

Tareas reproducibles

Un punto a favor del lenguaje es que nuestros análisis y resultados puedan ser reproducidos y corroborados.

Si entendemos los códigos escritos como textos que nos permite comunicar nuestros procedimientos y contar un razonamiento.

Existen muchos lenguajes de programación, cada uno con sus reglas semánticas y gramaticales. No importa lo complicado que sea el código, el razonamiento atrás es la misma.

Al escribir código generamos una receta o bitácora explícita de nuestros procedimientos, permitiendo que sean replicados.

Además, podemos incluir mensajes que sean fáciles de leer pero que la computadora ignore.

En R esto se hace con el uso del símbolo de número o *hashtag* (#).

Todo el texto a la derecha de ese símbolo será ignorado por R, pero brinda información a la persona que esté leyendo el código, por ejemplo:

```
# Sumamos dos más dos  
2 + 2 # También puedo comentar en la misma línea
```

Comentar el código nos ayuda a explicar las operaciones que hacemos.

Esto es útil al compartir nuestro código con alguien más, pero también al regresar a un código viejo.

Proyectos de RStudio

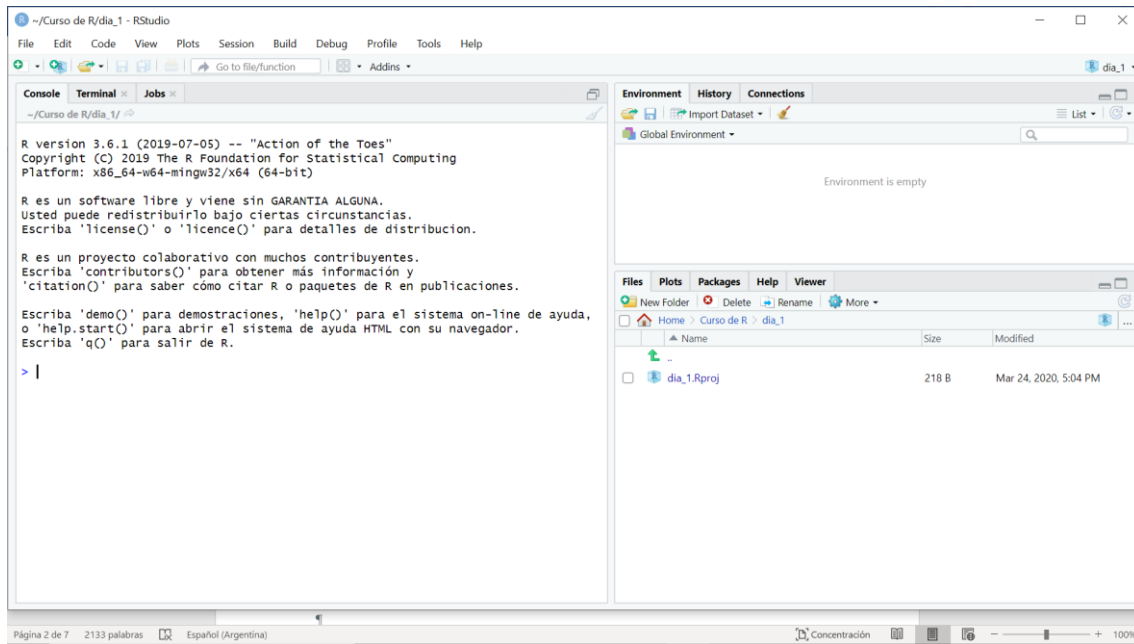
R se basa en el concepto de proyecto y el directorio del proyecto mantiene todos los archivos relacionados en una dirección de trabajo estable.

El directorio de trabajo no es más que la localización de todos los archivos y carpetas.

El comando `getwd()` (*get working directory*) te mostrará el directorio de trabajo.

Al efecto abrí en mis documentos una carpeta que llamé “curso de R” – después iremos abriendo nuevas carpetas en función del avance ahí.

Entonces con el R abierto voy a file y le digo nuevo proyecto y le digo que en la carpeta de curso R me nombre al proyecto como día_1



Fíjense que a la derecha en files aparece el nombre del proyecto como “día_1.Rproj”

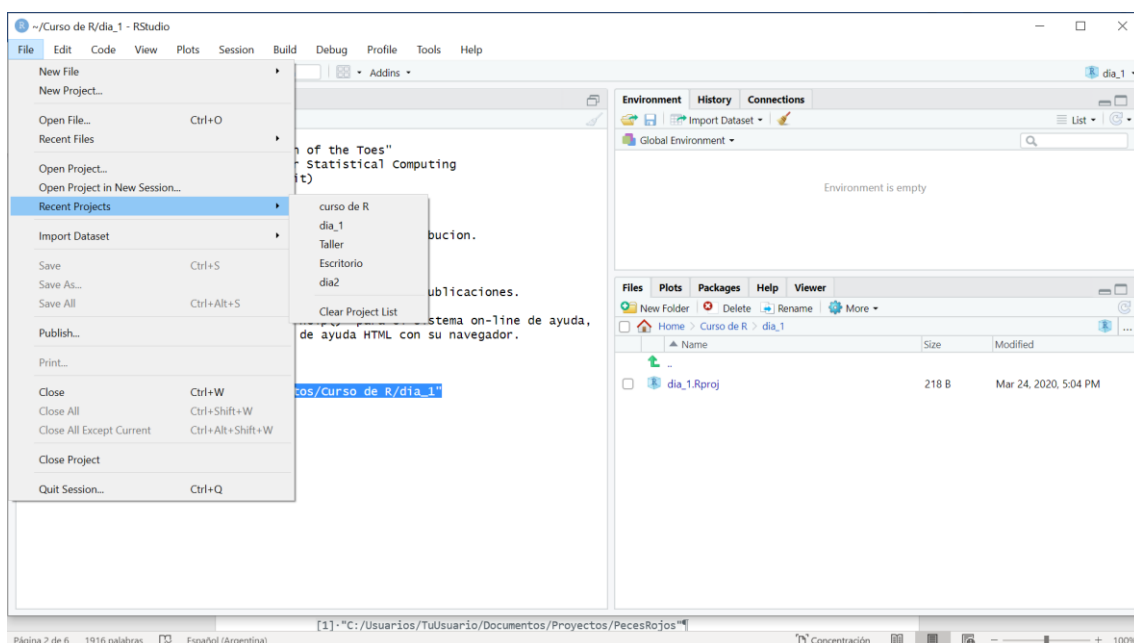
En la consola escribo

`getwd()`

y obtengo por resultado lo siguiente

`[1] "C:/Users/Usuario/OneDrive/Documentos/Curso de R/día_1"`

Observen que también puedo abrir otros proyectos anteriores.



Todo esto parece obvio, pero es la organización de trabajo sobre la que trabaja R

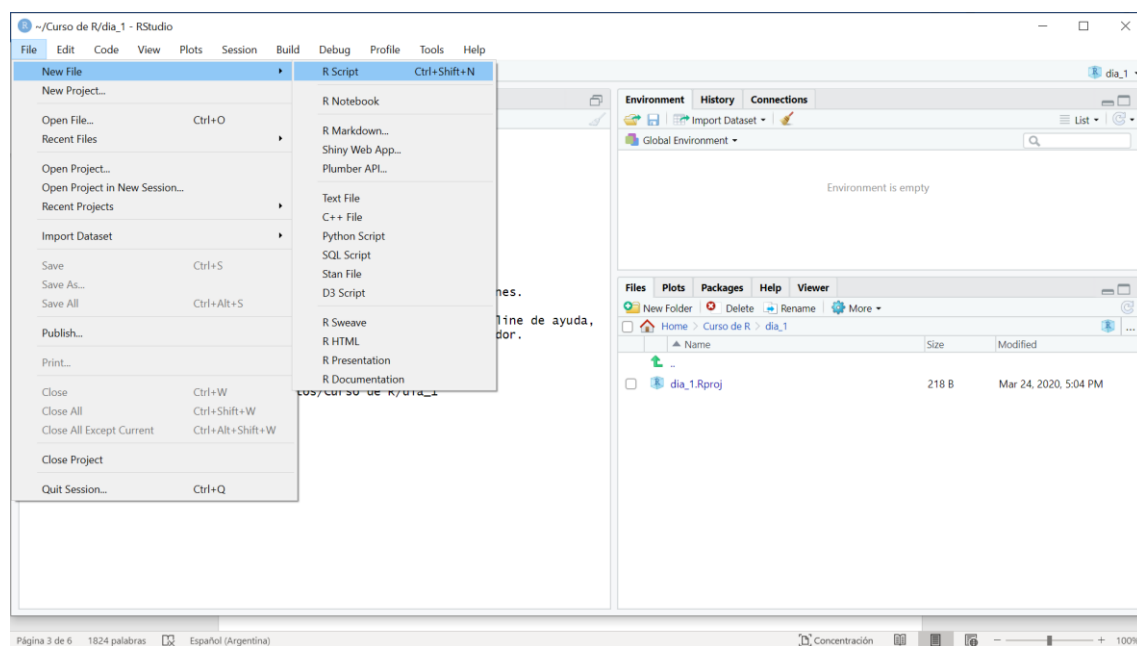
Tener un directorio asociado a nuestro proyecto nos permite cargar y guardar datos, figuras y tablas más fácilmente sin tener que estar cambiando el directorio.

Scripts

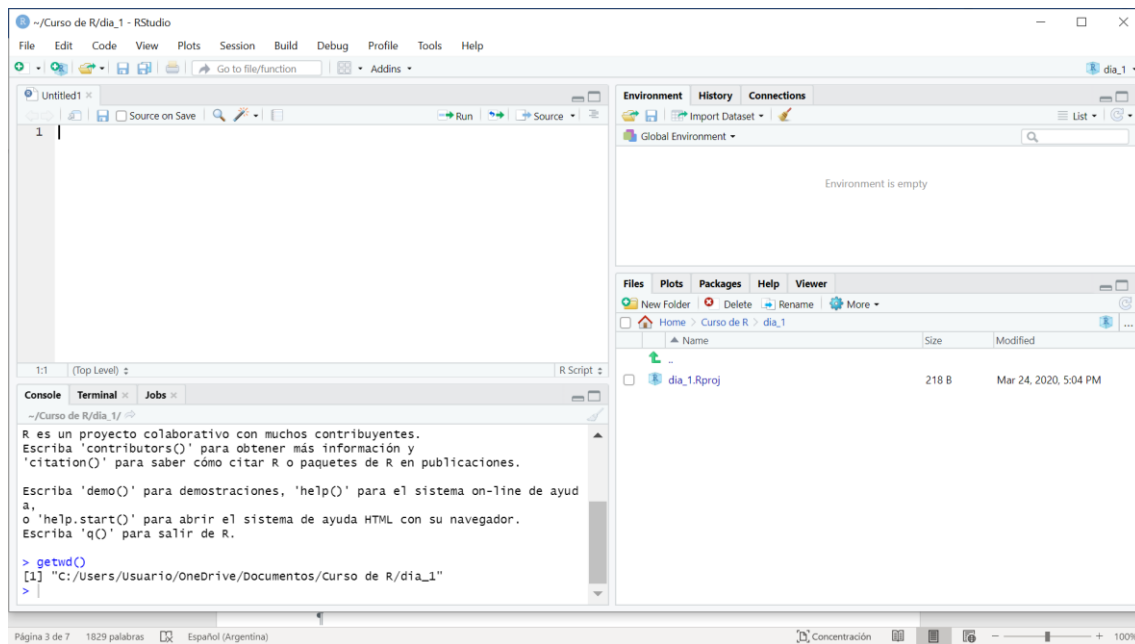
Como vimos anteriormente, podemos ejecutar comandos directamente en la consola. Sin embargo, la mayoría de las veces realizaremos varias operaciones simultáneamente y necesitamos tener todo el código en un mismo lugar. Esto nos ahorra tener que escribir todas las operaciones cada vez, pero también nos da un documento que podemos archivar para futuras referencias o compartir. El medio para hacer eso son los

Scripts

La forma más sencilla es con el uso de *scripts* de R. Estos no son más que documento de texto con extensión `.R` en vez de `.txt`.



La pantalla resultante ahora es:



También puedes utilizar la combinación de teclas `Ctrl+Shift+N`. Esto abrirá otra hoja

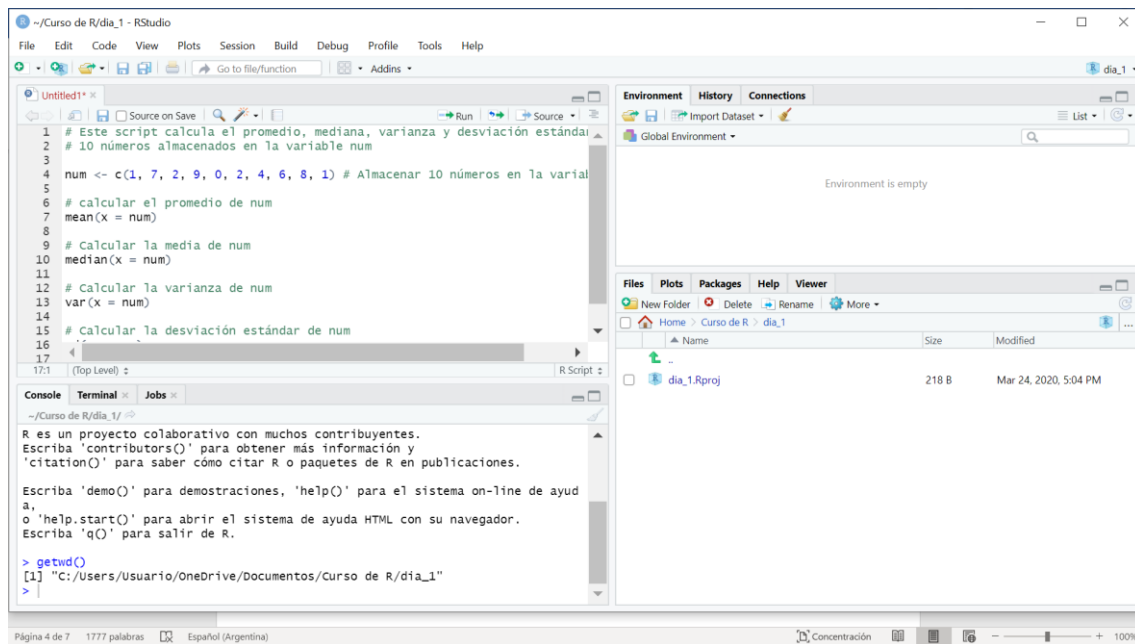
“Untitled2”

Cerremos la 2 y mantengamos la 1

Aquí podemos escribir varias operaciones seguidas, pero R no las ejecutará hasta que se lo indiquemos.

Por ejemplo, el siguiente código genera una serie de 10 números almacenados en la variable `num` y después calcula el promedio, la mediana, la varianza y la desviación estándar.

Copia y pega el código en tu script.



Este script calcula el promedio, mediana, varianza y desviación estándar de
10 números almacenados en la variable num

```
num <- c(1, 7, 2, 9, 0, 2, 4, 6, 8, 1) # Almacenar 10 números en la variable num
```

```
# calcular el promedio de num
mean(x = num)
```

```
# Calcular la media de num
median(x = num)
```

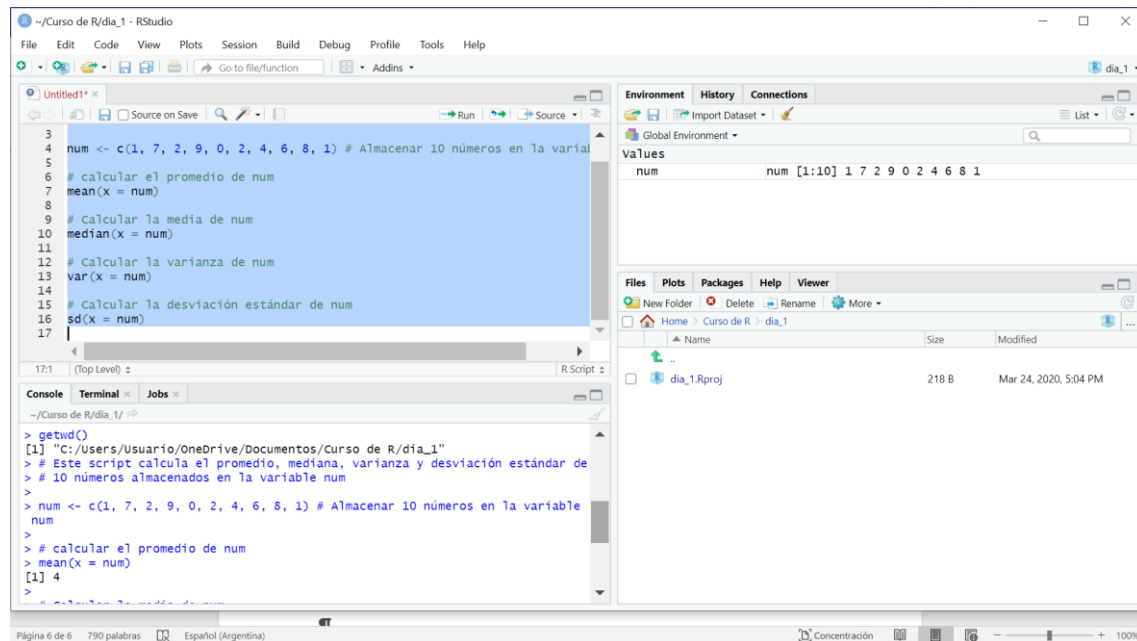
```
# Calcular la varianza de num
var(x = num)
```

```
# Calcular la desviación estándar de num
sd(x = num)
```

Si presionas `Enter`, lo único que lograrás es agregar una nueva línea.

Para ejecutar el código deberás colocar tu cursor al principio o al final de una línea y presionar `Ctrl+Enter` (También puedes hacer click en el ícono que dice *Run* en la esquina superior derecha del *script*).

Prueba que sucede pintando todo el script y run.



Observemos los cambios

- En Enviroment vemos que se creo un vector numérico con 10 elementos
- Abajo en la consola vemos que se calculo la media, la mediana, la varianza y la desviación del vector de números.

Probemos cerrar y guardar y volver a abrir.

Observen que hay un archivo de .Rhistory que guarda todo lo que hicimos. Hay un archivo .Rdata que guarda los datos del vector y el proyecto que guarda el script.

