

The R logo is a large, stylized graphic in the background of the slide. It is composed of several blue and yellow rectangular blocks arranged in a way that suggests the letter 'R'.

# ANÁLISIS DE DATO CON R



contáctenos: [enei@inei.gob.pe](mailto:enei@inei.gob.pe) / 433-3127



Pasaje Hernán Velarde 285 Lima.

Entre la cuadra 01 y 02 de la Av. Arequipa.

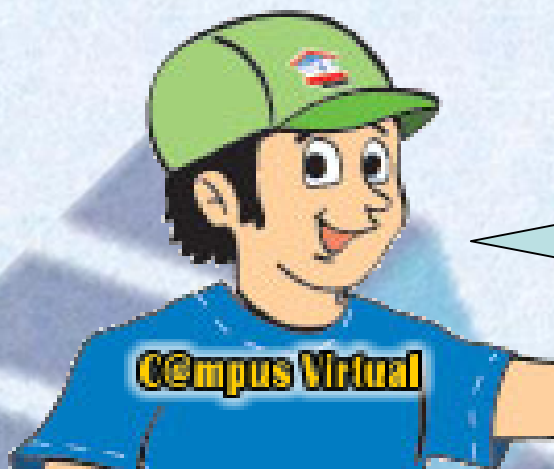
Correo: [enei@inei.gob.pe](mailto:enei@inei.gob.pe) / [campusvirtual@inei.gob.pe](mailto:campusvirtual@inei.gob.pe)

Teléfonos: 433-3127 - 332-4650

## Centro Andino de Formación y Capacitación en Estadística

Cursos Especializados en Estadística e Informática

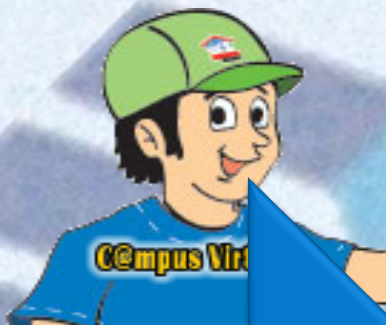




**Estimado alumno,  
buen día. Cualquier  
consulta no dudes en  
comentarme o  
avisarme.**

**Este curso es  
netamente práctico y  
se que lograremos  
objetivos importantes.**

**Recuerda siempre nuestro correo  
para cualquier consulta**

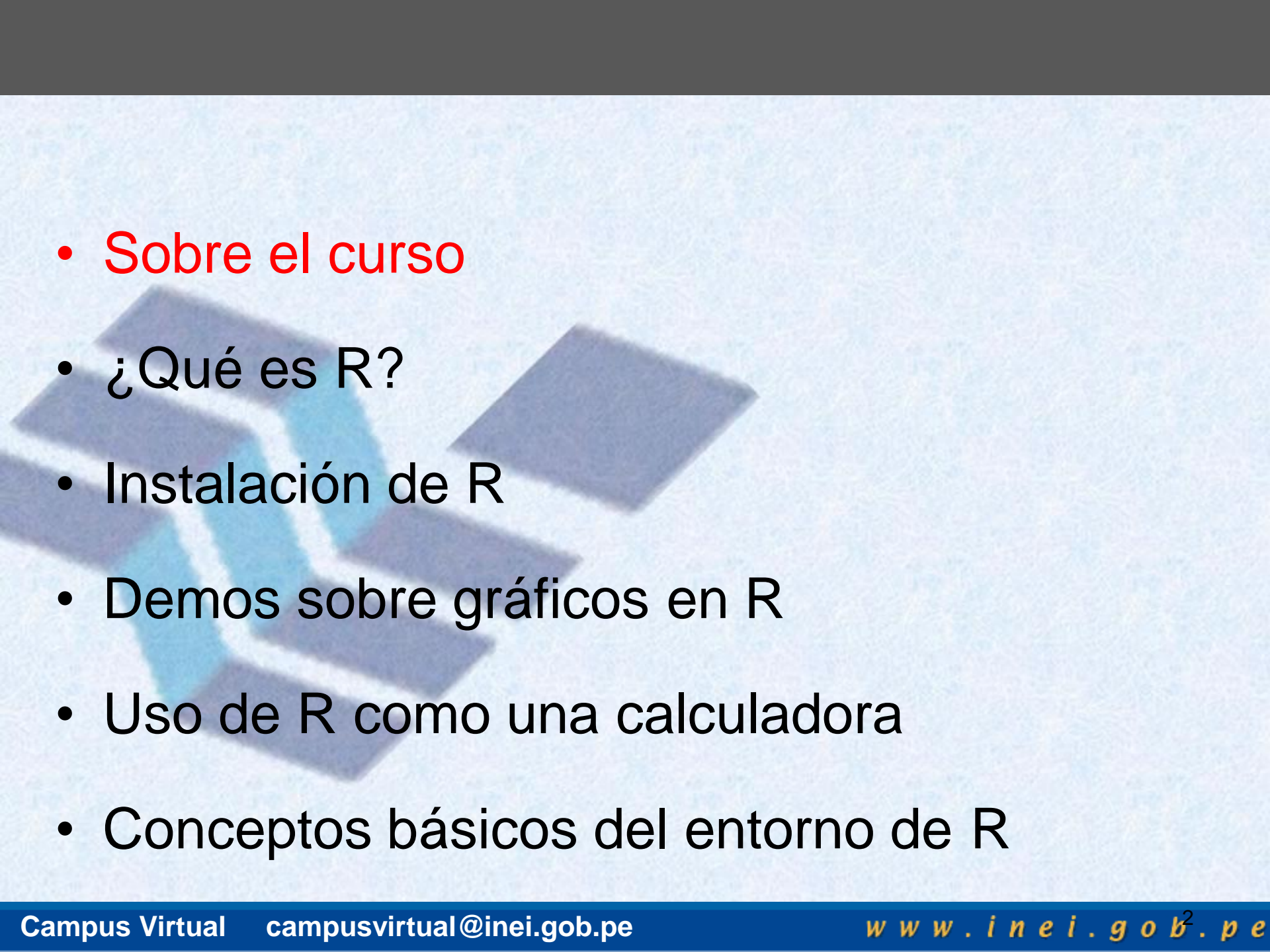


**campusvirtual@inei.gob.pe**





# Introducción al lenguaje R

- 
- Sobre el curso
  - ¿Qué es R?
  - Instalación de R
  - Demos sobre gráficos en R
  - Uso de R como una calculadora
  - Conceptos básicos del entorno de R



# Sobre el curso

- 30 horas (1 sesión de 2h - 7 sesiones de 4 h/cs)
- 02 prácticas calificadas
- Cubre los elementos básico para usar R
- Enfoque práctico dirigido al autoaprendizaje

# Sobre el curso

## Contenido temático:

- Introducción y Preliminares
- Manipulaciones simples: números y vectores
- Arrays y Matrices
- Listas y data frames
- Lectura de datos de ficheros externos
- Distribuciones de probabilidad
- Introducción a la estadística básica y gráficos
- Programación en R
- Procedimientos gráficos en R



- Sobre el curso
- ¿Qué es R?
- Instalación de R
- Demos sobre gráficos en R
- Uso de R como una calculadora
- Conceptos básicos del entorno de R

# ¿Qué es R?

- Es un lenguaje de programación y un entorno para el cálculo de estadísticas y elaboración de gráficas
- Es una implementación de código abierto del lenguaje S, desarrollado por los Laboratorios Bell
- Escrito inicialmente por Ross Ihaka y Robert Gentleman a mediados de los 90s



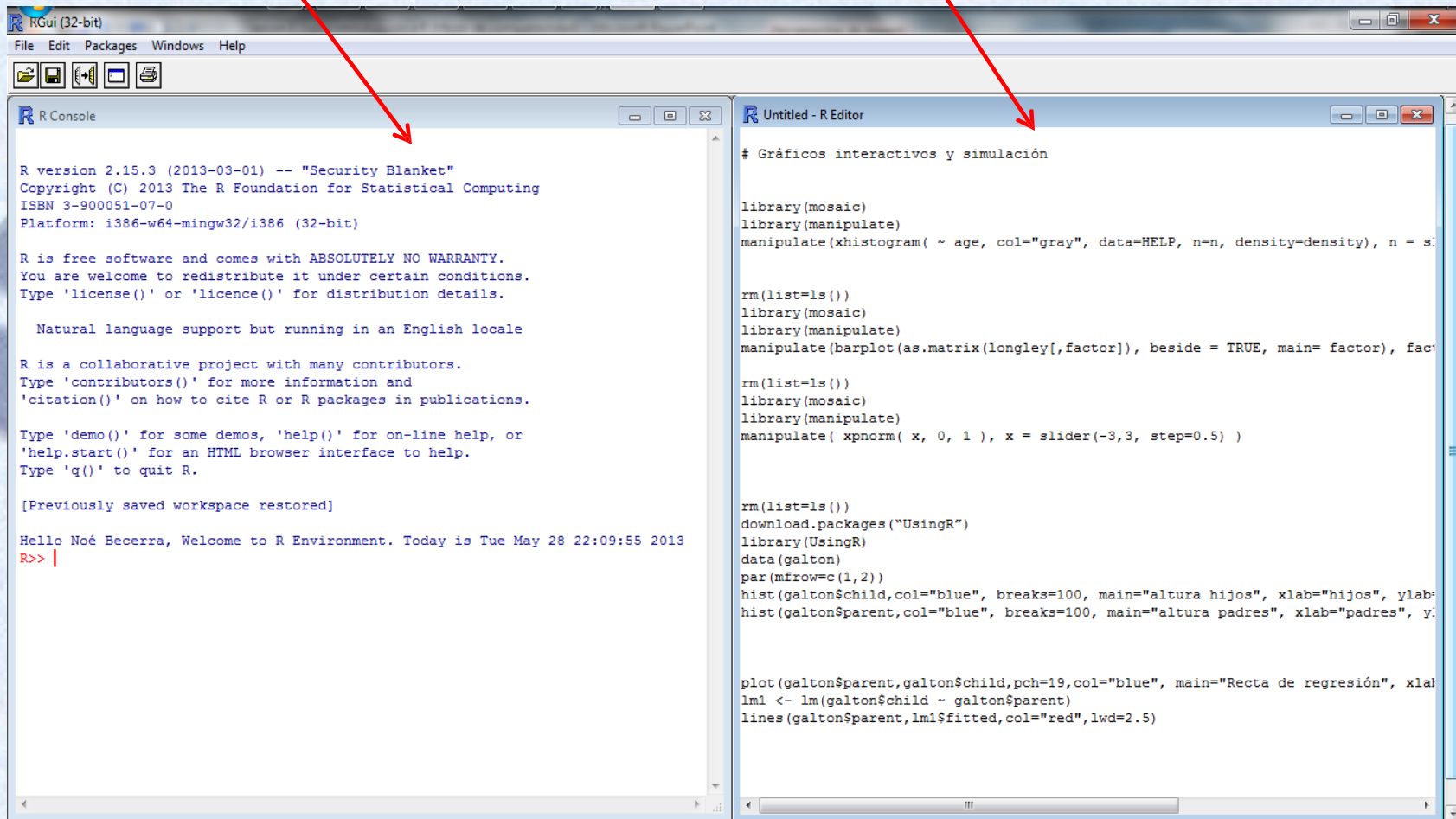
# ¿Qué es R?

- Pros:
  - Gratuito
  - Gran cantidad de paquetes disponibles y muy flexible
  - Una variedad de bases de datos incluidas
  - Amplio soporte en internet
- Cons:
  - Mucho más orientado hacia programación
  - Interface gráfica mínima
  - Curva inicial de aprendizaje muy inclinada

# ¿Qué es R?

Consola

Area de script

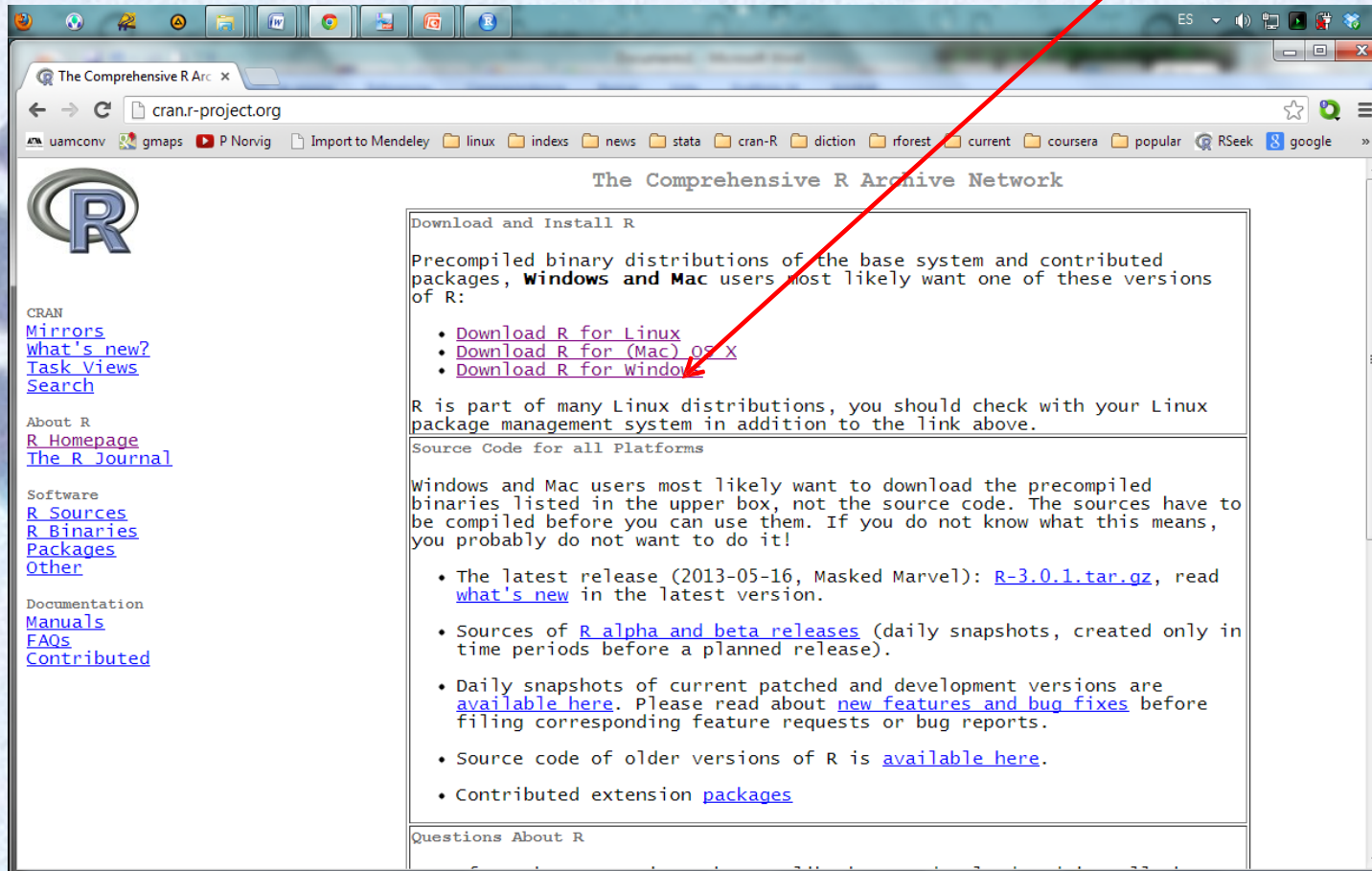




- Sobre el curso
- ¿Qué es R?
- **Instalación de R**
- Demos sobre gráficos en R
- Uso de R como una calculadora
- Conceptos básicos del entorno de R

# Instalar R - Windows

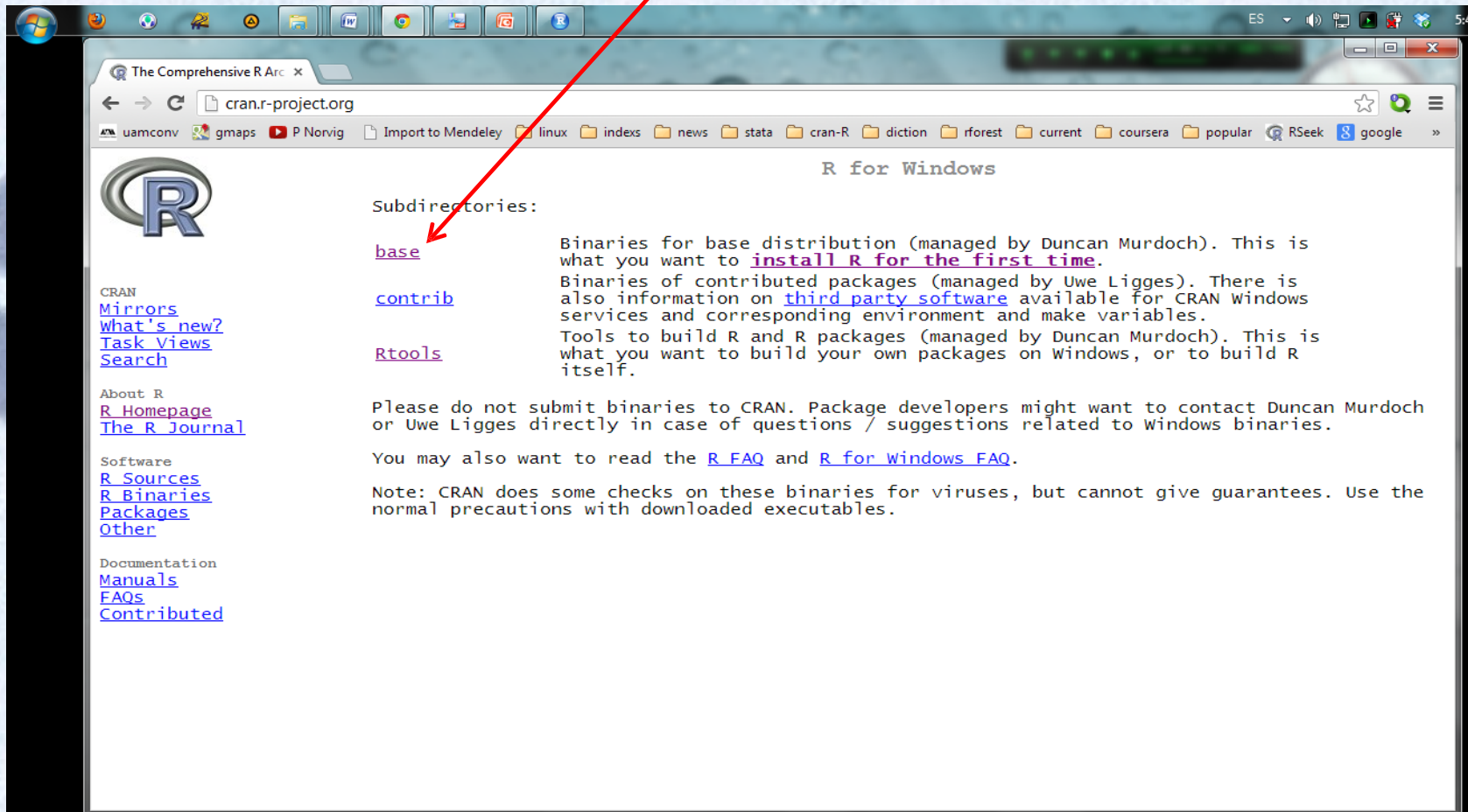
- <http://cran.r-project.org/>





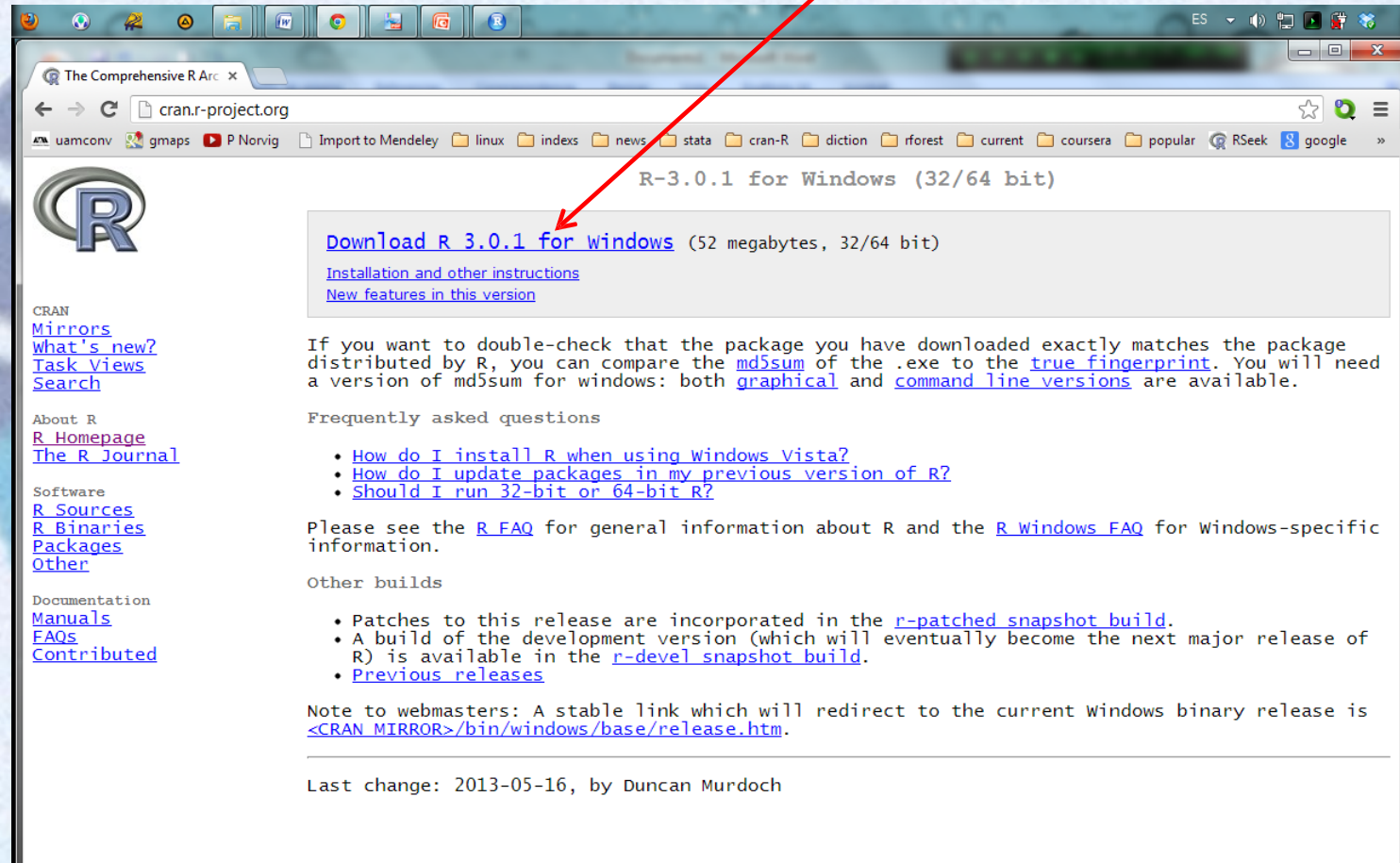
# Instalar R - Windows

- Windows: dar click en “base”



# Instalar R - Windows

- Dar click en download R 3.0.1





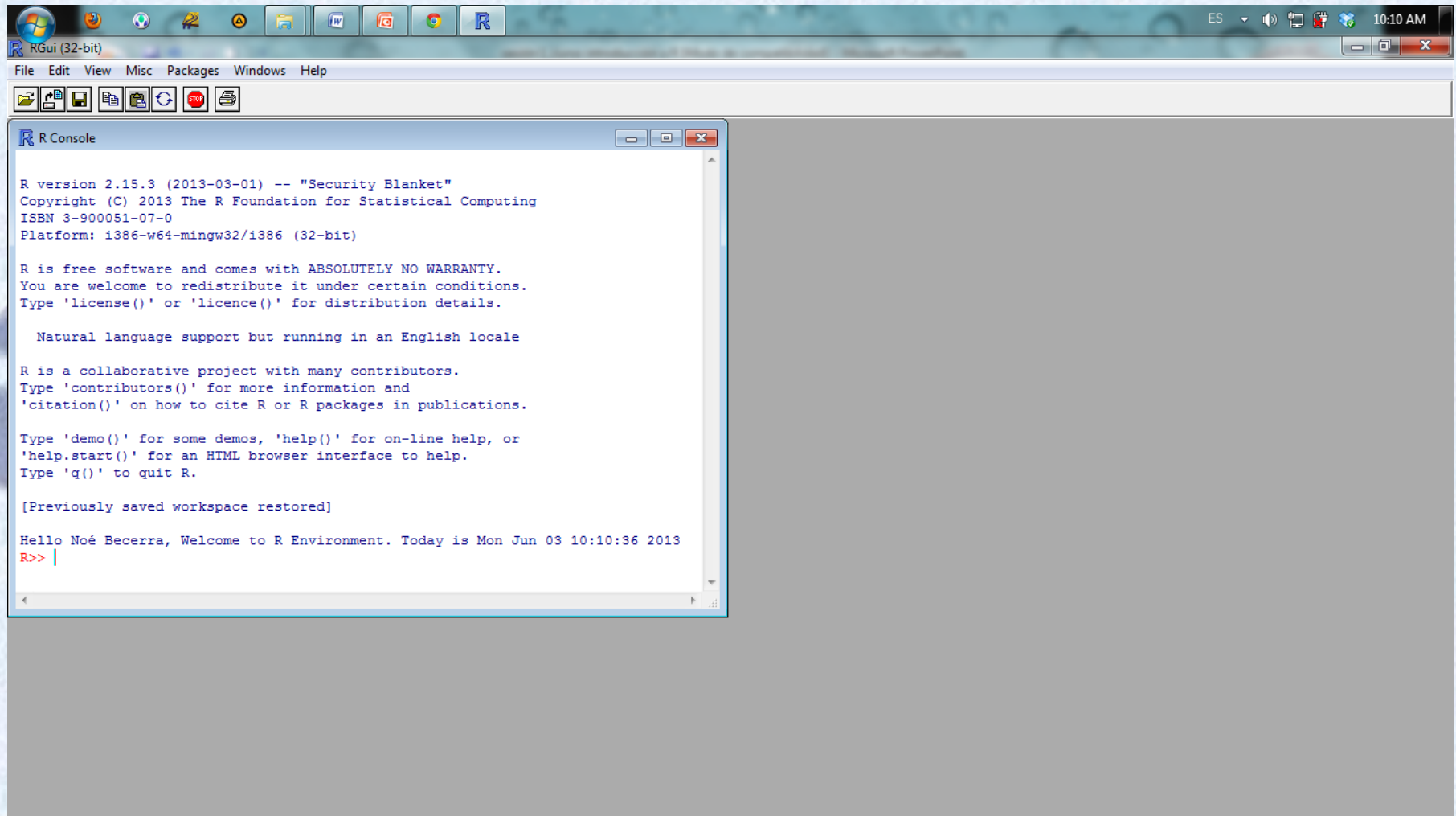
# Instalar R

- Doble click en el archivo que se bajó
- Dar “siguiente” unas pocas veces
- Usar la configuración por “*default*”
- Finalizar la instalación

- Sobre el curso
- ¿Qué es R?
- Instalación de R
- **Demos sobre gráficos en R**
- Uso de R como una calculadora
- Conceptos básicos del entorno de R



# Demos de gráficos en R



# Demos de gráficos en R

**> demo(graphics)**

Dar enter después de observar un poco la gráfica

**> demo(persp)**

Dar enter después de observar un poco la gráfica

Una gráfica interactiva en tres dimensiones

Es necesario primero bajar el paquete rgl

**> install.packages("rgl")**

**> library(rgl)**

**> open3d()**

**> plot3d( iris[,1:3], type="s", size=1.2, col=rainbow(3))**



- Sobre el curso
- ¿Qué es R?
- Instalación de R
- Demos sobre gráficos en R
- **Uso de R como una calculadora**
- Conceptos básicos del entorno de R

# Uso de R como calculadora

La consola de R funciona como una calculadora

**OJO con la precedencia de las operaciones matemáticas**

Además de las operaciones matemáticas ordinarias, R contiene las siguientes funciones:

# logaritmo base e  
**> log(20)**

# logaritmo base 10  
**> log10(234)**

# logaritmo de cualquier base  
**> log(x, base=a)**

# operación exponencial  
**> exp(x)**



# Operador de asignación

Para asignar valores a una variable se utilizan los símbolos:

-> , <- o bien =. Ejemplos

```
> x <- 20
```

```
> a = x+2
```

El comando “c” (concatenar) permite almacenar varios datos en una variable

```
> x=c(2,5,9,14,23, 29,35, 40)
```

```
> x
```

# Generación y simulación de datos

Muchas de las operaciones de R se realizan a través de funciones. Una función es una rutina ya construida en R que se ejecuta cuando se le “llama”. Las funciones tiene uno o varios argumentos donde algunos ya están dados por “default”.

Las siguientes funciones sirven para generar o simular datos

```
> x=c(1,2,3,4,5,6,7,8,9,10); x
```

```
> y= 4:40 ;y
```

```
> z=rep(1:2, c(10,15)) ;z
```

```
> w=seq(from = 1, to = 100, by =2) # genera números del 1 al 100 con  
incrementos de dos, también se puede seq(1,100,2)
```

```
> rnorm(100, mean=0, sd=1) # genera 100 números aleatorios con  
distribución normal estándar, alternatively rnorm(100,0,1)
```



- Sobre el curso
- ¿Qué es R?
- Instalación de R
- Demos de gráficos en R
- Uso de R como una calculadora
- Conceptos básicos del entorno de R

# R workspace (espacio de trabajo)

El espacio de trabajo es el conjunto todos los objetos que el usuario ha definido dentro de R (variables, vectores, funciones, bases de datos, etc)

Los comandos de R terminan casi siempre con paréntesis aunque no tengan ningún parámetro

Los siguientes comando son de uso común en el manejo del espacio de trabajo

# mostrar el directorio actual de trabajo  
> **getwd()**

# listar los objetos del espacio de trabajo  
> **ls()**

# cambiar el directorio de trabajo  
> **setwd("ruta del nuevo directorio")**



# R workspace

## Historia de la sesión

# mostrar los últimos 25 comandos

**> history()**

# mostrar todos los comandos previos

**> history(max.show=Inf)**

# guardar la historia de comandos en un archivo

**> savehistory(file="mi\_archivo.Rhistory")** # el default es ".Rhistory"

# cargar la historia de comandos

**> loadhistory(file="mi\_archivo.Rhistory")** # el default es ".Rhistory"

# R workspace

## Guardar y cargar el workspace

# guardar el workspace en el archivo ".RData" (default)  
> **save.image("mi archivo.Rdata")**

# cargar un espacio de trabajo en la sesión actual de R  
> **load("mi archivo.RData")**

# salir de R  
> **q( )**



# Packages (paquetes)

Los packages consisten de un conjunto de funciones pre-programadas que fueron desarrolladas para tareas específicas. Además, algunos paquetes contienen bases de datos.

Existen dos tipos de paquetes:

Los que vienen por default en la base de instalación de R

Los que están disponibles para descargar y que necesitan ser instalados manualmente

Para descargar paquetes de internet:

**> install.packages("tree")**

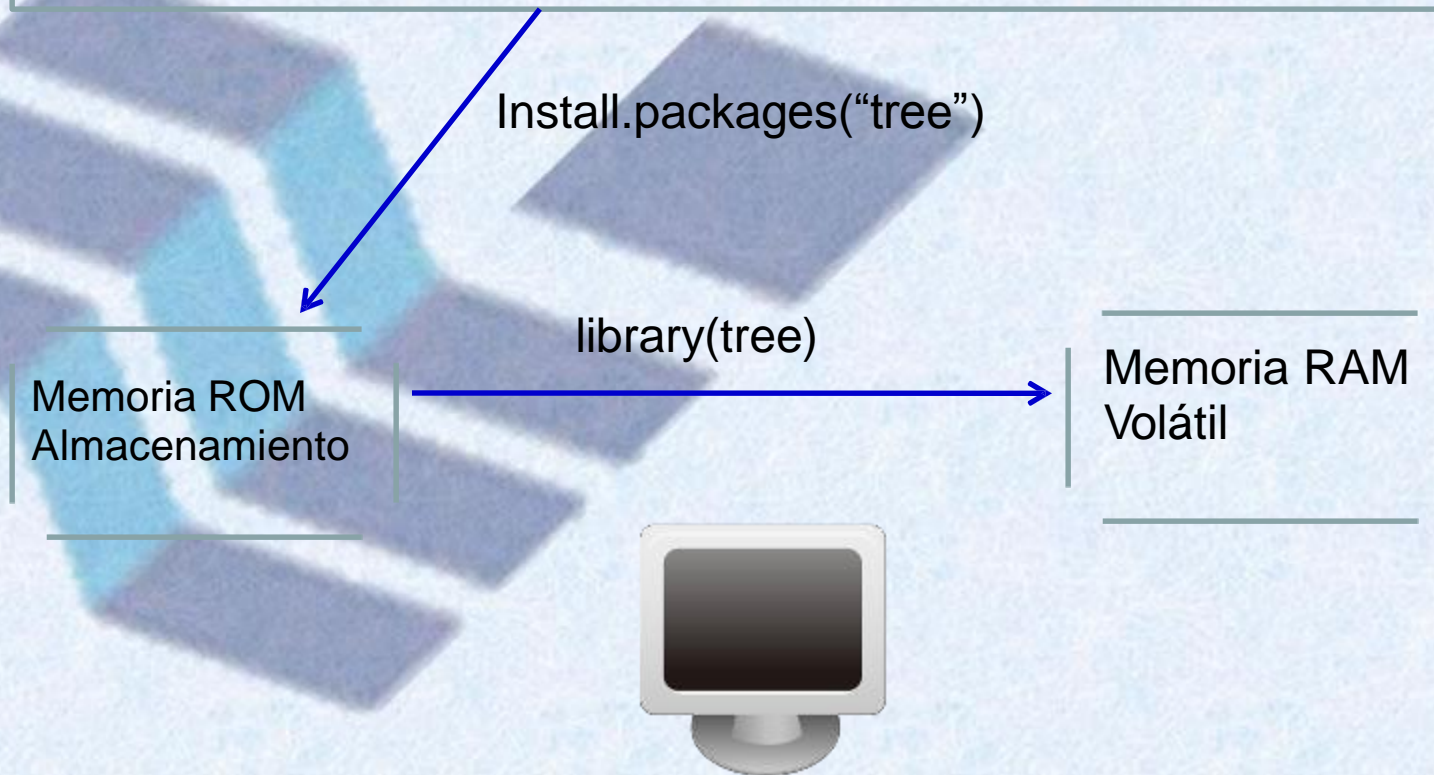
Es necesario cargar en memoria los paquetes para usar los comandos

**> library(tree)**

**> require(tree)**

# Packages R

## Mirrors





# Sobre scripts

Existen dos formas para trabajar con R: de forma interactiva y por medio scripts

De forma interactiva se escriben los comandos en la consola y éstos se ejecutan de forma inmediata en cuanto se presiona la tecla intro

Un script es un conjunto de comandos que se pueden ejecutar en la consola y que además se pueden guardar como archivo. En el menú file se da click en New Script y aparece una pantalla donde se pueden anotar todos los comandos o funciones que se van a ejecutar. Este modo de trabajo permite ejecutar los comandos uno por uno y ver los resultados

Los scripts se guardan con la extensión “.R” y pueden ser cargados nuevamente en sesiones subsiguientes

Recomendable anexar comentarios en los scripts y las funciones!  
# es el símbolo para introducir comentarios

# Sobre NaN y NA

NaN quiere decir que no es un número (Not a Number).

**> 3/0**

**> 0/Inf**

**> (0:3)\*\*Inf**

**> 0/0**

**> Inf-Inf**

**> Inf/Inf**

NA indica que es un valor faltante (missing value). El comando `is.na(x)` sirve para determinar si se tienen valores faltantes en un vector

**> w=c(1:6, NA, NA,9:12)**

**> w**

**> is.na(w)**

**> which(is.na(w))**



# ¿Cómo buscar ayuda?

La primera opción para buscar ayuda es abrir la pestaña Help en el menú de R.

Si nos encontramos en internet podemos entrar al sitio de CRAN-R y buscar el tema de interés. Otros sitios que pueden ser útiles son:

<https://stat.ethz.ch/mailman/listinfo/r-help>

<http://stackoverflow.com/>

<http://www.statmethods.net/>

<http://stats.stackexchange.com/>

<http://journal.r-project.org/>

<http://www.rseek.org/>

# ¿Cómo buscar ayuda?

Buscar ayuda cuando conocemos el nombre del comando

```
> ? read.table
```

```
> ?? read.table
```

Si no conocemos el nombre de la función pero sí conocemos el tema sobre el que trata esa función:

```
> help.search("data input")
```



# ¿Cómo buscar ayuda?

El comando `args()` muestra todos los parámetros que necesita una función específica

**> args(mean)**

La función `apropos("lm")` nos muestra una lista de todos los objetos que tienen en su nombre la cadena de caracteres que se encuentran entre las comillas.

**> apropos("lm")**

La siguiente función despliega el contenido del paquete que está en paréntesis

**> library(help="randomForest")**

# Errores comunes

Olvidar el/los paréntesis (o comillas, cuando se necesitan) al escribir los comandos

```
> library(rgl
```

+

```
> install.packages(foreign)
```

**Error in install.packages(foreign) : object 'foreign' not found**

Omitir la creación de un objeto antes de hacer alguna operación con ese objeto

```
> read.csv("base empresas_mayo_2013.csv")
```

```
> names(base)
```

**> Error: object 'base' not found**

```
> base = read.csv("base empresas mayo_2013.csv")
```

```
> names(base)
```

```
[1] "Folio"      "product_inno" "process_inno" "ti_low"      "ti_high"
```

```
[6] "services"   "ln_size"      "RandD"        "ln_age"      "links_univ"
```



# Errores comunes

Olvidar cargar el paquete necesario para realizar el procedimiento deseado

```
> open3d()
```

**Error: could not find function "open3d"**

```
> library(rgl)
```

```
> open3d()
```

Errores de escritura en los comandos

```
> getwe()
```

**Error: could not find function "getwe"**

```
> getwd()
```

```
[1] "C:/Users/Owner/Documents/_R stuff"
```

# Errores comunes

Omitir signo de operaciones matemáticas (producto)

```
> x=12; y=7; z=5
```

```
> (x-y)(x-z)
```

**Error: attempt to apply non-function**

```
> x=12; y=7; z=5
```

```
> (x-y)*(x-z)
```

```
[1] 35
```



# Errores comunes

## Ejercicio

Dados la siguiente serie de datos sobre peso y altura

Peso 60, 72, 57, 90, 95, 72

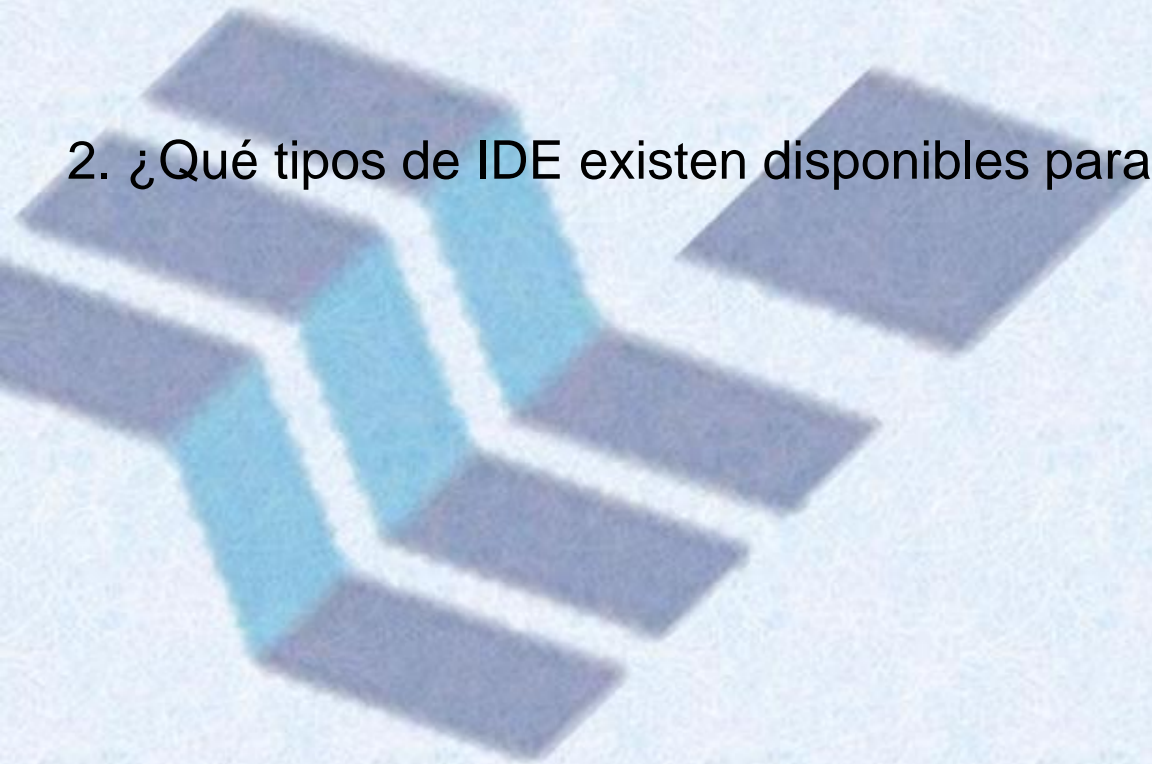
Altura 1.75, 1.80, 1.65, 1.90, 1.74, 1.91

realizar las operaciones siguientes en R

1. Calcular la media, la varianza y la desviación estándar
2. Calcular el índice de masa corporal ( $IMC = \text{peso} / \text{altura}^2$ )
3. Hacer un diagrama de dispersión de los datos

# Tarea

1. Investigar qué es un IDE (Integrated **D**evelopment **E**nvironment)
2. ¿Qué tipos de IDE existen disponibles para R?





## Comunicación constante con la Escuela del INEI

**Correo de la Escuela del INEI**  
**[enei@inei.gob.pe](mailto:enei@inei.gob.pe)**

**Área de Educación Virtual**  
**[campusvirtual@inei.gob.pe](mailto:campusvirtual@inei.gob.pe)**

