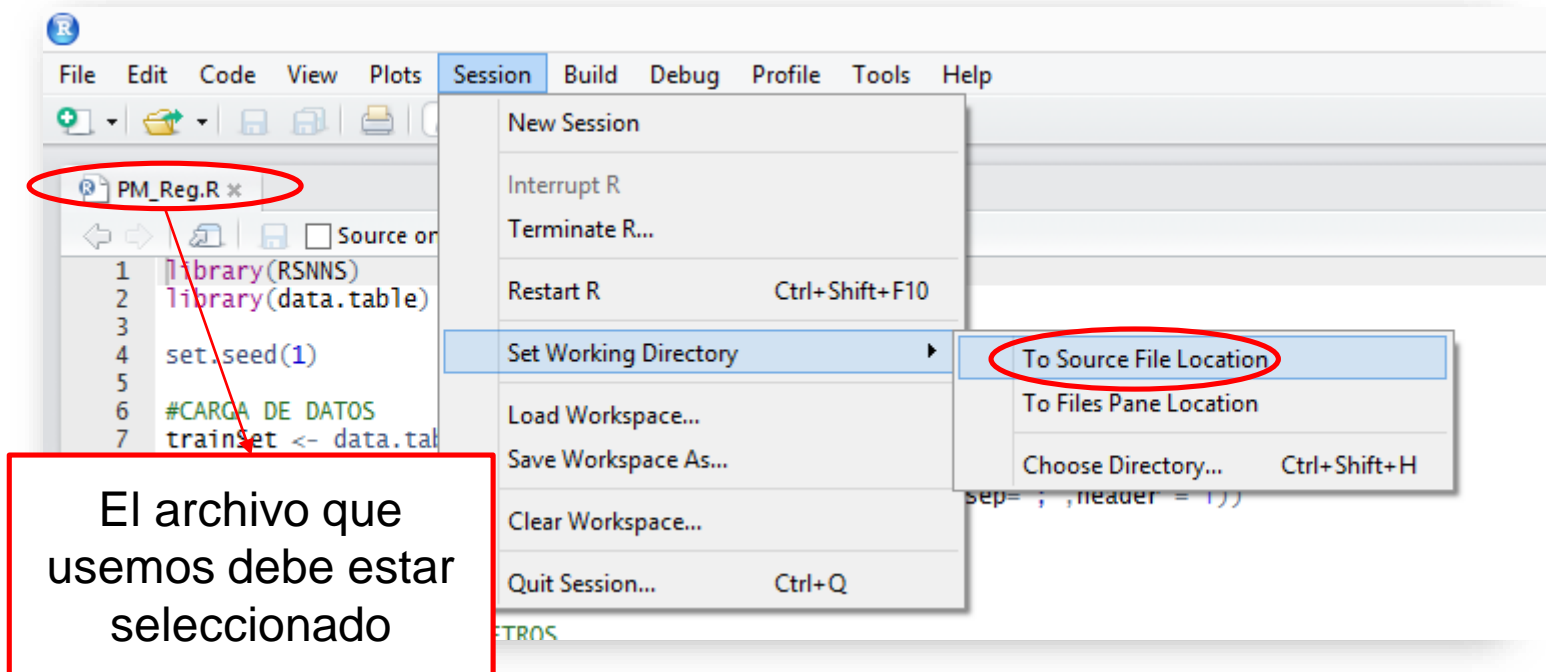
- Entorno de desarrollo para el lenguaje R con consola, desarrollo de scripts, visor de variables y diversas utilidades para facilitar el uso de R.
- Se puede encontrar en:
<https://www.rstudio.com/products/rstudio>

Preparación de RStudio

- Abrir RStudio en Windows
- Desde RStudio, abrir el archivo de la práctica PM_Reg_Colmenarejo.R
- Asegurarse de que los archivos de datos y el script estén en el mismo directorio

Preparación de RStudio

- Configurar el directorio de trabajo



Preparación de RStudio

- Instalar paquetes necesarios:
 - RSNNS
 - Es una adaptación para R del conocido simulador “The Stuttgart Neural Network Simulator” (SNNS)
 - Es un paquete que contiene muchas funciones para usar el simulador
 - <https://cran.r-project.org/web/packages/RSNNS/RSNNS.pdf>

Ficheros de datos

- El script está preparado para cargar los siguientes ficheros de datos:
 - Train.csv
 - Test.csv
 - Validacion.csv
- Los ficheros de datos deben tener el formato siguiente:
 - ‘,’ coma para marcar separación entre dos campos
 - ‘.’ punto para marcar el decimal de los números
 - Sin cabecera
- Pueden usarse otros formatos cambiando los valores de los parámetros dec, sep y header
- Si los campos están separados por espacios o tabulador puede usarse `read.table()` en lugar de `read.csv()`

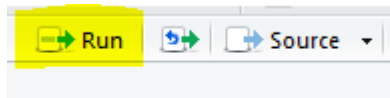
Ejecutar el script

- Ejecutar todo el script de una vez

- Pulsar “Source”



- También se puede ejecutar la parte del código que se desee seleccionándolo y pulsando “Run”.



Usar el MLP con RSNNS

- El código que ejecuta MLP es:

```
13  
14 #SELECCION DE LOS PARAMETROS  
15 topologia <- c(20)  
16 razonAprendizaje <- 0.01  
17 ciclosMaximos <- 200  
18
```

```
#EJECUCION DEL APRENDIZAJE Y GENERACION DEL MODELO  
# en Rsnns se llama test a nuestro fichero de validación  
model <- mlp(x= trainSet[, -target],  
             y= trainSet[, target],  
             inputsTest= validSet[, -target],  
             targetsTest= validSet[, target],  
             size= topologia,  
             maxit=ciclosMaximos,  
             learnFuncParams=c(razonAprendizaje),  
             shufflePatterns = F  
             )
```

- Los parámetros que hay que modificar son los indicados:
 - **topologia**: Define las neuronas ocultas del MLP, se define usando una concatenación de valores, indicando el número de neuronas ocultas por cada capa: c(20), c(10,20,15), etc.
 - **razonAprendizaje**: Define la razón de aprendizaje con un número entre 0 y 1
 - **ciclosMaximos**: Define el número máximo de iteraciones del algoritmo de aprendizaje

Usar el MLP con RSNNS

- A la derecha en Rstudio se muestra la evolución del error. La línea roja marca la validación y la línea negra representa el error de entrenamiento.
- Se pueden consultar:
 - Errores finales: *errors*
 - Evolución del MSE: *iterativeErrors*
 - Salidas de la red: *outputs*
- El resultado del entrenamiento se guarda al final en ficheros csv.

