
Universidad
Tecmilenio®

Universidad Tecmilenio

Fundamentos de programación para Big Data

Programación para Big Data en lenguaje R

Semana 4

Introducción

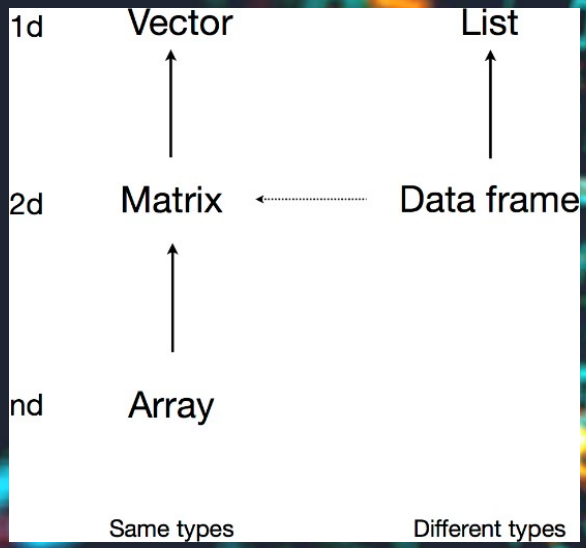
R es una excelente alternativa para ser más productivo en relación con las herramientas ofimáticas usuales, ya que día con día existen cada vez más datos que requieren un análisis sofisticado.

Uno de los componentes primordiales de esta productividad es la oportunidad de automatizar, mediante la programación, las tareas que sean tediosas y repetitivas (Gil, 2018).

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

En R se puede trabajar con distintas **estructuras de datos**, algunas son de una sola dimensión y otras permiten más, como se indica en el siguiente diagrama:



El lenguaje R está orientado a objetos, lo cual le proporciona flexibilidad y simplicidad. Los objetos creados se almacenan en la memoria activa del equipo de cómputo bajo el nombre señalado por la persona que edita el documento.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Observa la creación de los objetos “a” y “b”:

```
a <- 5
```

```
b <- 8
```

Mediante R es posible realizar operaciones aritméticas básicas, por ejemplo:

```
a+b
```

```
a-b
```

```
a*b
```

```
a/b
```

Puedes guardar los resultados de las operaciones que realices en R dentro de un objeto, por ejemplo:

```
c<-2*a + 100/b
```

Los **objetos** en R pueden ser datos, funciones, resultados, variables, entre otros. Además, cuenta con dos atributos intrínsecos que son los siguientes:

- ✓ **Longitud** (número de elementos en el objeto).
- ✓ **Tipo** (puede ser número, completo, carácter y lógico).

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

De acuerdo con Hernández (2019), los **vectores** son arreglos ordenados donde se puede almacenar información de diversos tipos:

- ✓ **Numérica** (variable cuantitativa).
- ✓ **Alfanumérica** (variable cualitativa).
- ✓ **Lógica** (falso o verdadero).

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

La función que permite crear un vector es **c()** y dentro de los paréntesis se localizará la información que se va a almacenar.

Debe tener un nombre pequeño y representativo del contenido de la información, que se asigna por medio del operador <-, por ejemplo:

```
edad <- c(15, 19, 13, NA, 20)
deporte <- c(TRUE, TRUE, NA, FALSE, TRUE)
comic.fav <- c(NA, 'Superman', 'Batman', NA, 'Batman')
```

El primero, contiene la edad de cinco personas, el símbolo NA significa que no se tiene registro de la edad de una persona. El segundo es un vector lógico que guarda las respuestas a la pregunta: ¿usted práctica algún deporte? El tercero contiene la historieta favorita de las personas entrevistadas.

Para extraer el valor de un vector se debe escribir el nombre del vector y la posición, por ejemplo:

```
edad[3]
```

```
## [1] 13
```

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Un **arreglo** (array) es una matriz con diversas dimensiones, que puede contener información numérica, alfanumérica o lógica. Para crear un arreglo, utilizamos la función **array()** .

El código para realizar un arreglo de 3 x 4 x 2 con las primeras 24 letras del alfabeto es el siguiente:

```
miarray <- array(data=letters[1:24], dim=c(3, 4, 2))
```

Las **matrices** son arreglos de forma rectangular de filas y columnas con información numérica, alfanumérica o lógica. Para crear una matriz, utilizamos la función **matrix()** .

El código para realizar una matriz de 4 filas y 5 columnas con los primeros 20 números positivos es el siguiente:

```
mimatriz <- matrix(data=1:20, nrow=4, ncol=5, byrow=FALSE)
```

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Universidad Tecmilenio

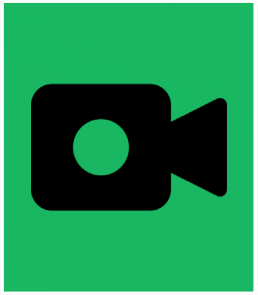
Dataframes

Muy frecuentemente, los datos se disponen en tablas: hojas de cálculo, bases de datos, ficheros csv, etc.

Además, casi todos los métodos estadísticos (como la regresión lineal) operan sobre información organizada en tablas.

Como consecuencia, gran parte del trabajo con R consiste en manipular tablas de datos para darles el formato necesario para acabar analizándolos estadística o gráficamente (Gil, 2018).

Para conocer más detalles sobre cada uno de los conceptos del tema, ve el siguiente video:



Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Funciones en Python* [Archivo de video]. Recuperado de https://bit.ly/2N2UmMY

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Cierre

Este tema es el punto medular del aprendizaje en R, pues todo lo que puede proseguir será conocimiento estadístico aplicado sobre estos objetos. Practica continuamente con múltiples casos de uso donde puedas utilizar este lenguaje.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Bibliografía

- Gil, C. (2018). *R para profesionales de los datos: una introducción* . Recuperado de https://www.datanalytics.com/libro_r/
- Hernández, F. (2019). *Manual de R* . Recuperado de https://fhernanb.github.io/Manual-de-R/objetos.html#vectores

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Universidad Tecmilenio

Fundamentos de programación para Big Data

Presentación de datos

Semana 4

Introducción

La visualización de los resultados es sumamente importante, ya que te ayuda a encontrar patrones entre los datos durante el análisis o a presentar la información de manera sencilla y entendible para el receptor final.

Los gráficos deben mostrar la información adecuada para tomar decisiones en el momento justo.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Universidad Tecmilenio

Visualización de datos en cada uno de los lenguajes aprendidos durante el curso:

R

Para conocer las funciones necesarias para importar, exportar, guardar y la lectura de datos en R (de distintas fuentes externas), te recomendamos leer el siguiente material desde la página 57 hasta la 70:

Charte, F. (2014). *Análisis exploratorio y visualización de los datos con R* . Recuperado de https://bit.ly/30zBZF2

Python

Para conocer el proceso necesario para importar, exportar, guardar y realizar la lectura de datos en Python, te recomendemos ver el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Funciones en Python* [Archivo de video]. Recuperado de https://bit.ly/2N2UmMY

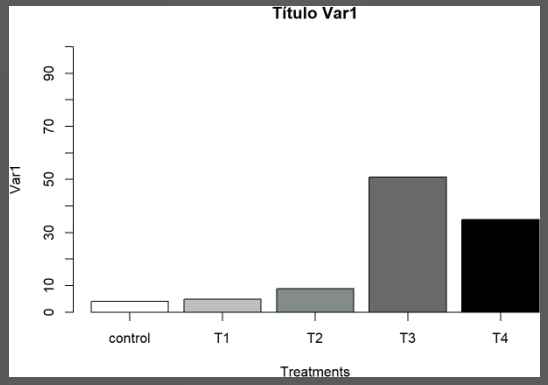
Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Gráficos

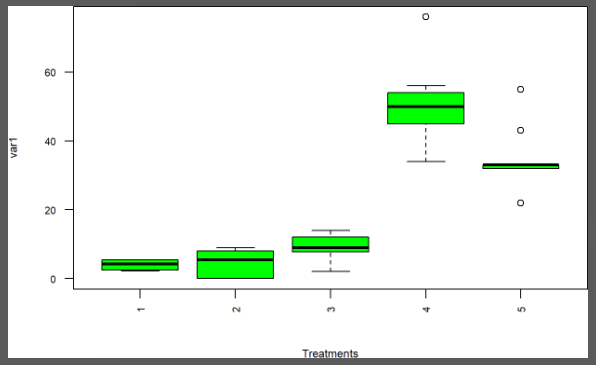
Los lenguajes **R** y **Python** tienen la capacidad de generar diferentes gráficas, las cuales te ayudan a comprender los datos con los que trabajas, al realizar un primer análisis visual y al formar conclusiones iniciales. A continuación, se presentan las gráficas para ambos lenguajes:

Barras



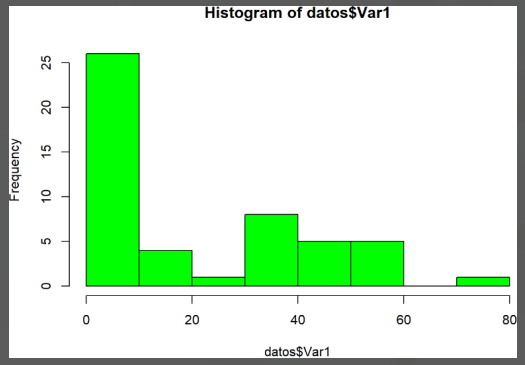
Para cuando tienes variables dependientes asociadas por tratamientos.

Boxplot



Señala la distribución de una variable mediante cuartiles.

Histograma



Señala la distribución de datos numéricos, en donde cada barra indica la frecuencia de datos en un rango.

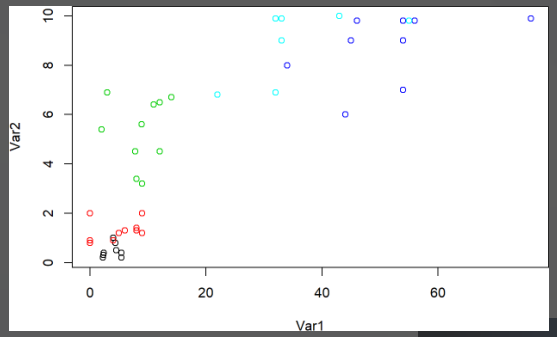
Fuente: Fernández, A. (2019). *Gráficos en R.* Recuperado de https://rpubs.com/aafernandez1976/graficos

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

Y no menos importante:

Dispersión



Señala la relación entre dos variables continuas mediante puntos.

Fuente: Fernández, A. (2019). *Gráficos en R.* Recuperado de https://rpubs.com/aafernandez1976/graficos

Existen diversas **librerías** para ambos lenguajes que te ayudan a realizar la visualización de datos, entre las cuales están las siguientes:

R	ggplot2
Python	Matplotlib
Python	Bokeh

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Explicación

R cuenta con diversas funciones para la creación de gráficos, un ejemplo es la función **plot()**, que se utiliza para generar diversos tipos de visualizaciones.

La estructura de esta función con los argumentos más comunes es la siguiente:

```
plot(x, y, type, main, sub, xlab, ylab)
```

Existen funciones que generan gráficos para variables cuantitativas y cualitativas. Para conocer sobre cada una de ellas, revisa los apartados dos y tres del siguiente libro electrónico:

Hernandez, F., y Correa, J. (2020). *Gráficos con R*. Recuperado de https://bit.ly/2R0ojj3

Para conocer un ejemplo de visualización de datos en R, ve el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Funciones en Python* [Archivo de video]. Recuperado de https://bit.ly/2N2UmMY

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®).

Explicación

Universidad Tecmilenio

Python

Para Python también existe la función **plot**, que permite realizar gráficas mediante la librería **Matplotlib**. Esta función, de acuerdo con Interactive Chaos (s.f.), "recibe un conjunto de valores x e y, y los muestra en el plano definido por los ejes como puntos unidos por líneas".

```
In [7]: x = [1, 2, 3, 4]
        y = [2, 5, 1, 7]
        plt.plot(x, y)
        plt.show()
```

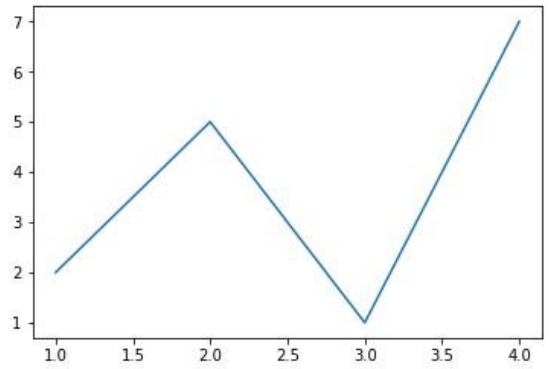


Figura 6. Fuente: InteractiveChaos. (s.f.). *La función plot*. Recuperado de https://bit.ly/2R2g71E

Para conocer un ejemplo de visualización de datos en Python, ve el siguiente video:

Dirección de Producción de Contenidos. (2019, 4 de diciembre). *Tema 5. Funciones en Python* [Archivo de video]. Recuperado de https://bit.ly/2N2UmMY

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®).

Cierre

La información se puede obtener correctamente usando fuentes verídicas con buena presentación, pero si no ha estado disponible en el momento que se necesitaba tomar una decisión, entonces no será útil.

Como analistas de datos es muy importante preservar el valor de estos en función del tiempo.

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .

Bibliografía

- Fernández, A. (2019). *Gráficos en R* . Recuperado de https://rpubs.com /aafernandez1976/graficos
- InteractiveChaos. (s.f.). *La función plot.* Recuperado de https://bit.ly/2R2g71E

Derechos de Autor Reservados. (UNIVERSIDAD TECMILENIO®) .