

Bài giảng 1:
Giới thiệu phần mềm R

ThS. Lý Sel

Faculty of Mathematics-Statistics

Ton Duc Thang University

- R có nguồn gốc từ S.
- R đầu tiên do Ross Ihaka và Robert Gentleman (ĐH Auckland, New Zealand) viết vào thập niên 1990s.
- R là "statistical and graphical programming language".
- Từ 1997: international "R-core", 15 người.

Tại sao R?

- Sử dụng miễn phí.
- Chạy trên Windows, Unix, MacOS.
- Rất nhiều phương pháp phân tích.
- Nhiều phương pháp "advanced" không có trong các phần mềm khác.
- Biểu đồ tuyệt vời.

Hay và yếu

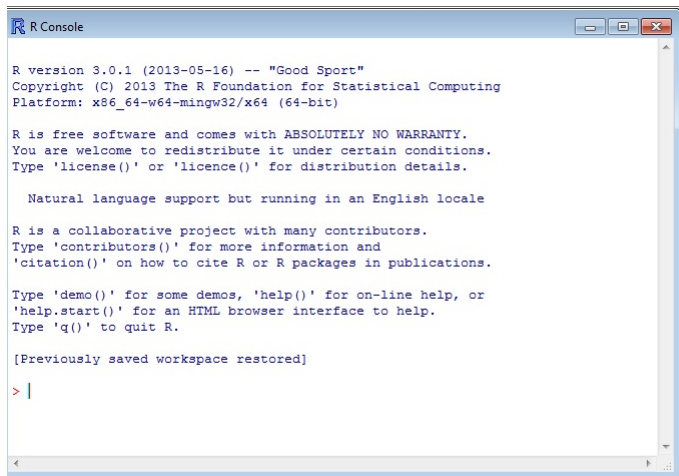
- Hay
 - Miễn phí.
 - Nhiều packages chuyên dụng.
 - Mã nguồn mở.
- Yếu
 - Thuật ngữ khó hiểu.
 - Dùng lệnh.
 - Ký hiệu.

R có thể làm được gì?

- R là một ngôn ngữ phân tích thống kê.
- Có thể thực hiện tất cả các mô hình phân tích.
- Mô phỏng (simulation).
- Vẽ đồ thị và biểu đồ (rất đẹp!!).
- Lập trình cho phương pháp mới.
- Khác...

- Truy cập: <http://cran.r-project.org>.
- Chọn Download R for windows.
- Chọn base.
- Chọn Download R ...
- Cài đặt thông thường trên máy tính (cứ Next và OK).

Cửa sổ làm việc của R



```
R Console

R version 3.0.1 (2013-05-16) -- "Good Sport"
Copyright (C) 2013 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Previously saved workspace restored]

> |
```

Object

- Mỗi object phải có tên.
- Tên có thể là chữ thường, chữ hoa, số, và ký hiệu dấu chấm hoặc dấu gạch ngang dưới "_".
Chú ý R phân biệt chữ hoa và chữ thường.
- Mỗi object dùng để lưu trữ các kết quả tính toán thông qua dấu
=
hay
<-

Ví dụ:

```
luong.thang = 8  
luongthang1 <- 2*4
```


- Cài đặt packages mới

```
install.packages("psych")
```

```
install.packages(c("psych", "Hmisc"))
```

- Load các packages đã cài

```
library(psych)
```

hoặc

```
require(psych)
```

- R là một trong những phần mềm mạnh trong khoa học thống kê.
- Hoàn toàn miễn phí.
- Sử dụng rộng rãi trên thế giới.

Bài giảng 2:
Nhập và xuất dữ liệu

ThS. Lý Sel

Faculty of Mathematics-Statistics

Ton Duc Thang University

Ví dụ: ta có bảng số liệu thu nhập sau (triệu đồng):

STT	Thu nhập	Giới tính	Trình độ
1	2.1	nữ	khác
2	2.5	nam	khác
3	2.0	nam	khác
4	4.1	nữ	phổ thông
5	4.2	nam	phổ thông
6	4.5	nữ	phổ thông
7	4.0	nam	phổ thông
8	11.0	nữ	ĐH, sau ĐH
9	8.0	nam	ĐH, sau ĐH
10	9.0	nữ	ĐH, sau ĐH

Có 3 cách thông dụng:

- Nhập trực tiếp sử dụng hàm: `c()` và `data.frame()`.
- Nhập trực tiếp sử dụng cửa sổ: `edit()`.
- Đọc bất kỳ file dữ liệu sẵn có như excel, spss, text,...

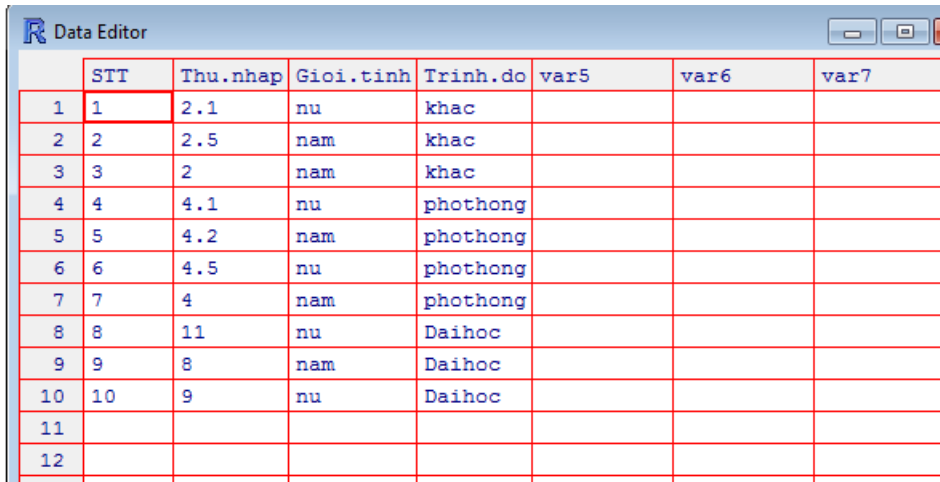
- Nhập trực tiếp sử dụng hàm: `c()` và `data.frame()`:

```
> STT=c(1:10)
> Thu.nhap=c(2.1,2.5,2.0, 4.1,4.2,4.5,4.0,11.0,8.0,9.0)
> Gioi.tinh=c("nu","nam","nam","nu","nam","nu","nam",
"nu","nam","nu")
> Trinh.do=c("khac","khac","khac","phothong","phothong",
"phothong","phothong","Daihoc","Daihoc","Daihoc")
> income=data.frame(STT, Thu.nhap,Gioi.tinh,Trinh.do)
```

Nhập dữ liệu...

- Nhập trực tiếp sử dụng cửa sổ: `edit()`.

```
> income2=edit(data.frame())
```



R Data Editor

	STT	Thu.nhap	Gioi.tinh	Trinh.do	var5	var6	var7
1	1	2.1	nu	khac			
2	2	2.5	nam	khac			
3	3	2	nam	khac			
4	4	4.1	nu	phothong			
5	5	4.2	nam	phothong			
6	6	4.5	nu	phothong			
7	7	4	nam	phothong			
8	8	11	nu	Daihoc			
9	9	8	nam	Daihoc			
10	10	9	nu	Daihoc			
11							
12							

Nhập dữ liệu...

- Đọc file dữ liệu sẵn có như excel, spss, text,...

- * **Đọc dữ liệu từ file excel:** 2 bước

- + Bước 1: Chuyển file excel định dạng "*.xls" sang định dạng "*.csv" bằng cách chọn Save as và chọn file type "*.csv" (CSV comma delimited).

- + Bước 2: Dùng lệnh `read.csv()`

Ví dụ: Đọc dữ liệu "whiteside insulation.csv" trong thư mục "Thuchanh_R" ở ổ đĩa C:

```
> whiteside = read.csv("C:/Thuchanh_R/whiteside  
insulation.csv",header=T)
```

Hoặc

```
> whiteside = read.csv(file.choose(),header=T)
```

Sau đó, chọn nơi lưu trữ file và tên file cần đọc.

* Đọc dữ liệu từ các file khác:

Sử dụng package "foreign":

`library(foreign)` hoặc `require(foreign)`

- Text file: `read.table()`
- SPSS file: `read.spss()`
- STATA file: `read.stata()`
-

* Lưu dữ liệu thành R data (*.rda):

```
setwd("C:/Thuchanh_R") # chỉ nơi lưu trữ
```

```
save(income, file="thunhap.rda") # đặt tên file thunhap.rda.
```

* Xuất dữ liệu thành các file khác: write.table()

- Text file:

```
write.table(income, "C:/Thuchanh_R/thunhap.txt", sep="\t",  
row.names=F)
```

- Excel file:

```
write.table(income, "C:/Thuchanh_R/thunhap.xls", sep="\t",  
row.names=F)
```

- SPSS file:

```
write.table(income, "C:/Thuchanh_R/thunhap.sav", sep="\t",  
row.names=F)
```

-

- Đọc dữ liệu vào R có 3 cách thông dụng: 2 cách trực tiếp `c()`, `edit()` hoặc gián tiếp từ các file khác.
- R có thể đọc bất cứ file gì. File csv: `read.csv()` (khuyến khích)
- R có thể xuất sang bất cứ loại file: `write.table()`

Bài giảng 3:
Biên tập dữ liệu

ThS. Lý Sel

Faculty of Mathematics-Statistics

Ton Duc Thang University

Phép toán số học

+	addition - cộng
-	subtraction - trừ
*	multiplication - nhân
/	division - chia
\wedge hoặc **	exponential - lũy thừa
$x \% \% y$	modulus ($x \bmod y$) - chia lấy phần dư. Vd: $7 \% \% 3 = 1$.
$x \% / \% y$	integer division - chia lấy phần nguyên. Vd: $7 \% / \% 3 = 2$.

Phép toán logic

$<$	less than - nhỏ hơn
$<=$	less than or equal to - nhỏ hơn hoặc bằng
$>$	greater than - lớn hơn
$>=$	greater than or equal to - lớn hoặc bằng
$==$	exactly equal to - bằng
$!x$	not x - phủ định của x
$x y$	x OR y - x hoặc y
$x \& y$	x AND y - x và y

Các hàm tính toán

<code>abs()</code>	Hàm trị tuyệt đối
<code>sqrt()</code>	Hàm căn bậc 2
<code>sin()</code> , <code>cos()</code> , <code>tan()</code>	các hàm lượng giác
<code>asin()</code> , <code>acos()</code> , <code>atan()</code>	các hàm ngược của hàm lượng giác
<code>exp()</code>	Hàm e mũ
<code>log()</code>	Hàm logarit tự nhiên (cơ số e)
<code>log10()</code>	Hàm logarit thập phân (cơ số 10)
<code>logb(x, a)</code>	Hàm logarit cơ số a của x
<code>sum()</code>	Hàm tổng
<code>prod()</code>	Hàm tích
<code>choose(n, k)</code>	Tổ hợp của n chập k
....	

Ví dụ: Trong dữ liệu whiteside, đổi đơn vị độ C sang độ F:

```
whiteside$Temp.in.F=whiteside$Temp*1.8+32
```

- Kiểm tra lại:

```
head(whiteside)
```

	Insul	Temp	Gas	Temp.in.F
1	Before	-0.8	7.2	30.56
2	Before	-0.7	6.9	30.74
3	Before	0.4	6.4	32.72
4	Before	2.5	6.0	36.50
5	Before	2.9	5.8	37.22
6	Before	3.2	5.8	37.76

Mã hóa dữ liệu

Tạo biến mới thông qua mã hóa.

* Chẳng hạn, trong dữ liệu income (thu nhập), mã hóa các giới tính lại:

```
attach(income)
income$gender[Gioi.tinh=="nam"]<-1
income$gender[Gioi.tinh=="nu"]<-0
```

* Mã hóa biến trình độ:

```
income$level[Trinh.do=="Daihoc"]<-2
income$level[Trinh.do=="phothong"]<-1
income$level[Trinh.do=="khac"]<-0
income
```

Lệnh: `order()`

- Sắp xếp theo thứ tự tăng dần: `order(x)`

```
income.increasing=income[order(Thu.nhap),]
```

- Sắp xếp theo thứ tự giảm dần: `order(-x)`

```
income.decreasing=income[order(-Thu.nhap),]
```

Một data frame xem như một matrix.

- Rút trích một số phần tử

`income[8,2]` # rút ra từ dữ liệu income phần tử ở dòng 8, cột 2

`income[8,]` # rút ra từ dữ liệu income dòng 8

`income[,2]` # rút ra từ dữ liệu income cột 2

`income[,c(2,4)]` # rút ra từ dữ liệu income cột 2 và cột 4

`income[,-1]` # rút ra toàn bộ dữ liệu income ngoại trừ cột 1

`income[,c("Thu.nhap", "Trinh.do")]` # rút ra cột thu nhập và cột trình độ

Rút trích dữ liệu...

- Rút trích một tập con dữ liệu: `subset()` hoặc `split()`

```
women=subset(income,Gioi.tinh=="nu") # điều kiện giới tính nữ
```

```
men=subset(income,Gioi.tinh=="nam") # đk giới tính nam
```

```
women.dh=subset(income,Gioi.tinh=="nu" & Trinh.do  
=="Daihoc") # đk giới tính nữ và trình độ là đại học.
```

```
men.pt=subset(income,Gioi.tinh=="nam" | Trinh.do  
=="phothong") # đk giới tính nam hoặc trình độ là phổ thông.
```

```
women.high=subset(income,Gioi.tinh=="nu" & Thu.nhap >=  
8 & Thu.nhap <= 15) # đk giới tính nữ và thu nhập từ 8 triệu đến  
15 triệu đồng.
```

- **Hợp nhất theo dòng:** Lệnh `rbind()`. Ta xét hai dữ liệu sau:

women

STT	Thu.nhap	Gioi.tinh	Trinh.do
1	2.1	nu	khac
4	4.1	nu	phothong
...

men

STT	Thu.nhap	Gioi.tinh	Trinh.do
2	2.5	nam	khac
3	2.0	nam	khac
...

```
income.new=rbind(women,men)
```

Hợp nhất dữ liệu...

- **Hợp nhất theo cột:** Lệnh `cbind()` hoặc `merge()`.

Ta xét hai dữ liệu sau:

women

STT	Thu.nhap	Gioi.tinh	Trinh.do
1	2.1	nu	khac
4	4.1	nu	phothong
...

age

STT	Gioi.tinh	Tuoi
1	nu	18
4	nu	25
...

```
women.new=merge(women,age,by=c("STT","Gioi.tinh"))
```

Chuyển đổi kiểu dữ liệu

- Chuyển sang kiểu số: `as.numeric()`
- Chuyển sang kiểu character: `as.character()`
- `as.vector()`, `as.matrix()`, `as.data.frame()`

Ví dụ:

```
id=as.character(STT)
```

Chuyển dữ liệu cột sang dòng

Lệnh: `melt()`

Packages: `reshape` hoặc `reshape2`

Ví dụ: Giả sử ta có dữ liệu `dat` sau:

```
dat
  id sex group day1 day2 day3
  1  F     1   13   15   16
  2  M     1   18   17   20
  3  F     2   22   20   19
  4  M     2   23   24   25
```


Chuyển dữ liệu cột sang dòng...

```
library(reshape2)
dat2=melt(dat,id=c("id","sex","group"), measure.vars =
c("day1", "day2","day3"))
```

dat2

	id	sex	group	variable	value
1	1	F	1	day1	13
2	2	M	1	day1	18
3	3	F	2	day1	22
4	4	M	2	day1	23
5	1	F	1	day2	15
6	2	M	1	day2	17
7	3	F	2	day2	20
8	4	M	2	day2	24
9	1	F	1	day3	16
10	2	M	1	day3	20
11	3	F	2	day3	19
12	4	M	2	day3	25

Chuyển dữ liệu dòng sang cột

Lệnh: `dcast()`

Package: `reshape2`

```
library(reshape2)
```

```
dat3=dcast(dat2,id+sex+group ~ variable)
```

`dat3`

id	sex	group	day1	day2	day3
1	F	1	13	15	16
2	M	1	18	17	20
3	F	2	22	20	19
4	M	2	23	24	25

Bài giảng 4: Làm sạch dữ liệu

ThS. Lý Sel

Faculty of Mathematics-Statistics

Ton Duc Thang University

* **Khái niệm:**

Làm sạch dữ liệu là quá trình kiểm tra, phát hiện và sửa chữa các lỗi trong bộ dữ liệu thô ban đầu.

* **Mục đích:**

- Cải tiến chất lượng dữ liệu (nhất quán và chính xác).
- Phân tích kết quả chính xác.
- Tiết kiệm thời gian, không phải thực hiện lại từ đầu.
- ...

- Lỗi do nhập liệu: sai giá trị, tên biến, thang đo, đơn vị đo, ...
- Lỗi do lưu trữ.
- Lỗi do tích hợp dữ liệu.
- Missing values: có các giá khuyết.
- Special values: có các giá trị đặc biệt như số vô cùng lớn, vô cùng bé,...
- Outliers: có các giá trị ngoại biên, vượt giới hạn (quá lớn hoặc quá nhỏ).

Data entry errors

Các lỗi do nhập liệu sai có thể phát hiện và khắc phục bằng cách cho từ 2 người nhập độc lập. Sau đó, ta kiểm tra lại hai data này khớp nhau hay không?

Ví dụ:

		age	gender			age	gender
dat1	1	18	M	dat2	1	18	M
	2	19	W		2	29	W
	3	30	M		3	30	W
	4	25	W		4	25	M
	5	28	M		5	28	M

`dat1==dat2`

→ Các vị trí chưa khớp sẽ trả về FALSE.

Data entry errors...

Ta có thể tạo một hàm kiểm tra 2 data có khớp nhau không? Nếu chưa khớp thì xuất ra các vị trí sai sót.

```
datacheck=function(dat1,dat2) {  
+ check=dat1==dat2  
+ col=names(dat1)  
+ r.num=nrow(dat1)  
+ c.num=ncol(dat1)  
+ for(i in 1:r.num)  
+ for(j in 1:c.num)if (check[i,j]==F)  
+ print(paste("Hang thu:",i,"", Cot:", col[j]","", File 1:",  
dat1[i,j]","", File 2:", dat2[i,j]))  
}  
  
datacheck(dat1,dat2)
```

Missing values

* **Missing values:** NA.

Ví dụ: Dữ liệu people có các biến age (năm), height (cm)
people

	age	gender	height
1	18	M	160
2	19	W	163
3	30	M	170
4	25	W	165
5	28	M	NA

* **Kiểm tra và phát hiện:** `is.na()` hoặc `complete.cases()`

`is.na(people)`

`complete.cases(people)`

Missing values...

* **Khắc phục:** có thể bỏ loại NA hoặc nhập lại, thay thế bằng trung bình, trung vị, mode, fit lại bằng hồi quy, nội suy ...

- **Lựa chọn 1:** Loại bỏ các giá trị khuyết `na.omit()`.

```
people.new1=na.omit(people)
na.action(people.new1)
```

- **Lựa chọn 2:** Nhập lại, thay thế

```
attach(people)
people=edit(people) # Nhập lại các giá trị cần sửa
tb <- mean(height,na.rm=T)
height[is.na(height)]<-tb # Thay thế bằng trung bình.
tv <- median(height,na.rm=T)
height[is.na(height)]<-tv # Thay thế bằng trung vị.
```

Missing values...

Ta có thể sử dụng lệnh `impute()` trong package `Hmisc`

```
library(Hmisc) # Chưa có package này, cài đặt thêm
height=impute(height,fun=mean)
height=impute(height,fun=median)
height=impute(height,fun="random") # Thay thế ngẫu nhiên
```

*** Fit sử dụng hồi quy:**

$$Y = a_n X_n + a_{n-1} X_{n-1} + \dots + a_1 X_1 + a_0$$

```
cor(height,age,use="na.or.complete") # Xem tương quan giữa
height và age
model=lm(height ~ age) # Thiết lập mô hình hồi quy
i=is.na(height)
height[i]=predict(model,newdata=people[i,])
height
```

Missing values...

* Thay thế các missing values theo Phương pháp Gower

Gower sử dụng k phần tử gần nhất (k nearest neighbors) theo nghĩa so sánh khoảng cách $d(i, j)$.

Package: VIM.

Lệnh: `kNN()`

Ví dụ: Sử dụng dữ liệu về hoa iris

```
install.packages("VIM")
library(VIM)
data(iris)
n <- nrow(iris)
# Tạo ngẫu nhiên một số missing values( 10 giá trị/cột)
for (i in 1: n){
  iris[sample(1:n, 10, replace = FALSE), i] <- NA}
iris2 <- kNN(iris)
```

* **Special values:** NA, NULL, +Inf, - Inf, NaN.

* **Check:** `is.na()`, `is.null()`, `is.finite()`, `is.nan()`.

* **Ví dụ:** Chạy thử các đoạn code sau

```
is.null(c())  
is.null(c(1,4,"A","B"))  
is.finite(c(1, Inf, NaN, NA))  
is.nan(c(0/0, Inf-Inf, NA,3))
```

* **Outliers:** $x_j > Q_3 + 1.5/IQR$ hoặc $x_j < Q_1 - 1.5/IQR$,
trong đó, Q_1, Q_3 là các phân vị mức 25% và 75%
 IQR là interquantile range = $Q_3 - Q_1$

* **Kiểm tra:** Ta có thể sử dụng các lệnh: `summary()`, `boxplot()`,
`boxplot.stats()` hoặc vẽ `scatterplot()` trong package "car"

* **Ví dụ:**

```
x <- c(1:10, 20, 30)
summary(x)
boxplot(x)
boxplot.stats(x)$out # Xuất ra các outliers
library(car)
id=c(1:length(x))
scatterplot(id,x)
```

Obvious inconsistencies

* **Obvious inconsistencies:** Chẳng hạn, age phải là số dương và nhỏ hơn 150; height > 0, trẻ em thì chưa kết hôn được; đàn ông thì ko thể có bầu,...

* **Ví dụ:**

people

age	agegroup	height	status	yearsmarried
21	adult	6.0	single	-1
2	child	3.0	married	0
18	adult	5.7	married	20
221	elderly	5.0	widowed	2
34	child	-7.0	married	3

```
library(editrules)
```

```
E <- editset(c("age >=0", "age <= 150"))
```

```
violatedEdits(E, people)
```

Obvious inconsistencies...

Ta có thể thiết lập các quy tắc, điều kiện để kiểm tra data một lượt.

- Bước 1: Tạo một file text tên edits.txt chứa các điều kiện.

```
# numerical rules
age >= 0
age <= 150
height > 0
age > yearsmarried
# categorical rules
status %in% c("married","single","widowed")
agegroup %in% c("child","adult","elderly")
if (status == "married") agegroup %in%
c("adult","elderly")
# mixed rules if ( status %in% c("married","widowed"))
age - yearsmarried >= 18
if ( age < 18 ) agegroup == "child"
if ( age >= 18 && age <65 ) agegroup == "adult"
if ( age >= 65 ) agegroup == "elderly"
```

Obvious inconsistencies...

- **Bước 2:** Sử dụng package editrules để tiến hành check.

```
E <- editfile(file.choose()) # Đọc file edits.txt vào E
plot(E)
ve <- violatedEdits(E, people)
ve
summary(ve)
plot(ve)
localizeErrors(E, people, method = "mip")
```

- **Bước 3:** Tiến hành sửa chữa

```
people[2, "status"] <- "single"
people[5, "height"] <- 7
people[5, "agegroup"] <- "adult"
summary(violatedEdits(E, people))
```


- Làm sạch dữ liệu: kiểm tra, phát hiện và sửa lỗi data.
- Lỗi nhập liệu: 2 người nhập độc lập và check sự khớp của 2 data.
- Missing, special values: xóa bỏ, nhập lại hoặc thay thế (cẩn thận).
- Outliers: sử dụng `summary()`, `boxplot()`, `boxplot.stats()`, `scatterplot()`. Việc loại bỏ hay giữ lại outliers nên xem xét kỹ càng.
- Obvious consistencies: Theo sự hiểu biết và kiến thức về data đó của người xử lý.

Bài giảng 5:
Phân tích mô tả

ThS. Lý Sel

Faculty of Mathematics-Statistics

Ton Duc Thang University

Các loại thang đo

- **Thang đo định danh (nominal):** chỉ thuộc tính, không có sự hơn kém, không có thứ bậc.
* **Ví dụ:** Giới tính: nam hoặc nữ; Tình trạng hôn nhân: Độc thân, có gia đình, ly dị, trường hợp khác,...
- **Thang đo thứ bậc (ordinal):** có sự hơn kém, có thứ bậc nhưng không có khoảng cách.
* **Ví dụ:** cấp bậc giáo dục: tiểu học, THCS, THPT, đại học, sau đại học
- **Thang đo khoảng (interval):** là thang đo thứ bậc có khoảng cách đều nhau, nhưng không có điểm gốc (số 0).
* **Ví dụ:** nhiệt độ, mức độ hài lòng, ...
- **Thang đo tỷ lệ (scale):** chỉ đặc tính số lượng, có số 0 làm gốc để so sánh hơn kém.
* **Ví dụ:** tuổi, chiều cao, cân nặng, huyết áp, lương,...

- **Dữ liệu định tính**

- Biến: biến phân loại. VD: giới tính, tôn giáo, hôn nhân, tên, ...
- Thang đo: định danh, thứ bậc.

- **Dữ liệu định lượng**

- Biến: liên tục hoặc biến rời rạc. VD: tuổi, chiều cao, cân nặng, số bệnh nhân, số ca, số lỗi, số lần thành công, số lần thất bại, ...
- Thang đo: khoảng, tỷ lệ.

- **Lập biểu bảng:** tần số, tần suất, tỷ lệ,...
- **Vẽ đồ thị, biểu đồ:** bar chart, pie chart, line chart, box plot, scatter plot, histogram, steam and leaf,...
- **Tính các chỉ số thống kê:** trung bình, trung vị, mốt, phương sai, độ lệch chuẩn, hệ số biến thiên, khoảng biến thiên, các phân vị, sai số chuẩn, khoảng tin cậy,...

Mô tả dữ liệu định tính: bằng bảng

* **Ví dụ:** Khảo sát những cái họ phổ biến ở Mỹ. Một mẫu 50 người được cho trong dữ liệu `lastnames.csv`.

- **Bảng tần số:** hàm `table()`
`tanso=table(lastnames)`
- **Bảng tần suất:** hàm `prop.table()`
`tansuat=prop.table(tanso)`
- **Bảng tần số-tần suất**
`bang=merge(tanso,tansuat,by='lastnames')`

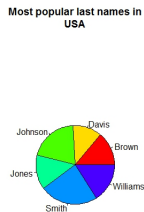
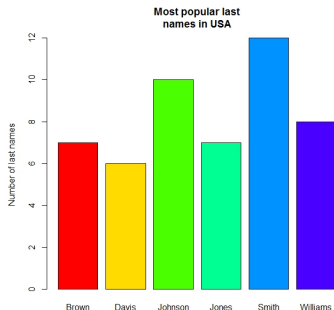
Mô tả dữ liệu định tính: bằng biểu đồ

- **Vẽ biểu đồ thanh:** `barplot()`

```
barplot(tanso,col=rainbow(7), xlab="Last names",  
ylab="Number of last names", main="Most popular last  
names in USA")
```

- **Vẽ biểu đồ tròn:** `pie()`

```
pie(tansuat,col=rainbow(7),main="Most popular last names in  
USA")
```



Mô tả dữ liệu định lượng: bằng bảng

* **Ví dụ:** Khảo sát kết quả học môn toán và môn văn của học sinh. Một mẫu gồm 29 người được cho trong dữ liệu diem.csv. Các biến gồm có: STT, Sex, Toan, Van.

* **Mục tiêu:** Lập bảng tổng hợp HS theo học lực môn toán: kém, yếu, trung bình, khá và giỏi.

- **Chia nhóm:** 5 nhóm theo các mốc điểm: 3.4, 4.9, 6.4, 7.9, và 10

```
diem=read.csv(file.choose(),header=T)
```

```
attach(diem)
```

```
chianhom=cut(Toan,c(0,3.4,4.9,6.4,7.9,10),include.lowest=T)
```

```
labels=c("kem","yeu","TB","kha","gioi"))
```


Mô tả dữ liệu định lượng: bằng bảng...

- **Bảng tần số:**

```
tanso.toan=table(chianhom)
```

- **Bảng tần suất:**

```
tansuat.toan=prop.table(tanso.toan)
```

- **Bảng tần số-tần suất:**

```
diem.toan=merge(tanso.toan,tansuat.toan,by="chianhom")  
tansuat.toan.new=round(tansuat.toan*100,1) # Làm tròn  
một chữ số  
phantram=paste(tansuat.toan.new,"%",sep="")# Gán dấu %  
diem.toan.new=cbind(tanso.toan,phantram)
```

Mô tả dữ liệu định lượng: bằng biểu đồ...

- Biểu đồ thanh:
`barplot(tanso.toan)`
- Biểu đồ tròn:
`pie(tansuat.toan)`
- Biểu đồ hộp: Kiểm tra sự phân tán dữ liệu
`boxplot(Toan)`
- Biểu đồ histogram: Kiểm tra phân bố của dữ liệu
`hist(Toan)`
- Đồ thị hàm mật độ:
`plot(density(Toan))`
- Biểu đồ nhánh-lá:
`stem(Toan)`

Mô tả dữ liệu định lượng: bằng các đặc trưng thống kê

- Trung bình: `mean(Toan)`
- Trung vị: `median(Toan)`
- Mode:
`freq=table(Toan)`
`names(freq)[freq==max(freq)]`
- Phương sai: `var(Toan)`
- Độ lệch chuẩn: `sd(Toan)`
- Sai số chuẩn: `sd(Toan)/sqrt(length(Toan))`
- Hệ số biến thiên: `sd(Toan)/mean(Toan)*100`
- Phân vị: `quantile(Toan, probs = 0.75)` # Phân vị mức 75%
- Độ trải giữa: `IQR(Toan)`
- Khoảng biến thiên: `range(Toan)`

- Thống kê tóm lược:
`summary(Toan)`
- Thống kê theo nhóm: chẳng hạn trung bình điểm toán theo giới tính
`tapply(Toan, Sex, mean)`
- Thống kê tổng hợp:
`by(diem, Sex, summary)`

Bài giảng 6: Biểu đồ

ThS. Thạc Thanh Tiền

Ton Duc Thang University

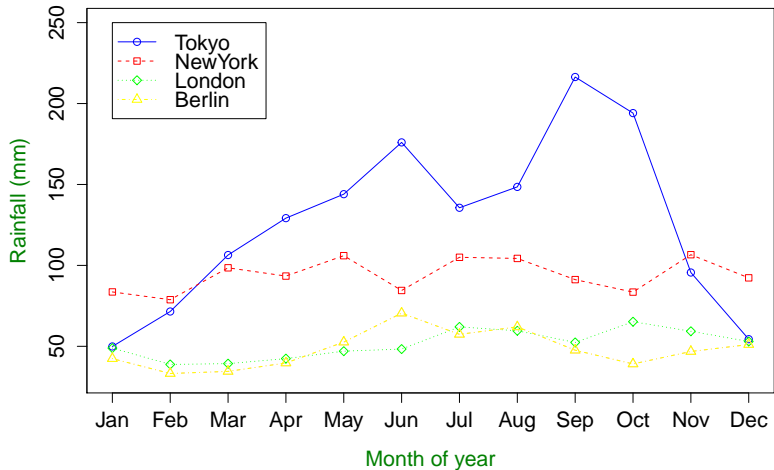
Line Charts

Vẽ biểu đồ line chart từ dữ liệu cityrain.txt

```
plot(Tokyo, type='o', col='blue',  
      ylim=c(0, 250), axes=F, ann=F)  
axis(1, at=1:12, lab=Month)  
axis(2, at=50*0:250)  
box()  
lines(NewYork, type='o', pch=22, lty=2, col='red')  
lines(London, type='o', pch=23, lty=3, col='green')  
lines(Berlin, type='o', pch=24, lty=4, col='pink')  
title(main='Monthly Rainfall in major cities',  
      col.main='red', font.main=4, xlab='Month of year',  
      ylab='Rainfall (mm)', col.lab=rgb(0,0.5,0))  
legend(1, 250, c('Tokyo','NewYork','London','Berlin'),  
      col=c('blue','red','green','pink'),  
      pch=21:24, lty=1:4)
```

Line Charts

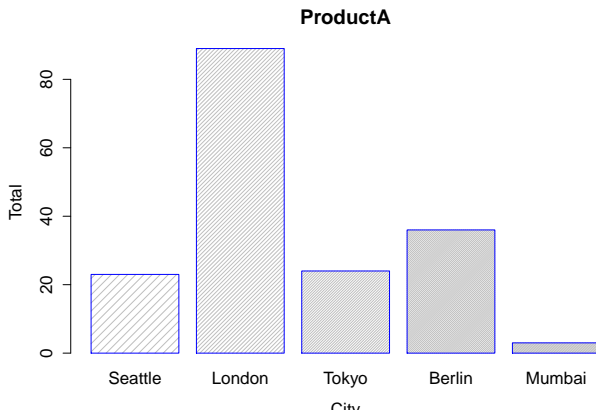
Monthly Rainfall in major city



Bar Charts

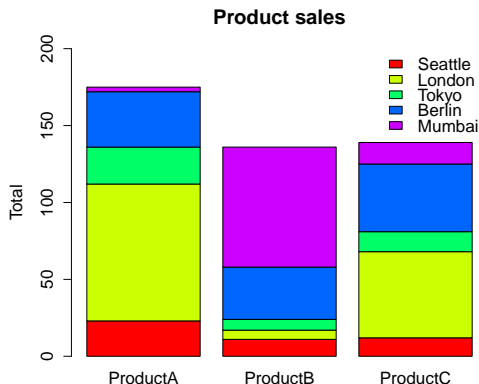
Vẽ biểu đồ Bar chart từ dữ liệu citysale.txt

```
barplot(ProductA, main="ProductA", names.arg=City, '  
        xlab="City", ylab="Total", border="blue",  
        density=c(10,20,30,40,50))
```

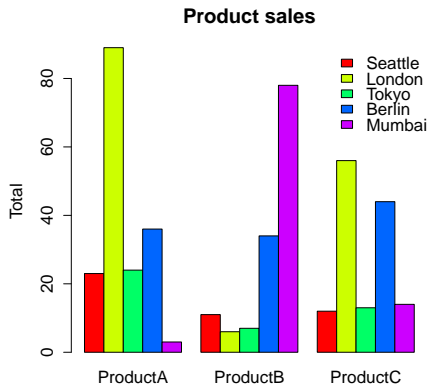


Bar Charts

```
barplot(as.matrix(data), main="Product sales",  
        ylab= "Total", col=rainbow(5), ylim=c(0,200))  
legend('topright',legend=City,bty='n',fill=rainbow(5))
```

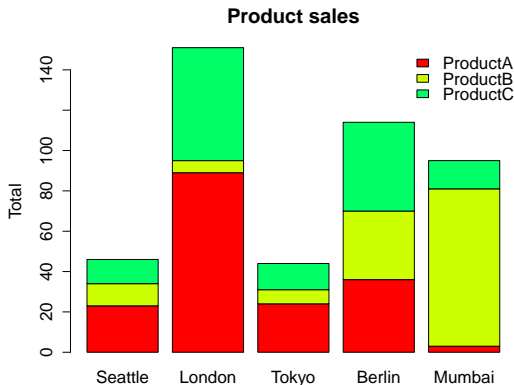


Bar Charts

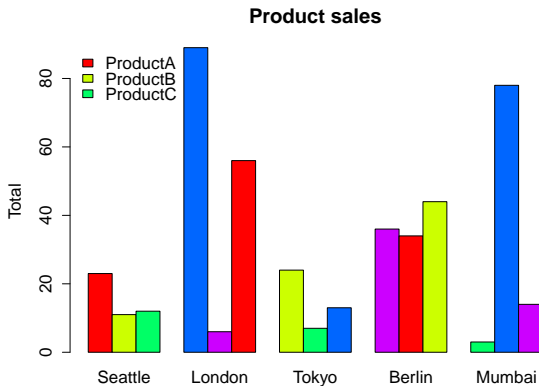


Bar Charts

```
barplot(t(data), main="Product sales", ylab= "Total",  
        col=rainbow(5), names.arg=City)  
legend('topright',names(data),bty='n',fill=rainbow(5))
```



Bar Charts



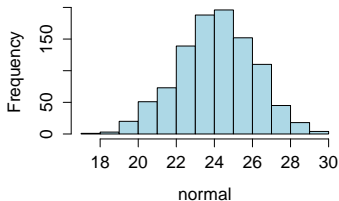
Histograms

Vẽ Histograms cho dữ liệu phân phối chuẩn với trung bình 24 và phương sai 2.

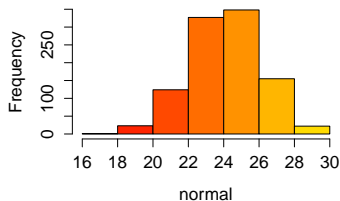
```
normal <- rnorm(1000, 24, 2)
hist(normal, col='lightblue')
hist(normal, breaks=5, main='Normal Histogram',
      col=heat.colors(5))
hist(normal, breaks=20, main='Normal Histogram',
      col=heat.colors(5))
hist(normal, breaks=c(17,20,22,25,28,30),
      main='Normal Histogram', col=heat.colors(5))
```

Histograms

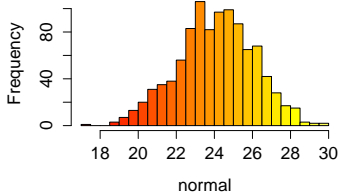
Histogram of normal



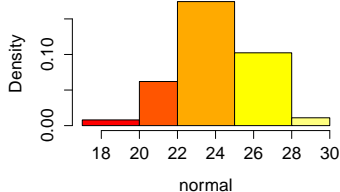
Normal Histogram



Normal Histogram

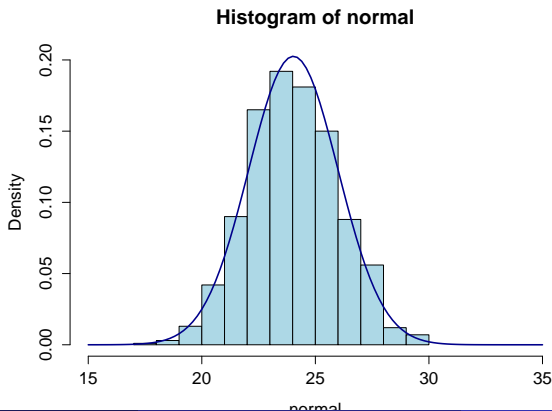


Normal Histogram



Histograms

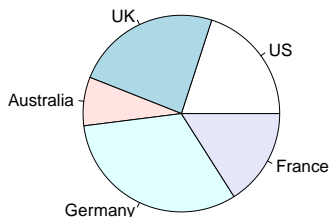
```
hist(normal, col='lightblue', freq=F,  
      xlim=c(15,35), ylim=c(0,0.20))  
curve(dnorm(x, mean=mean(normal), sd=sd(normal)),  
      add=TRUE, col="darkblue", lwd=2)
```



Pie Charts

```
slices <- c(10, 12, 4, 16, 8)
lbls <- c('US', 'UK', 'Australia', 'Germany', 'France')
pie(slices, labels = lbls,
    main="Pie Chart of Countries")
```

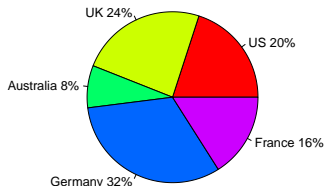
Pie Chart of Countries



Pie Charts

```
slices <- c(10, 12, 4, 16, 8)
lbls <- c('US', 'UK', 'Australia', 'Germany', 'France')
pct <- round(slices/sum(slices)*100)
lbls <- paste(lbls, pct)
lbls <- paste(lbls, "%", sep="")
pie(slices, labels = lbls, col=rainbow(length(lbls)),
    main="Pie Chart of Countries")
```

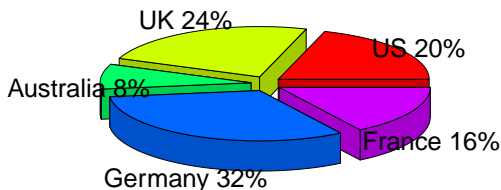
Pie Chart of Countries



Pie Charts 3D

```
library(plotrix)
slices <- c(10, 12, 4, 16, 8)
lbls <- c('US', 'UK', 'Australia', 'Germany', 'France')
pie3D(slices, labels=lbls, explode=0.1,
      main="Pie Chart of Countries")
```

Pie Chart of Countries

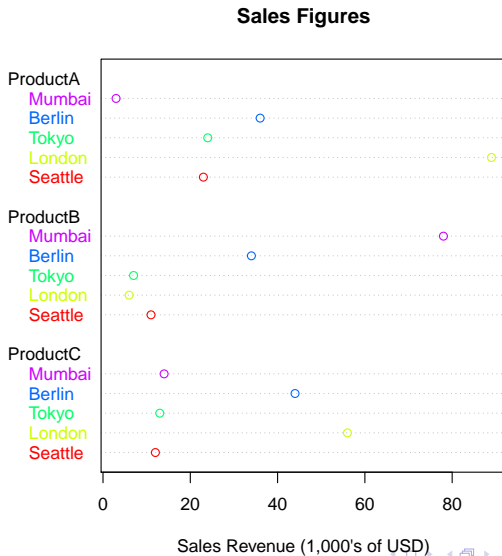


Dot Charts

Vẽ biểu đồ Dot Charts từ dữ liệu citysale.csv

```
install.packages('reshape')  
library(reshape)  
sale <- melt(citysale)  
dotchart(sale[,3], labels=sale[,1], groups=sale[,2],  
          col=rainbow(5), main="Sales Figures",  
          xlab="Sales Revenue (1,000's of USD)")
```

Dot Charts



Bài giảng 7:

Kiểm định giả thuyết

ThS. Thạc Thanh Tiền

Ton Duc Thang University

Kiểm định trung bình cho một mẫu

Biết phương sai tổng thể

```
library(BSDA)
z.test(x, y = NULL, alternative = "two.sided", mu = 0,
       sigma.x = NULL, sigma.y = NULL, conf.level = 0.95)
```

Không biết phương sai tổng thể

```
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)
```

Kiểm định tỷ lệ cho một mẫu

```
prop.test(x, n, p = NULL,  
          alternative = c("two.sided", "less", "greater"),  
          conf.level = 0.95, correct = TRUE)
```

Khi $n.p < 5$ và $n.q < 5$

```
binom.test(x, n, p = 0.5,  
            alternative = c("two.sided", "less", "greater"),  
            conf.level = 0.95)
```

Kiểm định sự khác biệt giữa hai trung bình

Mẫu độc lập và biết phương sai

```
z.test(x, y, alternative = , mu = 0,  
       sigma.x = , sigma.y = , conf.level = )
```

Mẫu độc lập và không biết phương sai

```
t.test(x, y = NULL,  
       alternative = c("two.sided", "less", "greater"),  
       mu = 0, paired = FALSE, var.equal = FALSE,  
       conf.level = 0.95, ...)
```

Mẫu không độc lập

```
t.test(x, y = NULL,  
       alternative = c("two.sided", "less", "greater"),  
       mu = 0, paired = FALSE, var.equal = FALSE,  
       conf.level = 0.95, ...)
```


Kiểm định sự khác biệt giữa hai tỉ lệ

Với cỡ mẫu đủ lớn

```
prop.test(x=c(X1,X2),n=c(n1,n2),  
          alternative = c('two.sided','less','greater'),  
          conf.level=0.95,correct=FALSE)
```

Với cỡ mẫu nhỏ

```
prop.test(x=c(X1,X2),n=c(n1,n2),  
          alternative = c('two.sided','less','greater'),  
          conf.level=0.95,correct=TRUE)
```

Khi cỡ mẫu nhỏ cũng không cần thiết dùng `correct=TRUE`, vì nó làm tăng p-value.

Kiểm định sự khác biệt giữa hai phương sai

Nhiều bài toán thực tế thường dẫn đến việc kiểm định sự khác biệt giữa hai phương sai tổng thể.

Để kiểm định sự khác biệt giữa hai phương sai tổng thể ta dùng hàm kiểm định `var.test()`.

```
var.test(x, y, ratio = 1,  
         alternative = c('two.sided', 'less', 'greater'),  
         conf.level = 0.95, ...)
```

Bài giảng 8:

Phân tích phương sai-ANOVA

ThS. Lý Sel

Faculty of Mathematics-Statistics

Ton Duc Thang University

* **Bài toán 1:** Số liệu năng suất (tạ/ha) của 3 loại giống lúa A, B, và C.

A	B	C
65	69	75
74	72	70
64	68	78
83	78	76

* **Câu hỏi nghiên cứu:**

- 1) Có sự khác biệt nào về năng suất của ba giống lúa?
- 2) Nếu có sự khác biệt, giống lúa nào khác với giống lúa nào?

* **Công cụ:** Phân tích phương sai một nhân tố (One-way ANOVA).

* **Mô hình ANOVA:** Kiểm định sự khác biệt về trung bình của nhiều nhóm (≥ 3 nhóm).

Dưới giả thuyết thống kê:

$$H_0 : \mu_A = \mu_B = \mu_C$$

H_1 : Có ít nhất một cặp nhóm khác biệt.

* **Giả định ANOVA:**

- (1) Dữ liệu các mẫu độc lập và có phân phối chuẩn.
- (2) Phương sai các nhóm bằng nhau.

* Các bước tiến hành:

- **Bước 1:** Kiểm định lại các giả định trước khi phân tích.

```
A=c(65,74,64,83)
```

```
B=c(69,72,68,78)
```

```
C=c(75,70,78,76)
```

```
nangsuat=c(A,B,C)
```

```
gionglua=c(rep("A",4),rep("B",4),rep("C",4))
```

```
tapply(nangsuat,gionglua,shapiro.test) # Kiểm định tính  
chuẩn của dữ liệu
```

```
bartlett.test(nangsuat ~ gionglua) # Kiểm định sự đồng  
nhất phương sai của các nhóm.
```

- **Bước 2:** Phân tích với lệnh `aov(formula, data=)`

```
model=aov(nangsuat ~ gionglua)
summary(model)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
gionglua	2	26.2	13.08	0.354	0.711
Residuals	9	332.5	36.94		

- **Bước 3:** Diễn giải kết quả:

Ta có, $P\text{-value} = 0.771 > 0.05$.

Do đó, ta chưa có cơ sở bác bỏ giả thuyết H_0 . Nghĩa là ta chấp nhận nhận định các giống lúa có năng suất như nhau.

- **Bước 4:** Phân tích hậu định.

Nếu ở Bước 3 ta kết luận có sự khác biệt giữa các nhóm thì Bước 4 cụ thể hóa hơn theo nghĩa cho biết nhóm này khác với nhóm nào?

Ở đây, ta dùng phương pháp: Tukey's Honest Significant Difference.

`TukeyHSD(model)`

- **Bước 4:** Phân tích hậu định.

***Ví dụ:** Nghiên cứu các loại thức ăn như: casein, horsebean, linseed, meatmeal, soybean và sunflower ảnh hưởng đến trọng lượng gà.

Dữ liệu: chickwts

```
with(chickwts)
model1=aov(weigh ~ feed)
summary(model1)
TukeyHSD(model1)
```

feed	diff	...	p adj
horsebean-casein	-163.383333	...	0.0000000
linseed-casein	-104.833333	...	0.0002100
meatmeal-casein	-46.674242	...	0.3324584
...

Two-way ANOVA (Không tương tác)

* **Bài toán 2:** Thí nghiệm với 3 loại giống lúa (A, B, và C) và 4 loại phân bón (I, II, III và IV). Số liệu năng suất lúa qua 4 năm như sau:

	A	B	C
I	65	69	75
II	74	72	70
III	64	68	78
IV	83	78	76

* **Câu hỏi nghiên cứu:**

- 1) Có sự khác biệt nào về năng suất của ba giống lúa?
- 2) Có sự khác biệt nào về sự ảnh hưởng của bốn loại phân bón đến năng suất?
- 3) Nếu có sự khác biệt, giống lúa nào khác với giống lúa nào? Phân bón nào khác với phân bón nào?

* **Công cụ:** Phân tích phương sai hai nhân tố (Two-way ANOVA).

* Phân tích bằng R

```
A=c(65,74,64,83)
B=c(69,72,68,78)
C=c(75,70,78,76)
nangsuat=c(A,B,C)
gionglua=c(rep("A",4),rep("B",4),rep("C",4))
phanbon=c(rep(c("I","II","III","IV"),3))
lua=data.frame(nangsuat,gionglua,phanbon)
attach(lua)
model2=aov(nangsuat ~ phanbon+gionglua)
summary(model2)
TukeyHSD(model2)
```

Two-way ANOVA (Tương tác)

* **Bài toán 3:** Khảo sát sự tương tác giữa các phân bón và giống lúa:

	A	B	C
I	65 68 62	69 71 67	75 75 78
II	74 79 76	72 69 69	70 69 75
III	64 72 65	68 73 75	78 82 80
IV	83 82 84	78 78 75	76 77 75

* **Câu hỏi nghiên cứu:**

- 1) Có sự khác biệt nào về năng suất của ba giống lúa?
- 2) Có sự khác biệt nào về sự ảnh hưởng của bốn loại phân bón?
- 3) Nếu có sự khác biệt, giống lúa nào khác với giống lúa nào? Phân bón nào khác với phân bón nào?
- 4) Có sự tương tác nào giữa các phân bón và giống lúa?

Two-way ANOVA (Tương tác)

* Phân tích bằng R

```
A =c(65,68,62,74,79,76,64,72,65,83,82,84)
B=c(69,71,67,72,69,69,68,73,75,78,78,75)
C=c(75,75,78,70,69,75,78,82,80,76,77,75)
nangsuat=c(A,B,C)
gionglua=c(rep("A",12),rep("B",12),rep("C",12))
phanbon2=c(rep("II",3))
phanbon3=c(rep("III",3))
phanbon4=c(rep("IV",3))
phanbon=c(rep(c(phanbon1,phanbon2,phanbon3,phanbon4),3))
lua2=data.frame(nangsuat,gionglua,phanbon)
detach(lua)
attach(lua2)
model3=aov(nangsuat ~ gionglua+phanbon+gionglua*phanbon)
summary(model3)
TukeyHSD(model3)
interaction.plot(gionglua,phanbon,nangsuat,pch=8,type="b")
```

- ANOVA: Phương pháp so sánh sự khác biệt của nhiều nhóm (≥ 3).
- Giả định: Dữ liệu phân phối chuẩn, phương sai các nhóm bằng nhau. Điều kiện này bị vi phạm, sử dụng kiểm định Kruskal-Wallis `kruskal.test(formula,data)` (tìm hiểu thêm).
- One-way ANOVA: `aov(x ~ factor)`
- Two-way ANOVA: `aov(x ~ factor1 + factor2)`
- Two-way ANOVA with interaction: `aov(x ~ factor1 + factor2 + factor1*factor2)`
- Post-hoc analysis: `TukeyHSD(model)`