

Ejercicios de evaluación

Ejercicios de evaluación

Sitio: [eGela UPV/EHU](#)
Curso: Software matemático y estadístico
Libro: Ejercicios de evaluación
Imprimido por: Clara Benlloch Coscollà
Día: viernes, 25 de septiembre de 2020, 19:10

Tabla de contenidos

[1. Ejercicios de evaluación](#)

[1.1. Ejercicio 1.1](#)

[1.2. Ejercicio 1.2](#)

1. Ejercicios de evaluación

Te propongo a continuación dos ejercicios.

La escala de evaluación tiene cuatro valores: *, **, ***, ****.

Las evaluaciones de ambos valdrán para la nota final.

Observaré la fidelidad alcanzada con las propuestas, así como la claridad con la que has escrito el código.

Tienes que enviarme desde las tareas *Entrega del ejercicio* de la página principal del aula, los códigos que reproducen los resultados propuestos, antes de las **7:00 del miércoles día 30 de septiembre** en el editor dispuesto para ello, de forma que yo los pueda ejecutar directamente en **R**. Para ello no debes enviarme un *copiar y pegar* de la [RConsole](#) sino del fichero de textos del script o código, es decir, sin que aparezca '**>**'. Procura que el nombre del fichero tenga un nombre identificativo tuyo y del ejercicio; por ejemplo, 'yy.1.1.txt'.

1.1. Ejercicio 1.1

Carga la base de datos '[iris](#)' y responde a las siguientes cuestiones mediante el código de *R*:

1. ¿Cuáles son los identificadores de las plantas que tienen estas dos características a la vez: largura del pétalo entre 1.6 y 4.0, y anchura del pétalo entre 1.2 y 1.5? (sin incluir esos valores puntuales: (1.6, 4.0), (1.2, 1.5)).
2. ¿Cuántas son las plantas del punto anterior 1.?
3. ¿De qué especies son las plantas del punto anterior 1?
4. ¿Cuáles son los valores correspondientes a las demás características (largura del sépalo, anchura del sépalo) de las plantas del punto anterior 1.?
5. Ordena el conjunto de todas las plantas según la anchura del pétalo, de mayor a menor: la planta que tenga la anchura más grande será la primera, y la que tenga la más pequeña la última.
6. Asigna a cada planta (de la 1ª a la 150ª) el lugar o *puesto* que ocupa en el ordenamiento anterior, es decir, establece un [ranking](#) (ten en cuenta los empates). Añade una columna al *dataframe* '[iris](#)' en la que aparezca en cada planta el *puesto* asignado (se trata de una [variable](#) numérica).
7. Considera los *puestos* que han resultado para las plantas de la especie 'setosa' (recuerda que las plantas están ordenadas según la anchura del pétalo), y calcula la suma de los mismos. Haz lo mismo con las especies 'virginica' y 'versicolor'. Al ser la cantidad de plantas igual para cada especie, la suma ordena las especies según la anchura del pétalo. Si las cantidades no fueran las mismas (50 de cada especie), el orden entre las especies se calcularía mediante la *media aritmética* de cada especie. En cualquier caso, se podría luego decir qué especie tiene, en general (o *en términos medios*), la mayor anchura del pétalo. A partir de aquí entramos en cuestiones de estadística.

1.2. Ejercicio 1.2

El mus es un juego de naipes o cartas (<http://es.wikipedia.org/wiki/Baraja>) que es muy popular entre los vascos, pero no sólo entre vascos. De hecho, hay incluso un campeonato mundial.

La modalidad del juego más habitual es la de 4 jugadores (llamados 'muslari's, que en lengua vasca quiere decir jugador de mus), dos contra dos, alrededor de una mesa, intercalados, o cada pareja situada un miembro en frente del otro.

Se juega con 40 cartas, que están repartidas en 4 palos (oros, copas, espadas, y bastos). Cada palo consta de 10 naipes, que están numerados del 1 al 7 y del 10 al 12. Digamos, por ejemplo, que los números del 1 al 10 se corresponden al 1 de oros, 2 de oros, ..., 7 de oros, 10 de oros, 11 de oros, 12 de oros; los números del 11 al 20, 1 de copas, 2 de copas, ..., 7 de copas, 10 de copas, 11 de copas, 12 de copas; del 21 al 30, 1 de espadas, ... 12 de espadas; del 31 al 40, 1 de bastos, ..., 12 de bastos.

El juego consiste en sucesivos repartos de 4 cartas a cada jugador. En cada reparto se juegan 4 lances ('grande', 'pequeña', 'pares', y 'juego'), y en cada lance se hacen apuestas de puntos entre las parejas, al modo del póker. Al final de cada reparto cada pareja obtiene una cantidad de puntos que se acumula a las obtenidas en anteriores repartos. La pareja que antes llegue a 30 puntos (en una de las modalidades del mus) se anota un punto general (al modo del juego del tenis). La pareja que obtenga antes 3 puntos generales (en una de las modalidades del mus) es la ganadora de la partida (<http://es.wikipedia.org/wiki/Mus>).

En este ejercicio, más que entrar en el detalle del devenir de una partida, se trata de hacer la simulación de un reparto de naipes (4 a cada uno de los cuatro jugadores), y de determinar cuál es la disposición de los jugadores en el lance del 'juego' (otra cosa será quién ganará, y cuántos puntos se llevará la pareja correspondiente, pues esto depende del proceso de apuestas, donde la situación de la partida y la 'psicología' de los jugadores es una componente muy importante).

Por tanto, el ejercicio consiste en dar el código de los comandos que llevan a dar los siguientes resultados:

1. Define la `baraja` de naipes o cartas de tal forma que:

```
> names(baraja[ 7])
[1] "Oros.7"
> names(baraja[33])
[1] "Bastos.3"
```

2. Reparte 16 de las 40 cartas entre los 4 jugadores ('A1', 'B1', 'A2', 'B2'), y determina las cartas que ha recibido cada jugador. Por ejemplo, usando la semilla de azar '2012' ('`set.seed(2012)` ', en 'Constantes, vectores y factores' del capítulo de 'Estructuras de datos')

```
> names(A1)
[1] "Oros.11" "Espadas.12" "Copas.1" "Bastos.5"
> names(B1)
[1] "Espadas.7" "Bastos.1" "Copas.11" "Bastos.7"
> names(A2)
[1] "Espadas.11" "Bastos.2" "Oros.10" "Copas.10"
> names(B2)
[1] "Oros.12" "Copas.12" "Copas.7" "Oros.2"
```

3. En el lance del 'juego' las cartas que son figuras ('sota', 'caballo' y 'rey', es decir, las que llevan respectivamente los números 10, 11 y 12 de cada palo) valen 10 puntos. Cada jugador suma los puntos de sus 4 cartas. Construye un *dataframe* que dé los puntos de cada jugador:

```
> juego.df
  juego.A1 juego.B1 juego.A2 juego.B2
1       26       25       32       29
```

4. El lance del 'juego' tiene dos modalidades, dependiendo de los puntos sumados por los jugadores. La modalidad principal sucede cuando al menos uno de los jugadores consigue sumar 31 puntos o más. Añade una nueva columna o variable al *dataframe* anterior, de modo que se refleje esta situación.

```
> juego.df
  juego.A1 juego.B1 juego.A2 juego.B2    J
1       26       25       32       29 TRUE
```

5. Procede a los repartos correspondientes con las semillas '2013', '2014', '2015', y añade nuevas filas al *dataframe* 'juego.df':

```
> juego.df
  juego.A1 juego.B1 juego.A2 juego.B2      J
1       26      25      32      29  TRUE
2       40      25      26      23  TRUE
3       23      21      21      24 FALSE
4       10      25      19      24 FALSE
```