



Bài kiểm tra 1: Lập trình cơ bản trong và biến ngẫu nhiên.

Ngày 28 tháng 03 năm 2023

Thời gian làm bài: 30 phút.

Note:

- Sinh viên làm bài trên **R script**, lưu lại với tên có dạng: `"LTTK_HọTên_MSSV_Test1.R"`. Sau khi hoàn thành bài làm, copy phần code bài làm trong  sang file text `.txt` để backup. Nộp bài cả file **R script** và file text `.txt` (lưu với tên có dạng `"LTTK_HọTên_MSSV_Test1.txt"`).
- trong quá trình làm bài kiểm tra, sinh viên có thể tham khảo tài liệu: giáo trình thực hành và tài liệu "Giới thiệu về R" (đã được giới thiệu là tài liệu tham khảo môn học).
- Dùng lệnh `help(ten_ham)` để biết cú pháp và cách sử dụng một command trong .
- Bài làm cần trình bày như sau:

```
##
## Bai kiem tra 1 - Thuc hanh Ly thuyet Thong ke
## Nhom2 - Thu ... - tiet ....
##
## Ho ten: ..... - MSSV: .....
##
##*****

## Bai 1:

##-----
## Bai 2:

##-----
```

Bài 1 (3đ)

Gọi $\mathcal{M}_{m \times n}$ là tập hợp các ma trận có các phần tử là số thực và có chiều là $m \times n$.

Tạo các ma trận A và B như sau (0,5đ):

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 2 & 2 \\ 3 & 2 & 3 \end{pmatrix} \quad \text{và} \quad B = \begin{pmatrix} 1 & -1 & 3 & 6 & -2 \\ -3 & 5 & 1 & 0 & 2 \\ 7 & 4 & 0 & 1 & 5 \end{pmatrix}.$$

1.1. Viết câu lệnh để (1đ):

- trích ra cột thứ 3 của ma trận B ;
- ở hàng thứ 2 của B , trích ra các phần tử thứ 1, 4, 5 và gán vào 1 vector tên `B.row2.vec`.

1.2. (0,5đ) Tính ma trận tích của A và B (phép nhân giữa 2 ma trận trong đại số tuyến tính).

1.3. (0,5đ) Tìm ma trận chuyển vị của A và B . Sau đó, tính ma trận tích của A^\top và B^\top .

1.4. (0,5đ) Tìm ma trận $X \in \mathcal{M}_{3 \times 1}$ sao cho: $A.X = (1 \ 1 \ 1)^\top$.

Bài 2 (4đ)

2.1.(1đ) Dùng vòng lặp `while` và các câu lệnh điều kiện phù hợp, viết hàm `count.nb.ex2(Nmax.ex2,a,b)` để đếm các số tự nhiên liên tiếp (bắt đầu từ 1 và không vượt quá `Nmax.ex2`) chia hết cho `a` và chia hết cho `b`. Kết quả trả ra là đếm được bao nhiêu số và số tự nhiên cuối cùng đếm được.

Áp dụng cho `N.ex2 = 150`, `a = 3` và `b = 4`.

2.2.(1đ) Tạo các vector `Month = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)` và `Nb = (55,62,24,34,81,29,65,48,75,82,90,66)`.

Sau đó, tạo dataframe tên `mydf` gồm hai cột chính là 2 vectors `Month` và `Nb`.

2.3.(2đ) Load file dataset `data01.xls`.

Với dữ liệu trong file `data01.xls`, sử dụng vòng lặp thích hợp và các lệnh điều kiện nếu cần thiết, hãy viết (các) câu lệnh để đếm số người thuộc các nhóm `group1`, `group2`, `group3` và tạo ra vector `Number <- c(group1, group2, group3)`, trong đó

- `group1` là số người từ 60 tuổi trở xuống,
- `group2` là số người từ 61 đến 80 tuổi,
- `group3` là số người trên 80 tuổi.

Lưu ý, không cần gán vector `Number` vào dataframe.

Bài 3 (2đ)

Cho biến ngẫu nhiên X nhận các giá trị $\{0; 1; 2; 3; \dots; n\}$ (với $n \in \mathbb{N}, n \geq 1$) và có hàm trọng lượng xác suất được xác định như sau, với $k = 0; 1; 2; \dots; n$ và $0 < p < 1$:

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

3.1. (0,5đ) Viết hàm `MassProb.ex3.1(k,n,p)` để tính hàm trọng lượng xác suất của X , với n, k, p tổng quát.

3.2. (0,5đ) Viết hàm `cdf.ex3.1` (sinh viên tự xác định các tham số cần thiết) tính phân phối xác suất của X , tức là ứng với hàm F được xác định bởi, với $x \in \mathbb{R}$:

$$F_X(x) \equiv F(x) := \mathbb{P}(X \leq x).$$

3.3. (1đ) Bây giờ, áp dụng cho $n = 10$ và $p = 0.25$.

- Dùng hàm `cdf.ex3.1` để tính $F_X(5)$ và $F_X(8.5)$.
- Tính kỳ vọng $\mathbb{E}(X)$ và phương sai $\text{Var}(X) := \mathbb{E}([X - \mathbb{E}(X)]^2)$.
- Vẽ đồ thị phù hợp thể hiện hàm trọng lượng xác suất của biến ngẫu nhiên X .
- Vẽ đồ thị phù hợp thể hiện hàm phân phối xác suất F_X của biến ngẫu nhiên X .

Bài 4 (1đ)

4.1. Viết hàm với tên `findmax` (dùng câu lệnh `function`) để tìm giá trị lớn nhất trong các phần tử của một vector v chứa các số thực.

Hàm `findmax` sẽ trả ra một list với tên `mylist.ex4` gồm 2 variables (2 thành phần) là `val` và `val.ind`, trong đó

- `val` là giá trị lớn nhất của vector v ;
- `val.ind` là vị trí của (các) phần tử mang giá trị lớn nhất trong vector v .

Hướng dẫn: để tạo một list trong thư viện của , dùng lệnh như trong ví dụ sau

```
x <- c(1:4)
y <- FALSE
z <- matrix(c(1:4), nrow=2, ncol=2)
myList <- list(x,y,z)
```


Vậy đối với câu hỏi bài 4 này, chẳng hạn, với vector $v = (1, 2.7, 4, -3, 4, 1.5)$, thì hàm `findmax` phải trả ra

```
> mylist
$val
[1] 4

$val.index
[1] 3 5

>
```

4.2. Attach dataset `quakes`, đây là bộ dữ liệu ghi nhận vị trí của 1000 sự kiện địa chấn với cường độ ít nhất là 4.0 độ Richter, ở các địa điểm gần Fiji kể từ năm 1964.

Hướng dẫn: để load dataset có sẵn trong thư viện của , dùng lệnh sau

```
data(quakes)
```

- Sử dụng hàm `findmax` vừa viết để tìm ra số liệu ứng với trận động đất có cường độ Richter mạnh nhất được ghi nhận trong dataset `quakes`.
- Viết (các) câu lệnh xuất ra vị trí (vị trí theo kinh độ và vĩ độ) tương ứng với trận động đất với cường độ Richter mạnh nhất trong dataset `quakes`, và xuất ra kết quả có bao nhiêu trạm đã báo cáo về trận động đất này.

- - - Good luck! - - -