ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH

CẤU TRÚC RỜI RẠC CHO KHMT (CO1007)

Ứng dụng thống kê
khảo sát kết quả của bài tập online cho phép nộp bài nhiều lần

GVHD: Huỳnh Tường Nguyên
Trần Tuấn Anh
Nguyễn Ngọc Lễ
Nhóm: 25
SV thực hiện: Tô Hòa – 1910198
Trương Vĩnh Phước – 1910473
Nguyễn Huỳnh Đức – 1910137
Nguyễn Hoàng Trung – 1910644
Ngô Lê Quốc Dũng – 1910101
Lại Đức Anh Khoa – 1910265

Tp. Hồ Chí Minh, Tháng 06/2020

Mục lục

1 Động cơ nghiên cứu

Trong mùa dịch Covid-19, trường Đại học Bách Khoa, ĐHQG-HCM đã triển khai giảng dạy trực tuyến và yêu cầu sinh viên thực hiện các bài tập nhỏ để thu nhận phản hồi về việc học tập và hiểu biết của các bạn thông qua các tài nguyên online được cung cấp.

Phân tích & thống kê dữ liệu qua các lần nộp bài của sinh viên không những giúp giáo viên có những hướng đúng trong việc phát hiện ra những kiến thức mà sinh viên chưa chắc chắn, cũng như có hướng để cải thiện bổ sung phần học liệu trong tương lai để phù hợp với hơn người học.

2 Mục tiêu

Khai phá dữ liệu từ hệ thống nộp bài online có ý nghĩa quan trọng trong việc đánh giá chất lượng của sinh viên. Ngoài ra, những đánh giá kết quả nộp bài của từng sinh viên, hay từng bài tập sẽ góp phần xác định những điểm mạnh, điểm yếu của sinh viên để giáo viên có phương pháp phù hợp trong việc cải thiện kỹ năng của sinh viên.

Trong bài tập lớn này, các sinh viên sẽ bắt đầu với các bài toán thống kê đơn giản từ những dữ liệu được cung cấp. Qua đó, các em sẽ tìm ra những con số thú vị, có ý nghĩa đối với các dữ liệu thực tế trong quá khứ của hệ thống chấm bài online. Những kết quả mà các em tìm ra sẽ là bước khởi đầu cho việc khai phá nguồn dữ liệu của hệ thống sau này, nhằm đạt tới mục tiêu nâng cao kỹ năng lập trình, kỹ năng giải quyết vấn đề cho người học cũng như hướng tới mục tiêu cao hơn khi tích hợp với các hệ thống quản lý và cải thiện chất lượng dạy và học.

3 Mô tả dữ liệu

Đính kèm đề bài tập lớn là 24 files **filename.xlsx** ("CO1007_TV_HK192-Quiz...xlsx") trong đó chứa thông tin về điểm qua các lần nộp các bài Quiz của các sinh viên trên BKEL. Thông tin bài tập lớn:

1. *tid* là mã số bài tập (gồm có 24 file dữ liệu nên mã bài tập là các số từ 1 đến 24 thay cho tên file)
2. Mã số ID ta gọi là *uid* là mã số định danh sinh viên nộp bài, mỗi sinh viên có một *mã số id duy nhất và không trùng với một mã số id của các sinh viên khác*
3. *Tình trạng: Đã hoàn thành hoặc chưa bao giờ gởi*
4. *Đã bắt đầu vào lúc, Đã hoàn thành: Thời gian theo dạng "d B Y I:M p" là thời gian bắt đầu và kết thúc làm bài. Trong đó, "e" là ngày (1..31) "B" tên tháng đầy đủ, "Y" là năm (0..9999),"I" là giờ (01..12), "M" là phút (00–59), "p" chỉ định AM/PM*
5. *Thời gian thực hiện: Khoảng thời gian làm bài*
6. *Điểm/10: Tổng số điểm của các quiz cộng lại thấp nhất là 0, tối đa là 10*
7. *Q.i/1 là điểm số của bài quiz chỉ 0 hoặc 1.*

4 Nhiệm vụ

4.1 Thông tin chung

Nhóm: 25

Mã đề: $MD = 2907$

Bài tập cần làm: **1, 2, 3, 4, 5, 7, 9**

Bài tập làm thêm: **12**

Các file cần xử lý:

- **File 1:** CO1007_TV_HK192-Quiz 1.4-điểm.xlsx
- **File 2:** CO1007_TV_HK192-Quiz 1.5-điểm.xlsx

- **File 3:** CO1007_TV_HK192-Quiz 3.3-điểm.xlsx
- **File 4:** CO1007_TV_HK192-Quiz 4.2-điểm.xlsx

4.2 Đọc dữ liệu

Dữ liệu được đọc từ các file excel được lưu trữ trong một dataframe có tên là *data*. Dữ liệu trong *data* bào gồm:

- Cột ID lưu trữ tất cả Mã số ID của các lần nộp bài.
- Cột Status lưu trữ Tình trạng của các lần nộp bài bao gồm: *Done* đại diện cho *Đã hoàn thành* và *Not done* đại diện cho *Chưa bào giờ gửi*.
- Data frame Start gồm 5 cột: Start.day, Start.month, Start.year, Start.hour và Start.minute lưu trữ thời gian (ngày, tháng, năm, giờ và phút) của Đã bắt đầu vào lúc (thời gian thực hiện bài thi).
- Data frame Finish gồm 5 cột: Finish.day, Finish.month, Finish.year, Finish.hour và Finish.minute lưu trữ thời gian (ngày, tháng, năm, giờ và phút) của Đã hoàn thành (thời gian nộp bài).
- Cột Duration lưu trữ Thời gian thực hiên theo đơn vị giây.
- Cột Total lưu trữ Điểm/10 (Tổng điểm của các Quiz).
- 10 cột Q1 đến Q10 lưu trữ Q.i/1 (điểm số của bài Quiz).

4.3 Xử lý dữ liệu

Bài 1: Xác định số lượng sinh viên trong tập mẫu

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Mỗi sinh viên có một mã số sinh viên (MSSV) riêng, do đó số lượng sinh viên sẽ bằng số lượng phần từ của tập hợp các mã số sinh viên.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *unique()* để lấy tập giá trị của tập mã số sinh viên và sử dụng hàm *length()* để lấy số lượng tập giá trị đó.

```
student_num <- length(unique(data$ID))
```

- Kết quả:
  - Số lượng sinh viên ứng với mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	344 sinh viên
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	343 sinh viên
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	280 sinh viên
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	260 sinh viên

Bài 2: Nhóm câu hỏi liên quan đến điểm số của các sinh viên

a) Xác định điểm số là điểm tổng của các bài làm với mỗi câu hỏi đơn vị đều có điểm tối đa là 1 điểm.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta tính tổng tất cả các giá trị tổng điểm của các bài làm trong tập dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *sum()* để tính tổng điểm của tất cả các bài làm.
- Kết quả:
  - Điểm tổng của tất cả các bài làm của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	5512.25
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	5668.5
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	3410
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	4406

b) Xác định điểm số thấp nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách các bài nộp, ta chọn ra giá trị điểm số thấp nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta dùng hàm *min()* để tìm giá trị nhỏ nhất của 1 vector. Ở đây, *K* là một data frame đã lọc ra các dữ liệu thừa hay lỗi.

```
Least.Total <- min(K[,6])
```

- Kết quả:
  - Điểm số thấp nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	4.5 điểm
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	0.5 điểm
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	0 điểm
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	0 điểm

c) Xác định danh sách các sinh viên có ít nhất một bài có số điểm thấp nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta đã biết được số điểm thấp nhất qua câu b, nên ta có thể lập danh sách các sinh viên có ít nhất một bài toán có số điểm thấp nhất dựa vào giá trị vừa tìm được.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Dùng hàm subset để trích ra một data frame mới với yêu cầu là có số điểm Total bằng số điểm thấp nhất.

```
List.Least.Total <- subset(K, K$Total == Least.Total)
```

  - Tuy nhiên nếu chỉ lọc như thế này có khả năng một sinh viên có thể xuất hiện nhiều hơn một lần trong danh sách này, nên ta dùng thêm hai hàm *match()* và *unique()*.

```
List.Least.Total.Unique <- List.Least.Total[match(unique(
List.Least.Total$ID), List.Least.Total$ID),]
```

- Kết quả:
  - Danh sách các sinh viên có ít nhất một bài có điểm số thấp nhất của mỗi file:

```
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"   1913315
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"   1915775
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"   1914661
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"   1914661
```

d) Xác định phổ theo số lần nộp bài của các sinh viên có ít nhất một bài có số điểm thấp nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách các bạn sinh viên có ít nhất một bài có số điểm thấp nhất và danh sách ban đầu, ta tạo được một danh sách mới với các bạn sinh viên có it nhất một bài có số điểm thấp nhất và số lần làm bài của các bạn ấy.

Hiện thực trên R

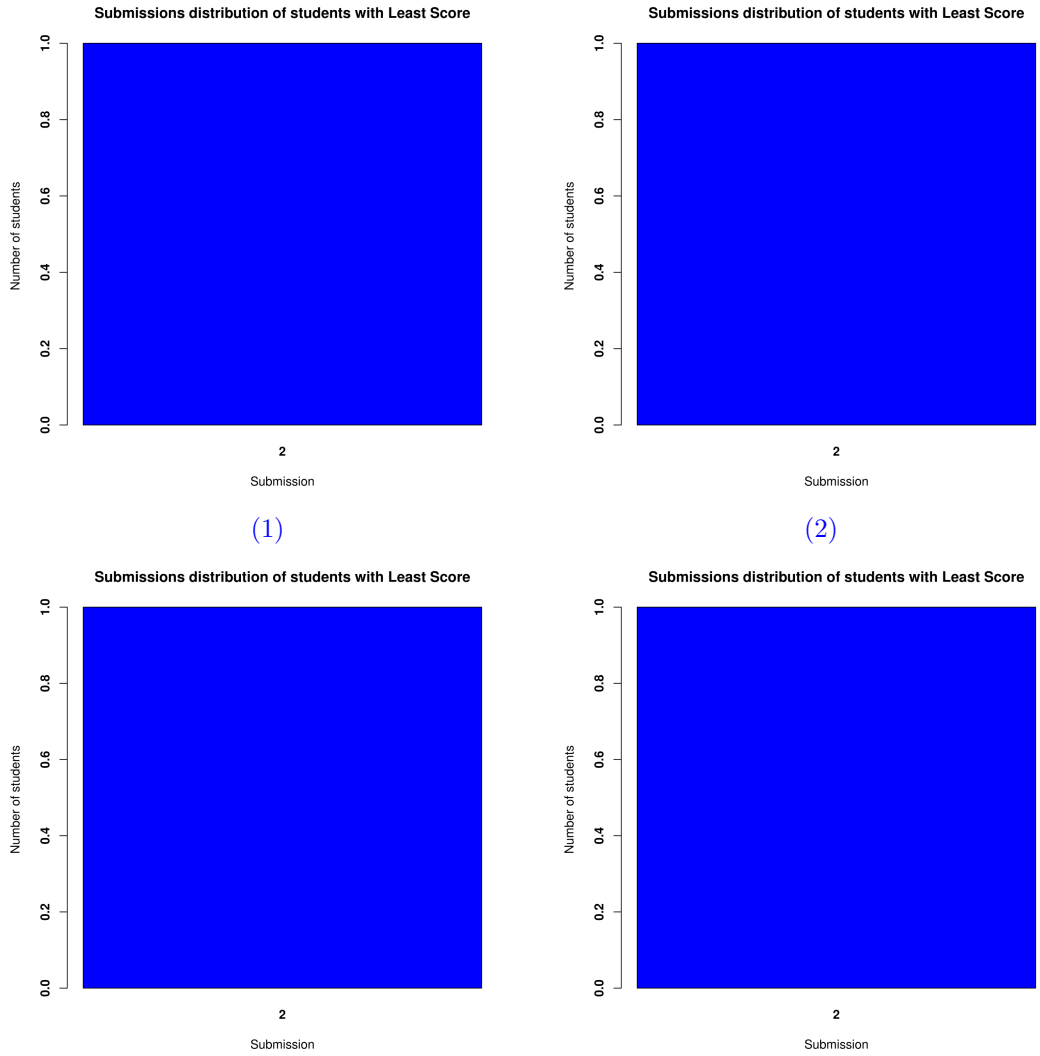
- Ý tưởng thực hiện:
  - Ta tạo một dataframe mới (gọi tạm là *List.Least.Total2*) là một subset của dataframe ban đầu kèm xét thêm điều kiện là ID phải tồn tại trong dataframe *List.Least.Total.Unique* (tức là các sinh viên có ít nhất một bài có số điểm thấp nhất).

    ```
    List.Least.Total2 <- subset(K, ID %in% List.Least.Total.Unique$ID)
    ```

  - Sau đó tạo thêm dataframe List.Least.Total.Freq để tính tần số xuất hiện (cũng là số lần làm bài) của các bạn sinh viên.

    ```
    List.Least.Total.Freq <- data.frame(table(List.Least.Total2$ID))
    ```

  - Cuối cùng ta sử dụng hàm *barplot()* để vẽ phổ theo số lần nộp bài.
- Biểu đồ:



Hình 2.1: Phổ theo số lần nộp bài của các sinh viên có ít nhất một bài có số điểm thấp nhất
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

e) Xác định điểm số tổng kết thấp nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta lập danh sách các sinh viên với điểm số tổng kết của mỗi sinh viên, sau đó chọn ra số điểm tổng kết thấp nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sắp xếp dữ liệu theo cột điểm từ cao đến thấp, và sử dụng hàm *match()* và *unique()* để giữ lại giá trị đầu tiên tính từ trên xuống (cũng là điểm tổng kết của các bạn).
  - Bước đầu tạo một dataframe *K.Descend* được sắp thứ tự từ cao đến thấp của dữ liệu ban đầu.

```
K.Descend <- K[order(-K$Total),]
```

- Tạo tiếp một dataframe *K.Descend.Unique* để lọc ra những lần làm bài có điểm cao nhất của mỗi bạn sinh viên từ *K.Descend*.

```
K.Descend.Unique <- K.Descend[match(unique(K.Descend$ID),
K.Descend$ID),]
```

Vậy điểm tổng kết nhỏ nhất sẽ là điểm có giá trị nhỏ nhất trong *K.Descend.Unique*.

```
Least.Final.Total <- min(K.Desecend.Unique$Total)
```

- Kết quả:
  - Điểm số tổng kết thấp nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	8 điểm
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	7 điểm
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	8 điểm
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	7 điểm

f) Xác định danh sách các sinh viên có điểm số tổng kết thấp nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Dựa vào điểm số tổng kết nhỏ nhất tính được ở câu *e* và bảng dữ liệu, ta lập danh sách sinh viên có điểm tổng kết thấp nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Lưu ý rằng *K.Descend.Unique* là danh sách điểm tổng kết của các bạn sinh viên.

```
List.Of.Final.Total <- K.Descend.Unique
```

  - Vậy *List.Of.Final.Total* là danh sách điểm tổng kết. Như vậy ta có thể tạo một subset là danh sách các bạn sinh viên từ dataframe *List.Of.Final.Total* và điểm tổng kết thấp nhất (*Least.Final.Total*) từ câu trên.

```
List.Least.Final.Total <- subset(List.Of.Final.Total, Total ==
Least.Final.Total)
```

  - *List.Least.Final.Total* là danh sách các sinh viên có điểm số tổng kết thấp nhất cần tìm.
- Kết quả:
  - Danh sách sinh viên có điểm tổng kết thấp nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1913334	1915928	1915275	1914661
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1812478	1812257	1915275	1914079
	1911975			
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1912523	1913218	1912966	
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1914210			

g) Xác định phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết thấp nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách tìm được ở câu *f* và danh sách ban đầu tìm được số lần nộp bài của các sinh viên, ta vẽ phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết thấp nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng các hàm *subset*() và *table*() để lấy danh sách các lượt làm bài của các bạn có điểm tổng kết nhỏ nhất.

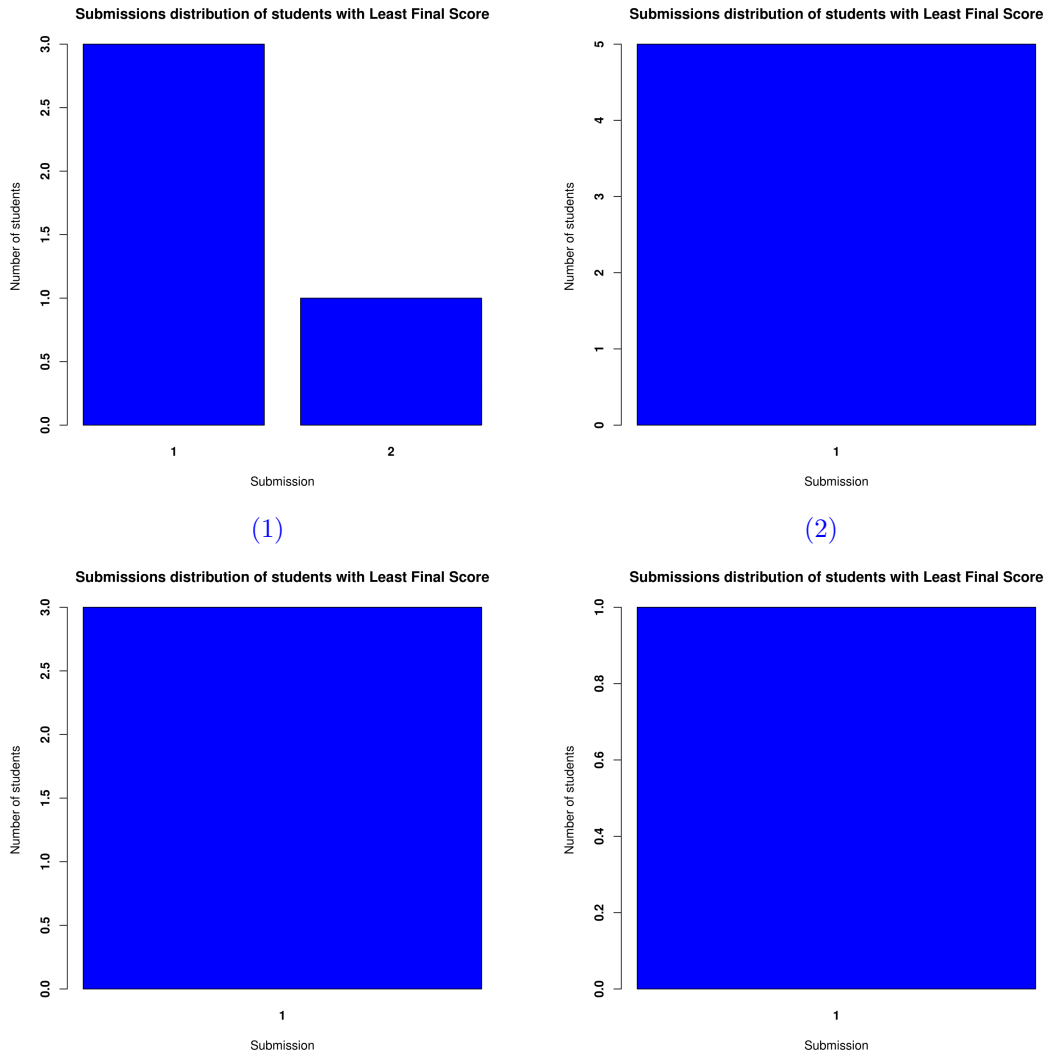
```
List.Least.Final.Total2 <- subset(K, ID %in% List.Least.Final.Total$ID)
```

– *List.Least.Final.Total2* là danh sách các lượt làm bài của các bạn có điểm tổng kết nhỏ nhất.

```
List.Least.Final.Total.Freq <- data.frame(table(List.Least.Final.Total2$ID))
```

– *List.Least.Final.Total.Freq* là danh sách số lần nộp bài của sinh viên có điểm số tổng kết thấp nhất. Ta dùng hàm *barplot()* để vẽ phổ số lần nộp bài của nhóm sinh viên trên.

- Biểu đồ:



(1) (2)

(3) (4)

Hình 2.2: Phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết thấp nhất

(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

h) Xác định điểm số cao nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Dựa vào danh sách dữ liệu điểm, ta tìm được điểm số cao nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta dùng hàm *max()* để tìm giá trị lớn nhất của một vector.
    ```
    Highest.Total <- max(K[,6])
    ```
- Kết quả:
  - Điểm số cao nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	10 điểm
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	10 điểm
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	10 điểm
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	10 điểm

i) Xác định danh sách các sinh viên có tối thiểu một bài nộp có số điểm số cao nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Dựa vào giá trị tìm được ở câu *h*, ta lập danh sách những sinh viên có ít nhất một bài nộp có số điểm cao nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *subset()* để lọc ra các danh sách các bài nộp có điểm số bằng số điểm cao nhất. Sau đó, ta sử dụng hàm *unique()* để lấy danh sách sinh viên từ dữ liệu vừa lọc ra được.
    ```
    List.Highest.Total <- subset(K, K$Total == Highest.Total)
    List.Highest.Total.Unique <- List.Highest.Total[match(unique(
    List.Highest.Total$ID), List.Highest.Total$ID), ]
    ```
- Kết quả:
  - Danh sách các sinh viên có ít nhất một bài có điểm số cao nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1915562	1913355	1914038	1913464
	1913186	1915919	1912041	1911591
	1910916	1914845	1915329	1911704
	1910666	1910351	1913467	1915323
	1914055	1915268	1914864	1910032
	...			
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1913094	1914807	1914352	1913844
	1913464	1915323	1911478	1913186
	1915822	1913014	1937019	1911591
	1911704	1914845	1913336	1910202
	1913355	1915540	1911186	1910101
	...			
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1914720	1911591	1913566	1912817
	1915482	1913775	1913355	1915329
	1911704	1910666	1913186	1914845
	1915541	1914474	1911136	1915473
	1911837	1912980	1914003	1911881
	...			
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1911881	1913355	1913186	1913014
	1915482	1910666	1911591	1911704
	1913241	1911314	1914845	1913123
	1914003	1911136	1913075	1912817
	1915541	1914720	1912811	1913396
	...			

j) Xác định phổ theo số lần nộp bài của các sinh viên có tối thiểu một bài nộp có điểm số cao nhất

Kiến thức chuẩn bị

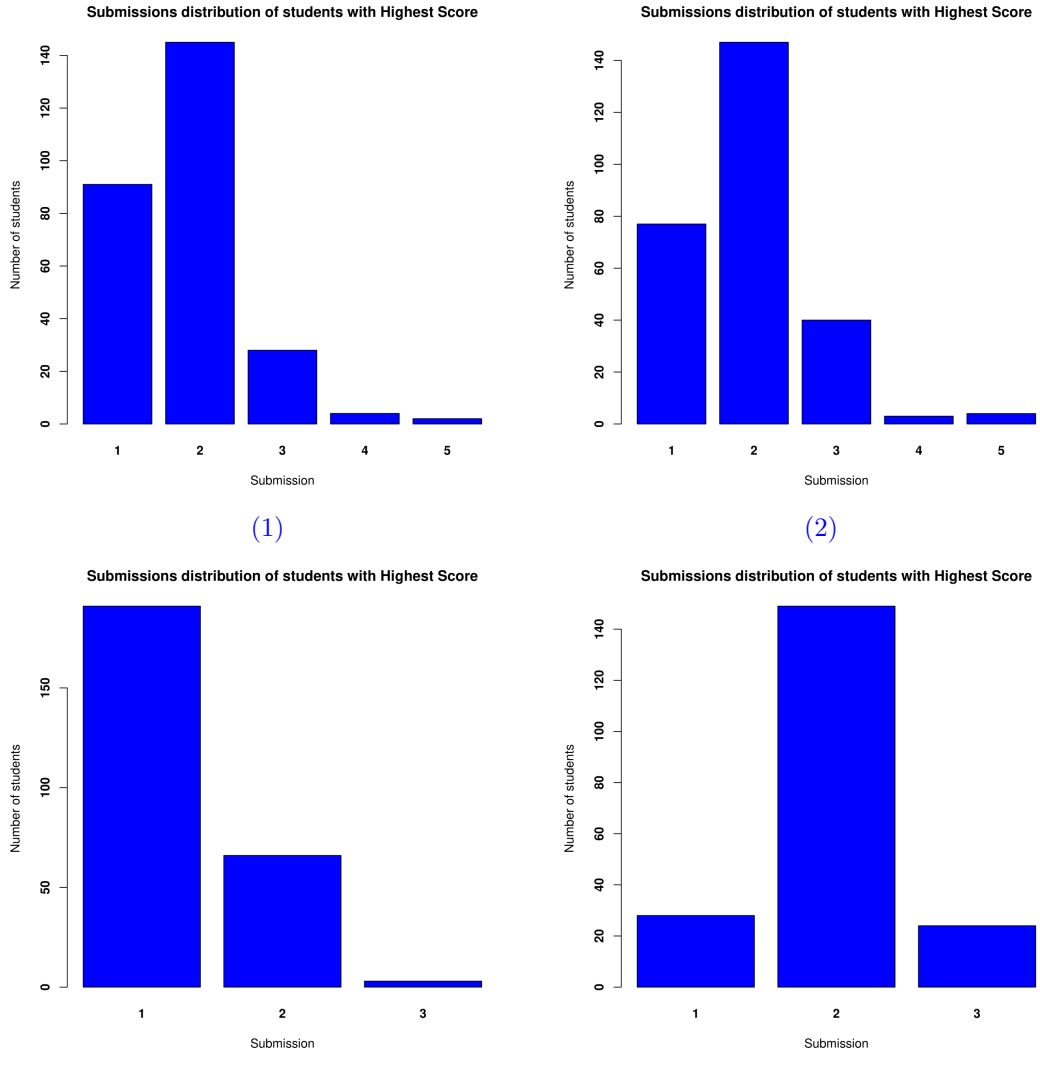
- Cách giải truyền thống:
  - Từ danh sách các bạn sinh viên có ít nhất một bài có số điểm cao nhất và danh sách ban đầu, ta tạo được một danh sách mới với các bạn sinh viên có ít nhất một bài có điểm số cao thấp và số lần làm bài của mỗi sinh viên, sau đó vẽ phổ theo số lần làm bài dựa trên tập dữ liệu vừa thu được.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *subset()* để lọc ra từ tập giá trị ban đầu những sinh viên có ít nhất một bài đạt điểm cao nhất, sau đó sử dụng *table()* để lập danh sách số lần nộp bài của mỗi bạn.

```
List.Highest.Total2 <-subset(K, ID %in% List.Highest.Total.Unique$ID)
List.Highest.Total.Freq <- data.frame(table(List.Highest.Total2$ID))
```

  - Cuối cùng, ta sử dụng hàm *barplot()* để vẽ phổ theo số lần nộp bài của nhóm sinh viên trên.
- Biểu đồ:



Hình 2.3: Phổ theo số lần nộp bài của các sinh viên có ít nhất một bài có số điểm cao nhất
(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"

k) Xác định điểm số tổng kết cao nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta lập danh sách điểm tổng kết của từng sinh viên, sau đó chọn ra điểm tổng kết coa nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Điểm tổng kết lớn nhất sẽ là điểm có giá trị lớn nhất trong *List.Of.Final.Total* đã được tính ở câu *f*.

    ```
    Highest.Final.Total <- max(List.Of.Final.Total$Total)
    ```
- Kết quả:
  - Điểm số tổng kết cao nhất của mỗi file:

    "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx" 10 điểm
    "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx" 10 điểm
    "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx" 10 điểm
    "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx" 10 điểm

l) Xác định danh sách các sinh viên có điểm số tổng kết cao nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Dựa vào điểm tổng kết cao nhất đã được xác định, ta lập danh sách những sinh viên có điểm tổng kết bằng điểm tổng két cao nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta tạo một subset là danh sách các bạn sinh viên từ dataframe *List.Of.Final.Total* và điểm tổng kết cao nhất (*Highest.Final.Total*) từ câu *k*.

    ```
    List.Highest.Final.Total <- subset(List.Of.Final.Total, Total ==
    Highest.Final.Total)
    ```
  - *List.Highest.Final.Total* là danh sách các sinh viên có điểm số tổng kết cao nhất cần tìm.
- Kết quả:
  - Danh sách các sinh viên có điểm số tổng kết cao nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1915562	1913355	1914038	1913464
	1913186	1915919	1912041	1911591
	1910916	1914845	1915329	1911704
	1910666	1910351	1913467	1915323
	1914055	1915268	1914864	1910032
	...			
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1913094	1914807	1914352	1913844
	1913464	1915323	1911478	1913186
	1915822	1913014	1937019	1911591
	1911704	1914845	1913336	1910202
	1913355	1915540	1911186	1910101
	...			
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1914720	1911591	1913566	1912817
	1915482	1913775	1913355	1915329
	1911704	1910666	1913186	1914845
	1915541	1914474	1911136	1915473
	1911837	1912980	1914003	1911881
	...			
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1911881	1913355	1913186	1913014
	1915482	1910666	1911591	1911704
	1913241	1911314	1914845	1913123
	1914003	1911136	1913075	1912817
	1915541	1914720	1912811	1913396
	...			

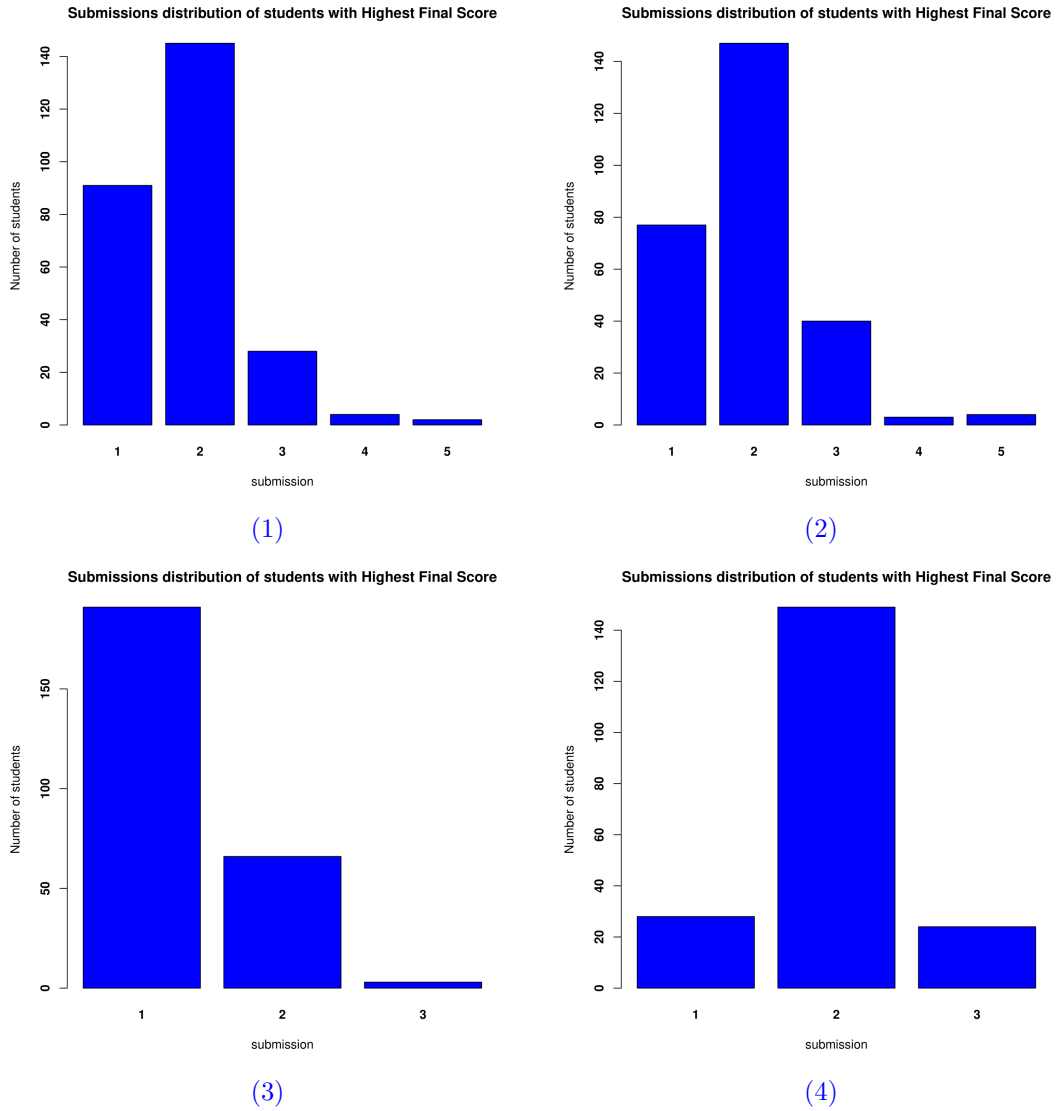
m) Xác định phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết cao nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách các sinh viên tìm được ở câu *l* và danh sách ban đầu, ta tìm được số lần nộp bài của các sinh viên theo yêu cầu bài toán. Sau đó, ta vẽ phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết cao nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng các hàm *subset()* và *table()* để lập danh sách số lần nộp bài của sinh viên có điểm số tổng kết cao nhất.
    ```
    List.Highest.Final.Total2 <-subset(K, ID %in% List.Highest.Final.Total$ID)
    ```
  - *List.Highest.Final.Total2* là danh sách các lượt làm bài của các bạn có điểm tổng kết cao nhất.
    ```
    List.Highest.Final.Total.Freq <- data.frame(table(
    List.Highest.Final.Total2$ID))
    ```
  - *List.Highest.Final.Total.Freq* là danh sách số lần nộp bài của sinh viên có điểm số tổng kết cao nhất. Dựa vào dữ liệu vừa thu được, ta dùng hàm *barplot()* để vẽ phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết cao nhất.
- Biểu đồ:



Hình 2.4: Phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết cao nhất
(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"

n) Xác định điểm số trung bình của của các sinh viên trong mẫu

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Điểm số trung bình được tính bằng tổng điểm tổng kết chia cho số lượng sinh viên trong tập.

$$\overline{x} = \frac{\sum_{i=1}^{k} x_i}{k}$$

trong đó: x_i là điểm tổng kết của sinh viên thứ i, k là số lượng sinh viên.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *mean()* để tính giá trị trung bình của điểm tổng kết.

```
K.mean <- mean(List.Of.Final.Total$Total)
```

- Kết quả:
  - Điểm số trung bình của các sinh viên trong mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	9.8
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	9.8
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	9.9
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	9.8

o) Xác định số lượng sinh viên có điểm số trung bình

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Dựa vào danh sách điểm tổng kết đã tính ở trên và điểm trung bình ở câu n, ta lập danh sách những bạn có điểm tổng kết bằng điểm trung bình, sau đó đếm số lượng sinh viên trong danh sách này.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Dùng hàm *subset()* tạo một danh sách *List.mean* là danh sách con của danh sách tổng kết với điều kiện là điểm tổng kết bằng với điểm trung bình.

```
List.mean <- subset(List.Of.Final.Total, Total == K.mean)
```

  - Sau đó, ta sử dụng hàm *length()* để lấy số lượng sinh viên thỏa mãn.
- Kết quả:
  - Số lượng sinh viên có điểm tổng kết bằng số điểm trung bình trong mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	0 sinh viên
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	0 sinh viên
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	0 sinh viên
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	0 sinh viên

p) Tính trung vị mẫu, cực đại mẫu, cực tiểu mẫu của trên.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Trung vị: Là một số tách giữa nửa lớn hơn và nữa bé hơn của một mẫu, một quần thể, nhưng ở đây ta nói đến là một tập hợp các giá trị là điểm.
  - Cực đại và cực tiểu: Là thành phần lớn nhất (hoặc cùng lớn nhất), nhỏ nhất (hoặc cùng nhỏ nhất) của một tập hợp các giá trị.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng các hàm *median()*, *min()*, *max()* để tìm các giá trị tương ứng.

```
print(median(K$Total))
print(min(K$Total))
print(max(K$Total))
```

- Kết quả:
  - Trung vị mẫu, cực đại mẫu và cực tiểu mẫu trong mỗi file:

	Trung vị	Cực đại	Cực tiểu
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	9.67	10	4.5
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	9.5	10	0.5
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	10	10	0
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	9.5	10	0

q) Hãy đo mức độ phân tán của điểm số (xung quanh giá trị trung bình) của mẫu

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Độ phân tán được xác định bằng phương sai - độ lệch chuẩn của mẫu.
  - Phương sai được tính bằng công thức:

$$s^2 = \frac{1}{n}\sum_{i=1}^{k} n_i x_i^2 - \overline{x}^2$$

trong đó: x_i là điểm số có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số bài làm
$\overline{x}$ là giá trị điểm trung bình

  - Độ lệch chuẩn được tính bằng công thức:

$$s = \sqrt{s^2}$$

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm $var()$ và $sd()$ để tính các giá trị tương ứng của mẫu.

```
print(var(K$Total))
print(sd(K$Total))
```

- Kết quả:
  - Phương sai, độ lệch chuẩn của mẫu trong mỗi file:

	Phương sai	Độ lệch chuẩn
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	0.8145266	0.9025113
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1.238846	1.113034
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	0.6682011	0.8174357
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1.874935	1.369283

.

r) Tính độ méo lệch (skewness), và độ nhọn (kurtosis) của dữ liệu trong mẫu trên.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Độ méo lệch là sự biến dạng sự bất đối xứng trong một phân phối hình chuông đối xứng hay phân phối chuẩn trong một tập dữ liệu, được tính bằng công thức:

$$\tilde{\mu}_3 = \frac{1}{n}\sum_{i=1}^{k} n_i \left(\frac{x_i - \overline{x}}{s}\right)^3$$

trong đó: x_i là điểm số có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số bài làm
$\overline{x}$ là giá trị điểm trung bình
s là độ lệch chuẩn

  - Độ nhọn là một đại lượng thống kê được sử dụng để miêu tả các phân phối, mô tả hình dạng của đuôi phân phối đó, tính bằng công thức.

$$\tilde{\mu}_4 = \frac{1}{n}\sum_{i=1}^{k} n_i \left(\frac{x_i - \overline{x}}{s}\right)^4$$

trong đó: x_i là điểm số có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số bài làm
$\overline{x}$ là giá trị điểm trung bình
s là độ lệch chuẩn

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *skewness*() và hàm *kurtosis*() để tính các giá trị tương ứng.
    ```
    print(skewness(List.Of.Final.Total$Total))
    print(kurtosis(List.Of.Final.Total$Total))
    ```
- Kết quả:
  - Độ méo lệch, độ nhọn của mẫu trong mỗi file:

	Độ méo lệch	Độ nhọn
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	-1.507457	5.730385
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	-2.62223	15.32753
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	-6.303646	64.30899
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	-1.601421	7.084802

s) Tính tứ phân vị (quartile) thứ nhất (Q_1) và thứ ba (Q_3) của mẫu.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Tứ phân vị thứ nhất Q_1: bằng trung vị phần dưới của một tập.
  - Tứ phân vị thứ hai Q_3: bằng trung vị phần trên của một tập.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *quantile*() với các tham số truyền vào là 0.25 và 0.75 tương ứng.
    ```
    print(quantile(K$Total, 0.25))
    print(quantile(K$Total, 0.75))
    ```
- Kết quả:
  - Tứ phân vị thứ nhất và thứ ba trong mỗi file:

	Thứ nhất Q_1	Thứ ba Q_3
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	9	10
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	9	10
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	9	10
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	8	10

t) Xác định số lượng sinh viên có điểm số nằm trong 2 mức điểm cao nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách điểm tổng kết, ta loại bỏ những bạn có số điểm cao nhất, sau đó tiến hành tìm những bạn có số điểm cao nhất trong danh sách này. Những bạn này chính là những bạn có số điểm cao thứ hai.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *max*() kèm theo điều kiện điểm số khác điểm cao nhất để xác định mức điểm cao thứ hai.
    ```
    Second.Highest.Final.Total <- max(List.Of.Final.Total$Total
    [List.Of.Final.Total$Total != Highest.Final.Total])
    ```
  - *Second.Highest.Final.Total* ở đây là điểm cao thứ hai trong danh sách.
  - Ta tiếp tục sử dụng hàm *subset*() để lập danh sách các bạn có điểm số lớn hơn hoặc bằng giá trị vừa tìm được.

```
List.2Highest.Final.Total <- subset(List.Of.Final.Total, Total ==
Highest.Final.Total | Total == Second.Highest.Final.Total)
```

  - *List.2Highest.Final.Total* là danh sách những bạn có điểm trong hai mức điểm cao nhất.
  - Sử dụng hàm *length()*, ta xác định được số lượng sinh viên trong nhóm này.
- Kết quả:
  - Số lượng sinh viên có điểm số nằm trong 2 mức điểm cao nhất trong mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	287 sinh viên
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	297 sinh viên
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	277 sinh viên
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	222 sinh viên

u) Xác định phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết ở 2 mức điểm cao nhất

Kiến thức chuẩn bị

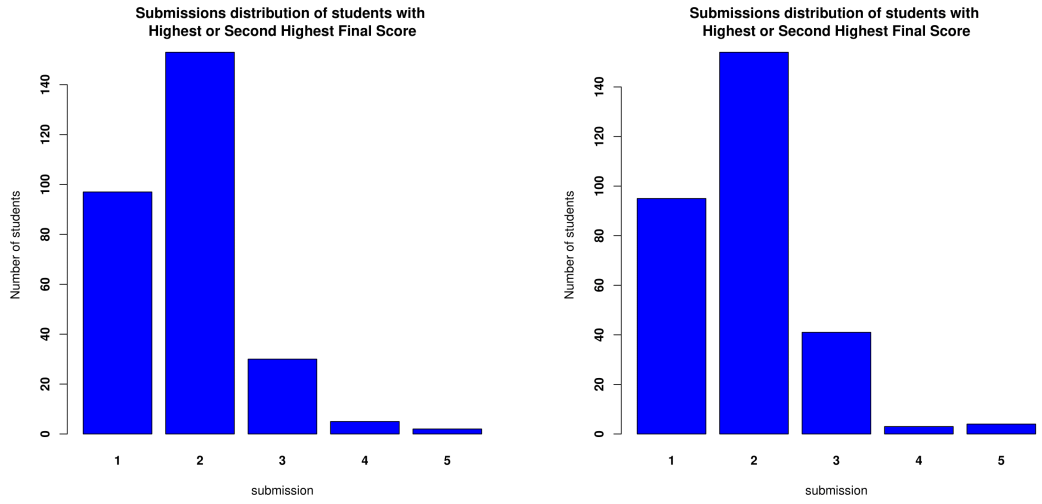
- Cách giải truyền thống:
  - Từ danh sách điểm tổng kết, kết hợp với hai giá trị là điểm cao nhất và điểm cao thứ hai, ta lọc ra những bạn thỏa điều kiện và vẽ phổ theo số lần nộp bài dựa trên dữ liệu vừa lọc được.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *subset()* để tạo danh sách mới thỏa mãn điểm số bằng điểm cao nhất hoặc coa thứ hai, sau đó sử dụng hàm *table()* để lập danh sách số lần nộp bài ứng với mỗi sinh viên.

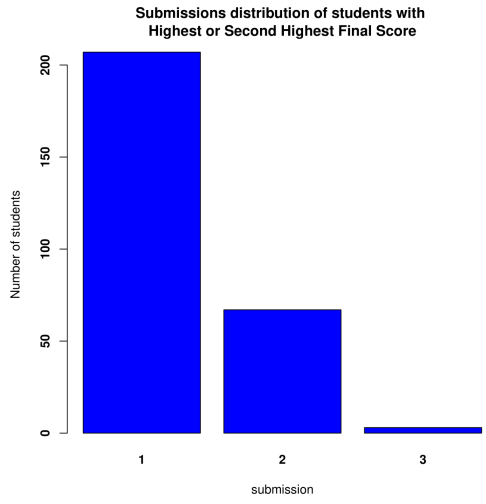
```
List.2Highest.Final.Total2 <-subset(K, ID %in%
List.2Highest.Final.Total$ID)
List.2Highest.Final.Total.Freq <- data.frame(table(
List.2Highest.Final.Total2$ID))
```

  - Sau đó ta dùng hàm *barplot()* để vẽ phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết ở 2 mức điểm cao nhất.
- Biểu đồ:

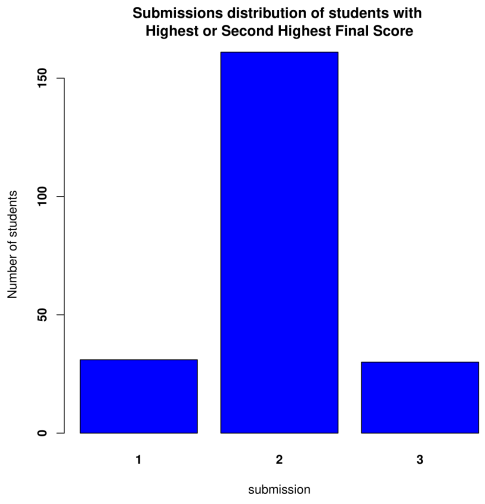


(1)

(2)



(3)



(4)

Hình 2.5: Phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết ở 2 mức điểm cao nhất
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

v) Xác định số lượng sinh viên có điểm số tổng kết ở mức điểm cao thứ k với k cho trước

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ dữ liệu ban đầu, ta lập dãy điểm tổng kết xếp từ cao xuống thấp, lấy giá trị thứ k của dãy ta vừa lập. Dựa vào giá trị này, ta lập danh sách các sinh viên có điểm tổng kết bằng giá trị này, sau đó tính số lượng các sinh viên thỏa mãn.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng tổ hợp hai hàm *unique*() và *order*() để lập dãy điểm tổng từ dữ liệu ban đầu, sau đó lấy phần tử thứ k từ dãy vừa lập.

```
Total.Descending <- unique(List.Of.Final.Total[order(
-List.Of.Final.Total$Total), ]$Total)
Certain.Total <- Total.Descending[k]
```

- Sau đó, ta sử dụng hàm *subset*() để lọc ra các sinh viên thỏa mãn điểm tổng kết bằng mức điểm vừa tìm được và sử dụng hàm *nrow*() để tính số lượng tương ứng.

- Kết quả:
  - Giả sử $k = 2$
  - Số lượng sinh viên có điểm số nằm trong mức điểm thứ k trong mỗi file:

`"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`	17 sinh viên
`"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`	26 sinh viên
`"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`	17 sinh viên
`"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`	21 sinh viên

w) Xác định phổ theo số lần nộp bài của các sinh viên có điểm số tổng kết ở mức điểm cao thứ k với k cho trước

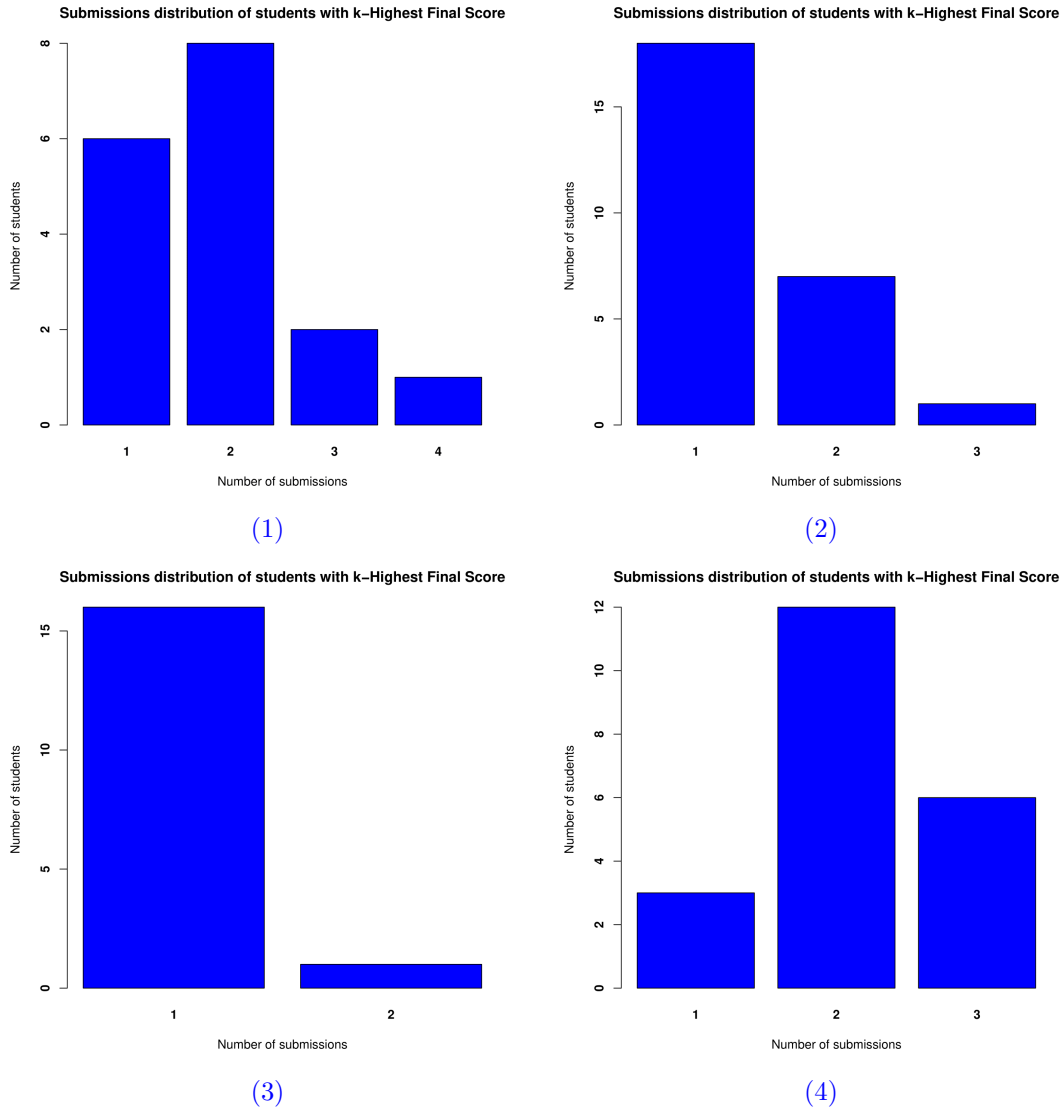
Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ dữ liệu ban đầu, ta lập danh sách toàn bộ các lần nộp của các sinh viên thỏa mãn điều kiện điểm tổng kết bằng mức điểm vừa tìm được, sau đó lập bảng các sinh viên đó kèm theo số lần nộp bài. Từ đó, ta vẽ được phổ theo số lần nộp bài của nhóm sinh viên này.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng tổ hợp hai hàm *unique*() và *order*() để lập danh sách toàn bộ các lần nộp của các sinh viên thỏa mãn điều kiện điểm tổng kết bằng mức điểm vừa tìm được. Sau đó, ta sử dụng hàm *table*() để tạo một dataframe mới gồm các sinh viên thỏa mãn cùng số lần mỗi sinh viên nộp bài.
  - Cuối cùng, ta sử dụng hàm *barplot*() để vẽ được phổ theo số lần nộp bài của nhóm sinh viên này.

- Biểu đồ:
  - Giả sử $k = 2$



Hình 2.7: Phổ số lần nộp bài của các sinh viên với điểm tổng kết ở mức điểm cao thứ k

(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

Bài 3: Nhóm câu hỏi liên quan đến số lần nộp bài

a) Xác định số lần nộp bài ít nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta đếm số lần lặp lại của các ID để biết được số lần nộp bài tương ứng với những Mã số ID đó rồi lập thành danh sách. Từ đó ta thấy được số lần nộp bài ít nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Trước hết ta cần xử lí dữ liệu, loại bỏ các sinh viên có Mã số ID không xác định và các sinh viên chưa làm bài quiz. Dùng lệnh *subset()* để tạo ra danh sách con của dữ liệu ban đầu với điều kiện ID không phải giá trị NA và trạng thái là "Đã hoàn thành":

```
clean_data <- subset(data, !is.na(ID) & Status == "Done")
```

- Dùng hàm *count*() để lập danh sách tần số xuất hiện của các ID trong dữ liệu đã xử lí. Đó cũng chính là số lần nộp bài tương ứng.

```
submission_table <- count(clean_data$ID)
```

- Dùng hàm *min*() để tìm ra số lần nộp bài ít nhất.

```
min_num <- min(submission_table$freq)
```

- Kết quả:
  - Số lần nộp bài ít nhất ứng với mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1 lần
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1 lần
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1 lần
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1 lần

b) Xác định danh sách các sinh viên có số lần nộp bài ít nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta đã biết số lần nộp bài ít nhất. Từ đó ta lọc được danh sách các sinh viên có số lần nộp bài ít nhất đó từ dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Để tiện cho các câu sau, ta xử lí dữ liệu một lần nữa. Vì ta chỉ cần xét điểm của lần nộp bài sau cùng nên ta loại bỏ các ID trùng lặp và chỉ giữ lại lần xuất hiện cuối cùng bằng lệnh *duplicated*(). Sau đó ta gộp danh sách số lần nộp bài của các sinh viên vào dữ liệu. Để đảm bảo chính xác, ta sắp xếp dữ liệu theo đúng thứ tự của các ID bằng lệnh *order*() trước khi gộp.

```
filtered_data <- clean_data[!rev(duplicated(rev(clean_data$ID))),]
arranged_data <- filtered_data[order(filtered_data$ID),]
arranged_data$submission = submission_table$freq
```

  - Từ dữ liệu đã được xử lí, ta lọc được danh sách các sinh viên có số lần nộp bài ít nhất tương ứng đã tìm được ở câu a bằng lệnh *subset*(). Từ danh sách này ta có được danh sách ID của các sinh viên đó.

```
least_subset <- subset(arranged_data, submission == min_num)
least_subset$ID
```

- Kết quả:
  - Danh sách các sinh viên có số lần nộp bài ít nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1812257	1812478	1813096	1813528
	1813681	1814611	1820028	1910076
	1910094	1910101	1910110	1910137
	1910224	1910238	1910339	1910346
	1910473	1910643	1910663	1910666
	...			
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1812257	1812478	1813528	1820028
	1910076	1910094	1910137	1910198
	1910224	1910265	1910339	1910346
	1910351	1910473	1910565	1910643
	1910650	1910735	1910984	1911000
	...			
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1613010	1812257	1812478	1813096
	1813681	1814096	1814518	1820028
	1910006	1910032	1910038	1910060
	1910076	1910094	1910101	1910110
	1910113	1910137	1910202	1910224
	...			
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1812257	1910094	1910110	1910402
	1910473	1910663	1910984	1911015
	1911056	1911185	1911283	1911285
	1911565	1911569	1911594	1911704
	1911837	1911841	1911931	1912041
	...			

c) Xác định phổ điểm của các sinh viên có số lần nộp bài ít nhất

Kiến thức chuẩn bị

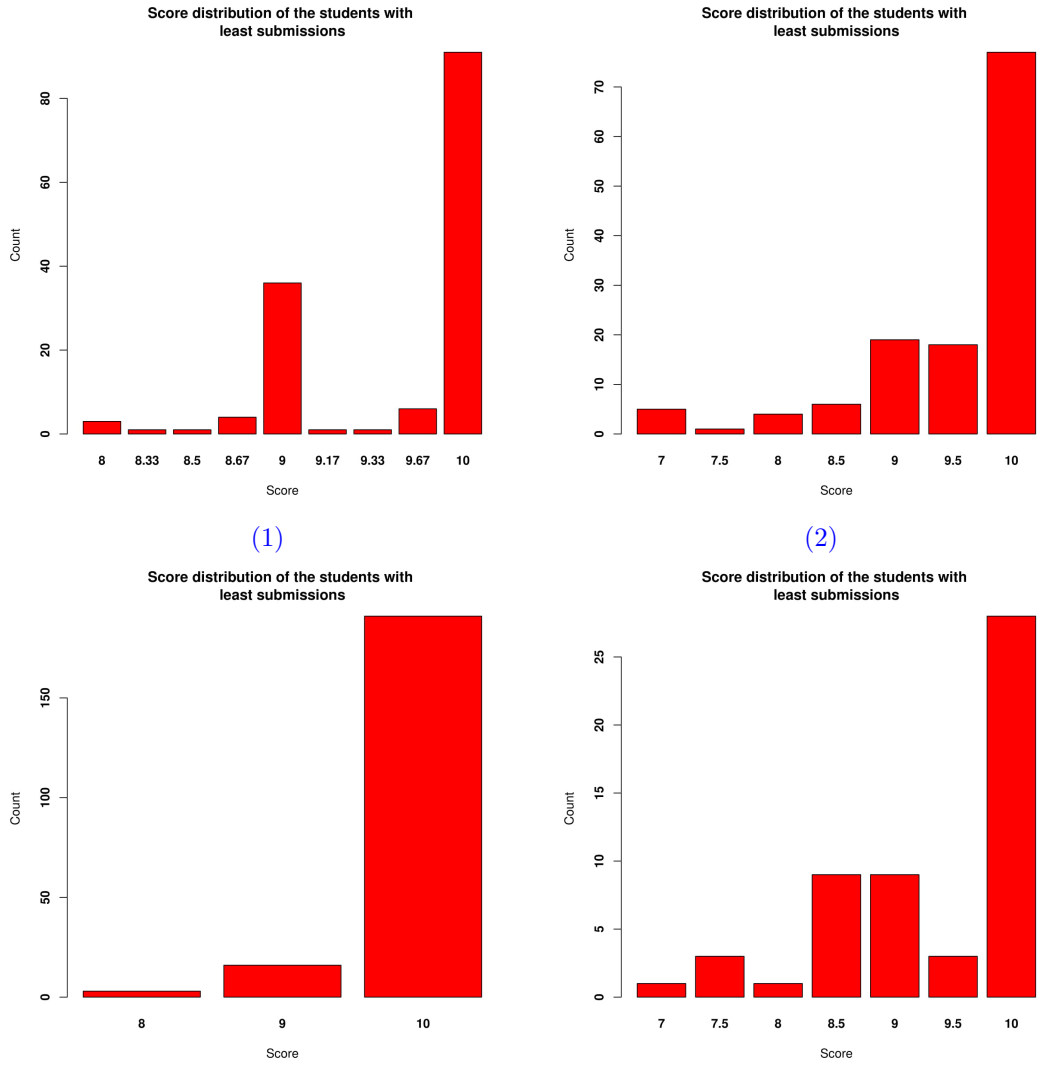
- Cách giải truyền thống:
  - Từ danh sách các sinh viên có số lần nộp bài ít nhất ở trên, ta vẽ được phổ điểm bằng cách thống kê tần số của các điểm số.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *table()* để thống kế tần số của các điểm số, dùng hàm *barplot()* để vẽ phổ điểm.

```
barplot(table(least_subset$Total), xlab = "Score", ylab = "Count", col
= "red", font = 2)
```

- Biểu đồ:



(1) (2)

(3) (4)

Hình 3.1: Phổ điểm của các sinh viên có số lần nộp bài ít nhất
(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"

d) Xác định số lần nộp bài nhiều nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách tần số của các Mã số ID ở câu a, ta xác định được số lần nộp bài nhiều nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Dùng hàm *max*() để tìm ra số lần nộp bài nhiều nhất.

```
max_num <- max(submission_table$freq)
```

- Kết quả:
  - Số lần nộp bài nhiều nhất ứng với mỗi file:

```
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"   5 lần
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"   5 lần
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"   3 lần
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"   3 lần
```

e) Xác định các sinh viên có số lần nộp bài nhiều nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta đã biết số lần nộp bài nhiều nhất. Từ đó ta lọc được danh sách các sinh viên có số lần nộp bài nhiều nhất đó từ dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Từ dữ liệu đã được xử lí, ta lọc được danh sách các sinh viên có số lần nộp bài nhiều nhất tương ứng đã tìm được ở câu a bằng lệnh *subset*(). Từ danh sách này ta có được danh sách ID của các sinh viên đó.

```
most_subset <- subset(arranged_data, submission == max_num)
most_subset$ID
```

- Kết quả:
  - Danh sách các sinh viên có số lần nộp bài cao nhất của mỗi file:

```
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"   1910038   1913756
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"   1912817   1913467   1914768   1915268
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"   1913045   1915520   1927007
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"   1910032   1910060   1910666   1911000
                                       1911136   1912056   1912539   1912676
                                       1913045   1913306   1913355   1913467
                                       1913566   1913775   1913918   1914003
                                       1914011   1914093   1914659   1914713
                                          ...
```

f) Xác định phổ điểm của các sinh viên có số lần nộp bài nhiều nhất

Kiến thức chuẩn bị

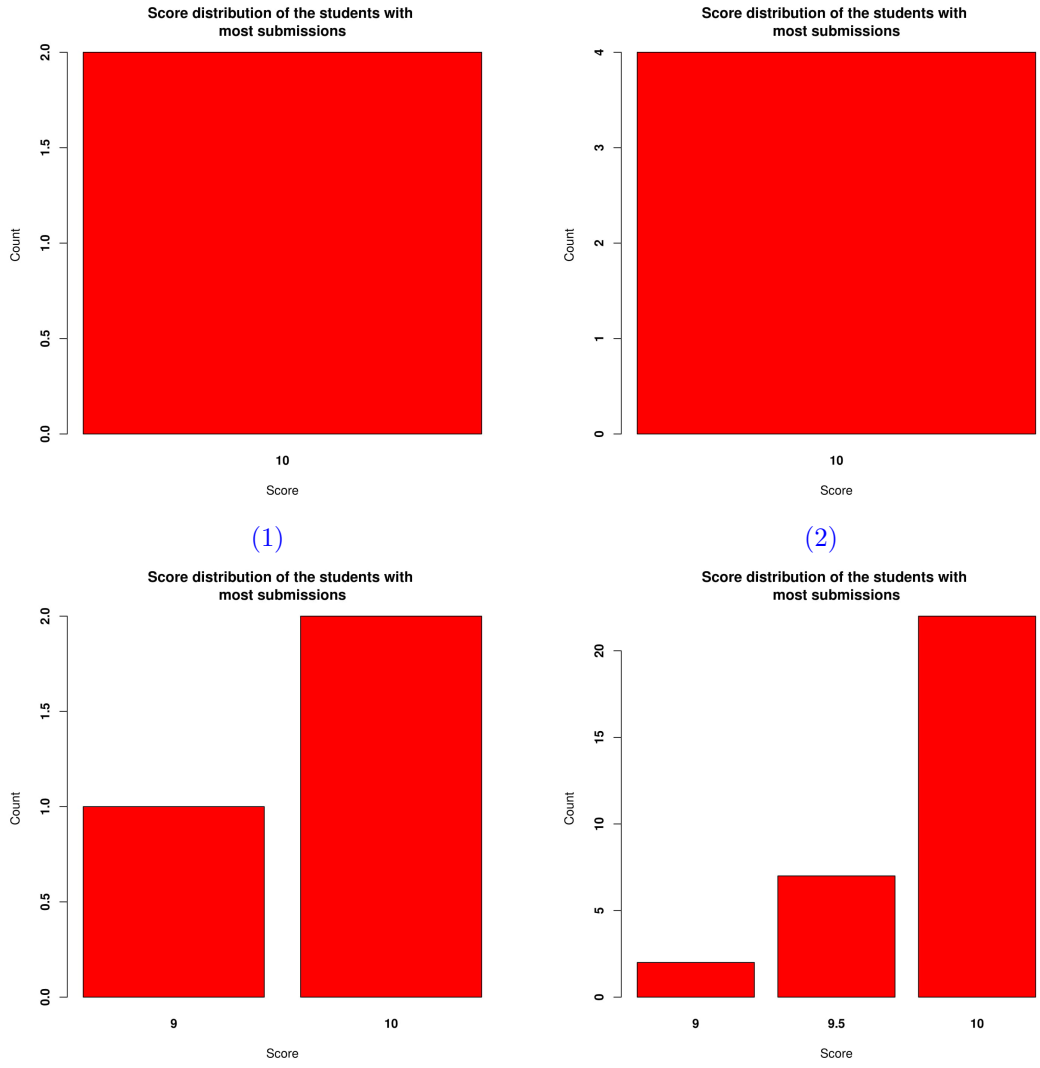
- Cách giải truyền thống:
  - Từ danh sách các sinh viên có số lần nộp bài nhiều nhất ở trên, ta vẽ được phổ điểm bằng cách thống kê tần số của các điểm số.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *table*() để thống kế tần số của các điểm số, dùng hàm *barplot*() để vẽ phổ điểm.

```
barplot(table(least_subset$Total), xlab = "Score", ylab = "Count", col
= "red", font = 2)
```

- Biểu đồ:



(1) (2)

(3) (4)

Hình 3.2: Phổ điểm của các sinh viên có số lần nộp bài nhiều nhất
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

g) Xác định số lần nộp bài trung bình của của các sinh viên

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách tần số của các Mã số ID ở câu a, ta xác định được số lần nộp bài trung bình.
  - Số lần nộp bài trung bình tính bởi:

$$\overline{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

trong đó: x_i là số lần nộp bài của sinh viên thứ i
n là tổng số sinh viên

Hiện thực trên R

- Ý tưởng thực hiện:

- Dùng hàm *avg*() để tìm ra số lần nộp bài trung bình, hàm *round*() để làm tròn số.

- Kết quả:
  - Số lần nộp bài trung bình ứng với mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	2 lần
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	2 lần
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1 lần
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	2 lần

h) Xác định số lượng sinh viên có số lần nộp trung bình

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta đã biết số lần nộp bài trung bình. Từ đó ta có thể đếm được số lượng sinh viên có số lần nộp bài trung bình từ dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Từ dữ liệu đã được xử lí, ta lọc được danh sách các sinh viên có số lần nộp bài trung bình tương ứng đã tìm được bằng lệnh *subset*(). Đếm số ID trong danh sách này bằng lệnh *length*() ta có được số lượng sinh viên có số lần nộp bài trung bình.

```
avg_subset <- subset(arranged_data, submission == avg_num)
length(avg_subset$ID)
```

- Kết quả:
  - Số lượng sinh viên có số lần nộp bài trung bình ứng với mỗi file

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	160 sinh viên
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	163 sinh viên
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	210 sinh viên
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	175 sinh viên

i) Xác định phổ theo điểm số của các sinh viên có lần nộp bài trung bình

Kiến thức chuẩn bị

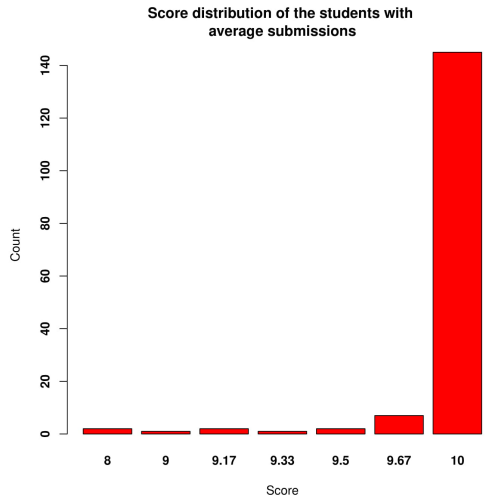
- Cách giải truyền thống:
  - Từ danh sách các sinh viên có số lần nộp bài trung bình ở trên, ta vẽ được phổ điểm bằng cách thống kê tần số của các điểm số.

Hiện thực trên R

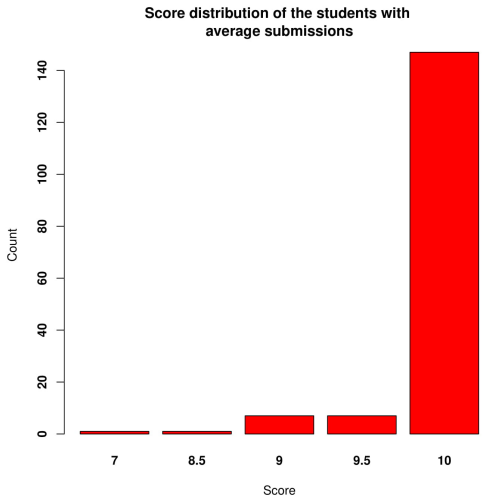
- Ý tưởng thực hiện:
  - Ta sử dụng hàm *table*() để thống kế tần số của các điểm số, dùng hàm *barplot*() để vẽ phổ điểm.

```
barplot(table(avg_subset$Total), xlab = "Score", ylab = "Count", col =
"red", font = 2)
```

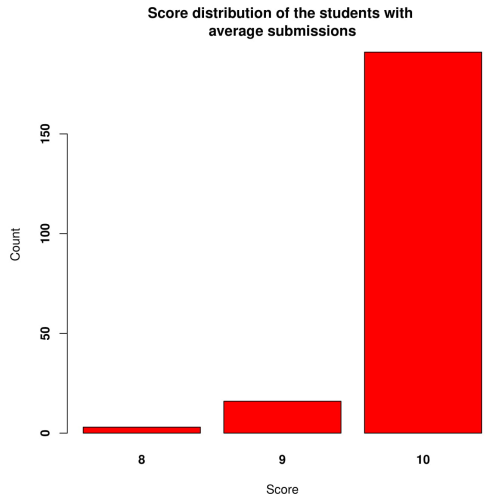
- Biểu đồ:



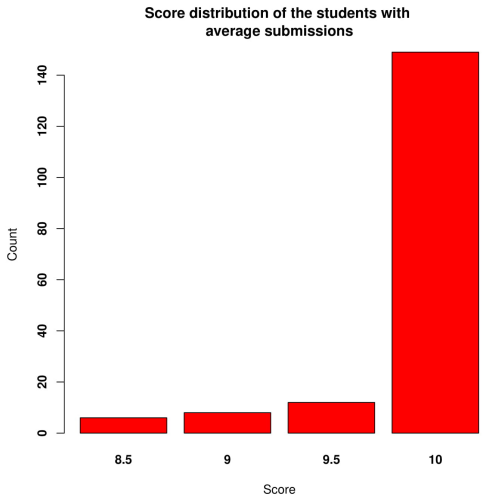
(1)



(2)



(3)



(4)

Hình 3.3: Phổ điểm của các sinh viên có số lần nộp bài trung bình
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

j) Tính trung vị mẫu, cực đại mẫu, cực tiểu mẫu của trên.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Trung vị: Là một số tách giữa nửa lớn hơn và nữa bé hơn của một mẫu, một quần thể, nhưng ở đây ta nói đến là một tập hợp các giá trị là điểm.
  - Cực đại và cực tiểu: Là thành phần lớn nhất (hoặc cùng lớn nhất), nhỏ nhất (hoặc cùng nhỏ nhất) của một tập hợp các giá trị.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Dùng các hàm *median*(), *max*(), *min*() để lần lượt tính trung vị, cực đại, cực tiểu của mẫu.

```
median(avg_subset$Total)
max(avg_subset$Total)
min(avg_subset$Total)
```

- Kết quả:
  - Trung vị - cực đại - cực tiểu (thứ tự lần lượt) của mẫu điểm số của các sinh viên có số lần nộp bài trung bình ứng với mỗi file:

	Trung vị	Cực đại	Cực tiểu
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	10	10	8
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	10	10	7
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	10	10	8
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	10	10	8.5

k) Hãy đo mức độ phân tán của điểm số (xung quanh giá trị trung bình) của mẫu.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Mức độ phân tán của điểm số (xung quanh giá trị trung bình) được thể hiện qua phương sai - độ lệch tiêu chuẩn của mẫu.
  - Phương sai được tính bằng công thức:

$$s^2 = \frac{1}{n}\sum_{i=1}^{k} n_i x_i^2 - \overline{x}^2$$

trong đó: x_i là điểm số có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số bài nộp trong mẫu
$\overline{x}$ là điểm số trung bình của mẫu

  - Độ lệch chuẩn được tính bằng công thức:

$$s = \sqrt{s^2}$$

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta dùng hàm *var()* và *sd()* để các giá trị tương ứng.

```
var(avg_subset$Total)
sd(avg_subset$Total)
```

- Kết quả:
  - Độ lệch tiêu chuẩn của mẫu điểm số của các sinh viên có số lần nộp bài trung bình ứng với mỗi file:

	Phương sai	Độ lệch chuẩn
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	0.07158138	0.267547
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	0.114936	0.3390221
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	0.1229437	0.3506333
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	0.1234319	0.3513287

l) Tính độ méo lệch (skewness), và độ nhọn (kurtosis) của dữ liệu trong mẫu trên.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Độ méo lệch là sự biến dạng sự bất đối xứng trong một phân phối hình chuông đối xứng hay phân phối chuẩn trong một tập dữ liệu, được tính bằng công thức:

$$\tilde{\mu}_3 = \frac{1}{n}\sum_{i=1}^{k} n_i \left(\frac{x_i - \overline{x}}{s}\right)^3$$

trong đó: x_i là điểm số có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số bài làm trong mẫu
$\overline{x}$ là giá trị điểm trung bình
s là độ lệch chuẩn

– Độ nhọn là một đại lượng thống kê được sử dụng để miêu tả các phân phối, mô tả hình dạng của đuôi phân phối đó, tính bằng công thức.

$$\tilde{\mu}_4 = \frac{1}{n}\sum_{i=1}^{k} n_i(\frac{x_i - \overline{x}}{s})^4$$

trong đó: x_i là điểm số có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số bài làm trong mẫu
$\overline{x}$ là giá trị điểm trung bình
s là độ lệch chuẩn

Hiện thực trên R

- Ý tưởng thực hiện:
  – Dùng hàm *skewness()* và *kurtosis()* để lần lượt tính độ méo lệch và độ nhọn của dữ liệu trong mẫu (Cần dùng package Moments).

```
skewness(avg_subset$Total)
kurtosis(avg_subset$Total)
```

- Kết quả:
  – Độ méo lệch - độ nhọn của dữ liệu trong mẫu điểm số của các sinh viên có số lần nộp bài trung bình ứng với mỗi file

	Độ méo lệch	Độ nhọn
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	-5.477856	36.59856
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	-5.241093	37.8015
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	-3.524972	15.58703
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	-2.775748	9.813872

m) Tính tứ phân vị (quartile) thứ nhất (Q_1) và thứ ba (Q_3) của mẫu.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Tứ phân vị thứ Q_1:bằng trung vị phần dưới của một tập.
  – Tứ phân vị thứ Q_3:bằng trung vị phần trên của một tập.

Hiện thực trên R

- Ý tưởng thực hiện:
  – Dùng hàm *quantile()* để tính tứ phân vị thứ nhất (Q_1) và thứ ba (Q_3) của mẫu.

```
quantile(avg_subset$Total, 0.25, na.rm = TRUE)
quantile(avg_subset$Total, 0.75, na.rm = TRUE)
```

- Kết quả:
  – Tứ phân vị thứ nhất (Q1) - thứ ba (Q3) của mẫu điểm số của các sinh viên có số lần nộp bài trung bình ứng với mỗi file:

	Tứ phân vị thứ nhất	Tứ phân vị thứ ba
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	10	10
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	10	10
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	10	10
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	10	10

n) Xác định danh sách các sinh viên nằm trong nhóm có số lần nộp bài nhiều nhì

Kiến thức chuẩn bị

- Cách giải truyền thống:

– Từ danh sách tần số của các Mã số ID (danh sách số lần nộp bài), ta xác định được số lần nộp bài nhiều nhì. Từ đó ta lọc được danh sách các sinh viên có số lần nộp bài nhiều nhất đó từ dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  – Để xác định số lần làm bài nhiều nhì, ta vẫn sử dụng hàm *max()* nhưng phải bỏ đi số lần làm bài nhiều nhất.
  – Từ đó, ta lọc được danh sách các sinh viên có số lần nộp bài nhiều nhì tương ứng bằng lệnh *subset()*. Từ danh sách này ta có được danh sách ID của các sinh viên đó.

```
second_max_num <- max(submission_table$freq[submission_table$freq
!= max_num])    most_2nd_subset <- subset(arranged_data, submission ==
second_max_num)
most_2nd_subset$ID
```

- Kết quả:
  – Danh sách các sinh viên có số lần nộp bài nhiều nhì của mỗi file:

```
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"  1910198  1911000  1913186  1913328
                                      1927007  1937019
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"  1913040  1914003  1914210
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"  1511191  1812477  1852443  1910123
                                      1910409  1910892  1911066  1911262
                                      1911363  1911441  1912123  1912288
                                      1912371  1912410  1912457  1912594
                                      1912602  1912676  1912713  1912761
                                          ...
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"  1613010  1812477  1812478  1813681
                                      1814096  1814518  1820028  1910006
                                      1910038  1910101  1910113  1910123
                                      1910137  1910202  1910224  1910238
                                      1910265  1910276  1910298  1910339
                                          ...
```

o) Xác định danh sách các sinh viên nằm trong nhóm có số lần nộp bài nhiều nhất hoặc nhiều nhì

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Ta đã biết số lần nộp bài nhiều nhất và nhiều nhì. Từ đó ta lọc được danh sách các sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì từ dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  – Từ dữ liệu, ta lọc được danh sách các sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì tương ứng bằng lệnh *subset()*. Từ danh sách này ta có được danh sách ID của các sinh viên đó.

```
most_group_subset <- subset(arranged_data, submission == max_num |
submission == second_max_num)
most_group_subset$ID
```

- Ý tưởng thực hiện:
  – Để xác định số lần làm bài nhiều nhì, ta vẫn sử dụng hàm *max()* nhưng phải bỏ đi số lần làm bài nhiều nhất.
  – Từ đó, ta lọc được danh sách các sinh viên có số lần nộp bài nhiều nhì tương ứng bằng lệnh *subset()*. Từ danh sách này ta có được danh sách ID của các sinh viên đó.

```
second_max_num <- max(submission_table$freq[submission_table$freq
!= max_num])    most_2nd_subset <- subset(arranged_data, submission ==
second_max_num)
most_2nd_subset$ID
```

- Kết quả:
  - Danh sách các sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1910038	1910198	1911000	1913186
	1913328	1913756	1927007	1937019
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1912817	1913040	1913467	1914003
	1914210	1914768	1915268	
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1511191	1812477	1852443	1910123
	1910409	1910892	1911066	1911262
	1911363	1911441	1912123	1912288
	1912371	1912410	1912457	1912594
	1912602	1912676	1912713	1912761
	...			
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1613010	1812477	1812478	1813681
	1814096	1814518	1820028	1910006
	1910032	1910038	1910060	1910101
	1910113	1910123	1910137	1910202
	1910224	1910238	1910265	1910276
	...			

p) Xác định số lượng sinh viên nằm trong nhóm có số lần nộp bài nhiều nhất hoặc nhiều nhì

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta đã biết số lần nộp bài nhiều nhất và nhiều nhì. Từ đó ta có thể đếm được số lượng sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì từ dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Từ câu trên, ta đã có danh sách các sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì. Đếm số ID trong danh sách này bằng lệnh *length()* ta có được số lượng sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì.

    ```
    length(most_group_subset$ID)
    ```

- Kết quả
  - Số lượng sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì tương ứng với mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	8 sinh viên
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	7 sinh viên
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	70 sinh viên
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	206 sinh viên

q) Xác định phổ theo điểm số của các sinh viên có lần nộp bài nhiều nhất hoặc nhiều nhì

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách các sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì ở trên, ta vẽ được phổ điểm bằng cách thống kê tần số của các điểm số.

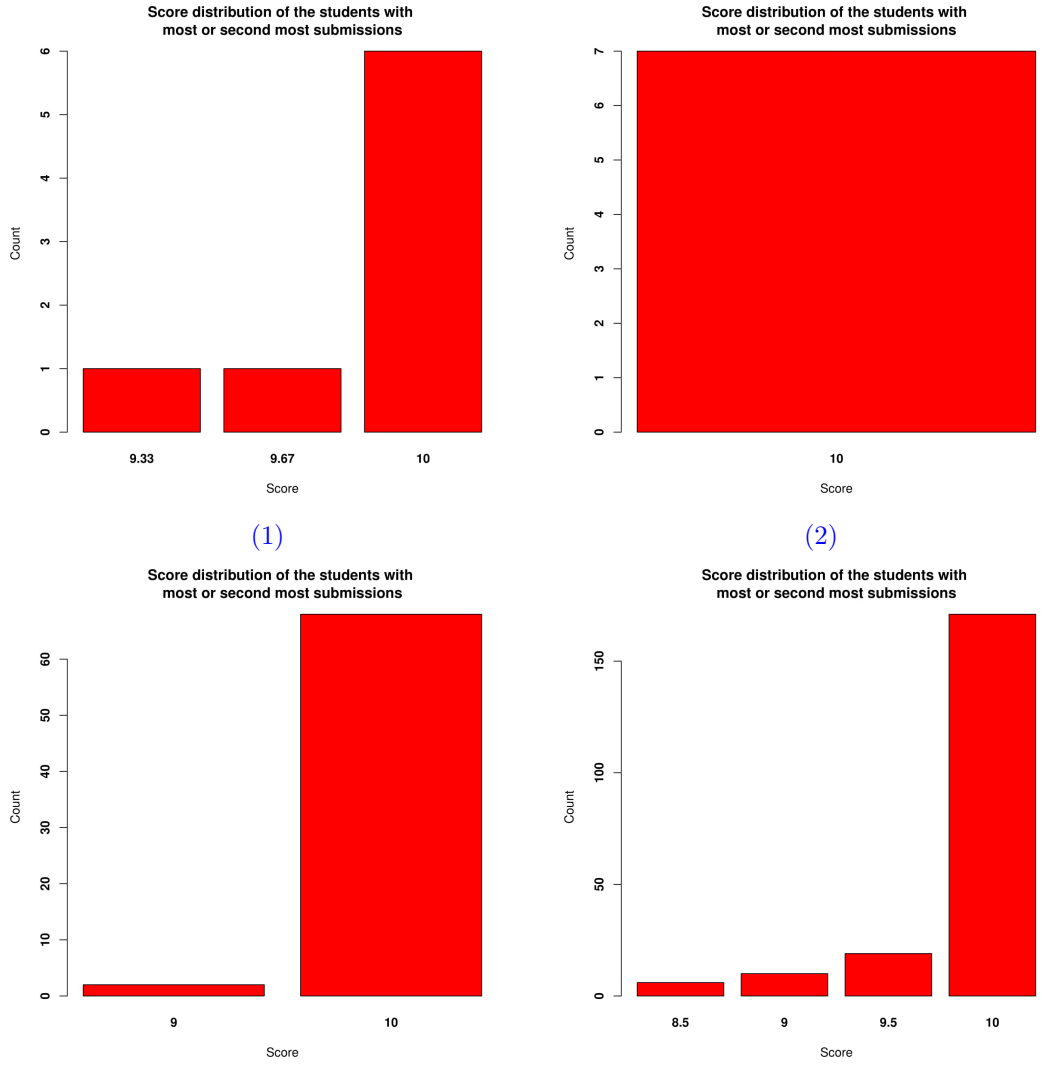
Hiện thực trên R

- Ý tưởng thực hiện:

– Ta sử dụng hàm *table()* để thống kế tần số của các điểm số, dùng hàm *barplot()* để vẽ phổ điểm.

```
barplot(table(most_group_subset$Total), xlab = "Score", ylab =
"Count", col = "red", font = 2)
```

- Biểu đồ:



(1) (2)

(3) (4)

Hình 3.4: Phổ điểm của các sinh viên có số lần nộp bài nhiều nhất hoặc nhiều nhì

(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

r) Xác định danh sách các sinh viên nằm trong nhóm một phần ba đầu theo thứ tự số lần nộp bài giảm dần

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Ta sắp xếp dữ liệu theo thứ tự giảm dần của số lần nộp bài rồi lấy một phần ba đầu tiên của dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Để sắp xếp dữ liệu theo thứ tự giảm dần của số lần nộp bài, ta dùng hàm *order*(). Sau đó ta lấy một phần ba đầu của danh sách bằng hàm *head*().

```
decreasing_data <- arranged_data[order(arranged_data$submission,
decreasing = TRUE),]
top_third <- head(decreasing_data, length(decreasing_data$submission)/3)
top_third$ID
```

- Kết quả:
  - Danh sách các sinh viên nằm trong top một phần ba theo số lần nộp bài giảm dần của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1910038	1913756	1910198	1911000
	1913186	1913328	1927007	1937019
	1712727	1910113	1910276	1910892
	1911066	1911185	1911565	1911704
	1912267	1912463	1912683	1912700
	...			
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1912817	1913467	1914768	1915268
	1913040	1914003	1914210	1712727
	1813503	1910123	1910563	1910666
	1911015	1911565	1911591	1911881
	1911900	1911907	1912046	1912457
	...			
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1913045	1915520	1927007	1511191
	1812477	1852443	1910123	1910409
	1910892	1911066	1911262	1911363
	1911441	1912123	1912288	1912371
	1912410	1912457	1912594	1912602
	...			
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1910032	1910060	1910666	1911000
	1911136	1912056	1912539	1912676
	1913045	1913306	1913355	1913467
	1913566	1913775	1913918	1914003
	1914011	1914093	1914659	1914713
	...			

s) Xác định số lượng các sinh viên nằm trong nhóm một phần ba đầu theo thứ tự số lần nộp bài giảm dần

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Đếm số lượng sinh viên trong danh sách ở câu trên ta có số lượng sinh viên nằm trong nhóm một phần ba đầu theo thứ tự số lần nộp bài giảm dần.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Từ câu trên, ta đã có danh sách các sinh viên nằm trong nhóm một phần ba đầu theo thứ tự số lần nộp bài giảm dần. Đếm số ID trong danh sách này bằng lệnh *length*() ta có được số lượng sinh viên nằm trong nhóm một phần ba đầu theo thứ tự số lần nộp bài giảm dần.

```
length(top_third$ID)
```

- Kết quả:
  - Số lượng sinh viên nằm trong nhóm một phần ba đầu theo thứ tự số lần nộp bài giảm dần tương ứng với mỗi file

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	114 sinh viên
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	114 sinh viên
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	93 sinh viên
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	86 sinh viên

t) Xác định phổ theo điểm số của các sinh viên nằm trong nhóm một phần ba đầu theo thứ tự số lần nộp bài giảm dần

Kiến thức chuẩn bị

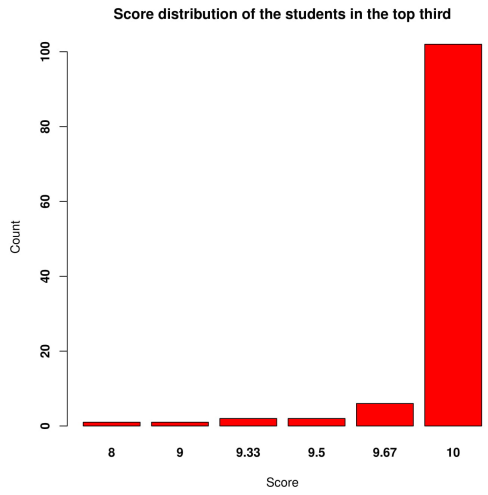
- Cách giải truyền thống:
  - Từ danh sách các sinh viên nằm trong nhóm một phần ba đầu theo thứ tự số lần nộp bài giảm dần ở trên, ta vẽ được phổ điểm bằng cách thống kê tần số của các điểm số.

Hiện thực trên R

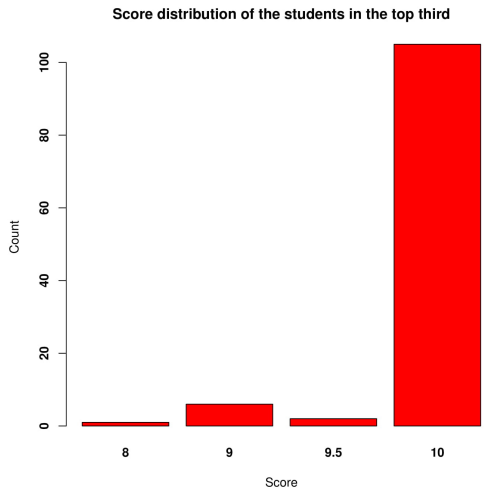
- Ý tưởng thực hiện:
  - Ta sử dụng hàm *table()* để thống kế tần số của các điểm số, dùng hàm *barplot()* để vẽ phổ điểm.

```
barplot(table(top_third$Total), xlab = "Score", ylab = "Count", col =
"red", font = 2)
```

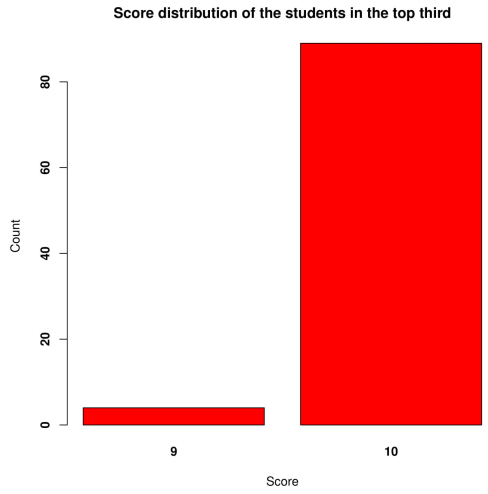
- Biểu đồ:



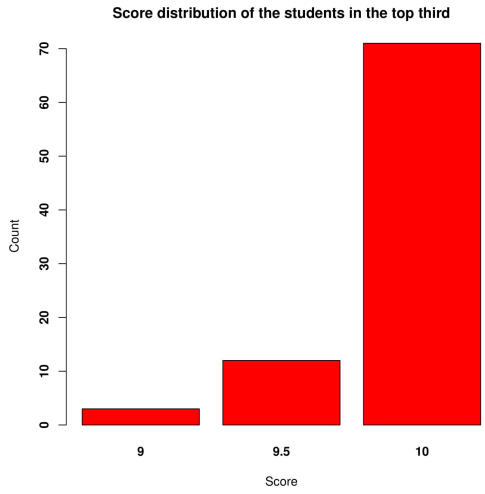
(1)



(2)



(3)



(4)

Hình 3.5: Phổ điểm của các sinh viên trong nhóm một phần ba đầu

(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"

u) Xác định phổ theo điểm số của các sinh viên nằm trong k nhóm đầu mà mỗi nhóm chứa các sinh viên có cùng số lần nộp bài và các nhóm được sắp xếp theo thứ tự giảm dần của có số lần nộp bài (với k cho trước).

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ bảng tần số của các ID, ta tìm ra được danh sách k nhóm đầu (k cho trước) theo thứ tự giảm dần của số lần nộp bài. Từ danh sách này, thống kê tần số của các điểm số, ta có được phổ điểm.

Hiện thực trên R

- Ý tưởng thực hiện:
  - k là số cho trước, được nhập từ bàn phím bằng hàm *readline*()

```
k <- readline(prompt = "Enter k: ")
k <- as.integer(k)
```

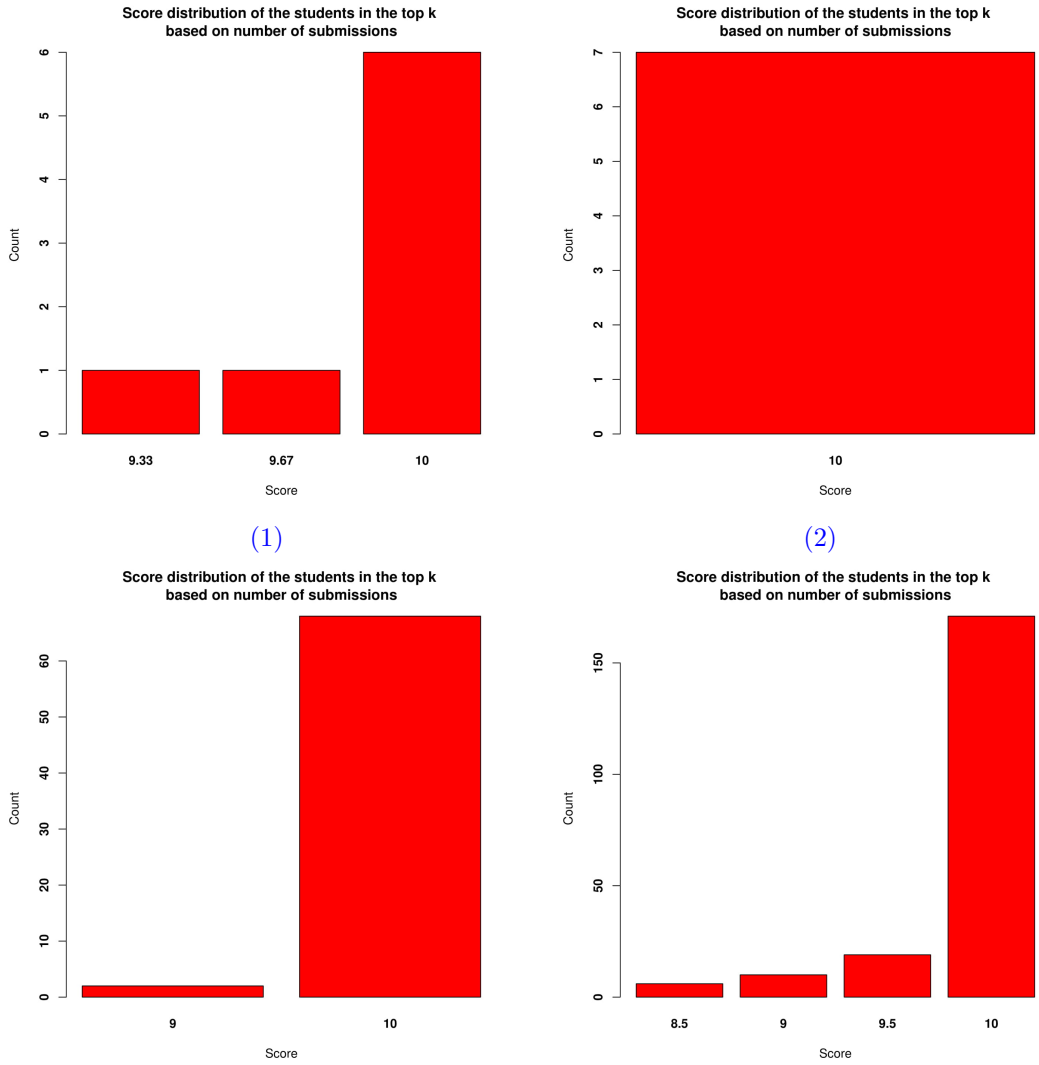
  - Trước hết ta cần tìm số lần nộp bài lớn thứ k bằng hàm *max*() và vòng lặp for.

```
if (k == 1) {
   k_th_max <- max_num
}
else{
   for (i in 2:k)
   {
         k_th_max <- max(submission_table$freq[submission_table$freq <
max_temp])
      max_temp <- k_th_max
   }
}
```

  - Số lần nộp bài lớn thứ k được lưu trong biến k_th_max
  - Sau đó ta lọc được danh sách k nhóm đầu bằng hàm *subset*(). Từ danh sách này ta có được phổ điểm của các học sinh trong k nhóm đầu. Ta sử dụng hàm *table*() để thống kế tần số của các điểm số, dùng hàm *barplot*() để vẽ phổ điểm.

```
top_k_group <- subset(arranged_data, submission >= k_th_max)
barplot(table(top_k_group$Total), xlab = "Score", ylab = "Count", col
= "red", font = 2)
```

- Biểu đồ:
  - Giả sử $k = 2$:



Hình 3.6: Phổ điểm của các sinh viên trong k nhóm đầu
(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"

Bài 4: Nhóm câu hỏi liên quan đến thời gian, tần suất nộp bài của các sinh viên

a) Với mỗi sinh viên, xác định thời gian dài nhất tính từ lần nộp bài đầu tiên đến lần nộp cuối.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Với mỗi sinh viên, ta lấy thời gian nộp bài cuối cùng của sinh viên đó trừ cho thời gian nộp bài đầu tiên của sinh viên đó.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta tạo 2 data frame lần lượt giữ các giá trị của các lần nộp bài đầu và cuối của sinh viên, sau đó lấy hiệu 2 vector ngày tháng trong 2 data frame ta sẽ được thời gian dài nhất tính từ lúc nộp bài lần đầu cho đến lúc nộp bài lần cuối.

- Sử dụng hàm *max()* để thu được giá trị thời gian dài nhất.

- Kết quả:
  - Thời gian dài nhất tính từ lần nộp bài đầu tiên đến lần nộp cuối tương ứng với mỗi file

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	73468 phút
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	67413 phút
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	32769 phút
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	30862 phút

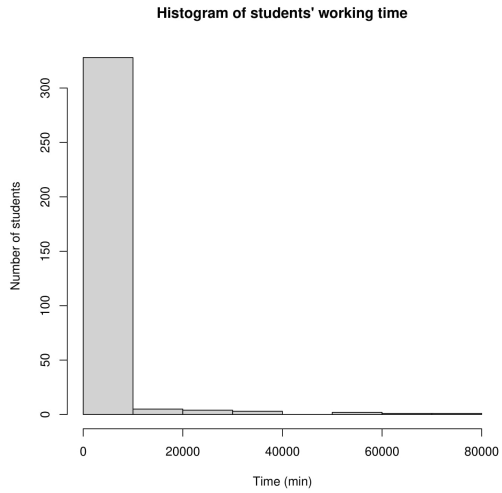
b) Xác định phổ thời gian làm việc (được tính từ lần nộp bài đầu tiên đến lần nộp cuối) của các sinh viên.

Kiến thức chuẩn bị

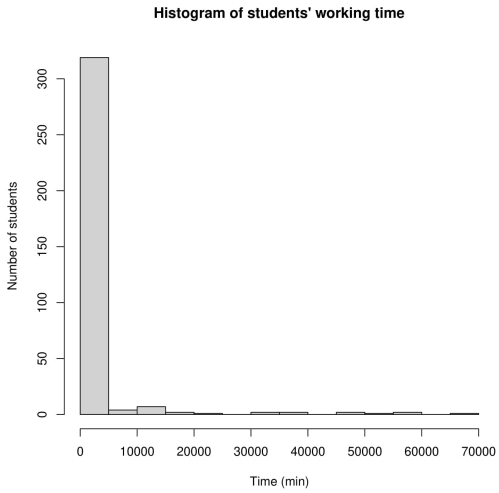
- Cách giải truyền thống:
  - Mỗi sinh viên có một khoảng thời gian học tập online là khác nhau, qua việc các sinh viên nộp bài lần đầu và lần cuối, ta thống kê được biểu đồ biểu diễn phổ thời gian làm việc của sinh viên (theo số lượng sinh viên và hiệu thời gian giữa lần nộp đầu và cuối)

Hiện thực trên R

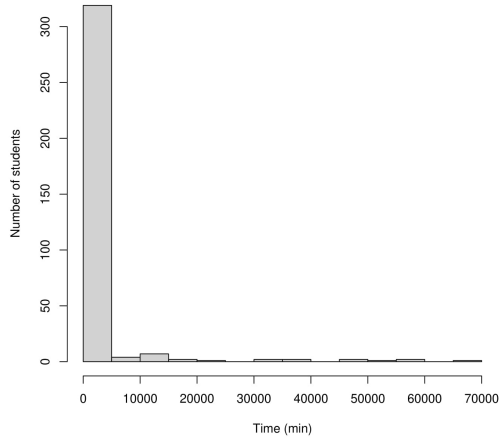
- Ý tưởng thực hiện:
  - Dùng hàm *hist()* để vẽ biểu đồ như hình bên dưới.
- Biểu đồ:



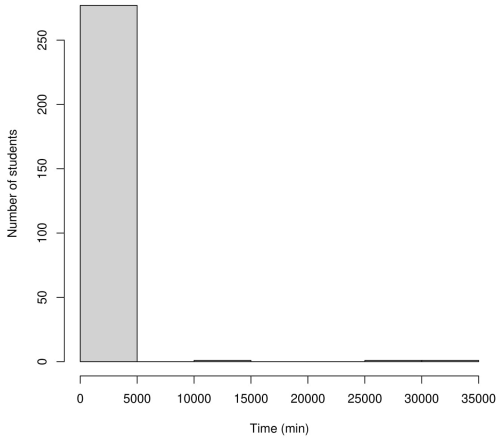
(1)



(2)



(3)



(4)

Hình 4.1: Phổ thời gian làm việc của các sinh viên ứng với mỗi file
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

c) Tần suất nộp bài được tính bằng phân số giữa khoảng thời gian tính từ lần nộp bài đầu tiên đến lần nộp cuối và số lần nộp bài.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Được tính bằng thương của hiệu thời gian nộp bài (như ở câu *a*) và số lần nộp bài.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Dùng hàm *table()* và *data.frame()* để trích ra một data frame mới, chứa ID của sinh viên và số lần nộp bài, sau đó thực hiện phép chia như đã nói ở trên, ta được dữ liệu cần tìm.
- Kết quả:
  - Danh sách sinh viên kèm theo tần suất nộp bài của mỗi file:

	Mã số ID	Tần suất (phút/bài nộp)
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1511191	1.500000e+00
	1613010	1.000000e+00
	1712727	2.000000e+00
	1812257	0.000000e+00
	1812477	2.500000e+00
	...	
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1511191	5.000000e-01
	1613010	1.500000e+00
	1712727	1.666667e+00
	1812257	0.000000e+00
	1812477	1.500000e+00
	...	
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1511191	5.000000e-01
	1613010	0.000000e+00
	1812257	0.000000e+00
	1812477	5.940000e+02
	1812478	0.000000e+00
	...	
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1613010	1.000000e+00
	1812257	0.000000e+00
	1812477	1.000000e+00
	1812478	2.000000e+00
	1813681	1.500000e+00
	...	

d) Xác định danh sách các sinh viên có tần suất nộp bài ít nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Lập danh sách sinh viên kèm theo tần suất nộp bài của mỗi người. Chọn ra những sinh viên có tần suất nộp bài ít nhất.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Trích một data frame mới chứa tần suất nộp bài của mỗi sinh viên từ data đã đọc được từ đề bài, sau đó ta chỉ trích lọc những dữ liệu có tần suất bằng tần suất nhỏ nhất.
- Kết quả:
  - Danh sách sinh viên có tần suất nộp bài ít nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1812257	1812478	1813096	1813528
	1813681	1814611	1820028	1910076
	1910094	1910101	1910110	1910137
	1910224	1910238	1910339	1910346
	1910351	1910473	1910643	1910663
	...			
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1812257	1812478	1813528	1820028
	1910076	1910094	1910113	1910137
	1910198	1910224	1910265	1910339
	1910346	1910351	1910473	1910565
	1910643	1910650	1910735	1910984
	...			
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1613010	1812257	1812478	1813096
	1813681	1814096	1814518	1820028
	1910006	1910032	1910038	1910060
	1910076	1910094	1910101	1910110
	1910113	1910137	1910202	1910224
	...			
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1812257	1910094	1910110	1910402
	1910473	1910663	1910984	1911015
	1911056	1911185	1911283	1911285
	1911565	1911569	1911594	1911704
	1911837	1911841	1911931	1912041
	...			

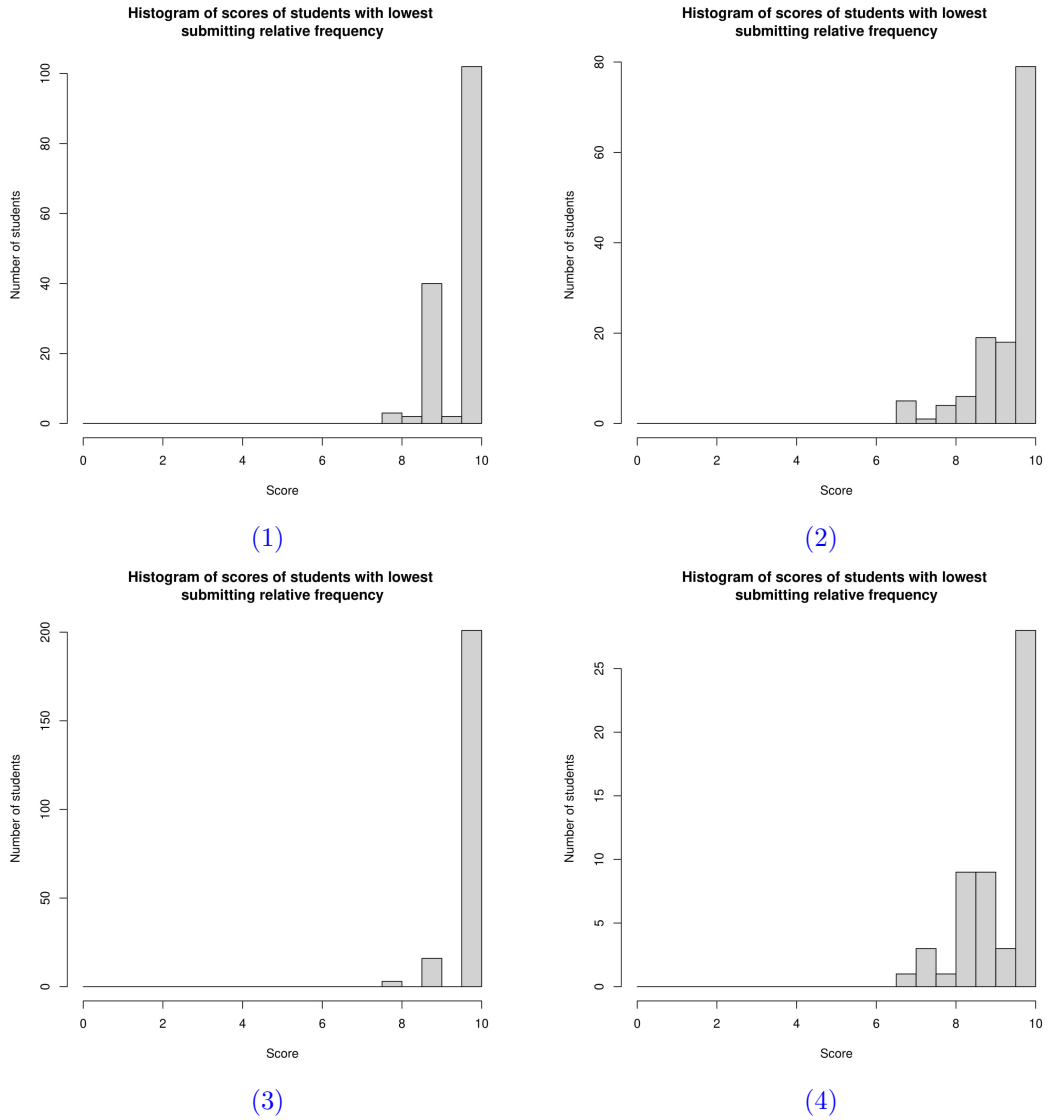
e) Xác định phổ điểm của các sinh viên có tần suất nộp bài ít nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Tương tự câu *d*, ta liệt kê các sinh viên có tần suất nộp bài ít nhất và vẽ phổ điểm của các sinh viên ấy.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Từ data rút ra một data frame (gọi là A) gồm ID của các sinh viên và điểm số cao nhất của mỗi sinh viên. Đồng thời ta đã có một data frame từ câu *c* và *d* (gọi là B). Rút từ B ra các sinh viên có tần suất nộp bài thấp nhất. Sau đó dùng hàm *subset()* trích lọc ra từ A, ta có được một data frame mới chứa dữ liệu của sinh viên có tần suất nộp bài ít nhất, từ đó in ra phổ điểm bằng hàm *hist()*.
  - Trong bài này ta sử dụng các hàm: *subset()*, *hist()*, *unique()*, *match()*.
- Biểu đồ:



Hình 4.2: Phổ điểm của các sinh viên với tần suất nộp bài thấp nhất
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

f) Xác định số lượng sinh viên có tần suất nộp bài nhiều nhất

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Lập danh sách sinh viên kèm theo tần suất nộp bài của mỗi người. Chọn ra những sinh viên có tần suất nộp bài nhiều nhất. Đếm số lượng sinh viên.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Tương tự với câu *d*, đối với bài này ta chỉ trích lọc những dữ liệu có tần suất bằng tần suất lớn nhất. Đếm số lượng sinh viên có trong tập này cho ta kết quả về số lượng sinh viên có tần suất nộp bài nhiều nhất.
- Kết quả:
  - Số lượng sinh viên có tần suất nộp bài lớn nhất đối với mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1 sinh viên
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1 sinh viên
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1 sinh viên
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1 sinh viên

g) Xác định các sinh viên có tần suất nộp bài nhiều nhất.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Làm tương tự câu f. Ta lập danh sách những sinh viên có tần suất nộp bài cao nhất từ dữ liệu đã trích lọc.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Sau khi đã trích lọc dữ liệu từ câu f, ta in ra bảng danh sách những sinh viên thỏa mãn tần suất nộp bài bằng tần suất nộp bài cao nhất.
  - Ta sử dụng các hàm *max*(), *subset*() trong bài này
- Kết quả:
  - Các sinh viên có tần suất nộp bài nhiều nhất của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1914477
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1911185
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1915442
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1936024

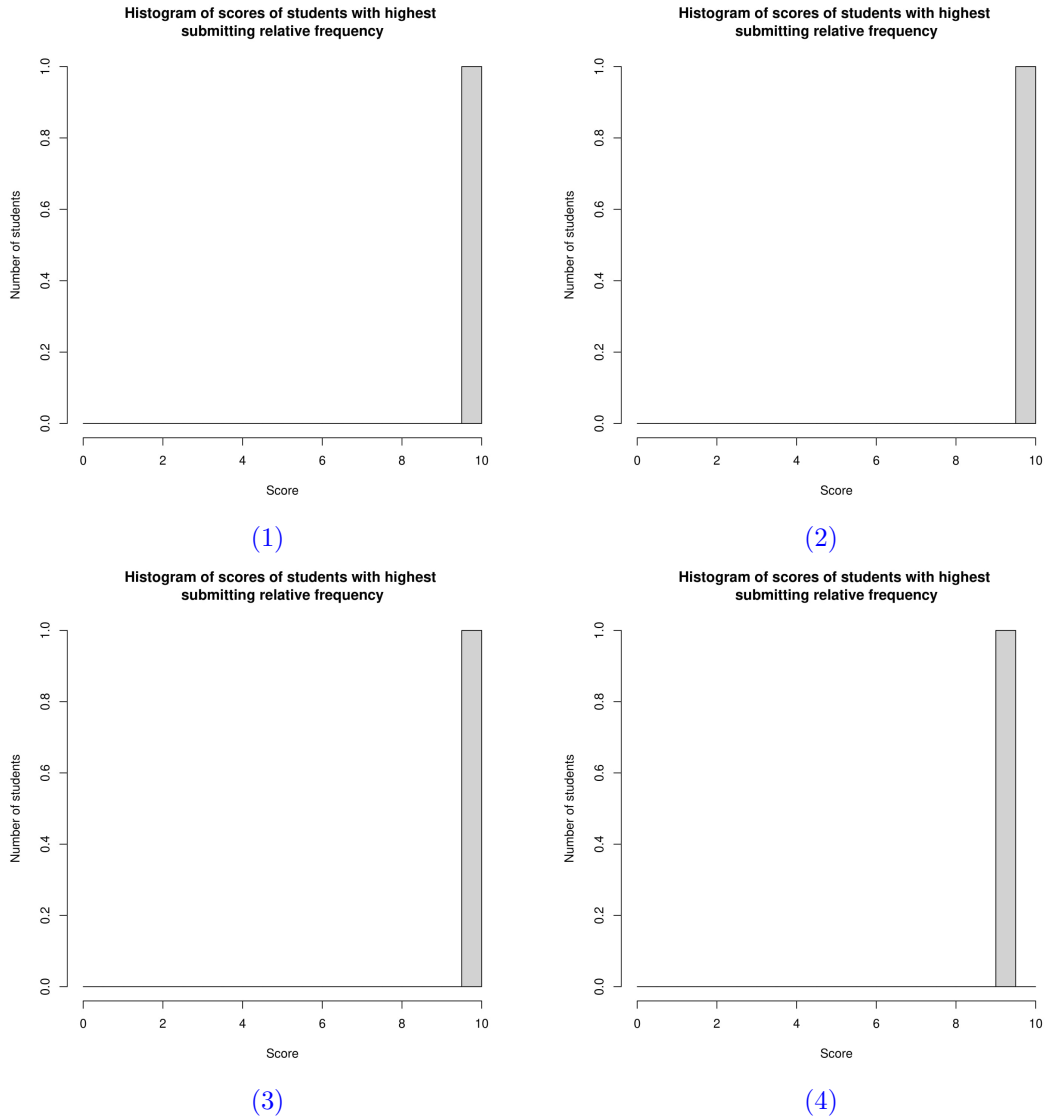
h) Xác định phổ điểm của các sinh viên có tần suất nộp bài nhiều nhất.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Tương tự câu f, ta liệt kê các sinh viên có tần suất nộp bài nhiều nhất và vẽ phổ điểm của các sinh viên ấy.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Các bước hiện thực tương tự câu f, ta lọc dữ liệu gồm những sinh viên có tần suất nộp bài cao nhất, sau đó vẽ phổ điểm dựa trên dữ liệu đã trích lọc bằng hàm *hist*().
  - Trong bài này ta sử dụng các hàm: *subset*(), *hist*(), *unique*(), *match*().
- Biểu đồ:



(1) (2)

(3) (4)

Hình 4.3: Phổ điểm của các sinh viên với tần suất nộp bài cao nhất
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

i) Xác định các sinh viên nằm trong nhóm có tần suất nộp bài nhiều nhì.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Sắp xếp lại các tần suất nộp bài của sinh viên từ cao xuống thấp, lọc ra những bạn sinh viên có tần suất nộp bài nhiều nhì.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Đầu tiên, ta dùng hàm *subset*() trích lọc data frame không có những sinh viên có tần suất nộp bài nhiều nhất. Sau đó lại dùng hàm *subset*() trích lọc data frame vừa thu được nhưng lần này là trích lọc những sinh viên có tần suất nhiều nhất. Suy ra, ta được danh sách những sinh viên có tần suất nộp nhiều nhì.
- Kết quả:

– Các sinh viên có tần suất nộp bài nhiều nhì của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1911185
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1914093
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1915520
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1915775

j) Xác định các sinh viên nằm trong nhóm có tần suất nộp bài nhiều nhất hoặc nhiều nhì.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Sắp xếp lại các tần suất nộp bài của sinh viên từ cao xuống thấp, lọc ra những bạn sinh viên có tần suất nộp bài nhiều nhất hoặc nhiều nhì.

Hiện thực trên R

- Ý tưởng thực hiện:
  – Để lọc ra danh sách các sinh viên có tần suất nộp bài nhiều nhất hoặc nhiều nhì, ta dùng lại dữ liệu vừa mới tạo từ câu i, sử dụng hàm *subset()* với dữ liệu trên với điều kiện giá trị của tần suất nộp bài lớn hơn hoặc bằng câu i. Như vậy, ta được danh sách các sinh viên có tần suất nộp bài nhiều nhất và nhiều nhì.
- Kết quả:
  – Các sinh viên có tần suất nộp bài nhiều nhất hoặc nhì của mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1911185	1914477
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1914093	1911185
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1915520	1915442
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1915775	1936024

k) Hãy tính thời gian trung bình (tính bằng giây) giữa hai lần nộp bài liền nhau của cùng một sinh viên trong mẫu đã chọn.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Để tính thời gian giữa các lần nộp bài trung bình, ta lấy trung bình của $n-1$ giá trị thời gian giữa hai lần nộp bài liên tiếp, với n là số lần nộp bài của sinh viên đó.
  – Giá trị này cũng có thể xác định bằng công thức:

$$\overline{\Delta t} = \frac{t_{last} - t_{first}}{n-1}$$

trong đó: t_{last} là thời gian nộp bài cuối cùng
t_{first} là thời gian nộp bài đầu tiên
n là số lần nộp bài

Hiện thực trên R

- Ý tưởng thực hiện:
  – Sử dụng công thức trên, ta tính được thời gian trung bình giữa các lần nộp bài của mỗi sinh viên.
- Kết quả:
  – Danh sách sinh viên kèm theo thời gian trung bình giữa hai lần nộp bài của mỗi file:

	Mã số ID	Thời gian trung bình (giây)
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	1511191	180
	1613010	120
	1712727	180
	1812477	300
	1813503	180
	...	
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1511191	60
	1613010	180
	1712727	150
	1812477	180
	1813096	240
	...	
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1511191	60
	1812477	71280
	1852443	60
	1910123	480
	1910409	60
	...	
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1613010	120
	1812477	120
	1812478	240
	1813681	180
	1814096	540
	...	

l) **Tính tần số, tần suất và tần suất tích lũy của mẫu trên.**

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Tần số: ta đếm tất cả các lần xuất hiện của từng giá trị tần suất nộp bài.
  - Tần suất: ta lấy tần số chia cho tổng tất cả các lần xuất hiện của các giá trị có thể có.

$$f_i = \frac{n_i}{\sum_{j=1}^{k} n_j}$$

trong đó: f_i là tần suất của giá trị tần suất nộp bài thứ i
n_i là tần số của giá trị tần suất nộp bài thứ i
k là số giá trị tần suất nộp bài

  - Tần xuất tích lũy: bằng tần suất của giá trị này cộng với tổng tần suất của các giá trị trước nó.

$$F_{C_i} = \sum_{j=1}^{i} f_i$$

trong đó: F_{C_i} là tần suất tích lũy của giá trị tần suất nộp bài thứ i
f_i là tần suất của giá trị tần suất nộp bài thứ i

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng chủ yếu là hàm *table()* để đếm số lần xuất hiện của các giá trị tần suất nộp bài, rồi thực hiện các phép tính như đã đề cập ở trên. Từ đó, ta lập được bảng tần số, tần suất và tần suất lũy thừa của từng giá trị.
- Kết quả:
  - Danh sách các giá trị tần suất nộp bài kèm theo tần số, tần suất và tần suất tích lũy xuất hiện của chúng trong mỗi file:

	Tần suất nộp bài	Tần số
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	0	149
	0.33	1
	0.5	47
	0.67	7
	0.75	2
	...	
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	0	132
	0.5	28
	0.67	6
	1	41
1.2	1	
	...	
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	0	220
	0.33	1
	0.5	38
	0.67	1
	1	7
	...	
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	0	54
	0.5	25
	0.67	2
	1	68
	1.33	4
	...	

	Tần suất nộp bài	Tần suất
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	0	0.433139535
	0.33	0.002906977
	0.5	0.136627907
	0.67	0.020348837
	0.75	0.005813953
	...	
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	0	0.384839650
	0.5	0.081632653
	0.67	0.017492711
	1	0.119533528
	1.2	0.002915452
	...	
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	0	0.785714286
	0.33	0.003571429
	0.5	0.135714286
	0.67	0.003571429
	1	0.025000000
	...	
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	0	0.207692308
	0.5	0.096153846
	0.67	0.007692308
	1	0.261538462
	1.33	0.015384615
	...	

	Tần suất nộp bài	Tần suất tích lũy
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	(0,1]	0.2848837
	(1,2]	0.4098837
	(2,3]	0.4534884
	(3,4]	0.4622093
	(4,5]	0.4709302
	...	
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	(0,1]	0.2186589
	(1,2]	0.3760933
	(2,3]	0.4373178
	(3,4]	0.4723032
	(4,5]	0.4897959
	...	
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	(0,1]	0.1678571
	(1,2]	0.1857143
	(2,3]	0.1857143
	(3,4]	0.1892857
	(4,5]	0.1928571
	...	
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	(0,1]	0.3653846
	(1,2]	0.6076923
	(2,3]	0.6653846
	(3,4]	0.7038462
	(4,5]	0.7384615
	...	

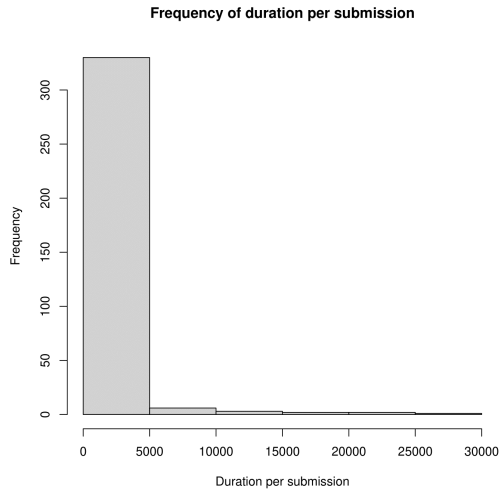
m) Vẽ biểu đồ tần số của mẫu trên. Hãy nhận xét về biểu đồ.

Kiến thức chuẩn bị

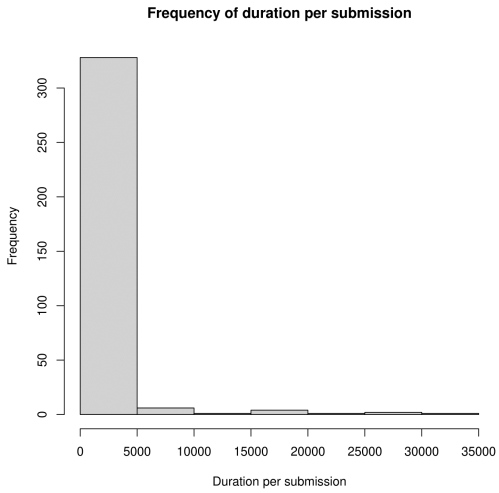
- Cách giải truyền thống:
  - Lập danh sách từng giá trị tần suất nộp bài với tần số của chúng, sau đó vẽ biểu đồ tương ứng từ dữ liệu vừa có.

Hiện thực trên R

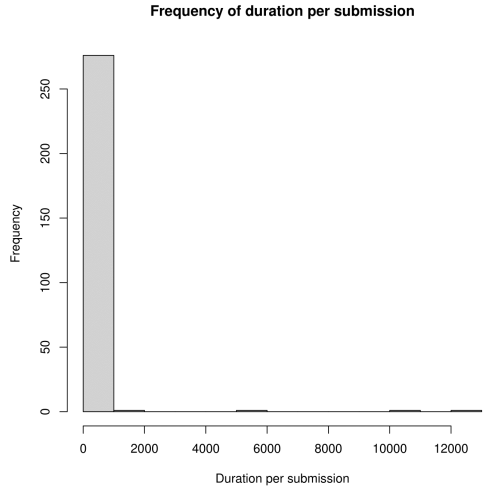
- Ý tưởng thực hiện:
  - Dùng hàm *barplot()* để vẽ các đồ thị, truyền đối số là các vector, ta vẽ được đồ thị tương quan của các giá trị tần suất nộp bài và tần số của chúng.
- Biểu đồ:



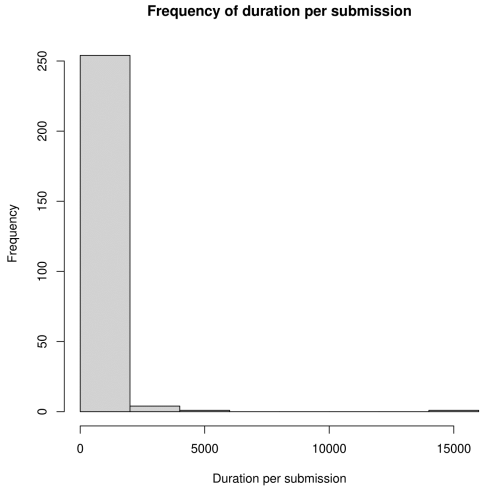
(1)



(2)



(3)



(4)

Hình 4.4: Biểu đồ các giá trị tần suất nộp bài và tần số
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

Nhận xét: Biểu đồ rất dốc, ta thấy được sự phân bố rất chênh lệch của tần suất nộp bài, có rất nhiều sinh viên của tần suất nộp bài trong khoảng 0 đến 5000, còn lại từ 5000 đến 15000 thì chỉ có rất ít sinh viên. Chứng tỏ tần suất nộp bài là rất ít, không có chênh lệch lớn giữa phần đông các sinh viên.

n) Vẽ biểu đồ tần suất của mẫu trên. Hãy nhận xét về biểu đồ.

Kiến thức chuẩn bị

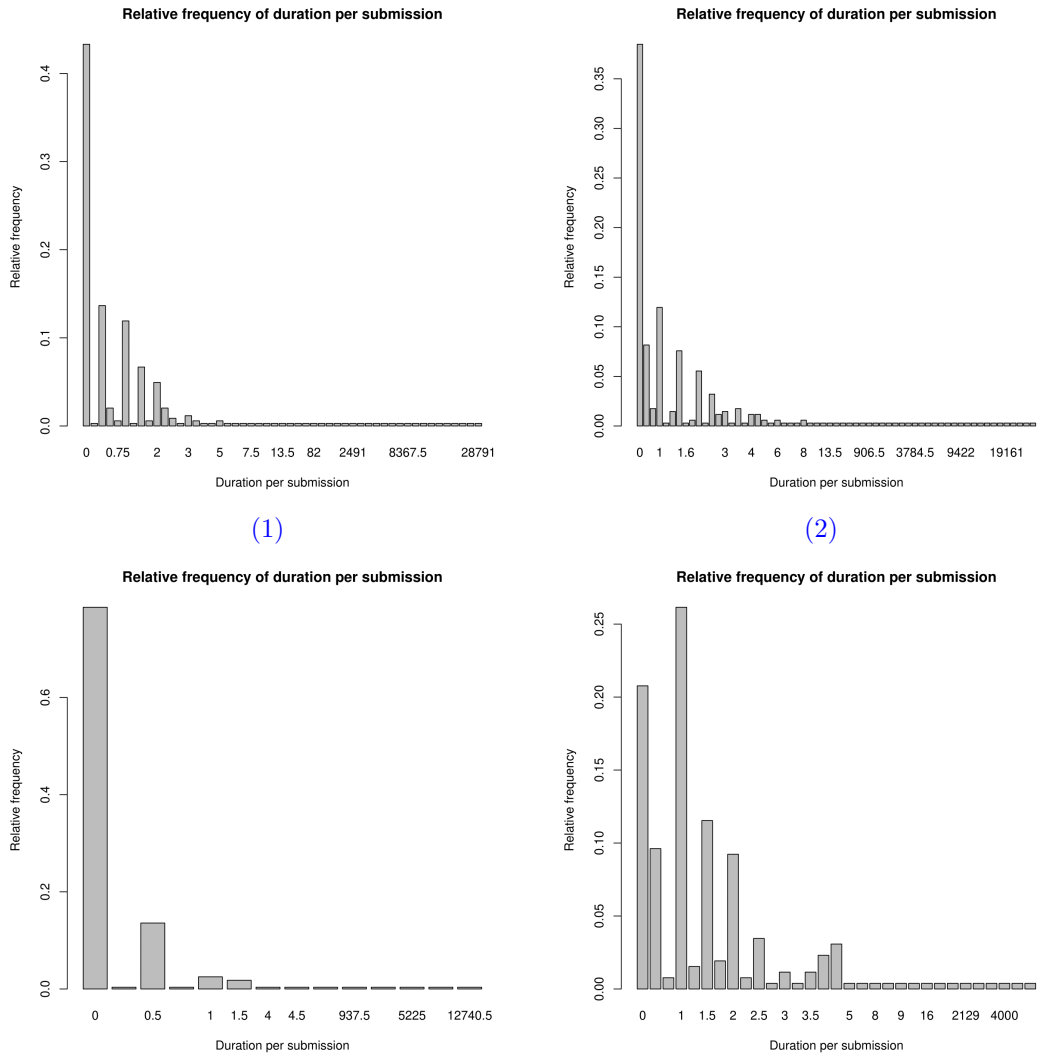
- Cách giải truyền thống:
  - Lập danh sách từng giá trị tần suất nộp bài với tần suất của chúng, sau đó vẽ biểu đồ tương ứng từ dữ liệu vừa có.

Hiện thực trên R

- Ý tưởng thực hiện:

– Dùng hàm *barplot()* để vẽ các đồ thị, truyền đối số là các vector, ta vẽ được đồ thị tương quan của các giá trị tần suất nộp bài và tần suất của chúng.

- Biểu đồ:



(1) (2)

(3) (4)

Hình 4.5: Biểu đồ các giá trị tần suất nộp bài và tần suất
(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"

Nhận xét: Biểu đồ có sự phân bố không đồng đều. Tần suất của mẫu là khả cao khi tần suất nộp bài trong khoảng nhỏ hơn 6. Nhưng nhìn chung thì tần suất của mẫu rất chênh lệch giữa các tần suất nộp bài có giá trị liền kề.

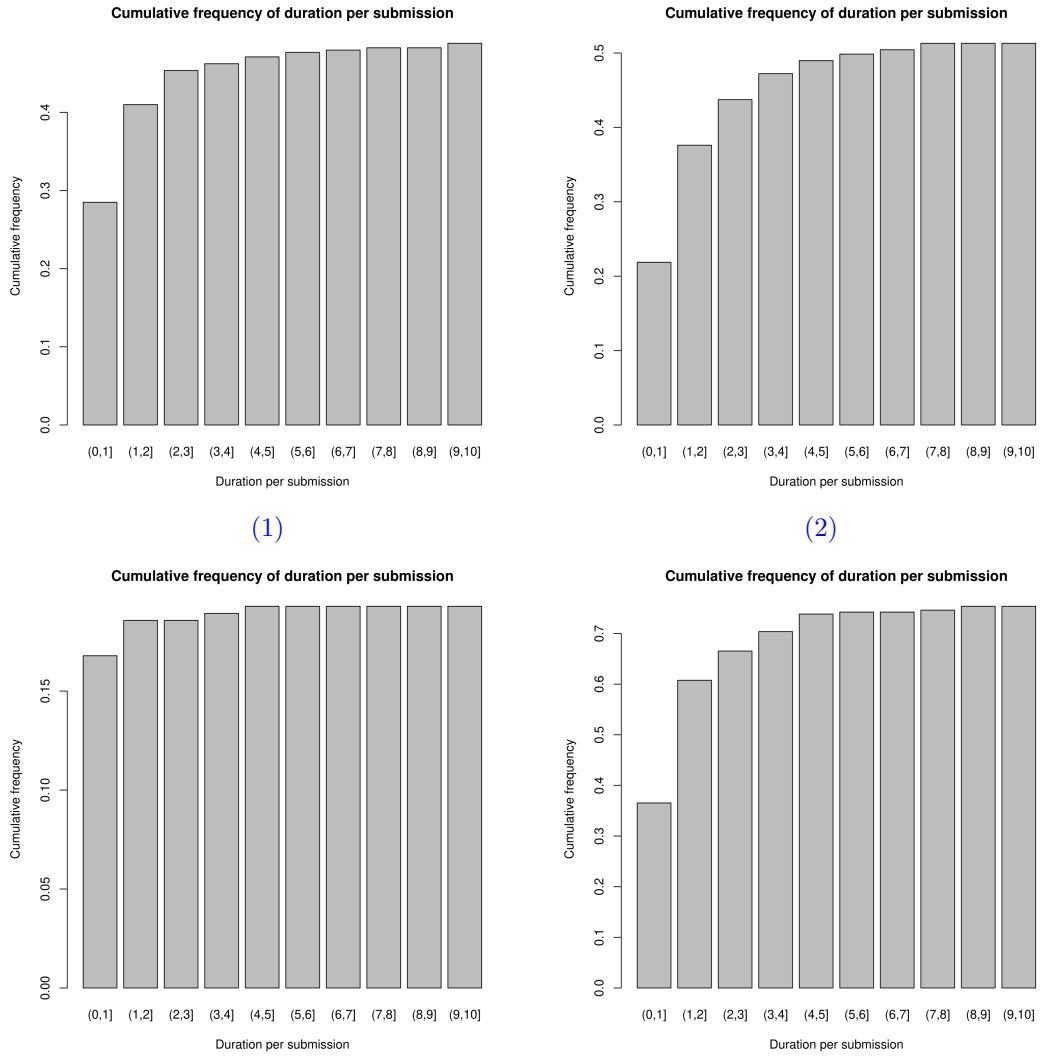
o) Vẽ biểu đồ tần suất tích lũy của mẫu trên. Hãy nhận xét về biểu đồ.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Lập danh sách từng giá trị tần suất nộp bài với tần suất lũy thừa của chúng, sau đó vẽ biểu đồ tương ứng từ dữ liệu vừa có.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Dùng hàm *barplot()* để vẽ các đồ thị, truyền đối số là các *vector()*, ta vẽ được đồ thị tương quan của các giá trị tần suất nộp bài và tần suất lũy thừa của chúng.
- Biểu đồ:



(1) (2)

(3) (4)

Hình 4.6: Biểu đồ các giá trị tần suất nộp bài và tần suất tích lũy
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

Nhận xét: Biểu đồ có sự phân bố liên tục, tần suất tích lũy tăng chậm dần khi tần suất nộp bài tăng, phù hợp với sự thay đổi giá trị của tần suất của mẫu.

p) Tính trung vị mẫu, cực đại mẫu, cực tiểu mẫu của trên.

Kiến thức chuẩn bị

- Cách giải truyền thống:

- Trung vị: Là một số tách giữa nửa lớn hơn và nữa bé hơn của một mẫu, một quần thể, nhưng ở đây ta nói đến là một tập hợp các giá trị là điểm.
- Cực đại và cực tiểu: Là thành phần lớn nhất (hoặc cùng lớn nhất), nhỏ nhất (hoặc cùng nhỏ nhất) của một tập hợp các giá trị.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng các hàm có sẵn như *subset()*, *median()*, *min()*, *max()* để lấy các giá trị trung vị, nhỏ nhất, lớn nhất liên quan đến tần suất nộp bài của từng sinh viên.
- Kết quả:
  - Các giá trị tương ứng theo tần suất nộp bài đối với mỗi file:

	Trung vị mẫu	Cực đại mẫu	Cực tiểu mẫu
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	0.5	28791	0
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	1	33706.5	0
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	0	12740.5	0
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1	15431	0

q) Hãy đo mức độ phân tán của điểm số (xung quanh giá trị trung bình) của mẫu.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Để đo mức độ phân tán, ta xem xét dữ liệu trên 2 yếu tố: phương sai và độ lệch chuẩn.
  - Phương sai được tính bằng công thức:

$$s^2 = \frac{1}{n}\sum_{i=1}^{k} n_i x_i^2 - \overline{x}^2$$

trong đó: x_i là tần suất nộp bài có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số bài làm
$\overline{x}$ là giá trị tần suất nộp bài trung bình

  - Độ lệch chuẩn được tính bằng công thức:

$$s = \sqrt{s^2}$$

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng các hàm có sẵn như *var()*, *sd()* để lấy các giá trị phương sai, độ lệch chuẩn liên quan đến tần suất nộp bài của từng sinh viên.
- Kết quả:
  - Các giá trị tương ứng theo tần suất nộp bài đối với mỗi file:

	Phương sai	Độ lệch chuẩn
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	9202857	3033.621
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	14189912	3766.95
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	1109978	1053.555
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	1175476	1084.194

r) Tính độ méo lệch (skewness), và độ nhọn (kurtosis) của dữ liệu trong mẫu trên.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Độ méo lệch là sự biến dạng sự bất đối xứng trong một phân phối hình chuông đối xứng hay phân phối chuẩn trong một tập dữ liệu, được tính bằng công thức:

$$\tilde{\mu}_3 = \frac{1}{n}\sum_{i=1}^{k} n_i(\frac{x_i - \overline{x}}{s})^3$$

trong đó: x_i là tần suất nộp bài có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số sinh viên
$\overline{x}$ là giá trị tần suất nộp bài trung bình
s là độ lệch chuẩn

– Độ nhọn là một đại lượng thống kê được sử dụng để miêu tả các phân phối, mô tả hình dạng của đuôi phân phối đó, tính bằng công thức.

$$\tilde{\mu}_4 = \frac{1}{n}\sum_{i=1}^{k} n_i(\frac{x_i - \overline{x}}{s})^4$$

trong đó: x_i là tần suất nộp bài có tần số n_i
k là số các các trị x_i phân biệt
n là tổng số sinh viên
$\overline{x}$ là tần suất nộp bài trung bình
s là độ lệch chuẩn

Hiện thực trên R

- Ý tưởng thực hiện:
  – Ta sử dụng các hàm có sẵn như *skewness()*, *kurtosis()* để lấy các giá trị độ lệch, độ nhọn liên quan đến tần suất nộp bài của từng sinh viên.
- Kết quả:
  – Các giá trị tương ứng theo tần suất nộp bài đối với mỗi file

	Độ lệch	Độ nhọn
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	6.28353	46.24756
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	5.845597	39.97973
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	10.48039	116.0303
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	11.65296	156.6423

s) Tính tứ phân vị (quartile) thứ nhất (Q_1) và thứ ba (Q_3) của mẫu.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Tứ phân vị thứ nhất Q1: bằng trung vị phần dưới của một tập.
  – Tứ phân vị thứ ba Q3: bằng trung vị phần trên của một tập.

Hiện thực trên R

- Ý tưởng thực hiện:
  – Ta sử dụng hàm *quartile()* để lấy các giá trị tương ứng.
- Kết quả:
  – Các giá trị tương ứng theo tần số điểm đối với mỗi file:

	Tứ phân vị thứ nhất	Tứ phân vị thứ 3
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	0	1.5
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	0	2
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	0	0
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	0.5	2

Bài 5: Nhóm câu hỏi liên quan đến điểm trung bình

Gọi điểm số lần nộp bài thứ k của sinh viên i với i là uid và $k \in (1, 2, 3, ...)$. Điểm tổng hợp của sinh viên tính tới lần nộp thứ k là điểm lớn nhất cho bài tập đó mà sinh viên đạt được cho tới lần nộp thứ k, tức là:

$$score_{ik} = max(s_{i1}, s_{i2}, ..., s_{ik})$$

Đối với sinh viên nộp ít hơn k lần thì vẫn tính theo công thức với giá trị khuyết xem như là 0. Gọi TB_k là điểm trung bình của các sinh viên tính tới lần nộp thứ k.

a) Hãy tính và vẽ biểu đồ sự phân bố về điểm đạt được của sinh viên sau $k = 6$ lần nộp bài.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Thống kê những lần nộp có cùng Mã số ID theo thứ tự thời gian.
  - Với mỗi Mã số ID lấy điểm số cao nhất của $k = 6$ lần nộp bài đầu tiên.
  - Tổng hợp số lượng những Mã số ID có mức điểm giống nhau và vẽ biểu đồ.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Dùng hàm *order()* Sắp xếp lại dữ liệu *data* theo thời gian nộp bài

    ```
    data <- data[order(data$Finish$year, data$Finish$month,
    data$Finish$day, data$Finish$hour, data$Finish$minute), ]
    ```

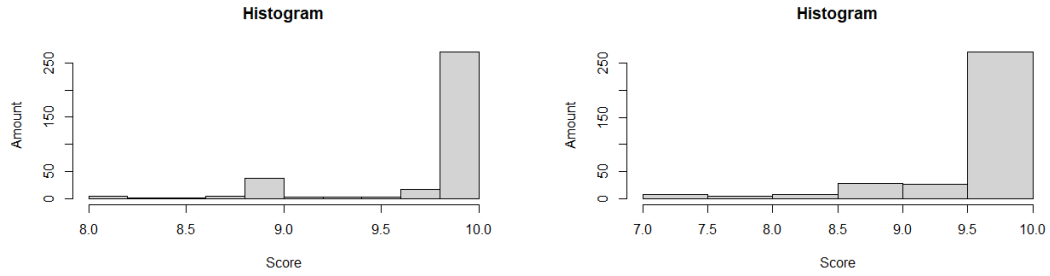
  - Lấy ra dữ liệu Mã số ID và Điểm/10,00 từ *data* và lưu vào 1 data frame có tên *data_stu*.
  - Sử dụng hàm *cblind()* và hàm *match()* sắp xếp điểm của mỗi sinh viên trong $k = 6$ lần nộp vào ma trận *data_submit*:

    ```
    data_submit <- cbind(data_submit, data_stu[match(unique(data$ID),
    data_stu$ID, nomatch = NA_integer_), ][["Total"]])
    data_stu <- data_stu[-match(unique(data$ID), data_stu$ID, nomatch =
    0),]
    ```

  - Tính giá trị lớn nhất của mỗi hàng trong ma trận *data_submit* và vẽ biểu thị sự phân bố điểm:

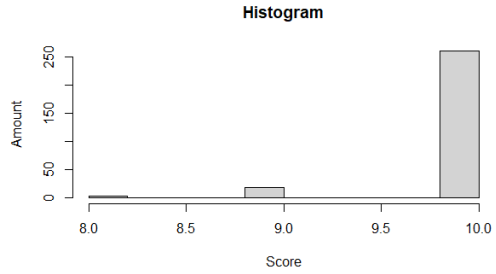
    ```
    hist(apply(data_submit, 1, max, na.rm =TRUE), main = "Histogram", xlab
    = "Score", ylab = "Amount")
    ```

- Biểu đồ:

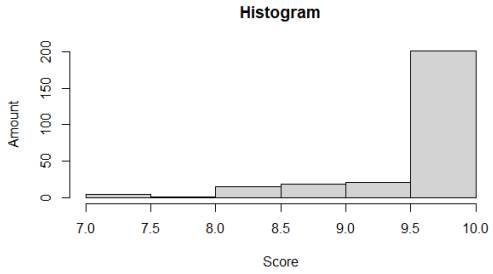


(1)

(2)



(3)



(4)

Hình 5.1: Phân bố điểm của các sinh viên sau $k = 6$ lần nộp
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

b) Áp dụng câu a với k được tính theo công thức sau:

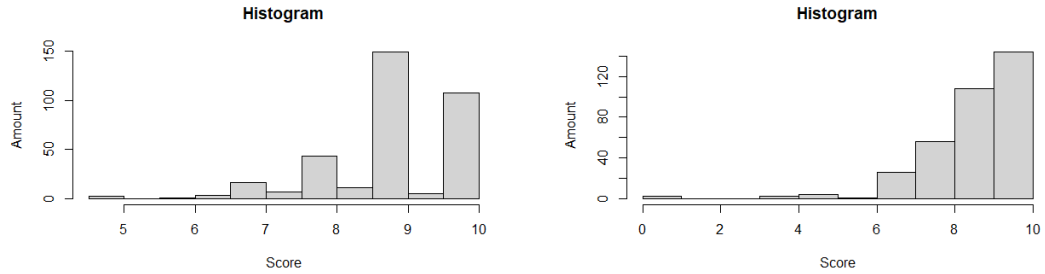
$$MD \ mod \ 3 + 1$$

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Tính toán giá trị $k = MD \mod 3 + 1$ với $MD = 2907$ ta tính được $k = 1$
  - Các bước còn lại hoàn toàn tương tự **Bài 5** câu a.

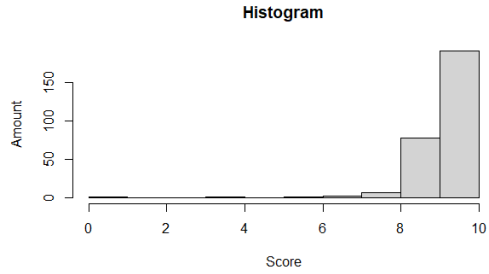
Hiện thực trên R

- Ý tưởng thực hiện:
  - Tương tự **Bài 5** câu a.
- Biểu đồ:

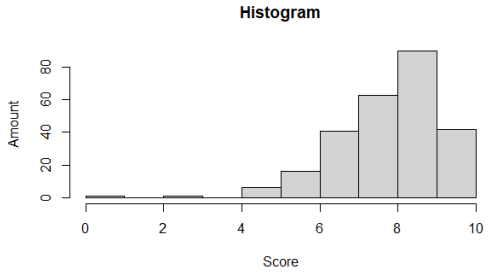


(1)

(2)



(3)



(4)

Hình 5.2: Phân bố điểm của các sinh viên sau $k = 1$ lần nộp
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

c) Hãy tính các giá trị TB_k và vẽ biểu đồ thể hiện sự thay đổi của các giá trị trung bình này với sự thay đổi của k. Hãy nhận xét về biểu đồ mà các em vừa vẽ được.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Thống kê những lần nộp có cùng Mã số ID theo thứ tự thời gian.
  - Tính toán với i từ 1 đến số lần nộp bài lớn nhất của tất cả Mã số ID: với mỗi Mã số ID lấy điểm cao nhất trong i lần nộp bài đầu tiên.
  - Tính trung bình điểm số của toàn bộ Mã số ID ứng với mỗi i
  - Vẽ đồ thị biểu diễn điểm số trung bình theo i

Hiện thực trên R

- Ý tưởng thực hiện:
  - Sắp xếp lại dữ liệu *data* đọc được ban đầu theo thứ tự thời gian nộp bài

    ```
    data<-data[order(data$Finish$year, data$Finish$month, data$Finish$day,
    data$Finish$hour, data$Finish$minute), ]
    ```

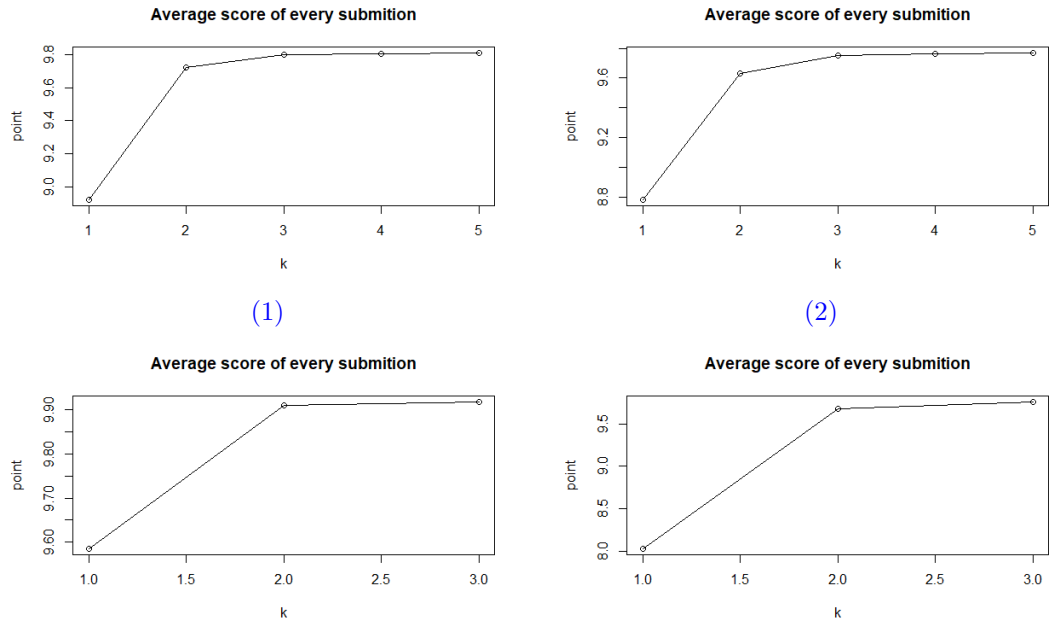
  - Lấy ra dữ liệu Mã số ID và Điểm/10,00 từ *data* và lưu vào 1 data frame có tên *data_stu*
  - Dùng hàm *cbind()* và *match()* liệu thực hiện thêm vào ma trận *data_submit* 1 cột mang giá trị Điểm/10,00 của lần nộp đó ứng với mỗi Mã số ID và đồng thời xóa các hàng đã lấy dữ liệu Điểm/10,00 ra khỏi *data_stu*:

    ```
    data_submit <- cbind(data_submit, data_stu[match(unique(data$ID),
    data_stu$ID, nomatch = NA_integer_),]["Total"]])
    data_stu <- data_stu[-match(unique(data$ID), data_stu$ID, nomatch =
    0),]
    ```

  - Dùng hàm *mean()* tính điểm trung bình của sinh viên đạt được sau mỗi lần nộp bài lưu vào biến *avrPoint* và vẽ đồ thị.

```
avrPoint <- c(avrPoint, mean(apply(data_submit[ ,1:i], 1, max, na.rm =
TRUE), na.rm = TRUE))
```

- Biểu đồ:



(1) (2)

(3) (4)

Hình 5.3: Điểm trung bình của sinh viên qua các lần nộp bài
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

Nhận xét: Sau các lần làm bài điểm số của sinh viên dần được cải thiện và hầu hết số điểm của sinh viên được cải thiệt nhiều nhất ở lần làm bài thứ 2.

d) Hãy cho biết trung bình điểm số mà các sinh viên đạt được qua bài tập tid_n này là bao nhiêu.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Sắp xếp các lần nộp bài theo thứ tự thời gian.
  - Thống kê những lần nộp có cùng Mã số ID theo thứ tự thời gian.
  - Tính điểm cao nhất của tất cả số lần nộp bài cho mỗi Mã số ID.
  - Tính trung bình của tất cả điểm số cáo nhất vừa tính được

Hiện thực trên R

- Ý tưởng thực hiện:
  - Sắp xếp lại dữ liệu *data* đọc được ban đầu theo thứ tự thời gian:
    ```
    data<-data[order(data$Finish$year, data$Finish$month, data$Finish$day,
    data$Finish$hour, data$Finish$minute), ]
    ```
  - Lấy ra dữ liệu Mã số ID và Điểm/10,00 từ *data* và lưu vào 1 data frame có tên *data_stu*
  - Dùng hàm *cbind()* và *match()* liệu thực hiện thêm vào ma trận *data_submit* 1 cột mang giá trị Điểm/10,00 của lần nộp đó ứng với mỗi Mã số ID và đồng thời xóa các hàng đã lấy dữ liệu Điểm/10,00 ra khỏi *data_stu*:

```
data_submit <- cbind(data_submit, data_stu[match(unique(data$ID),
data_stu$ID, nomatch = NA_integer_),][["Total"]])
data_stu <- data_stu[-match(unique(data$ID), data_stu$ID, nomatch =
0),]
```

Tính điểm trung bình và in ra kết quả.

- Kết quả:
  - Điểm trung bình của sinh viên trong các file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	9.81 điểm
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	9.77 điểm
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	9.92 điểm
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	9.76 điểm

Bài 7: Nhóm câu hỏi liên quan đến sinh viên học đối phó

Sinh viên học **đối phó** là sinh viên có nộp bài lần đầu tiên trễ hơn thời điểm t_2.

a) Hãy xác định thời điểm t_2 phù hợp.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta sắp xếp bảng giá trị chỉ gồm lần nộp đầu tiên theo thứ tự tăng dần thời gian nộp, sau đó chọn top 10% ở cuối bảng này là những sinh viên học đối phó. Thời điểm t_2 sẽ là thời điểm nộp bài lần đầu tiên của sinh viên đầu tiên trong nhóm này.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Sử dụng hàm *order()* để sắp xếp sinh viên theo thứ tự tăng dần thời gian nộp bài, sau đó sử dụng hàm *match()* để lấy danh sách những lần đầu mỗi MSSV xuất hiện (ứng với lần nộp đầu tiên của mỗi sinh viên).

```
the_lazy_as <- actual_data[order(actual_data$Start$year,
actual_data$Start$month, actual_data$Start$day,
actual_data$Start$hour, actual_data$Start$minute), ]
the_lazy_as <- the_lazy_as[match(unique(the_lazy_as$ID),
the_lazy_as$ID), ]
```

  - Lấy 10% số lượng sinh viên ở cuối bảng danh sách sau khi đã lọc, sau đó lấy giá trị thời gian nộp bài đầu tiên của sinh viên đầu tiên trong nhóm sinh viên trên.

```
the_lazy_as <- the_lazy_as[(nrow(the_lazy_as) - (nrow(the_lazy_as) %/%
10)):nrow(the_lazy_as), ]
```

- Kết quả:
  - Thời gian t_2 phù hợp ứng với mỗi file:

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	t_2 = 12:01 ngày 24/4/2020
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	t_2 = 18:16 ngày 27/4/2020
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	t_2 = 19:29 ngày 11/5/2020
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	t_2 = 20:19 ngày 12/5/2020

b) Xác định số lượng sinh viên học đối phó.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Sau khi đã lấy 10% sinh viên ở cuối bảng, ta tính số lượng sinh viên của nhóm này.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *nrow()* để tính số lượng sinh viên của bảng sau khi đã lấy 10% số lượng sinh viên cuối danh sách.
    ```
    print(cat("The number of lazy students:", nrow(the_lazy_as), "\n"))
    ```
- Kết quả:
  - Số lượng sinh viên học đối phó với mỗi file:

    "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx" 35 sinh viên
    "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx" 35 sinh viên
    "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx" 29 sinh viên
    "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx" 27 sinh viên

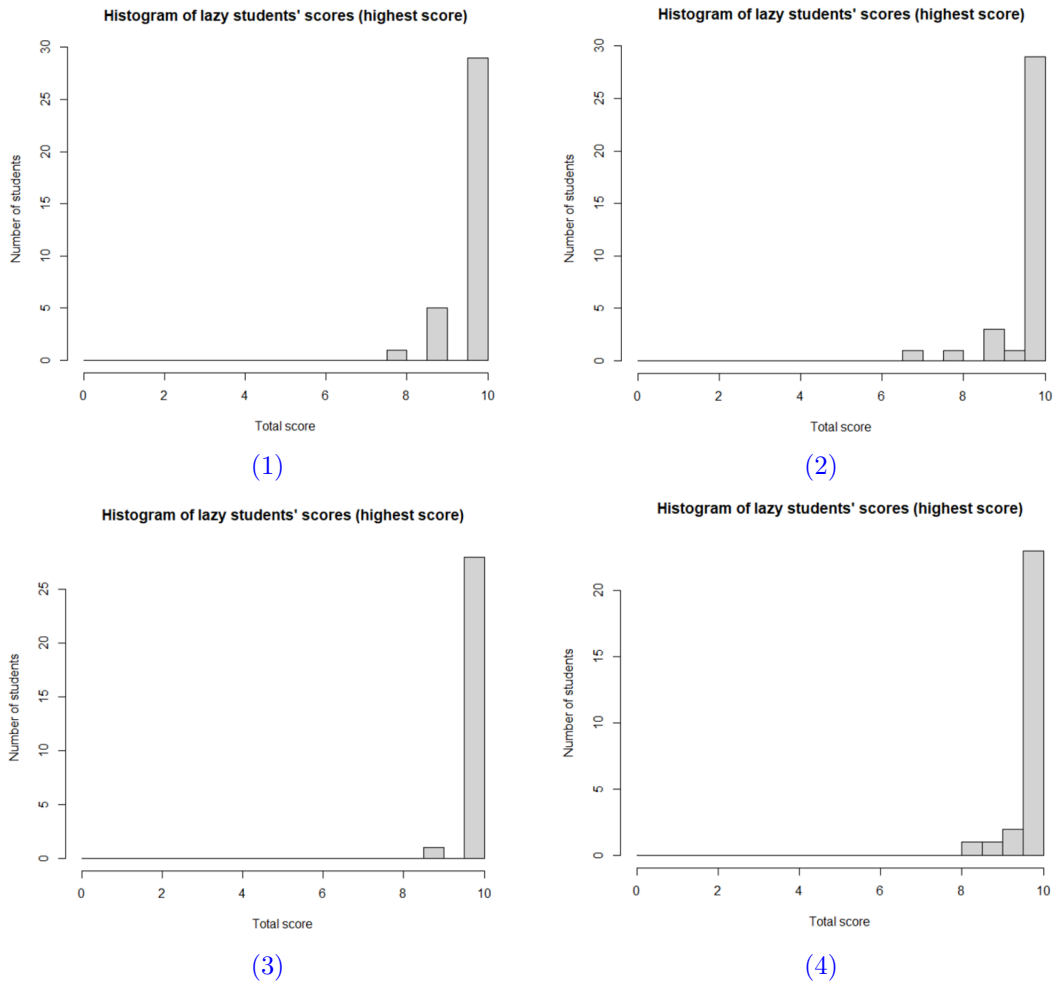
c) Xác định phổ điểm của các sinh viên học đối phó.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Dựa vào bảng điểm của các sinh viên học đối phó, ta vẽ phổ điểm và khảo sát.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm `hist()` để vẽ phổ điểm của nhóm sinh viên học đối phó, điểm của mỗi sinh viên là điểm cao nhất trong các lần nộp bài.
    ```
    hist(the_lazy_des$Total, main = "Histogram of lazy students' scores
    (highest score)", xlab = "Total score", ylab = "Number of students",
    xlim = c(0,10), breaks = ((0:20)*0.5))
    ```
- Biểu đồ:



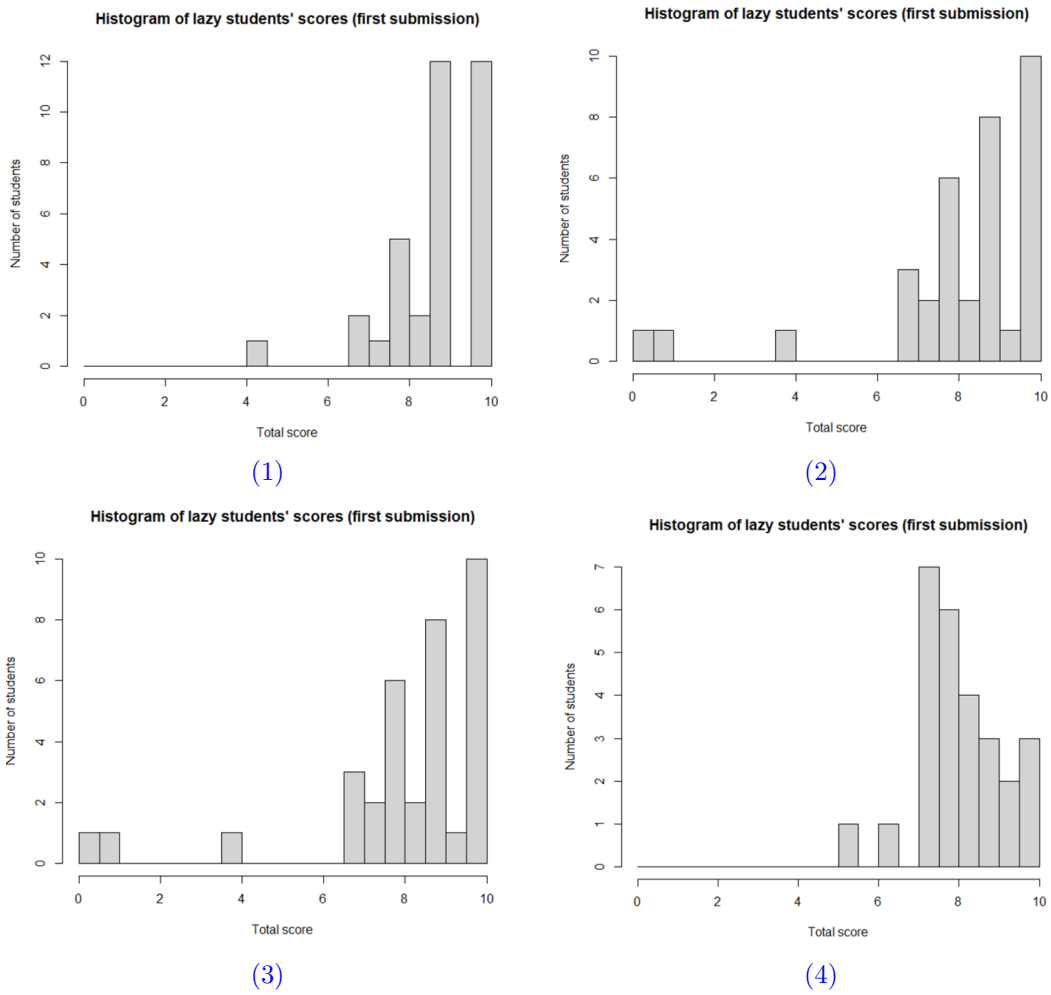
(1) (2)

(3) (4)

Hình 7.1: Phổ điểm của các sinh viên học đối phó dựa trên điểm cao nhất mỗi sinh viên đạt được

(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

– Tuy nhiên, cách tiếp cận này chưa thật sự hợp lý. Vì đây là bài tập có thể làm nhiều lần, các sinh viên có thể rút kinh nghiệm cho các lần làm bài tiếp theo. Vậy nên, khi khảo sát phổ điểm trên nhóm sinh viên học đối phó, ta chỉ nên dựa vào điểm của lần nộp đầu tiên.



(1) (2)

(3) (4)

Hình 7.1: Phổ điểm của các sinh viên học đối phó dựa trên điểm lần nộp bài đầu tiên mỗi sinh viên đạt được

(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

– Qua đó, ta thấy những phổ điểm sau hợp lý hơn. Phổ điểm lúc này có sự phân bố nhiều hơn ở các mức điểm thấp hơn 9, phù hợp với việc các sinh viên học đối phó thường không đạt điểm cao như các sinh viên chăm chỉ hơn.

Bài 9: Nhóm câu hỏi liên quan đến sinh viên thông minh

Sinh viên **thông minh** là sinh viên có kết quả tốt (điểm lớn hơn k) ngay từ n lần nộp đầu tiên.

a) Hãy xác định giá trị k và n phù hợp.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  – Ta có thể xác định điểm k bằng cách nhìn vào điểm số thấp nhất mà 20% sinh viên đứng ở top đầu đạt được, và số n là số lần nộp bài trung bình của tất cả sinh viên.
  – Đối với n: Ta thống kê tần số nộp bài của từng sinh viên, sau đó tìm số lần nộp bài trung bình, lấy giá trị trần (ceiling) làm số lần nộp n.
  – Đối với k: Ta lấy với mỗi sinh viên điểm của *lần nộp bài có kết quả tốt nhất*, sau đó lọc ra top 20% sinh viên với số điểm cao nhất và xem điểm số của sinh viên ở cuối danh sách này là k.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta lập một data frame chứa những sinh viên đã nộp bài, sau đó lập một bảng tần số của MSSV bằng hàm *table()* (dưới dạng một data frame), sau đó tính trung bình bằng hàm *mean()* và lấy giá trị trần bằng hàm *floor()*:

    ```
    actual.data <- subset(data, Status == "Done") num.of.sub <-
    data.frame(table(actual.data$ID))
    avg.num.of.sub <- floor(mean(num.of.sub$Freq))
    ```

  - Lúc này n chính là *avg.num.of.sub*.
  - Ta lại tạo một dataframe mới từ *actual.data* chỉ chứa Mã số ID và tổng điểm của lần nộp đó. Sau đó sắp xếp thứ tự theo Mã số ID và theo số điểm của lần nộp. Khi đó điểm cao nhất của lần nộp sẽ được đẩy lên trên.
  - Lúc này, ta chỉ cần loại bỏ những hàng trùng nhau về Mã số ID bằng hàm *match()*.
  - Để ý dấu "-" trong hàm *order()* là để sắp xếp theo thứ tự giảm dần.

    ```
    the.elite <- actual.data[order(actual.data$ID, -actual.data$Total), ]
    the.elite <- the.elite[match(unique(the.elite$ID), the.elite$ID), ]
    the.elite <- the.elite[order(-the.elite$Total), ]
    ```

  - Lấy phần tử nhỏ nhất từ 20% phần tử đầu tiên của cột Total trong *the.elite* ta được k, lưu trong biến *required.total*.

    ```
    required.total <- min(the.elite$Total[1 : (length(the.elite$Total) /
    5)])
    ```

- Kết quả:
  - Giá trị k và n đối với mỗi file:

    `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"` $k = 10, n = 1$
    `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"` $k = 10, n = 1$
    `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"` $k = 10, n = 1$
    `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"` $k = 10, n = 1$

b) Xác định số lượng sinh viên thông minh.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ kết quả đã có ở câu a, ta lọc ra những sinh viên thỏa mãn điều kiện về n và k, sau đó đếm số lượng sinh viên thỏa mãn.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Với số lần nộp bài đầu tiên n đã được xác định ở câu trước, ta sẽ tạo data frame mới bằng cách loại bỏ đi những lần nộp bài sau lần nộp bài thứ n.
  - Ý tưởng chính sẽ là từ danh sách những MSSV độc nhất trong *num.of.sub*, ta dùng vòng lặp for để với những MSSV trong $num.of.sub\$Var1$, ta loại bỏ những kết quả sau lần nộp thứ 2 của sinh viên đó.
  - Công việc còn lại sẽ là lọc ra những sinh viên có đủ điểm và loại bỏ đi những sinh viên bị lặp.
  - Để loại bỏ những lần nộp sau lần nộp thứ n:

```
the.elite <- actual.data[order(actual.data$ID), ]
for (ID in the.elite$ID)
{
   freq <- 0
   itr <- 1
   while (itr <= nrow(the.elite))
   {
      if (the.elite$ID[itr] == ID)
         freq <- freq + 1
      if (freq > avg.num.of.sub)
      {
         the.elite <- the.elite[-itr, ]
         break
      }
      itr <- itr + 1
   }
}
```

  - Để lọc lại những sinh viên đủ điểm:

```
the.elite <- the.elite[the.elite$Total >= required.total, ]
the.elite <- the.elite[match(unique(the.elite$ID), the.elite$ID), ]
```

- Kết quả:
  - Số lượng sinh viên thông minh trong mỗi file

"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	91 sinh viên
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	77 sinh viên
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	191 sinh viên
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	29 sinh viên

c) Xác định phổ điểm của các sinh viên thông minh.

Kiến thức chuẩn bị

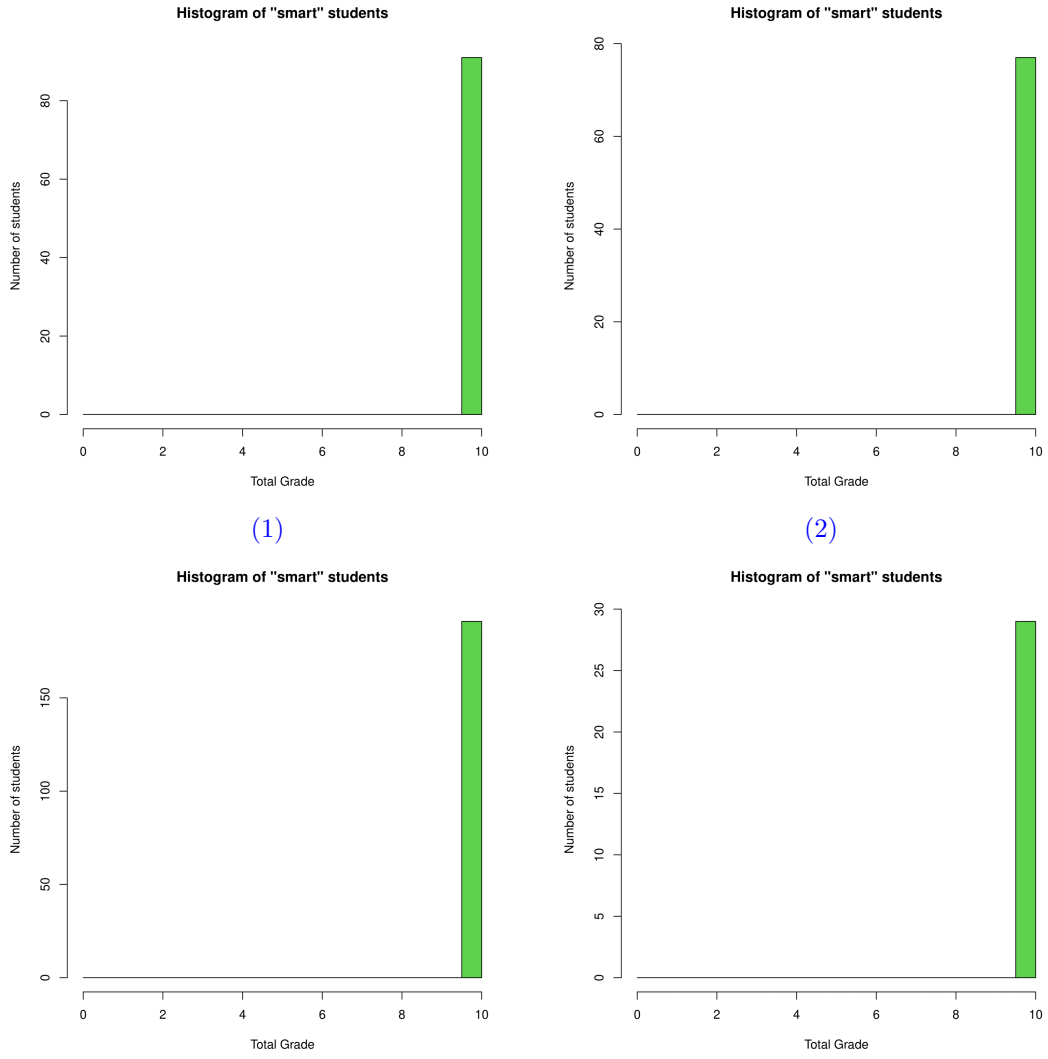
- Cách giải truyền thống:
  - Để vẽ phổ điểm cho các sinh viên thông minh, ta cần thống kê tần số của số điểm đạt được sau đó dùng biểu đồ cột để biểu diễn.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta dùng lệnh *hist()* để vẽ phổ điểm của những sinh viên thông minh.

```
hist(the.elite$Total, (0:20)*0.5, col = 0x87CEEB, xlab = "Total
Grade", ylab = "Number of students", main = paste("Histogram of
"smart" students"))
```

- Biểu đồ:



(1) (2)

(3) (4)

Hình 9.1: Phổ điểm của các sinh viên thông minh
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

Nhận xét: Việc phổ điểm không có dạng hình chuông, cụ thể là phổ điểm chỉ là một cột thẳng đứng, có nghĩa là đề đang chưa có độ khó để phân loại sinh viên.

Bài 10: Nhóm câu hỏi liên quan đến sinh viên chủ động

Sinh viên học **chủ động** là sinh viên thông minh hoặc sinh viên siêng năng mà có nộp bài nhiều lần để cải thiện điểm.

a) Hãy xác định các thông số phù hợp.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta xử lý tương tự bài 9 để tìm ra danh sách các sinh viên thông minh.
  - Để tìm ra các sinh viên siêng năng, ta sắp xếp dữ liệu theo chiều tăng dần thời gian nộp bài và lấy 10% đầu dữ liệu. Trong số này, lọc ra những sinh viên có số lần nộp bài thỏa mãn.

– Kết hợp hai danh sách này, ta được danh sách những sinh viên học chủ động.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Xử lý tương tự bài 9, ta tìm được các thông số về điểm và số lần nộp bài phù hợp.
  - Đối với sinh viên siêng năng, ta dùng hàm *order*() để sắp xếp dữ liệu theo chiều tăng dần thời gian nộp, sau đó lấy 10% đầu danh sách để lấy thông số thời gian nộp bài phù hợp.
    ```
    the.hard <- actual.data[order(actual.data$Start$year,
    actual.data$Finish$month, actual.data$Finish$day,
    actual.data$Finish$hour, actual.data$Finish$minute), ]
    the.hard <- the.hard[(1:(nrow(the.hard) %/% 10)), ]
    required.time <- the.hard[nrow(the.hard), "Finish"]
    ```
  - Ta làm tương tự bài 9 để tìm số lần nộp bài thích hợp, lần này sử dụng hàm *ceiling*() thay vì *floor*().
    ```
    req.num.of.sub <- ceiling(mean(num.of.sub$Freq))
    ```
  - Sử dụng hai hàm *subset*() và *rbind*(), ta tìm được danh sách những sinh viên học chủ động khi kết hợp các điều kiện trên.
- Kết quả:
  - Các thông số phù hợp trong mỗi file

`"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`	Thông minh: $k = 10, n = 1$ Siêng năng: $t_1 =$ 19:48 ngày 25/03/2020 Số lần nộp tối thiểu: 2
`"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`	Thông minh: $k = 10, n = 1$ Siêng năng: $t_1 =$ 22:46 ngày 27/03/2020 Số lần nộp tối thiểu: 2
`"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`	Thông minh: $k = 10, n = 1$ Siêng năng: $t_1 =$ 20:48 ngày 13/04/2020 Số lần nộp tối thiểu: 2
`"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`	Thông minh: $k = 10, n = 1$ Siêng năng: $t_1 =$ 13:34 ngày 17/04/2020 Số lần nộp tối thiểu: 2

b) Xác định số lượng sinh viên biết cách học chủ động.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách thu được, ta đếm số lượng sinh viên thuộc nhóm này.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *nrow*() để lấy số lượng sinh viên trong nhóm này.
    ```
    print(cat("The number of active students:", nrow(final.data), "\n"))
    ```
- Kết quả:
  - Số lượng sinh viên chủ động trong mỗi file:

`"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`	111 sinh viên
`"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`	101 sinh viên
`"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`	197 sinh viên
`"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`	49 sinh viên

c) Xác định phổ điểm của các sinh viên biết cách học chủ động.

Kiến thức chuẩn bị

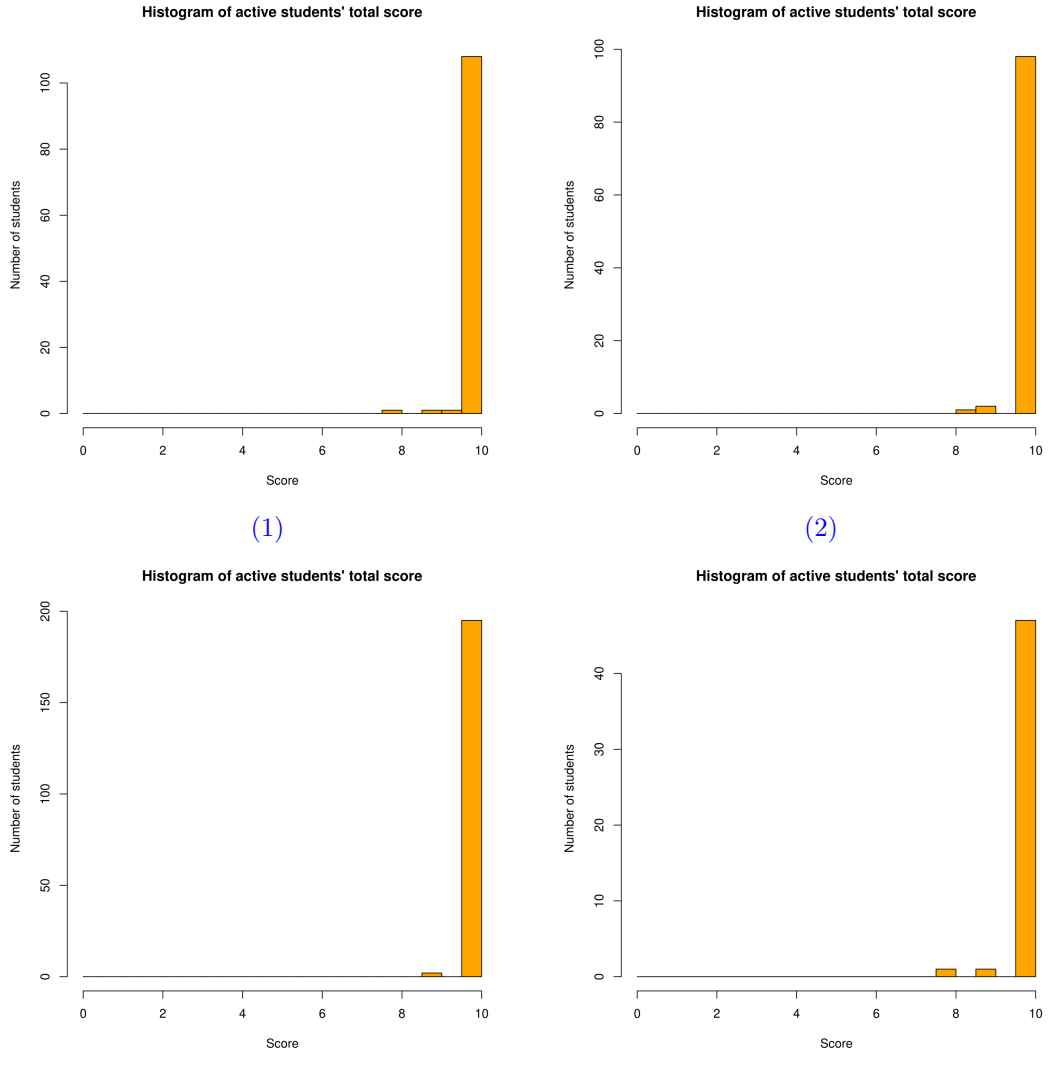
- Cách giải truyền thống:
  - Từ danh sách thu được, ta vẽ phổ điểm của các sinh viên thuộc nhóm này.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *hist*() để vẽ phổ điểm của các sinh viên trong nhóm này.

```
hist(final.data$Total, main = "Histogram of active students' total
score", xlab = "Score", ylab = "Number of students", col = "orange",
xlim = c(0,10), breaks = ((0:20)*0.5))
```

- Biểu đồ:



Hình 10.1: Phổ điểm của các sinh viên học chủ động
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

Bài 11: Tổng hợp các nhóm sinh viên

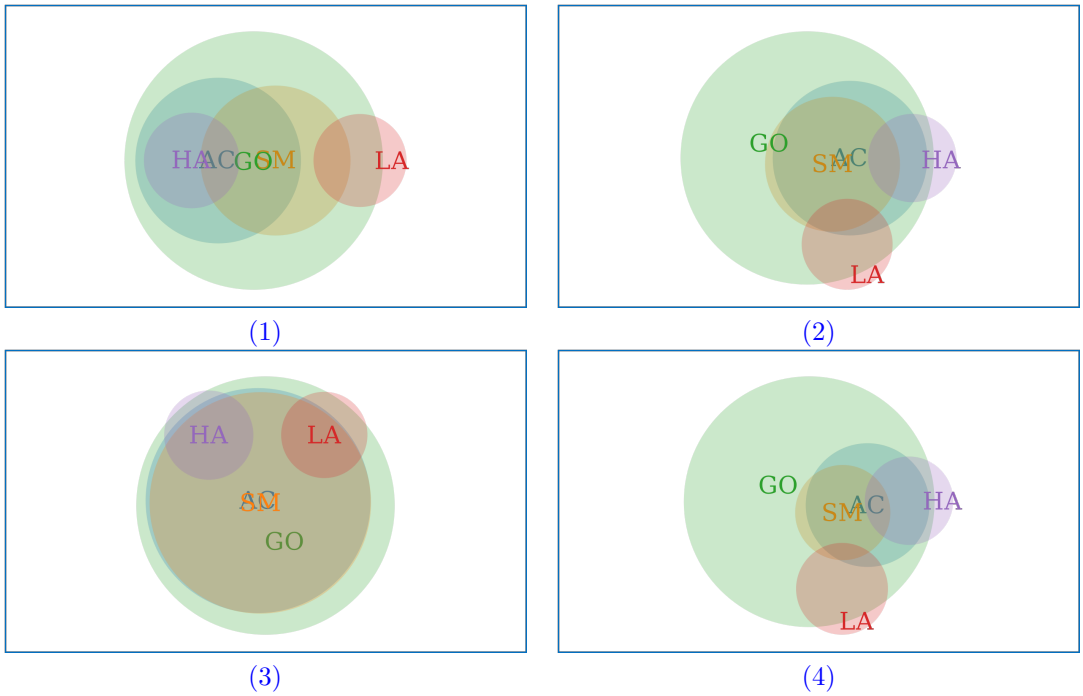
Xác định phần giao của các loại sinh viên đánh giá ở trên (từ câu 6 đến câu 10 và vẽ biểu đồ thống kê minh họa.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách các nhóm sinh viên, ta tìm giao của các nhóm này và vẽ giản đồ Venn tương ứng.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Sử dụng các hàm *subset()* và *nrow()* để đếm số lượng sinh viên của mỗi nhóm và phần giao của các nhóm.
- Biểu đồ:



Hình 11.1: Giản đồ Venn biểu diễn các tập sinh viên
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`
Các nhóm sinh viên: AC: chủ động; SM: thông minh; GO: giỏi; LA: đối phó; HA: siêng năng

Bài 12: Điểm thưởng

Nhóm có thể tự đề xuất và bổ sung thêm những giá trị thống kê hữu ích đối với tập dữ liệu điểm này.

Một số nhận xét:

Trong các vấn đề trên, giá trị *Thời gian làm bài* của các sinh viên chưa được đưa vào khảo sát. Đây cũng là một giá trị có thể cho biết tình hình học và hiểu bài của sinh viên. Do đó, nhóm xin được phép bổ sung một số thống kê liên quan đến giá trị này.

a) Thời gian làm bài trung bình của tổng các bài nộp, của các bài nộp lần đầu và của các bài nộp lần cuối.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Để tính giá trị trung bình, ta lấy tổng số thời gian làm bài của các bài nộp chia cho số bài nộp.
  - Lập danh sách các bài nộp lần đầu và lần cuối, ta tính được giá trị trung bình thời gian làm bài của các bài nộp lần đầu và lần cuối.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Ta sử dụng hàm *mean()* để tính giá trị trung bình cho thời gian làm bài.
  - Để lập danh sách các bài nộp lần đầu (cuối), ta sử dụng hàm *order()* để sắp xếp danh sách theo chiều tăng dần thời gian bắt đầu làm bài, sau đó sử dụng hàm *match()* để lấy các hàng có các ID xuất hiện lần đầu tiên, thu được danh sách các bài nộp lần đầu (cuối) của mỗi sinh viên. Sau đó, ta thực hiện tính trung bình trên tập dữ liệu thu được.

```
first_submit <- data[order(data$Start$year, data$Start$month,
data$Start$day, data$Start$hour, data$Start$minute),]
first_submit <- first_submit[match((unique(first_submit$ID)),
first_submit$ID),]
mean(first_submit$Duration)
```

- Kết quả:
  - Thời gian làm bài trung bình (tính theo s) của mỗi file:

	Tất cả	Lần nộp đầu	Lần nộp cuối
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	296.6886	435.2471	217.1047
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	449.4133	692.5131	342.9767
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	264.9887	318.375	261.15
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	300.4889	465.2231	215.3538

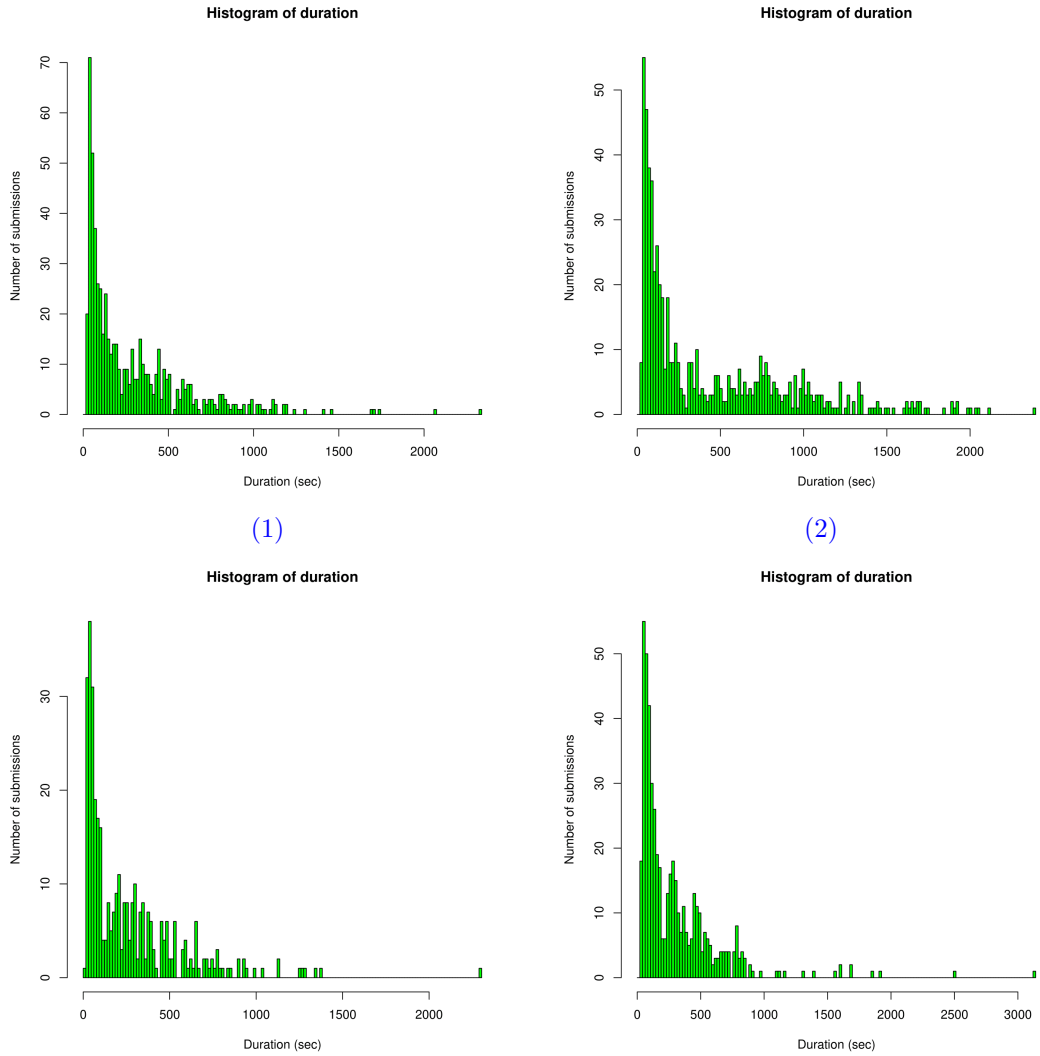
b) Lập phổ theo thời gian làm bài của tổng các lần nộp bài, các bài nộp lần đầu và các bài nộp lần cuối.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Từ danh sách đã lập, ta vẽ phổ theo thời gian làm bài ứng với các tập dữ liệu.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Sử dụng lệnh `hist()` để vẽ các phổ thời gian trên các tập dữ liệu tương ứng.
- Biểu đồ:



(1)

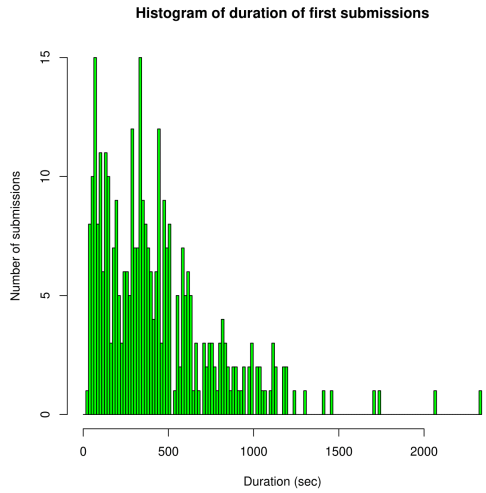
(2)

(3)

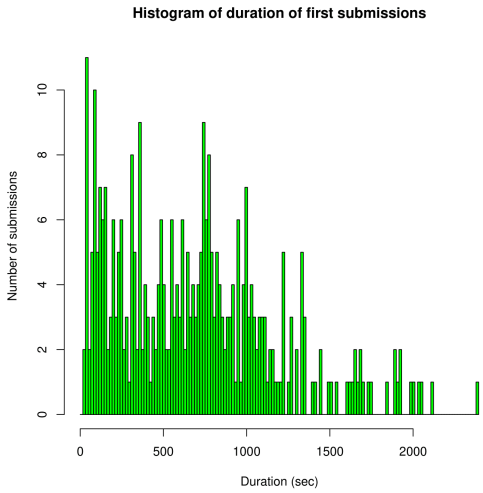
(4)

Hình 12.1: Phổ thời gian làm bài của tất cả bài làm

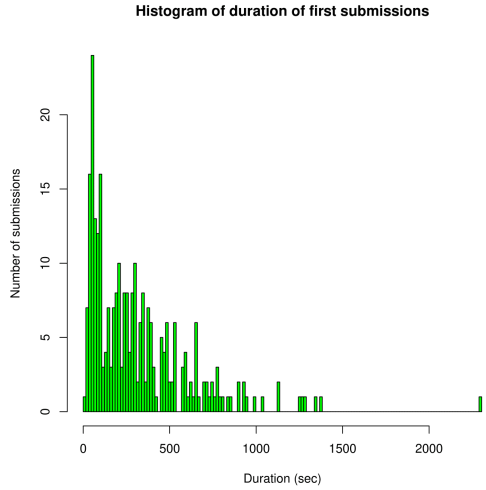
(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"



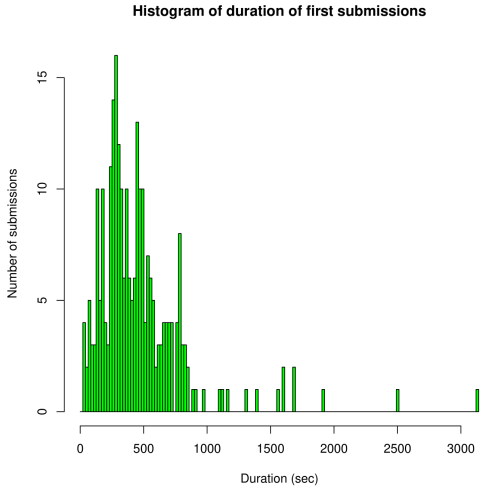
(1)



(2)



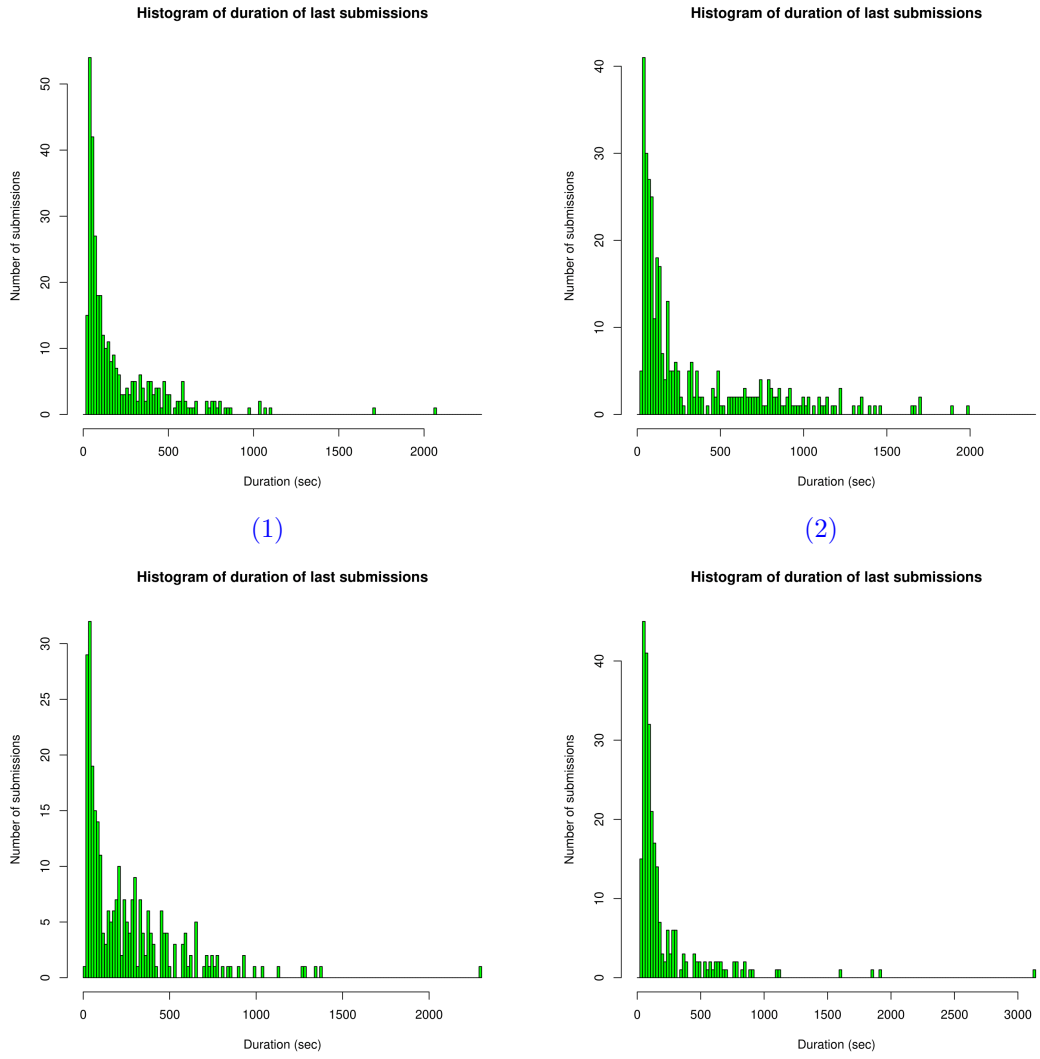
(3)



(4)

Hình 12.2: Phổ thời gian làm bài của các bài nộp lần đầu

(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"



(1) (2)

(3) (4)

Hình 12.3: Phổ thời gian làm bài của các bài nộp lần cuối
(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"

c) Thời gian làm bài trung bình của các bài nộp lần đầu và của các bài nộp lần cuối của các sinh viên có số lần nộp bài từ 2 trở lên.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta lập danh sách sinh viên có số lần nộp bài từ 2 trở lên, sau đó tính giá trị trung bình cộng thời gian làm bài các lần đầu và lần cuối.

Hiện thực trên R

- Ý tưởng thực hiện:
  - Sử dụng tập dữ liệu đã sắp xếp tăng dần theo thời gian bắt đầu làm bài, ta loại bỏ đi những lần nộp bài đầu tiên, sau đó sắp xếp tập dữ liệu giảm dần theo thời gian bắt đầu làm bài và lấy những lần ID xuất hiện đầu tiên, ta thu được danh sách những lần nộp cuối của các sinh viên có số lần nộp bài từ 2 lần trở lên.

  - Từ danh sách các bài nộp lần đầu, ta trích những bài nộp có số ID nằm trong tập dữ liệu vừa lập, ta thu được danh sách những lần nộp đầu của các sinh viên có số lần nộp bài từ 2 lần trở lên.
  - Ta tính toán giá trị trung bình thời gian làm bài trên hai tập dữ liệu này.
- Kết quả:
  - Thời gian làm bài trung bình (tính theo s) đối với các sinh viên có số lần nộp bài từ 2 trở lên của mỗi file:

	Lần nộp đầu	Lần nộp cuối
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	470.32	95.115
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	707.831	144.9624
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	288.2	60.65714
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	438.1214	123.8204

Nhận xét: Đối với các sinh viên có số lần làm bài từ 2 trở lên, ta nhận thấy rõ sự chênh lệch thời gian làm bài lần đầu và lần cuối. Với cùng một đề, khi số lần làm bài tăng, thời gian làm bài của mỗi lần giảm rõ rệt.

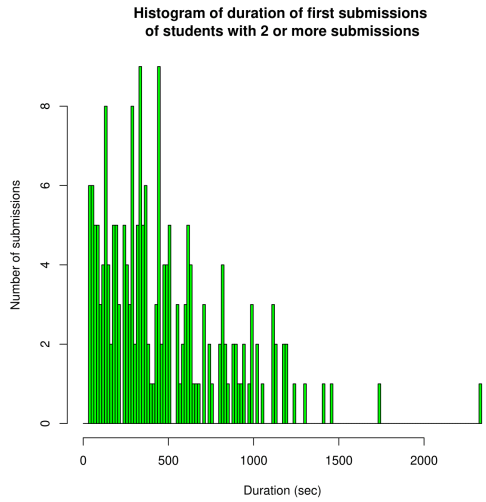
d) Lập phổ theo thời gian làm bài của các bài nộp lần đầu và các bài nộp lần cuối của các sinh viên có số lần nộp từ 2 trở lên.

Kiến thức chuẩn bị

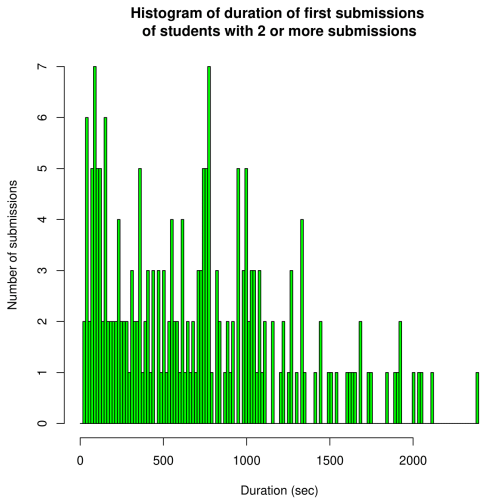
- Cách giải truyền thống:
  - Từ danh sách đã lập, ta vẽ phổ theo thời gian làm bài ứng với các tập dữ liệu.

Hiện thực trên R

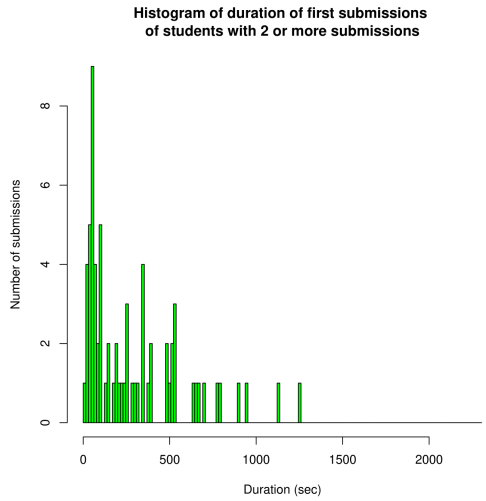
- Ý tưởng thực hiện:
  - Sử dụng lệnh *hist()* để vẽ các phổ thời gian trên các tập dữ liệu tương ứng.
- Biểu đồ:



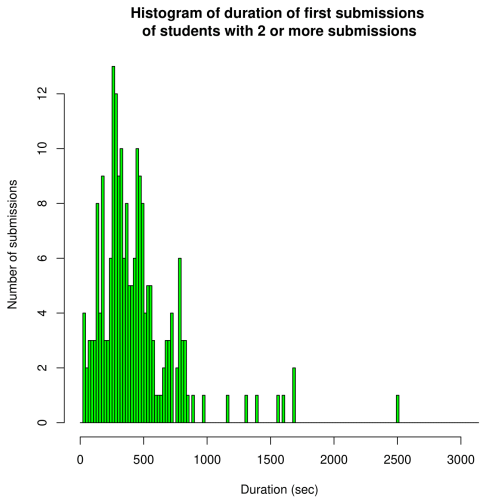
(1)



(2)



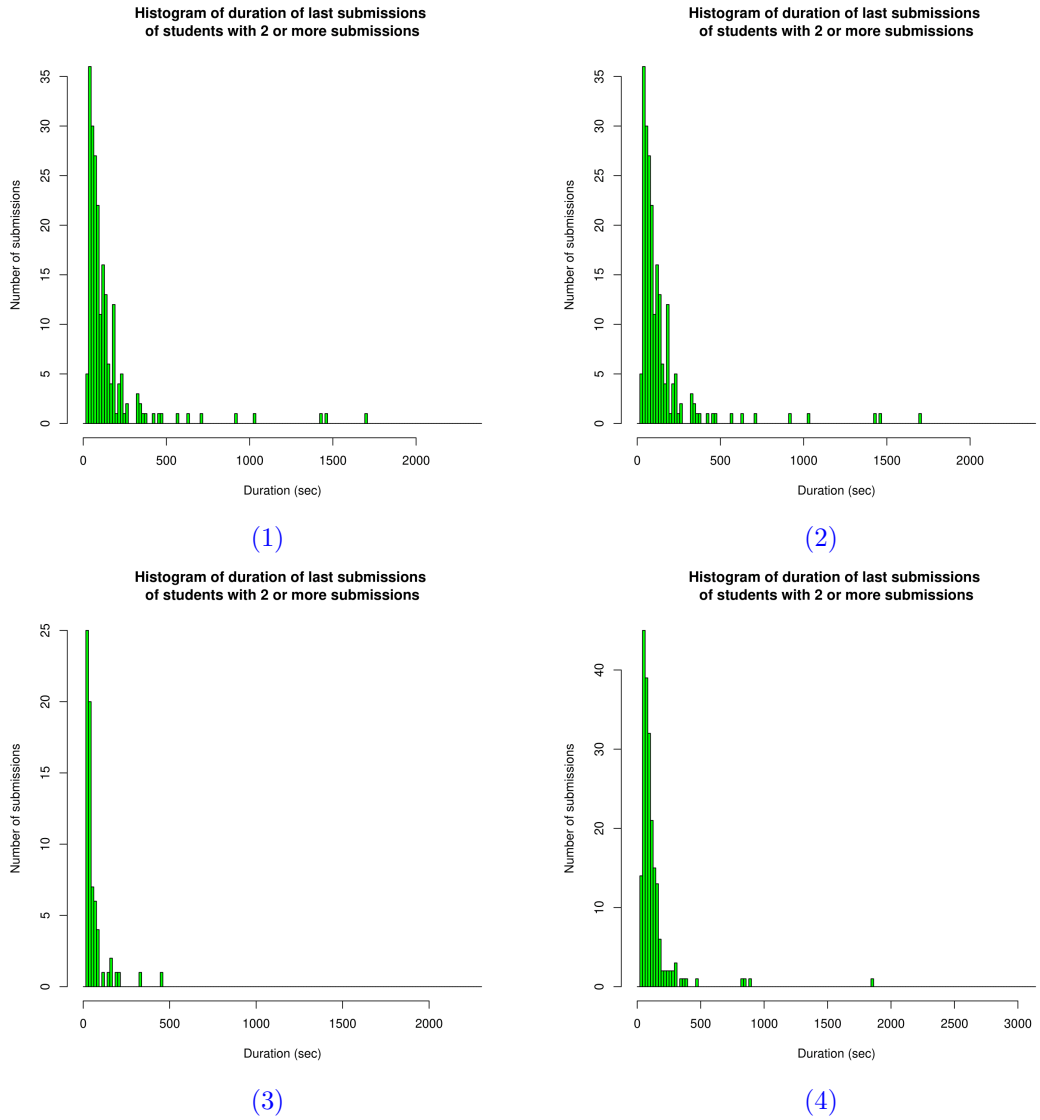
(3)



(4)

Hình 12.4: Phổ thời gian làm bài của các lần nộp đầu đối với các sinh viên có số lần nộp từ 2 trở lên

(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"



(1) (2)

(3) (4)

Hình 12.5: Phổ thời gian làm bài của các lần nộp cuối đối với các sinh viên có số lần nộp từ 2 trở lên

(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

e) Nhận xét tương quan giữa điểm đạt được và thời gian làm bài của các lần nộp bài đầu và cuối của các sinh viên có số lần nộp bài từ 2 trở lên.

Kiến thức chuẩn bị

- Cách giải truyền thống:
  - Ta tính giá trị trung bình chêch lệch thời gian làm bài và điểm đạt được giữa lần đầu và lần cuối của các sinh viên có số lần nộp bài từ 2 trở lên. Vẽ phổ điểm tương ứng để rút ra nhận xét.

Hiện thực trên R

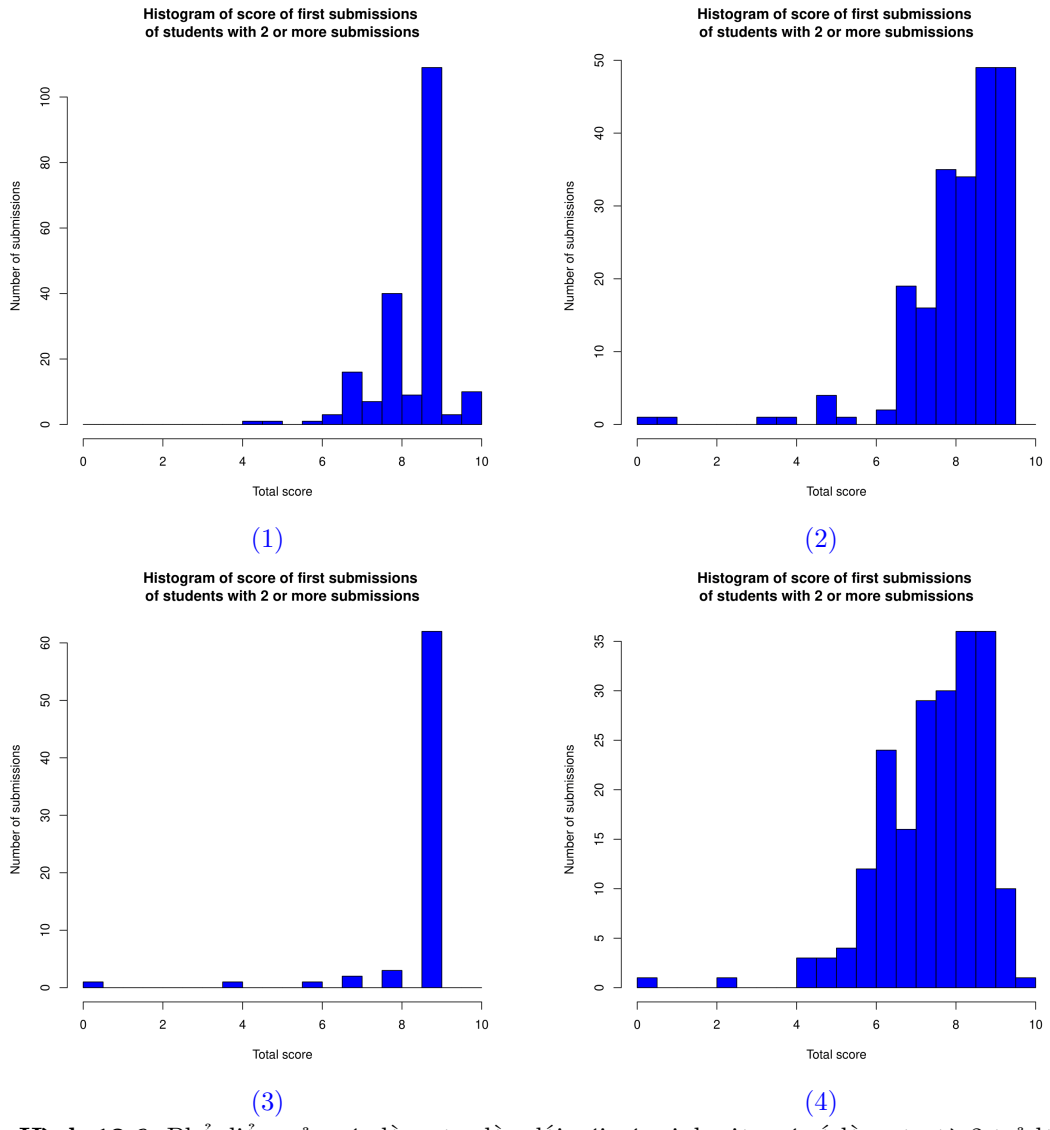
- Ý tưởng thực hiện:
  - Từ các tập dữ liệu có sẵn, ta lập được danh sách các sinh viên có số lần nộp bài từ 2 trở lên kèm theo chêch lệch thời gian làm bài và điểm đạt được giữa lần đầu và lần cuối của mỗi sinh viên.

– Ta tính giá trị trung bình của chêch lệch thời gian làm bài và điểm đạt được giữa lần đầu và lần cuối, sau đó vẽ phổ điểm của các lần nộp bài đầu và cuối.

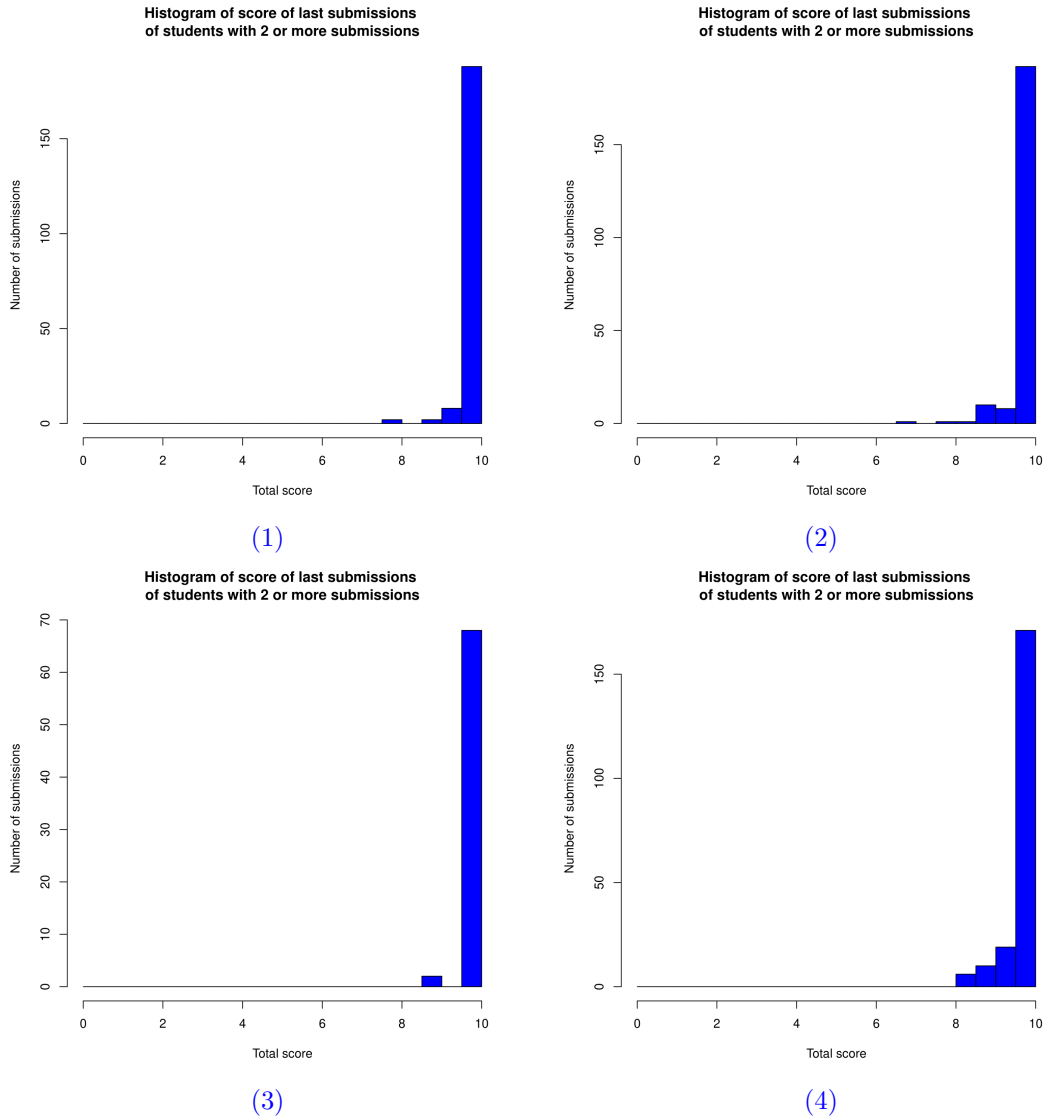
- Kết quả:
  – Các giá trị chênh lệch đối với các sinh viên có số lần nộp bài từ 2 trở lên của mỗi file:

	Thời gian làm bài (s)	Điểm đạt được
"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"	375.205	1.50525
"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"	562.8685	1.565728
"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"	227.5429	1.314286
"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"	314.301	2.165049

- Biểu đồ:



Hình 12.6: Phổ điểm của các lần nộp đầu đối với các sinh viên có số lần nộp từ 2 trở lên
(1) "CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"
(2) "CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"
(3) "CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"
(4) "CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"



Hình 12.7: Phổ điểm của các lần nộp cuối đối với các sinh viên có số lần nộp từ 2 trở lên
(1) `"CO1007_TV_HK192-Quiz 1.4-điểm.xlsx"`
(2) `"CO1007_TV_HK192-Quiz 1.5-điểm.xlsx"`
(3) `"CO1007_TV_HK192-Quiz 3.3-điểm.xlsx"`
(4) `"CO1007_TV_HK192-Quiz 4.2-điểm.xlsx"`

Nhận xét: Nhìn vào phổ điểm, ta thấy rõ phổ điểm của lần nộp cuối nhọn hơn và phân bố gần giá trị cực đại, trong khi phổ điểm của lần nộp đầu lại có hình dạng chuông (trừ quiz 3.3 hình dạng này không thể hiện rõ) và phân bố trên khoảng điểm rộng. Như vậy, so với lần nộp bài đầu, các bài nộp lần cuối có điểm cải thiện hơn rất nhiều.

4.4 Source Code

Source code của từng bài được đính kèm trong thư mục Source Code R, mỗi file tương ứng với một bài được đặt tên như sau: bài thứ i có tên file là `Ex`i`.R`.

Tài liệu

[Dal] Dalgaard, P. *Introductory Statistics with R.* Springer 2008.

[K-Z] Kenett, R. S. and Zacks, S. *Modern Industrial Statistics: with applications in R, MINITAB and JMP,* 2nd ed., John Wiley and Sons, 2014.

[Ker] Kerns, G. J. *Introduction to Probability and Statistics Using R,* 2nd ed., CRC 2015.