

Ejercicios de R

Curso: Introducción a la Estadística y Probabilidades CM-274

Lecturas Importantes

1. Notas sobre RStudio y paquetes importantes de R de Paul Hiemstra.
http://stcorp.nl/R_course/.
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. Escribe los programas correspondientes a los siguientes problemas
 - (a) Un palillo se rompe al azar en 3 piezas. Escribe una función en R que, basada en simulación, calcula y devuelve la probabilidad de que las piezas puedan formar un triángulo.
 - (b) Usa la función `curve` para mostrar el gráfico $f(x) = e^{-x^2}/(1+x^2)$ en el intervalo $0 \leq x \leq 10$. Entonces usa la función `integrate` para calcular el valor de la integral

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

- (c) Escribe una función llamada `norma` que calcula la norma Euclidea de un vector numérico. La norma Euclidea de un vector $x = (x_1, \dots, x_n)$ es

$$\|x\| = \sqrt{\sum_{i=1}^n x_i^2}.$$

Usa operaciones vectorizadas para calcular la suma. Prueba esta función sobre los vectores $(0, 0, 0, 1)$ y $(2, 5, 2, 4)$ para verificar que el resultado de la función es correcto.

- (d) Construye una matriz con 10 filas y 2 columnas conteniendo datos aleatorios normalmente estandarizados

```
> x <- matrix(rnorm(20), 10, 2)
```

Esta es una muestra aleatoria de 10 observaciones desde la distribución normal bivariada. Usa la función `apply` y la función `norma` del ejercicio anterior para calcular la norma Euclidea para esas 10 observaciones.

2. Desarrolla los siguientes problemas

- (a) El código produce un gráfico de dispersión

```
> plot.new()
> plot.window(range(pressure$temperature),
+             range(pressure$pressure))
> box()
```

```

> axis(1)
> axis(2)
> points(pressure$temperature, pressure$pressure)
> mtext("temperatura", side=1, line=3)
> mtext("presion", side=2, line=3)
> mtext("Presion de vapor \ncomo una funcion de la Temperatura ",
+       side=3, line=1, font=2)

```

- Describe completamente lo que cada llamada a la función en el código anterior hace, eso incluye una explicación del significado de cada argumento en las llamadas a funciones. Tu respuesta debe incluir una explicación de las diferentes regiones y sistemas de coordenadas creado por este código.
 - Describe cómo podría producir el mismo gráfico usando `viewports`, `layouts`, `units` en el sistema gráfico **grid**. Esta descripción debe incluir una mención de las funciones de `grid` que se requieren y lo que estas funciones hacen.
- (b) Usa la función `curve` para mostrar el gráfico de la función densidad **gamma** con parámetros 1 de forma y 1 de proporción. Usa ahora la función `curve` con el atributo `add = TRUE` para mostrar el gráfico de la densidad de la distribución Gamma, con parámetros de forma k y de proporción 1 para 2, 3, todos en la misma ventana.
- (c) Esta pregunta, es acerca de vectorización y reciclaje en R.
- Define por medio de una función que es vectorización en R.
 - Define por medio de una función que obedece la reglas de `recycling` en R.
 - Considera la función h definida por

$$h(x, y) = \sqrt{x^2 + y^2}$$

Escribe una función en R, llamada `hypot`, con argumentos x e y que implementa una versión de h que es vectorizada y que cumple las reglas del `recycling`.

- (d) Los datos de iris corresponden a las medidas en centímetros de las variables `length`, `width` de los sépalos y `length`, `width` de los pétalos respectivamente de 50 flores cada una de tres especies de iris. Hay cuatro variables numéricas correspondientes al sépalos y pétalos 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`).

3. Resuelve y escribe programa de R de

- (a) Sea X el número de "tres" obtenidos en 10 lanzamientos de un dado. Entonces X tiene una distribución Binomial ($n = 10, 1/6$). Calcula una tabla de probabilidades para $x = 0, 1, \dots, 10$ por dos métodos

- Usando la fórmula para la densidad de probabilidad

$$\mathbb{P}(X = k) = \binom{n}{k} p^k (1 - p)^{n-k}$$

y la aritmética vectorizada en R. Usa una secuencia para los valores de x y la función `choose` para calcular los coeficientes binomial $\binom{n}{k}$.

- Usa la función `dbinom` de R, para comparar ambos métodos.

- (b) Sea X el número de "tres" obtenidos en 10 lanzamientos de un dado. Entonces X tiene una distribución Binomial ($n = 10, 1/6$). Calcula una tabla de probabilidades binomiales acumulativas (`cdf`) para $x = 0, 1, \dots, 10$ por dos métodos

- Usando la función `cumsum` y el resultado anterior.
- Usando la función `pbinom`. Halla el valor de $\mathbb{P}(X > 5)$.

- (c) Usa la función `curve` para mostrar el gráfico de la densidad de $\chi^2(1)$. La función densidad de una distribución chi-cuadrado es `dchisq`.

(d) Supongamos que lanzamos un par de dados 1000 veces.

- 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`.
- Para cada par de lanzamientos, calcular la suma de los lanzamientos, y almacena la suma en la variable `suma-dado`.
- Utilice la función `table` para tabular los valores de la suma de lanzamientos. Calcule las proporciones para cada valor de la suma y compare esas proporciones con las probabilidades exactas de la suma de dos lanzamientos de dados.

4. Analiza y resuelve los siguientes problemas

- (a) La función `rpois` genera observaciones aleatorias desde una distribución de Poisson. Usa esta función para simular valores grandes ($n = 1000, 10000$) para una distribución de Poisson con parámetro $\lambda = 0.61$. Encuentra la distribución de frecuencia, media, varianza para la muestra.
- (b) Explica y corrige los siguientes códigos.

```
> x=seq(0,10,by=.025)
> 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
> 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)
```

- (c) La función `qnorm` retorna los percentiles (cuantiles) de una distribución normal. Usa la función `qnorm` para encontrar los cuantiles de la distribución normal estándar.

(d) Explica el siguiente código acerca de la Ley de los grandes números y la librería ggplot2.

```
> n <- 10000
> media <- cumsum(rnorm(n))/(1:n)
> g <- ggplot(data.frame(x = 1:n, y = media), aes(x = x, y = y))
> g <- g + geom_hline(yintercept = 0) + geom_line(size = 2)
> g <- g + labs(x = "Numero de obs", y = "media cumulativa")
> g
```

