

## Lecturas Importantes

1. Un artículo de KDnuggets sobre manipulación de datos usando dplyr.  
<http://www.kdnuggets.com/2017/08/next-generation-data-manipulation-dplyr.html>.
- 

## Preguntas

1. Escribe en un archivo con la extensión sh, las respuesta a las siguientes preguntas:
  - (a) Enumera todas las líneas de un archivo que no terminan con un punto y coma.
  - (b) Redirecciona el número de usuarios actualmente conectados a un archivo usuarios.
  - (c) Encuentra desde /bin y /usr/sbin todos los nombres de archivos que comienzan con s.
  - (d) Reporte el número de líneas que contienen un número dado, digamos 60, en todos los archivos del directorio actual.
2. Escribe operaciones en R, para generar cada uno de los siguientes vectores
  - El vector conteniendo los valores  $1, -2, 3, -4, \dots, 99, -100$ .
  - El vector conteniendo los primeros 100 valores del factorial.
  - El vector conteniendo las primeras 100 potencias de 2.
3. (a) Demuestra que, cuando  $a$  es un escalar y  $x$  es un vector, `match(a, x)` es equivalente a `min(which(x == a))`. Discute y muestra ejemplos de los usos de `match` y `which` en el caso de la comparación de dos vectores. Compare esto con el uso de `%in%`.  
(b) Utiliza la función `system.time` para comparar los tiempos de ejecución de los siguientes comandos de R.

```
• > y=c();for (t in 1:100) y[t]=exp(t)
```

```
• > y=exp(1:100)
```

```
• > y=sapply(1:100,exp)
```

- (c) Este es el texto del famoso trabalenguas sea shells:

```
> sea_shells <- c(  
+ "She", "sells", "sea", "shells", "by", "the", "seashore",  
+ "The", "shells", "she", "sells", "are", "surely", "seashells",  
+ "So", "if", "she", "sells", "shells", "on", "the", "seashore",  
+ "I'm", "sure", "she", "sells", "seashore", "shells"  
+ )
```

Utiliza la función `nchar` para calcular el número de letras en cada palabra. Ahora realiza un bucle sobre las longitudes de las palabras, mostrando un mensaje sobre la longitud de esas palabras. Por ejemplo, en palabras de longitud seis, debe decir que las palabras `shell` y `surely` tienen seis letras.

4. Observa la siguiente función `tst()` que calcula una cadena de calificación para un número único  $x$ .

```
> tst4 <- function(x) {  
+   if (x < -2) "muy negativo"  
+   else if (x < 1) "cercano a cero"  
+   else if (x < 3) "in [1, 3)"  
+   else "large"  
+ }
```

- (a) Corre la función para  $x = 0$ ,  $x = \text{Inf}$  y  $x = 2$ .
- (b) Escribe una nueva función `Tst4()` dando la misma salida que `tst4()`, pero define la función usando el comando `ifelse()`. De esta manera, la función no sólo funcionará para números individuales sino también para un vector  $x$ . Prueba tu función `Tst4()` en el vector de entrada `c(-Inf, -3: 5)`.
5. Un famoso juego del lanzamiento de monedas tiene las siguientes reglas: El jugador lanza una moneda repetidamente hasta que aparece un sello o la lanza un máximo de 1000 veces si no aparece sello. La apuesta inicial comienza en 2 soles y se duplica cada vez que aparecen caras.
- La primera vez que aparece sello, el juego termina y el jugador gana si está participando. Por lo tanto, el jugador gana 2 soles si sello aparece en el primer lanzamiento, 4 soles si cara aparece en el primer lanzamiento y sellos en el segundo, 8 soles si caras aparecen en los dos primeros lanzamientos y sellos en el tercero, y así sucesivamente.
- Matemáticamente, el jugador gana  $2^k$  soles, donde  $k$  es igual al número de lanzamientos hasta el primer sello. ¿Cuál es la probabilidad de tener beneficios si cuesta 15 soles participar?

### 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 Practica1.
  - Dentro de la carpeta Practica1, se deben crear las carpetas Ejercicio1, Ejercicio2, Ejercicio3, Ejercicio4, Ejercicio5 donde se deben alojar las soluciones con la extensión `sh` para el ejercicio 1 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, 0, 1, 1, 1), 9, 4)
>
> a.svd <- svd(a)
> a.svd$d
```

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