

Explorando el Transcriptoma con Datos de Expresión Genética

Introducción a R & RStudio

Yered Pita-Juárez

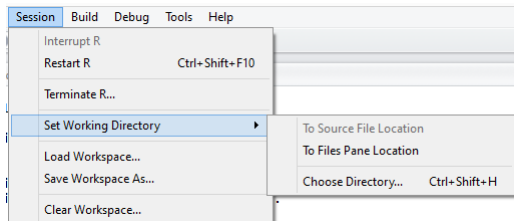
3/1/2015

Working Directory

- Working directory es el folder en tu computadora que R está usando
- Cuando intentes abrir un archivo, R lo buscare en el working directory
- Cuando guardes un archivo desde R, lo guardará en el working directory
- Antes de que empieces a trabajar, asegúrate de que el working directory sea el folder donde están tus datos o donde quieres guardar los resultados
- ¿Cuál es el working directory?
`getwd()`

Working Directory

- Para cambiar el working directory desde la línea de comando usa `setwd("nombredelfolder")`. Por ejemplo
`setwd("C:/UsuarioResignado/Aqui/Mismo/")`
- Usa / en lugar de \
- No olvides usar los apóstrofes ""
- En RStudio
Session/Set working directory



Working Directory

Ejercicio

- ¿Cuál es el `working directory`? Usa el comando `getwd()`.
- Crea un folder llamado `CdeC` y hazlo el `working directory` usando el comando `setwd()`.

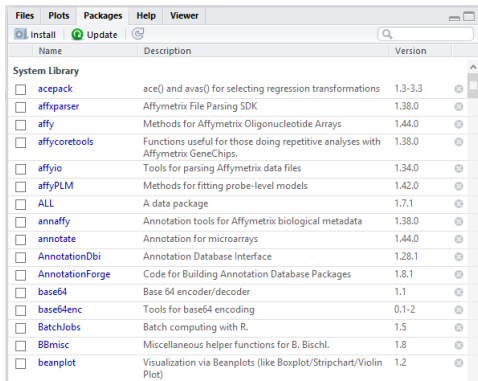


Packages

- Las funciones de R estan organizadas en packages
- ¿Qué packages están instalados?

`library()`

- En RStudio el panel de packages

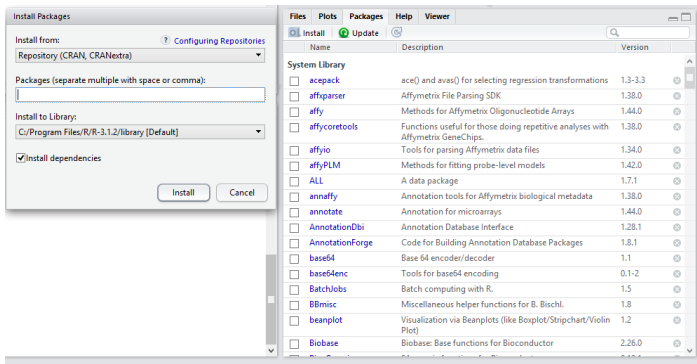


The screenshot shows the RStudio interface with the 'Packages' tab selected. It displays a list of installed and available packages. The table has columns for Name, Description, and Version. A search bar is at the top right of the panel. The 'System Library' section is expanded, showing a list of packages with checkboxes for installation and update actions.

Name	Description	Version
<input type="checkbox"/> acepack	ace() and avas() for selecting regression transformations	1.3-3.3
<input type="checkbox"/> affxparser	Affymetrix File Parsing SDK	1.38.0
<input type="checkbox"/> affy	Methods for Affymetrix Oligonucleotide Arrays	1.44.0
<input type="checkbox"/> affycoretools	Functions useful for those doing repetitive analyses with Affymetrix GeneChips.	1.38.0
<input type="checkbox"/> affyio	Tools for parsing Affymetrix data files	1.34.0
<input type="checkbox"/> affyPLM	Methods for fitting probe-level models	1.42.0
<input type="checkbox"/> ALL	A data package	1.7.1
<input type="checkbox"/> annaffy	Annotation tools for Affymetrix biological metadata	1.38.0
<input type="checkbox"/> annotate	Annotation for microarrays	1.44.0
<input type="checkbox"/> AnnotationDbi	Annotation Database Interface	1.28.1
<input type="checkbox"/> AnnotationForge	Code for Building Annotation Database Packages	1.8.1
<input type="checkbox"/> base64	Base 64 encoder/decoder	1.1
<input type="checkbox"/> base64enc	Tools for base64 encoding	0.1-2
<input type="checkbox"/> BatchJobs	Batch computing with R.	1.5
<input type="checkbox"/> BBmisc	Miscellaneous helper functions for B. Bischl.	1.8
<input type="checkbox"/> beanplot	Visualization via Beanplots (like Boxplot/Stripchart/Violin Plot)	1.2

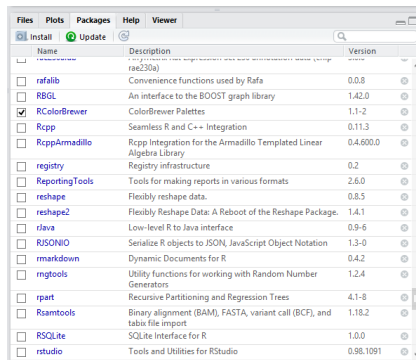
Packages

- Para instalar un package
`install.packages("nombredelpackage")`
- En RStudio



Packages

- Para cargar un package ya instalado
`library("nombredelpackage")`
- En RStudio selecciona el package desde el panel



Packages

Ejercicio

- Instala el paquete RColorBrewer
- Carga el paquete RColorBrewer



Packages

Ejercicio

- Si lo hiciste correctamente, el siguiente comando debe generar un gráfico

```
display.brewer.pal(7,"Accent")
```



Accent (qualitative)

Comandos

- R puede ser usado como una calculadora
- Escribe la ecuación despues de `>` , y presiona `enter`
`> 10^2 + 36`
- R dará la respuesta
`[1] 136`



Comandos

Ejercicio

Calcula la diferencia entre el 2014 y el año en el que empezaste la carrera, divide este número por la diferencia entre el 2014 y el año que naciste. Multiplica este número por 100 para obtener el porcentaje de tu vida que has pasado en la universidad.



Workspace

- Puedes asignar un número a una variable
- Para crear una variable

```
> a <- 4
```

```
0
```

```
> a = 4
```

- En el panel de workspace aparece la variable a
- Para ver el valor asignado a esta variable

```
> a
```

```
[1] 4
```

Workspace

- También, puedes usarla para hacer cálculos

```
> a * 5  
[1] 20
```

- Si vuelves a asignarle un valor a la variable, se pierde el valor que tenía asignado previamente

```
> a = a * 5  
> a  
[1] 20
```

- R es un lenguaje sensible a mayúsculas: A y a son dos símbolos diferentes
- Los nombres de las variables no pueden empezar con un número

Workspace

- Para ver que objetos están en la memoria
`objects()`
- En RStudio: en el panel de `workspace`
- Para eliminar un objeto de la memoria
`rm(nombreobjeto)`
- Para eliminar todos los objetos de la memoria en R
`rm(list=ls())`
- En RStudio usa el botón de `clear` en el panel `workspace`

Ejercicio

Repite el ejercicio anterior usando varias variables para calcular el porcentaje de tu vida que has pasado en la universidad



Workspace

Ejercicio

- Elimina todas los objetos de la memoria de R
- Crea las siguientes variables: $a = 1$, $b = 2$, $c = 3$
- Usa R para calcular $a + b + c$
- Elimina la variable b
- Repite la suma anterior, ¿cuál es el mensaje de error?



Escalares, Vectores y Matrices

- Los valores numéricos en R están organizados en
 - ▶ escalares
 - ▶ vectores
 - ▶ matrices
- Las variables que definimos anteriormente fueron escalares
- Para definir un vector necesitamos la function `c`
`b=c(3,4,5)`

Funciones

- Para calcular el promedio de los elementos del vector `b`
`> (3+4+5)/3`
- Para vectores mas largos, esto no es ideal
- Funciones: operaciones de uso frecuente
- Para calcular el promedio
`> mean(x=b)`
- Dentro del paréntesis están los argumentos de la función
- No siempre es necesario incluir los nombres de los argumentos
`> mean(b)`

Funciones

Ejercicio

- Calcula la edad promedio de la clase.
- Primero preguntale a tus compañer@s su edad, y crea un vector para guardar las edades de la clase
- Usa la función `sum` para calcular el promedio
- Usa la función `mean` para calcular el promedio




Funciones

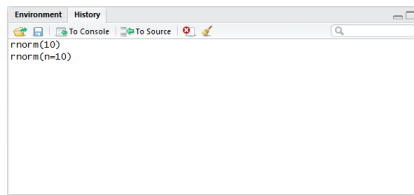
- `rnorm` es una función estándar de R para generar números aleatorios de la distribución normal

```
> rnorm(10)
[1] -0.949 1.342 -0.474 0.403
[5] -0.091 -0.379 1.015 0.740
[9] -0.639 0.950
```

- la primera línea contiene el comando
- Las líneas 2-4 contienen los resultados: un vector con 10 números aleatorios
- ¿Cuál es el comando?
- ¿Cuál es el argumento?
- También podemos usar
`rnorm(n=10)`

Funciones

- Usa  para acceder a comandos usados anteriormente
- En RStudio, usa el panel de `history`



Funciones

- Para generar 10 números aleatorios de una distribución normal con media 1.2 y desviación estándar de 3.4

```
> rnorm(10, mean=1.2, sd=3.4)
```
- La función `rnorm` tiene varios argumentos
- Solo requerimos un argumento, `n`, para usar `rnorm`
- Los demás argumentos son opcionales, y R tiene valores asignados a ellos por default
- En RStudio presiona Tab después de `rnorm(` para ver los argumentos de la función.

```
> rnorm(0)
```

<code>n =</code>	<code>n</code>
<code>mean =</code>	
<code>sd =</code>	number of observations. If length(n) > 1, the length is taken to be the number required.

Press F1 for additional help

Gráficas

- R tiene la capacidad para generar gráficas
- Un ejemplo muy simple

```
> x = rnorm(100)
> plot(x)
```
- Primero generamos 100 números aleatorios de una distribución normal
- En la segunda línea graficamos estos valores

Gráficas

Ejercicio

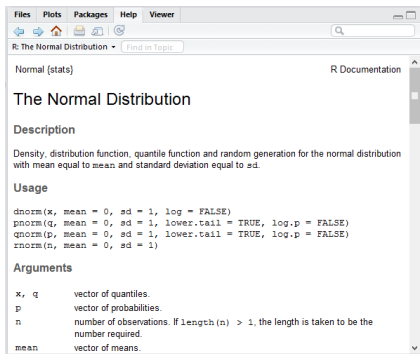
- Las funciones `hist` y `boxplot` son muy útiles para ver la distribución de una variable
- Usa las funciones `hist` y `boxplot` para generar gráficas acerca de la edad de la clase



Ayuda

- R incluye archivos de ayuda para sus funciones

> help(rnorm)



- La ayuda da una descripción de la función y de sus argumentos, así como de los valores default
- Algunas funciones incluyen ejemplos de su uso

> example(rnorm)

Ayuda

- Los archivos de ayuda para una función se pueden acceder también usando

`?nombrefuncion`

- Por ejemplo

`?rnorm`

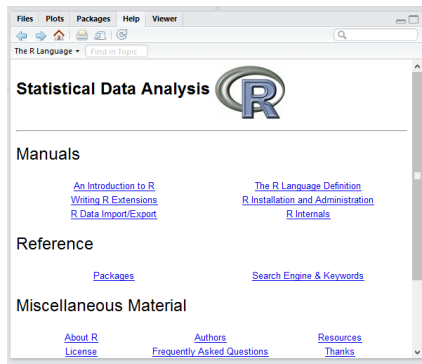
- Para buscar dentro de los archivos de ayuda podemos usar

`??tema`

o

`help.search("tema")`

- El manual de ayuda se puede acceder mediante
> `help.start()`



Ayuda

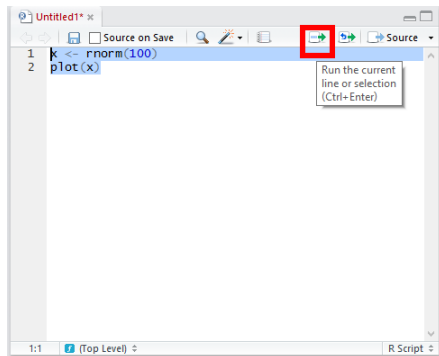
Ejercicio

- Busca los archivos de ayuda relacionados a `hypergeometric`
- ¿Qué función encontraste?
- ¿Qué ejemplos hay disponibles?



Scripts

- R es un lenguaje de línea de comando
- Scripts: archivos para guardar comandos
- Por lo general el nombre es `miscript.R`
- En RStudio, en el panel script puedes correr líneas de comando seleccionandolas y después presionando **CTRL** + **ENTER** o presionando el botón de run



Scripts

Ejercicio

Crea un script llamado `miprimeroscript.R` que genere y grafique 100 números aleatorios

