

Lecturas Importantes

1. Artículo de Ari Lamstein <http://www.kdnuggets.com/2016/11/data-science-101-good-at-r.html> de como empezar a programar en R para ciencia de datos.
2. Artículo sobre el ciclo de desarrollo de ciencia de datos de Vincent Granville.
<http://www.datasciencecentral.com/profiles/blogs/life-cycle-of-data-science-projects>.

Preguntas

1. Dada una lista cuyos miembros son vectores de cadena de caracteres de diferentes longitudes, utilice una función con `lapply` para pegar un signo de exclamación al final de cada elemento de cada miembro, con una cadena vacía como carácter de separación (tenga en cuenta que el comportamiento predeterminado de `paste` cuando se aplica a los vectores de caracteres). Ejecuta tus líneas de código en la lista dada por:

```
> f3 <- list("a",c("b","c","d","e"),"f",c("g","h","i"))
```

2. (a) Fijar cada uno de los siguientes error y corregirlos

```
> mtcars[-1:4, ]  
> mtcars[mtcars$cyl<= 5]  
> mtcars[mtcars$cyl== 4 | 6, ]
```

- (b) Por qué `x <- 1:5; x[NA]` produce 5 valores NA?
(c) Por qué `mtcars[1:20]` retorna un error. Cómo difiere del similar `mtcars[1:20,]`?
(d) Qué hace `df [is.na (df)] <- 0`?. Como funciona?
3. Escribe una función llamada `geolista` que puede buscar a través de una lista especificada y calcula las medias geométricas de cada miembro según las siguientes pautas:
 - La función debe definir y utilizar una función de ayuda interna que devuelve la media geométrica de un argumento vectorial.
 - Suponga que la lista sólo puede tener vectores numéricos o matrices numéricas como sus miembros. La función debe contener un bucle apropiado para inspeccionar a cada miembro de la lista a la vez.
 - Si el miembro de la lista es un vector, calcula la media geométrica de ese vector, sobrescribiendo el miembro con el resultado, que debe ser un solo número.
 - Si el miembro de la lista es una matriz, utilice un bucle implícito para calcular la media geométrica de cada fila de la matriz, sobrescribiendo el miembro con los resultados.
 - La lista final debe ser devuelta al usuario.

Ahora, como una prueba rápida, compruebe que tu función coincida con las siguientes dos llamadas:

```
f4 <- list(1:3,matrix(c(3.3,3.2,2.8,2.1,4.6,4.5,3.1,9.4),4,2),  
           matrix(c(3.3,3.2,2.8,2.1,4.6,4.5,3.1,9.4),2,4))  
geolista(f4)  
[[1]]  
[1] 1.817121  
[[2]]  
[1] 3.896152 3.794733 2.946184 4.442972  
[[3]]  
[1] 3.388035 4.106080
```

```
f5 <- list(1:9,matrix(1:9,1,9),matrix(1:9,9,1),matrix(1:9,3,3))
geolista(f5)
[[1]]
[1] 4.147166
[[2]]
[1] 4.147166[[3]]
[1] 1 2 3 4 5 6 7 8 9
[[4]]
[1] 3.036589 4.308869 5.451362
```

4. (a) ¿Qué produce los siguientes códigos y las propiedades que muestran

```
> f1 <- function(x = {y <- 1; 2}, y = 0) {
+   x + y
+ }
> f1()
```

- (b) ¿Por qué las siguientes dos invocaciones de `lapply` son equivalentes?

```
> trims <- c(0, 0.1, 0.2, 0.5)
> x <- rcauchy(100)
>
> lapply(trims, function(trim) mean(x, trim = trim))
> lapply(trims, mean, x = x)
```

- (c) Considera el siguiente problema : Dada una matriz numérica X , determina el índice de la primera fila de números positivos que no contiene NA . Resuelve el problema usando `for` y la función `apply()`.

- (d) ¿Cómo se determina el entorno desde el que se llama una función?

5. Dado un vector (x_1, x_2, \dots, x_n) , la autocorrelación muestral es definida como

$$r_k = \frac{\sum_{i=k+1}^n (x_i - \bar{x})(x_{i-k} - \bar{x})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

- (a) Escribe una función `tmpFn(xVec)` que toma un sólo argumento `xVec` que es un vector y retorna una lista de dos valores r_1 y r_2 . En particular encuentra los valores de r_1 y r_2 para el vector $(2, 5, 8, \dots, 53, 56)$.
- (b) Generaliza la función de manera que toma dos argumentos: el vector `xVec` y el entero k que está entre 1 y $n - 1$ donde n es la longitud de `xVec`. La función debe retornar un vector de los valores $(r_0 = 1, r_1, \dots, r_n)$.
6. El conjunto de datos *iris* contiene las medidas de la longitud y el ancho (en cm) de pétalos y sépalos de tres especies: 1: Setosa, 2: versicolor y 3: Virginica.

- Considera el objeto *iris*. ¿ Como está estructurado?. ¿ Cuantas observaciones(lineas) contiene?. ¿ Cuantas variables (columnas) contiene?.
- Para tener una visión general del conjunto de valores, utiliza la función `summary()` del conjunto de dato. ¿Qué información sobre el conjunto de datos proporciona?.
- Para la variable `Sepal.Length` verifica los resultados dados, usando las funciones `min()`, `max()`, `mean()`, `median()`, `quantile()`. Si es necesario usa la ayuda de `?quantile`.

Notas importantes

- Todos los ejercicios resueltos serán subidos a los repositorios de github creado por el estudiante. La forma en la que el estudiante debe presentar este laboratorio es:
 - Debes crear una carpeta local llamada Practica3.
 - Dentro de la carpeta Practica1, se deben crear las carpetas Ejercicio1, Ejercicio2, Ejercicio3, Ejercicio4, Ejercicio5 donde se deben alojar las soluciones con la extensión Rmd para los comentarios que son parte de las preguntas y .R del lenguaje R para los otros. Cualquier otra extensión, incluyendo letras minúsculas, será motivo de una rebaja de puntos en el ejercicio.
- Los archivos de respuesta deben llevar un comentario inicial con tu nombre y código. Por ejemplo.

```
> # Nombre : Cesar Lara Avila 20122345J
> # Respuesta1: El codigo muestra el uso del SVD en el calculo
> #de la inversa de matrices
>
>
> library(MASS)
>
> a <- matrix(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0,
+             0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1), 9, 4)
>
> a.svd <- svd(a)
> a.svd$d
```

- En esta práctica no se permite utilizar sitios de internet.