**Introducción a R y RStudio**

**¿Qué es R?**

Empezaremos diciendo que R es un lenguaje de programación interpretado,

de distribución libre, bajo Licencia GNU, y se mantiene en un ambiente

para el cómputo estadístico y gráfico. Este software corre en distintas plataformas

Linux, Windows, MacOS, e incluso en PlayStation 3.

R fue creado en 1992 en Nueva Zelanda por Ross Ihaka y Robert Gentleman

(Ihaka [1998]). La intención inicial con R, era hacer un lenguaje didáctico,

para ser utilizado en el curso de Introducción a la Estadística de la Universidad

de Nueva Zelanda.

**Ross y Robert**

A modo de broma Ross y Robert, comienzan a llamar “R” al lenguaje que implementaron, por las iniciales de sus nombres, y desde entonces así se le conoce en la muy extendida comunidad amante de dicho lenguaje.

Luego de la creación de R (en 1992), se da un primer anuncio al público del software R en 1993. En el año de 1995 Martin Mächler, de la Escuela Politécnica Federal de Zúrich, convence a Ross y Robert a usar la Licencia GNU para hacer de R un software libre.

Como consecuencia, a partir de 1997, R forma parte del proyecto GNU.

**Comenzando con las Hojas de Trucos**



Esta hoja es de las primeras que vamos a estudiar y aprender, conforme estén usando el software se pueden apoyar en la misma, tiene la descripción de todo lo que ustedes están usando en su escritorio y que significa.

Tomémonos el tiempo suficiente para leerla, estudiarla y practicarla, el curso será completamente auto didáctico, sin embargo, como equipo dependemos de todos, vamos a ir a la velocidad del más lento.

En esta dirección podrá encontrar las hojas de ayuda con las que cuenta R actualmente:

<https://posit.co/resources/cheatsheets/>

Como primer ejercicio busque la de RStudio IDE cheatsheet y descárguela.

**Comencemos**

A esta altura ya tenemos y habremos visualizado y practicado un poquito con RStudio, comenzaremos con un ejemplo sencillo e iremos escalando de a poco así que no se preocupe “aun”.

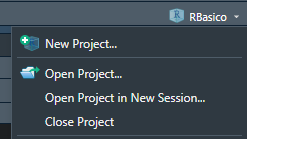
**Estableciendo un Directorio de Trabajo**

Recomiendo que, al comenzar, creemos un directorio de trabajo con carpetas ordenadas, en inicio el curso consta de 3 sesiones o niveles, por lo que les recomendaría abrir la carpeta de Mis Documentos en Windows.

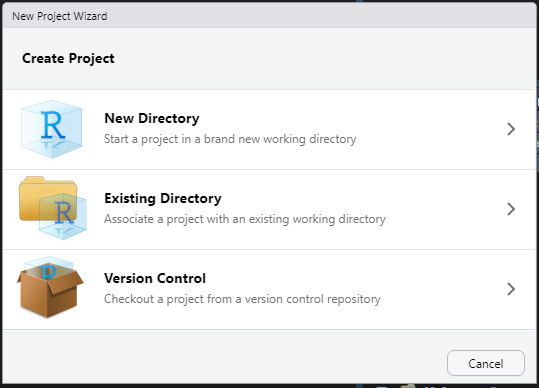
Allí podemos crear una carpeta que se llame “CIENCIA\_DATOS”, procuremos en todos los nombres no dejar espacio, ya que mas adelante se nos facilitara manejarlo así.

**Creando un nuevo proyecto**

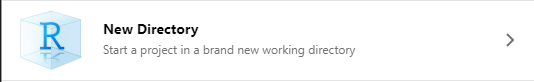
En R crearemos nuestro primer proyecto en donde guardaremos las sesiones de trabajo.



Al darle clic izquierdo a nuevo proyecto nos aparece la pantalla siguiente:



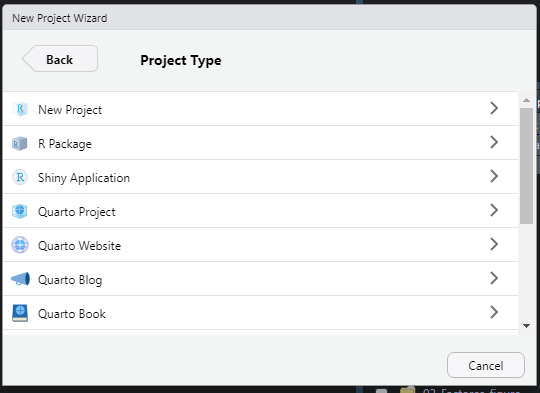
Seleccionaremos nuevo directorio:



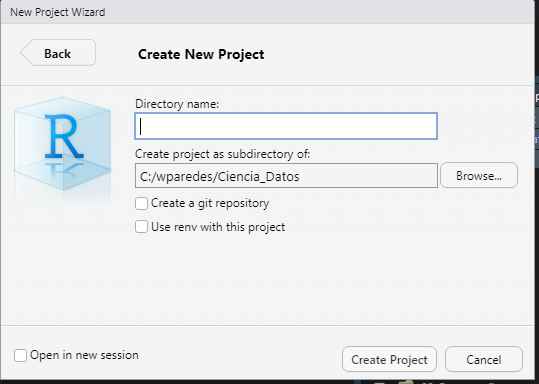
Aquí nos aparecerá una lista de opciones y seleccionaremos la primera que en este caso es nuevo proyecto.



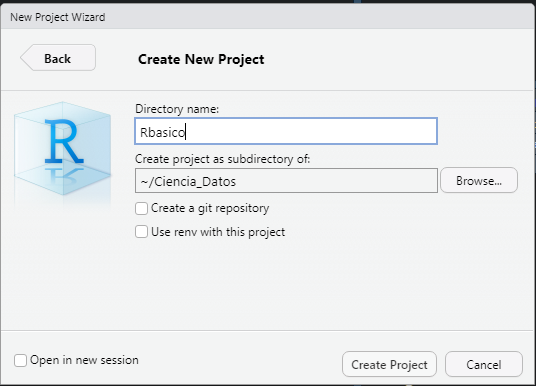
Ejemplo de pantalla de inicio de proyecto.



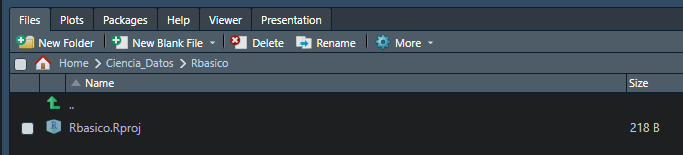
Pulsamos el botón de buscar para seleccionar la carpeta de archivos que previamente creamos.



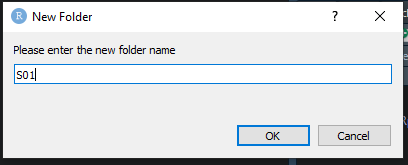
Posteriormente colocamos el nombre del directorio como se llamará el proyecto, en este caso RBasico.



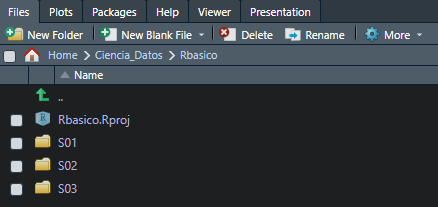
Al pulsar crear proyecto, este nos crea un archivo de proyecto de R en la ubicación seleccionada para que podamos comenzar a trabajar.



El curso lo dividiremos en 3 secciones y si presionamos dentro del icono de nuevo folder podremos crear 3 carpetas las cuales se llamarán S01, S02 y S03.



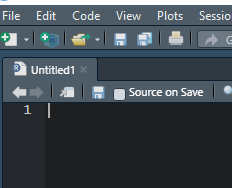
Lo hacemos de una en una para que quede de la siguiente forma:



Ahora presionaremos las teclas:

CTRL+SHIFT+N

El cual nos creará un documento nuevo en donde pondremos el siguiente fragmento de código:





En seguida asignaremos mediante código de R el valor 2023 a una variable llamada 'x':



**En R todo es un objeto**

Ahora se imprimirá el valor de la variable dos veces, la primera vez se hará de manera explícita por medio de la función print(), como sigue:



También lo podemos llamar de manera implícita únicamente tecleando la “X”

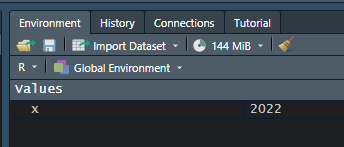


Finalmente haremos una multiplicación de x por 2

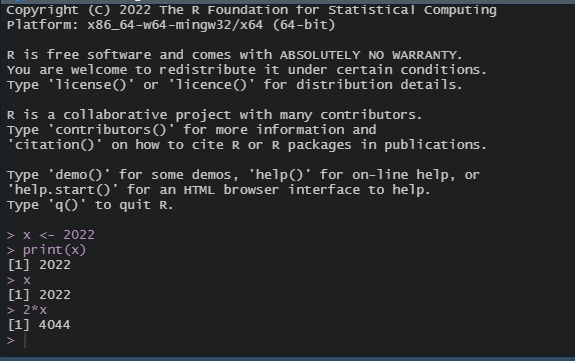


Aquí veremos varias cosas dentro de nuestro software.

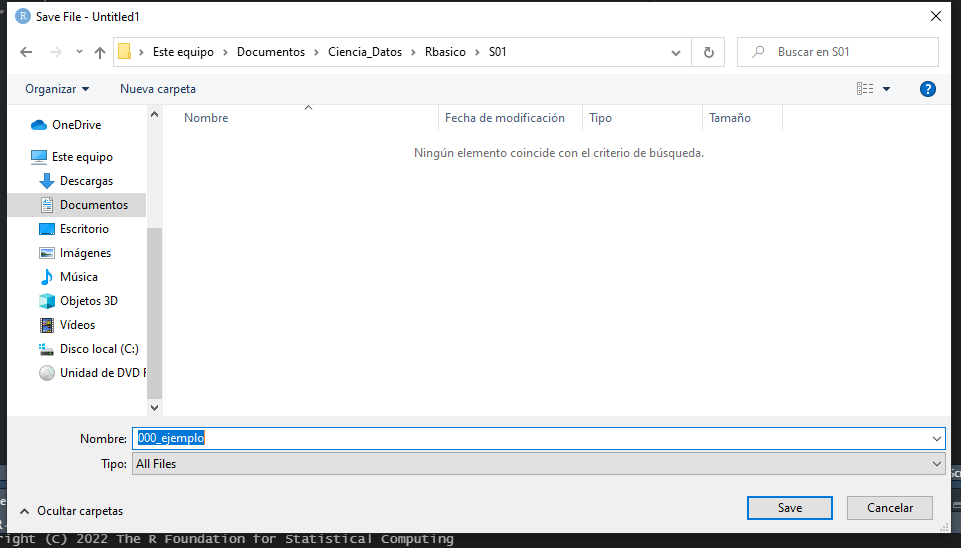
La primera es que en la carpeta de entorno aparece la x con el valor 2023, en conjunto del espacio ocupado en memoria.



En la consola abajo nos aparecen las operaciones anteriormente realizadas:



Vamos a guardar nuestro primer documento, pulsamos el botón guardar del menú y seleccionamos la carpeta “S01” y allí podremos nombrar a nuestro trabajo “000\_ejemplo” para guardar este pequeño ejercicio, de esa forma procederemos con cada uno de los ejemplos que estaremos realizando.



**Algunas características importantes de R**

El sistema R está dividido en dos partes conceptuales:

1. El sistema base de R, que es el que puedes bajar de <https://cran.r-project.org/bin/windows/base/>
2. Todo lo demás.

R se caracteriza por tener varios paquetes modulares, (hoy más de 25,000) los importantes son los estadísticos y los de inteligencia artificial, sin embargo, existen para hacer conexiones por ejemplo a redes sociales, hacer particulares usos de mapas o geometrías, lo importante es antes de comenzar un proyecto plantearnos que queremos hacer o que retos nos va a imponer y posterior buscar las librerías que nos puedan ayudar a cumplir con nuestro objetivo.

Si tenemos curiosidad en el siguiente enlace podremos ver cada uno de los paquetes que ya han pasado por varios filtros de revisión y se encuentran disponibles para la descarga directa:

<https://cran.r-project.org/web/packages/available_packages_by_name.html>

Nuestro sistema base de R contiene el paquete básico que se requiere para su ejecución y la mayoría de las funciones fundamentales. Los otros paquetes contenidos en la “base” del sistema incluye a utils, stats, datasets, graphics, grDevices, grid, tools, parallel, compiler, splines, tcltk, stats4.

**Software Libre**

Debido a que no hay que preocuparse por licencias y cuenta con la libertad que garantiza GNU.

Es decir, con R se tiene la libertad de:

1. Correrlo para cualquier propósito.
2. Estudiar cómo trabaja el programa y adaptarlo a sus necesidades, pues se tiene acceso al código fuente.
3. Redistribuir copias.
4. Mejorar el programa y liberar sus mejoras al público en general.

A las empresas o instituciones el código libre les ahorra millones de dólares al año, muchas de las instituciones del primer mundo hoy por hoy están pasando todas sus aplicaciones a ambientes de código libre.

**Importante**

Debido a su estructura, R consume mucho recurso de memoria, por lo tanto, si se utilizan datos de tamaño enorme, el programa se alentaría o, en el peor de los casos, no podría procesarlos.

En la mayoría de los casos, sin embargo, los problemas que pudieran surgir con referencia a la lentitud en la ejecución del código tienen solución, principalmente teniendo cuidado de vectorizar el mismo adecuadamente; ya que esto permitiría particionarlo y aprovechar en procesamiento paralelo en equipos con multinúcleos.

**Ayuda en R**

R cuenta con una muy buena ayuda en el uso de funciones de manera muy

similar a la función de man de UNIX. para obtener información de cualquier función en específico, por ejemplo, si nosotros quisiéramos saber cómo se utiliza la función para calcular modelos lineales “lm()” el comando es:

help(lm)

O abreviandolo:

¿lm()

Este ultimo es el que personalmente prefiero.