Laboratorio de R

Curso: Introducción a la Estadística y Probabilidades CM-274

Lecturas Importantes

1. `http://composingprograms.com/` es un libro online sobre paradigmas y conceptos de introducción a la Ciencia de la COmputación, basados en el Libro de de Abelson y Sussman `Structure and Interpretation of Computer Programs (SICP)`.

2. Curso de Inteligencia Artificial de la Universidad de Berkeley `http://inst.eecs.berkeley.edu/~cs188/sp09/lectures.html` soportado por John DeNero.

Preguntas

1. ¿Cuál es la diferencia entre los objetos `data.frame` y `data,table`?.

2. ¿ Cuál es la diferencia entre las funciones `read.table()` y `read.ftable()`?.

3. Los siguientes son una muestra de observaciones sobre la radiación solar entrante en un invernadero:

   11.1 10.6 6.3 8.8 10.7 11.2 8.9 12.2

   (a) Asigna los datos a un objeto `solar.radiacion`.

   (b) Encontrar la media, mediana y la varianza de las observaciones obtenidas sobre la radiación solar.

   (c) Agregar 10 a cada observación de `solar.radiacion` y asigna el resultado a `sr10`. Encontrar la media, la mediana y la varianza de `sr10`. Cuál de las estadística cambia y por cuanto?.

   (d) Multiplica cada observación por -2 y asigna el valor a `srm2`. Encontrar la media, la mediana y la varianza de `srm2`. Como las estadísticas cambian?.

4. Considera el conjunto de datos `islands` y prueba el siguiente código

```
> islands
> hist(log(islands,10), breaks="Scott", axes=FALSE, xlab="area",
+ main="Histograma de Areas de Islas")
> axis(1, at=1:5, labels=10^(1:5))
> axis(2)
> box()
```

   (a) Explica que está ocurriendo en cada paso del código de anterior.

   (b) Modifica el código, incorporando la regla de Sturges en lugar de la regla de Scott. En este caso, se debe usar la función `round` para asegurar que los excesivos números de digitos no son usados en las etiquetas de los ejes.

5. Los datos de `iris` corresponden a las medidas en centímetros de las variables `length, width sepal` y `length , width petal` es decir ancho y altura de los pétalos y sépalos respectivamente, de 50 flores cada una de tres especies de iris. Hay cuatro variables numéricas correspondientes al sépalo y pétalo y un factor `Species`. Muestra una tabla de medias para `Species` (donde las medias se deben calcularse por separado para cada una de las tres `Species`).

6. Supongamos que rodamos un par de dados 1000 veces.

   (a) Se puede simular 1000 lanzamientos de un dado utilizando la función de R `sample(6, 1000, replace = TRUE)`. Utilizando esta función dos veces, almacena 1000 lanzamientos simulados del primer dado en la variable `dado1` y 1000 lanzamientos simulados del segundo dado en la variable `dado2`.

   (b) Para cada par de lanzamientos, calcular la suma de los lanzamientos, y almacena la suma en la variable `suma-dado`.

   (c) Utilice la función `table` para tabular los valores de la suma de lanzamientos. Calcular las proporciones para cada valor de la suma y compara esas proporciones con las probabilidades exactas de la suma de dos lanzamientos de dados.

7. The National Institute of Standards and Technology tiene una página web que lista los primeros 500 dígitos del numero irracional π. Podemos leer esos dígitos en R, por medio del script

```
> pidigits =
+ read.table("http://www.itl.nist.gov/div898/strd/univ/data/PiDigits.dat",
+ skip=60)
```

   Usa la función `table` para construir una tabla de frecuencias de los dígitos del 1 al 9

8. Visualiza los datos `mtcars` incluidos con R y lee la documentación usando `?mtcars`. Muestra diagramas de caja paralelas de las variables cuantitativas. Encuentra una manera revelar alguna relación entre las variables?.

9. Encuentra el conjunto de datos `accident` del paquete `hmmm`. ¿Cuántas filas tiene el conjunto de datos?.

10. Comparar los tiempos de ejecución de los tres órdenes, usando `system.time`

```
> y<-c();for (t in 1:100) y[t]=exp(t)
> y<-exp(1:100)
> y<-sapply(1:100,exp)
```

11. Un Sudoku es una cuadrícula 9×9 que se completa de números del 1 al 9 de manera que cualquier número entre 1 y 9 sólo aparece una vez en una fila, una columna o un bloque de 3×3 de la cuadrícula. Este ejercicio resuelve una cuadrícula de Sudoku simple donde las entradas vacías se llenan una a la vez mediante la exclusión de todas las posibilidades una vez.

    El Sudoku que resolvemos está dada por

```
> s=matrix(0,ncol=9,nrow=9)
> s[1,c(6,8)]=c(6,4)
> s[2,c(1:3,8)]=c(2,7,9,5)
> s[3,c(2,4,9)]=c(5,8,2)
> s[4,3:4]=c(2,6)
> s[6,c(3,5,7:9)]=c(1,9,6,7,3)
> s[7,c(1,3:4,7)]=c(8,5,2,4)
> s[8,c(1,8:9)]=c(3,8,5)
> s[9,c(1,7,9)]=c(6,9,1)
```

   (a) Imprime la cuadrícula en la pantalla.

   (b) Definimos el array `pool= array (TRUE, dim = c(9,9,9))` de posibles valores para cada entrada (i, j) de la cuadrícula, `pool[i, j, k]` es FALSE si el valor de k puede ser excluido. Escribir el código R que actualiza `pool` para las entradas ya llenas.

   (c) Si i es un entero entre 1 y 81, explica el significado de $s[i]$.

(d) Demuestra que, para una entrada dada (a, b), los índices de los números enteros en el mismo cuadro de 3×3 como (a, b) se definen por

```
> boxa=3*trunc((a-1)/3)+1
> boxa=boxa:(boxa+2)
> boxb=3*trunc((b-1)/3)+1
> boxb=boxb:(boxb+2)
```

(e) Deducir que los valores de una entrada (a, b) que todavia no están determinados se puede excluir por

```
> for (u in (1:9)[pool[a,b,]])
+   pool[a,b,u]=(sum(u==s[a,])+sum(u==s[,b])+
+       sum(u==s[boxa,boxb]))==0
```

y que ciertas entradas corresponden a

```
> if (sum(pool[a,b,])==1) s[i]=(1:9)[pool[a,b,]]
```

(f) Resolver la cuadrícula con una exploración aleatoria de entradas (a, b) que sigue, siempre y cuando la `sum(s == 0)> 0`.

12. Usa la función `curve` para mostrar el gráfico de la función $f(x) = e^{-x^2}/(1 + x^2)$ sobre el intervalo $0 \leq x \leq 10$. Entonces usa la función `integrate` para calcular el valor de la integral

$$\int_0^\infty \frac{e^{-x^2}}{1+x^2} dx$$

El límite superior en el infinito es especificado por `upper=Inf` en la función `integrate`.

13. ¿Qué retorna `DF <- D(expression(cos(x)/sin(x)), "x")`. Si ejecutamos el comando `x <- pi/4`, que retorna `eval(DF)`?.

14. ¿Qué produce el siguiente código?.

```
> f1 <- function(x) f(x-1)
> f2 <- function(x) f(x/2)/2
> f3 <- function(x) 2*x*f(x^2)
> f4 <- function(x) f(1/x)/x^2
> f5 <- function(x) f(exp(x))*exp(x)
> f6 <- function(x) f(log(x))/x
> x=seq(0,10,by=.025)
> plot(x,f(x), ylim=c(0, 1.3), xlim=c(0, 10), main="Densidades Teoricas",
+      lwd=2, type="l", xlab="x", ylab="")
> lines(x,f1(x), lty=2, lwd=2)
> lines(x,f2(x), lty=3, lwd=2)
> lines(x,f3(x), lty=4, lwd=2)
> lines(x,f4(x), lty=1, col="grey", lwd=2)
> lines(x,f5(x), lty=2, col="grey", lwd=2)
> lines(x,f6(x), lty=3, col="grey", lwd=2)
> legend("topright", lty=1:4, col=c(rep("black", 4), rep("grey", 3)),
+        leg=c("X","X+1","2X", "sqrt(X)", "1/X", "log(X)", "exp(X)"))
```

y

```
> set.seed(123)
> x <- rgamma(100, 2)
> x1 <- x+1
```

```
> x2 <- 2*x
> x3 <- sqrt(x)
> x4 <- 1/x
> x5 <- log(x)
> x6 <- exp(x)
> plot(density(x), ylim=c(0, 1), xlim=c(0, 10), main="Densidades Empiricas",
+ lwd=2, xlab="x", ylab="f_X(x)")
> lines(density(x1), lty=2, lwd=2)
> lines(density(x2), lty=3, lwd=2)
> lines(density(x3), lty=4, lwd=2)
> lines(density(x4), lty=1, col="grey", lwd=2)
> lines(density(x5), lty=2, col="grey", lwd=2)
> lines(density(x6), lty=3, col="grey", lwd=2)
```

15. Explica, el siguiente script en *R*.

```
> m = 5000; a = 0; b = 1 # constantes
> w = (b - a)/m     # ancho de cada rectangulo
> g = seq(a + w/2, b - w/2, length=m) # m puntos en la particion
> const = 1/sqrt(2 * pi)      # const. para la funcion densidad
> h = const * exp(-g^2 / 2)
> sum(w * h)                   # area total (approx. prob.)
```