ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KĨ THUẬT MÁY TÍNH

XÁC SUẤT THỐNG KÊ (MT2013)

BÀI TẬP LỚN - ĐỀ 1

Nhóm MT25

GVHD: Nguyễn Kiều Dung

Thành viên nhóm MT25: Đỗ Đình Phú Quí - 2014289 - L07
Đặng Kim Phú - 2012523 - L07
Nguyễn Thị Minh Uyên - 2012538 - L07
Đoàn Duy Long - 2013653 - L17
Mã Hoàng Khôi Nguyên - 2013917 - L07

Thành phố Hồ Chí Minh, tháng 04 năm 2022

Mục lục

Phần A
CƠ SỞ LÝ THUYẾT

1 Khái quát chung

Phân tích hồi quy tuyến tính là một phương pháp phân tích quan hệ giữa biến phụ thuộc Y với một hay nhiều biến độc lập X. Mô hình hóa sử dụng hàm tuyến tính (bậc 1). Các tham số của mô hình (hay hàm số) được ước lượng từ dữ liệu.

Hệ số hồi quy phản ánh độ dốc của đường hồi quy tuyến tính cho thấy sự thay đổi của biến phụ thuộc, hệ số hồi quy thực ra là hệ số của phương trình phân tích hồi quy. Những hằng số này thu được bằng phương pháp bình phương cực tiểu, thông thường được gọi là hệ số hồi quy ước lượng được.

Gọi $\widehat{\beta}_1$, $\widehat{\beta}_0$ là các ước lượng của $\beta_0 v \beta_1$

Đường thẳng hồi quy với các hệ số ước lượng (fitted regression line) có dạng:

$$\widehat{Y} = \widehat{\beta_0} + \widehat{\beta_1} x$$

Một đường thẳng ước lượng tốt phải "gần với các điểm dữ liệu".

Tìm $\widehat{\beta_0}$ và $\widehat{\beta_1}$: dùng phương pháp bình phương cực tiểu.

Có rất nhiều hệ thống tương quan trong thống kê, song có 3 hệ thống thông dụng nhất là: Pearson r, Spearman ρ ,và Kendall τ.

1.1 Hệ số tương quan Pearson r

Hệ số tương quan Pearson (Pearson correlation coefficient, ký hiệu r) là số liệu thống kê kiểm tra đo lường mối quan hệ thống kê hoặc liên kết giữa các biến phụ thuộc với các biến liên tục.

Để đo lường mức độ mạnh yếu của mối quan hệ giữa hai biến số, chúng ta sử dụng hệ số tương quan. Hệ số tương quan có giá trị trong khoảng [-1.0;1.0]. Kết quả được tính ra lớn hơn 1 hoặc nhỏ hơn -1 có nghĩa là có lỗi trong phép đo tương quan.

- $r < 0$: Hệ số tương quan âm. Nghĩa là giá trị biến x và y nghịch biến với nhau.
- $r > 0$: Hệ số tương quan dương. Cho thấy mối quan hệ đồng biến hoặc tương quan đương.
- $r = 0$: Hai biến không có tương quan tuyến tính.
- $r = 1 \cup r = -1$: Hai biến có mối tương quan tuyến tính tuyệt đối.

Hệ số tương quan pearson (r) chỉ có ý nghĩa khi và chỉ khi mức ý nghĩa quan sát (sig.) nhỏ hơn mức ý nghĩa $\alpha = 5$
Ta có chi tiết hơn như sau:

- $r \in [0.5; 1] \cup [-1; -0.5]$ thì nó được cho là tương quan mạnh.
- $r \in [0.3; 0.5) \cup (-0.5; -0.3]$ thì nó được gọi là tương quan trung bình.
- Trong khoảng còn lại, thì được gọi là một mối tương quan yếu.

Cho 2 biến số x và y từ n mẫu, hệ số tương quan Pearson được ước tính theo công thức sau :

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}}$$

Trong R chúng ta sử dụng hàm $cor(x, y)$ để tính giá trị này.

1.2 Hệ số tương quan Spearman

Hệ số tương quan Spearman ρ được sử dụng khi hai biến x và y không tuân theo luật phân phối chuẩn, trái ngược với hệ số tương quan Pearson. Đôi khi đây còn được gọi là hệ số của phương pháp phân tích phi tham số. Hệ số này được ước tính bằng cách biến đổi biến x, y thành biến có thứ bậc (rank), sau đó xem xét độ tương quan giữa hai dãy số có bậc này.

Tương quan hạng Spearman được sử dụng thay thế tương quan Pearson để kiểm tra mối quan hệ giữa hai biến được xếp hạng hoặc một biến được xếp hạng và một biến đo lường không yêu cầu có phân phối chuẩn. Tương quan hạng Spearman là xem xét tính đơn điệu của 2 biến x và y với nhau. Nếu hệ số tương quan dương thì đồng biến. Nếu hệ số tương quan âm thì kết luận là nghịch biến.

Ta có cách tính hệ số tương quan Spearman ρ như sau :

$$R_s = 1 - \frac{6\sum D^2}{n^3 - n}$$

D là hiệu của hạng hai biến. Trong R sử dụng phương thức sau:

$$cor.test(x, y, method = "spearman")$$

1.3 Hệ số tương quan Kendall

Hệ số tương quan Kendall τ cũng là một phương pháp phân tích phi tham số được ước tính bằng cách tìm các cặp số (x,y) song hành với nhau. Một cặp (x,y) song hành ở đây được định nghĩa là có hiệu số (độ khác biệt) trên trục hoành có cùng dấu hiệu (dương hay âm) với hiệu trên trục tung. Nếu hai biến số x, y không có liên hệ với nhau thì số cặp song hành bằng hay tương đương với số cặp không song hành.

Trong R, để tính toán hệ số này. Chúng ta có thể sử dụng phương thức :

$$cor.test(x, y, method = "kendall")$$

2 Mô hình hồi quy tuyến tính đơn

Một mô hình thống kê tuyến tính đơn (simple linear regression model) liên quan đến một biến ngẫu nhiên Y và một biến giải thích x là phương trình có dạng

$$Y = \beta_0 + \beta_1 X + \epsilon$$

trong đó :

- β_0, β_1 là các tham số chưa biết, gọi là các hệ số hồi quy.
- X là biến độc lập, giải thích cho y.
- ϵ là thành phần sai số.

Các sai số ngẫu nhiên ϵ_i, i = 1, . . . , n trong mô hình được giả sử thỏa các điều kiện sau :

- Các sai số ϵ_i độc lập với nhau
- $E(\epsilon_i) = 0$ và $Var(\epsilon_i) = \sigma^2$
- Các sai số có phân phối chuẩn: $\epsilon_i \sim N(0, \sigma^2)$ với phương sai không đổi.

Cho trước X = x, ta có:

$$E(Y|X = x) = \beta_0 + \beta_1 x$$

Suy ra phân phối có điều kiện của Y cho trước X = x là:

$$Y|X = x \sim N(\beta_0 + \beta_1 x)$$

Ta có cách tính mô hình hồi quy tuyến tính đơn có dạng y = ax + b như sau :

$$S = \sum_{i=1}^{n}(v^2) = \sum_{i=1}^{n}(ax_i + b_i - y_i)^2 = min$$

Vậy a,b phải thỏa mãn hệ phương trình sau:

$$\begin{cases} \dfrac{\partial S}{\partial a} = 0 \\ \dfrac{\partial S}{\partial b} = 0 \end{cases} \Leftrightarrow \begin{cases} \dfrac{\partial S}{\partial a} = 2\sum_{i=1}^{n}(ax_i + b_i - y_i).x_i = 0 \\ \dfrac{\partial S}{\partial b} = 2\sum_{i=1}^{n}(ax_i + b_i - y_i) = 0 \end{cases}$$

Rút gọn lại ta có a, b thỏa mãn :

$$a = \frac{n\sum_{i=1}^{n}x_i y_i - \sum_{i=1}^{n}x_i \sum_{i=1}^{n}y_i}{n\sum_{i=1}^{n}x_i^2 - (\sum_{i=1}^{n}x_i)^2}, b = \frac{\sum_{i=1}^{n}y_i \sum_{i=1}^{n}x_i^2 - \sum_{i=1}^{n}x_i \sum_{i=1}^{n}x_i y_i}{n\sum_{i=1}^{n}x_i^2 - \sum_{i=1}^{n}(x_i)^2}$$

Đây là hệ 2 phương trình hai ẩn số a và b,với n là số lần làm thí nghiệm.

3 Hồi quy tuyến tính đa biến

Trong mô hình hồi quy tuyến tính đơn, $Y = \beta_0 + \beta_1 X + \epsilon$, có một yếu tố duy nhất là x, thực tế chúng ta có thể sử dụng nhiều biến hơn ví dụ như :

$$y_i = \alpha + \beta_1 x_{1i} + \beta_2 x_{2i} + \beta_3 x_{3i} + ... + \beta_k x_{ki} + \epsilon_i$$

Do trong mô hình chúng ta sử dụng nhiều biến x_i và nhiều thông số ước tính β_j (với i, j = 1, 2, ..., k)cho nên chúng được gọi là mô hình tuyến tính đa biến.

β_j cũng được ước tính chủ yếu bằng phương pháp bình phương cực tiểu và tương tự với hồi quy tuyến tính đơn. Ta có thể gọi như sau : $\widehat{y_i}$ là ước tính của y_i với

$$\widehat{y_i} = \widehat{\alpha} + \widehat{\beta_1}x_{1i} + \widehat{\beta_2}x_{2i} + \widehat{\beta_3}x_{3i} + ... + \widehat{\beta_k}x_{ki} + \widehat{\epsilon_i}$$

Phương pháp bình phương cực tiểu là tìm các hệ số $\widehat{\alpha}$, $\widehat{\beta_j}$ sao cho

$$\sum_{i=1}^{n}(y_i - \widehat{y})^2 = min$$

Đối với mô hình hồi quy tuyến tính đa biến, sử dụng ma trận sẽ đơn giản hơn rất nhiều, có thể viết lại như sau :

$$Y = X\beta + \epsilon$$

Trong đó ta có :

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ ... \\ y_i \end{bmatrix}, X = \begin{bmatrix} 1 & x_{11} & x_{21} & ... & x_{k1} \\ 1 & x_{12} & x_{22} & ... & x_{k2} \\ ... & ... & ... & ... & ... \\ 1 & x_{1i} & x_{2i} & ... & x_{ki} \end{bmatrix}, \beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ ... \\ \beta_i \end{bmatrix}, \epsilon = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ ... \\ \epsilon_i \end{bmatrix}$$

Phương pháp bình phương cực tiểu giải vecto

$$r\widehat{\beta} = (X^T X)^{-1} X^T Y$$

Và tổng phần dư là

$$\epsilon^T \epsilon = \left\| Y - \widehat{Y} \right\|^2$$

Phần B
HOẠT ĐỘNG 1

Tập tin "gia_nha.csv" chứa thông tin về giá bán ra thị trường (đơn vị đô la) của 21613 ngôi nhà ở quận King nước Mỹ trong khoảng thời gian từ tháng 5/2014 đến 5/2015. Bên cạnh giá nhà, dữ liệu còn bao gồm các thuộc tính mô tả chất lượng ngôi nhà.
Dữ liệu gốc được cung cấp tại: https://www.kaggle.com/harlfoxem/housesalesprediction.
Các biến chính trong bộ dữ liệu:

- price: Giá nhà được bán ra.
- floors: Số tầng của ngôi nhà được phân loại từ 1-3.5.
- condition: Điều kiện kiến trúc của ngôi nhà từ 1-5, 1: rất tệ và 5: rất tốt.
- view: Đánh giá cảnh quan xung quanh nhà theo mức độ từ thấp đến cao: 0-4.
- sqft_above: Diện tích ngôi nhà.
- sqft_living: Diện tích khuôn viên nhà.
- sqft_basement: Diện tích tầng hầm.

1 Đọc dữ liệu (Import)

1.1 Lời giải R

Dùng lênh read.csv() để đọc tệp tin.

```
data <- read.csv("gia_nha.csv")
```

1.2 Kết quả thực nghiệm

Sau khi chạy câu lệnh R để đọc dữ liệu, ta thu được bảng sau trong R:

	X.2	X.1	X	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	fl
1	1	1	1	7129300520	20141013T000000	221900	3	1.00	1180	5650	
2	2	2	2	6414100192	20141209T000000	538000	3	2.25	2570	7242	
3	3	3	3	5631500400	20150225T000000	180000	2	1.00	770	10000	
4	4	4	4	2487200875	20141209T000000	604000	4	3.00	1960	5000	
5	5	5	5	1954400510	20150218T000000	510000	3	2.00	1680	8080	
6	6	6	6	7237550310	20140512T000000	1225000	4	4.50	5420	101930	
7	7	7	7	1321400060	20140627T000000	257500	3	2.25	1715	6819	
8	8	8	8	2008000270	20150115T000000	291850	3	1.50	1060	9711	
9	9	9	9	2414600126	20150415T000000	229500	3	1.00	1780	7470	
10	10	10	10	3793500160	20150312T000000	323000	3	2.50	1890	6560	
11	11	11	11	1736800520	20150403T000000	662500	3	2.50	3560	9796	
12	12	12	12	9212900260	20140527T000000	468000	2	1.00	1160	6000	
13	13	13	13	114101516	20140528T000000	310000	3	1.00	1430	19901	
14	14	14	14	6054650070	20141007T000000	400000	3	1.75	1370	9680	

2 Làm sạch dữ liệu (Data cleaning)

2.1 Trích dữ liệu

Trích ra một dữ liệu con đặt tên là new_DF chỉ bao gồm các biến chính mà ta quan tâm như đã trình bày trong phần giới thiệu dữ liệu. Từ câu hỏi này về sau, mọi yêu cầu xử lý đều dựa trên tập dữ liệu con new_DF này.

2.1.1 Lời giải R

```
new_DF <- subset(data, select=c("price", "floors", "condition", "view",
"sqft_living", "sqft_above", "sqft_basement"))
```

2.1.2 Kết quả thực nghiệm

Sau khi chạy câu lệnh R, ta được bảng new_DF chỉ gồm những biến ta quan tâm:

	price	floors	condition	view	sqft_living	sqft_above	sqft_basement
24	252700	1.0	3	0	1070	1070	0
25	329000	2.0	4	0	2450	2450	0
26	*NA*	1.5	5	0	1710	1710	0
27	937000	2.0	3	0	2450	1750	700
28	667000	1.5	5	0	1400	1400	0
29	438000	1.0	3	0	1520	790	730
30	719000	2.0	3	0	2570	2570	0
31	580500	2.0	3	0	2320	2320	0
32	280000	3.0	3	0	1190	1190	0
33	687500	1.5	4	0	2330	1510	820
34	535000	1.5	4	0	1090	1090	0
35	322500	1.0	3	0	2060	1280	780
36	696000	1.5	3	0	2300	1510	790
37	550000	1.0	1	0	1660	930	730

2.2 Thay thế dữ liệu bị khuyết

Kiểm tra các dữ liệu bị khuyết trong tập tin. Nếu có dữ liệu bị khuyết, đề xuất phương pháp thay thế cho những dữ liệu bị khuyết này.

2.2.1 Tìm giá trị bị khuyết

- Lời giải R:

```
apply(is.na(new_DF), 2, sum)
```

- Sau khi chạy R ra được kết quả:

```
      price       floors    condition         view  sqft_living   sqft_above
         20            0            0            0            0            0
sqft_basement
          0
```

Nhận xét
Trong bảng dữ liệu trích lọc new_DF vẫn còn những giá trị bị khuyết (NA). Nhận thấy những giá trị khuyết này nằm ở cột price. Do số lượng NA là 20 (chiếm tỷ lệ $< 10\%$) nên ta sẽ xử lý bằng cách xóa các quan sát bị khuyết của price.

2.2.2 Xóa các hàng có giá trị bị khuyết

- Lời giải R

```
new_DF <- na.omit(new_DF)
View(new_DF)
```

- Sau khi chạy R thu được bảng:

	price	floors	condition	view	sqft_living	sqft_above	sqft_basement
24	252700	1.0	3	0	1070	1070	0
25	329000	2.0	4	0	2450	2450	0
27	937000	2.0	3	0	2450	1750	700
28	667000	1.5	5	0	1400	1400	0
29	438000	1.0	3	0	1520	790	730
30	719000	2.0	3	0	2570	2570	0
31	580500	2.0	3	0	2320	2320	0
32	280000	3.0	3	0	1190	1190	0
33	687500	1.5	4	0	2330	1510	820
34	535000	1.5	4	0	1090	1090	0
35	322500	1.0	3	0	2060	1280	780
36	696000	1.5	3	0	2300	1510	790
37	550000	1.0	1	0	1660	930	730
38	640000	2.0	4	0	2360	2360	0

3 Làm rõ dữ liệu (Data visualization)

3.1 Chuyển đổi biến

Đổi các biến price, sqft_living, sqft_above, sqft_basement về log của nó. Nếu nhà không có tầng hầm thì gán lại sqft_basement = 0

3.1.1 Lời giải R

```
new_DF$ price <- log(new_DF$ price )
new_DF$ sqft_living <- log(new_DF$ sqft_living)
new_DF$ sqft_above <- log(new_DF$ sqft_above)
new_DF$ sqft_basement <- log(new_DF$ sqft_basement)
for (i in 1:nrow(new_DF)){
  if (new_DF[i,7]==-Inf){
    new_DF[i,7]=0
  }
}
```

3.1.2 Kết quả thực nghiệm

Sau khi chạy các lệnh trên, ta thu được bảng new_DF mới:

	price	floors	condition	view	sqft_living	sqft_above	sqft_basement
1	12.30998	1.0	3	0	7.073270	7.073270	0.000000
2	13.19561	2.0	3	0	7.851661	7.682482	5.991465
3	12.10071	1.0	3	0	6.646391	6.646391	0.000000
4	13.31133	1.0	5	0	7.580700	6.956545	6.813445
5	13.14217	1.0	3	0	7.426549	7.426549	0.000000
6	14.01845	1.0	3	0	8.597851	8.266164	7.333023
7	12.45877	2.0	3	0	7.447168	7.447168	0.000000
8	12.58400	1.0	3	0	6.966024	6.966024	0.000000
9	12.34366	1.0	3	0	7.484369	6.956545	6.593045
10	12.68541	2.0	3	0	7.544332	7.544332	0.000000
11	13.40378	1.0	3	0	8.177516	7.528332	7.438384
12	13.05622	1.0	4	0	7.056175	6.756932	5.703782
13	12.64433	1.5	4	0	7.265430	7.265430	0.000000
14	12.89922	1.0	4	0	7.222566	7.222566	0.000000

3.2 Thống kê mô tả

3.2.1 Tính các giá trị thống kê mô tả của biến liên tục

Đối với các biến liên tục, tính các giá trị thống kê mô tả bao gồm: trung bình, trung vị, độ lệch chuẩn, giá trị lớn nhất và giá trị nhỏ nhất.

3.2.1.a Lời giải R

```
thongke<-as.data.frame (
    rbind(
        apply(new_DF[c(1 ,5 ,6, 7)],MARGIN=2,mean),
        apply(new_DF[c(1 ,5 ,6, 7)],MARGIN=2,median),
        apply(new_DF[c(1 ,5 ,6, 7)],MARGIN=2,sd),
        apply(new_DF[c(1 ,5 ,6, 7)],MARGIN=2,min),
        apply(new_DF[c(1 ,5 ,6, 7)],MARGIN=2,max)
      ),
    row.names=c("Trung binh","Trung vi","Do lech chuan","GTNN","GTLN")
)
```

3.2.1.b Kết quả thực nghiệm

Sau khi chạy câu lệnh R, ta được bảng các giá trị thống kê của các biến liên tục như sau:

	price	sqft_living	sqft_above	sqft_basement
Trung binh	13.047841	7.5503286	7.3948826	2.528378
Trung vi	13.017003	7.5548585	7.3524411	0.000000
Do lech chuan	0.526574	0.4247722	0.4276433	3.169678
GTNN	11.225243	5.6698809	5.6698809	0.000000
GTLN	15.856731	9.5134035	9.1495282	8.480529

3.2.2 Lập bảng thống kê số lượng cho biến phân loại

3.2.2.a Lời giải R

```
floors_table <- table(new_DF$floors, dnn = "floors")
View(floors_table)

condition_table <- table(new_DF$condition, dnn = "condition")
View(condition_table)

view_table <- table(new_DF$view, dnn = "view")
View(view_table)
```

3.2.2.b Kết quả thực nghiệm

- Đối với biến floors

	floors	Freq
1	1	10672
2	1.5	1909
3	2	8230
4	2.5	161
5	3	613
6	3.5	8

- Đối với biến condition

	condition	Freq
1	1	30
2	2	172
3	3	14016
4	4	5677
5	5	1698

- Đối với biến view

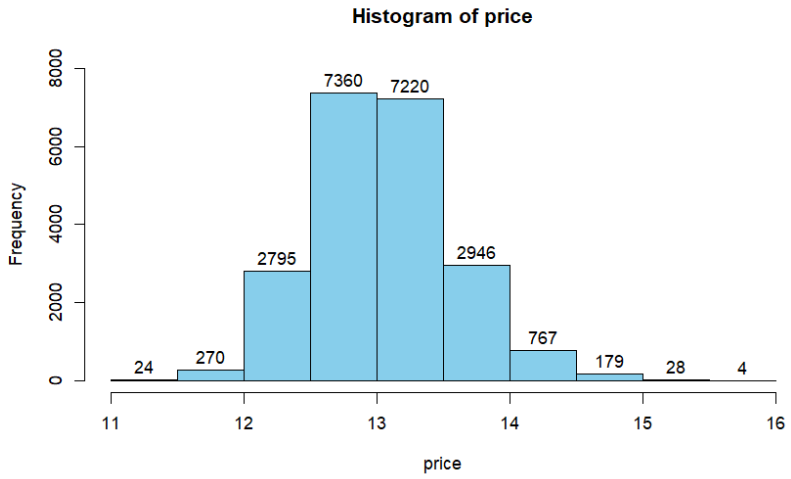
	view	Freq
1	0	19472
2	1	331
3	2	962
4	3	509
5	4	319

3.2.3 Đồ thị phân phối của biến price

3.2.3.a Lời giải R

```
hist(new_DF$price, main="Histogram of price", xlab="price",
ylab="Frequency", xlim=c(11,16), ylim=c(0,8000), labels = T,
col = "skyblue")
```

3.2.3.b Kết quả thực nghiệm



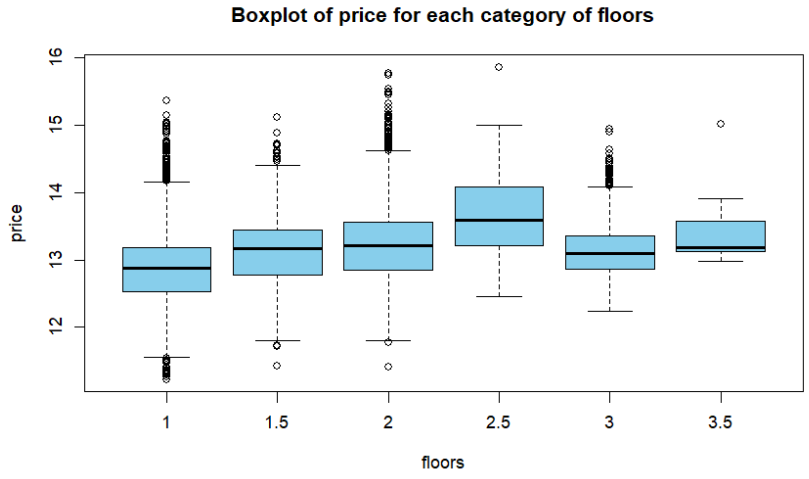
3.2.4 Đồ thị phân phối của biến price cho từng biến phân loại

3.2.4.a Lời giải R

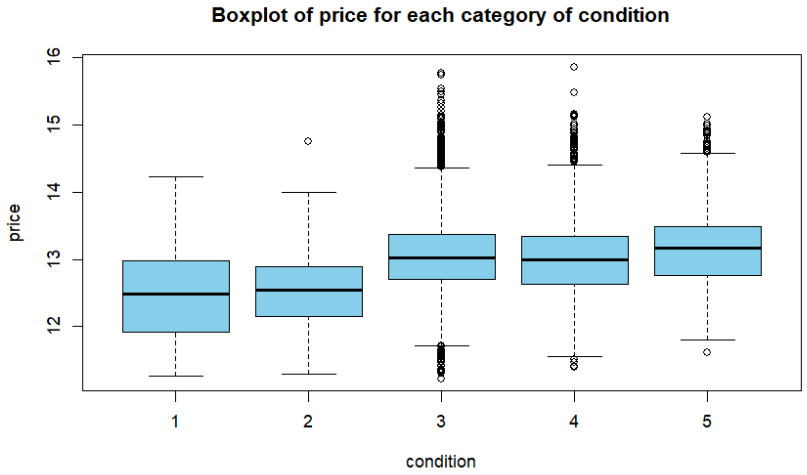
```
boxplot(main="Boxplot of price for each category of floors",
price~floors, xlab="floors", new_DF, ylab="price", col = "azure1")

boxplot(main="Boxplot of price for each category of condition",
price~condition, new_DF, xlab="condition", ylab="price", col = "azure1")

boxplot(main="Boxplot of price for each category of view",
price~view, new_DF, xlab="view", ylab="price", col = "azure1")
```

3.2.4.b Kết quả thực nghiệm

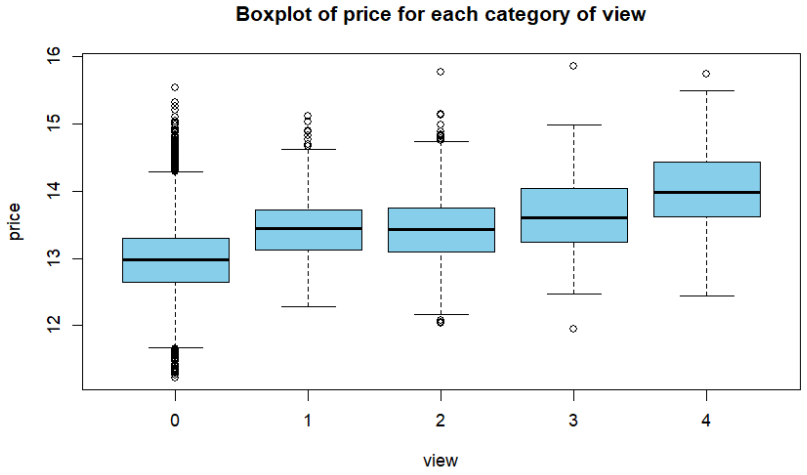
- Đối với biến floors



- Đối với biến condition



- Đối với biến view



3.2.5 Đồ thị phân phối của biến price theo từng biến liên tục

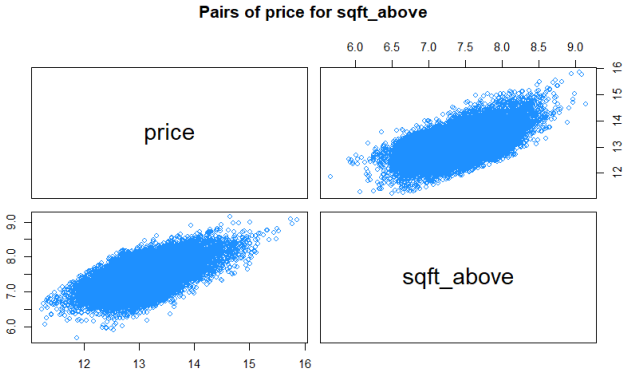
3.2.5.a Lời giải R

```
pairs (price~sqft_living, main ="Pairs of price for sqft_living",
new_DF, col="dodgerblue")
pairs (price~sqft_above, main ="Pairs of price for sqft_above",
new_DF, col="dodgerblue")
pairs (price~sqft_basement, main ="Pairs of price for sqft_basement",
new_DF, col="dodgerblue")
```

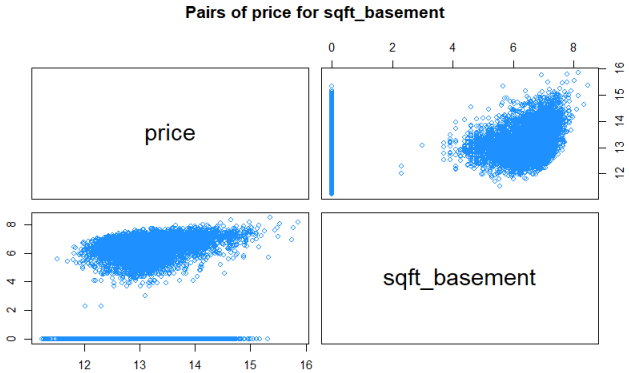
3.2.5.b Kết quả thực nghiệm

- Đối với biến sqft_living

- Đối với biến sqft_above



- Đối với biến sqft_basement



4 Xây dựng các mô hình hồi quy tuyến tính (Fitting linear regression models)

Xét mô hình hồi quy tuyến tính bao gồm biến price là một biến phụ thuộc và tất cả các biến còn lại đều là biến độc lập. Dùng lệnh lm() để thực thi mô hình hồi quy tuyến tính bội.

4.1 Lời giải R

```
HoiQuy<-lm(price~floors + condition + view + sqft_living + sqft_above +
sqft_basement, data=new_DF)
summary(HoiQuy)
```

4.2 Kết quả thực nghiệm

```
Call:
lm(formula = price ~ floors + condition + view + sqft_living +
    sqft_above + sqft_basement, data = new_DF)

Residuals:
     Min       1Q   Median       3Q      Max
-1.21679 -0.27522  0.01534  0.24743  1.45544

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   7.166402   0.051850 138.213  < 2e-16 ***
floors        0.102742   0.005834  17.611  < 2e-16 ***
condition     0.075289   0.004014  18.757  < 2e-16 ***
view          0.125296   0.003405  36.803  < 2e-16 ***
sqft_living   0.172634   0.029214   5.909 3.49e-09 ***
sqft_above    0.544931   0.029260  18.624  < 2e-16 ***
sqft_basement 0.043005   0.001976  21.760  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3671 on 21586 degrees of freedom
Multiple R-squared:  0.5141,    Adjusted R-squared:  0.514
F-statistic:  3807 on 6 and 21586 DF,  p-value: < 2.2e-16
```

Nhận xét:

- Từ cột **Estimate** ta có phương trình hồi quy:
  price = 7.166402+0.102742*floors+0.075289*condition+0.075289*view+0.172634*sqft_living+0.544931*sqft_above+0.043005*sqft_basement
- Ta nhận thấy hệ số của tất cả các biến đều dương có nghĩa là giá trị của các biến tỉ lệ thuận với giá nhà.
- Để đánh giá sự tác động của các biến lên giá nhà, ta quan tâm các hệ số hồi quy P_{value} tương ứng. Ta thấy P_{value} của các biến floors, condition, view, sqft_above, sqft_basement đều $<$ 2e-16, điều này nói lên rằng ảnh hưởng của các biến này có ý nghĩa rất lớn đến giá nhà. Ta còn nhận thấy sự ảnh hưởng của biến sqft_living ít hơn các biến còn lại.

5 Dự báo (Predictions)

Theo công thức có được từ hồi quy tuyến tính, ta có thể thực hiện dự báo giá nhà theo nhưng trường hợp cụ thể
Dự báo giá nhà ở quận King khi floors=2, condition=3, view=4, sqft_living=10, sqft_above=10, sqft_basement=10.

5.1 Lời giải R

```
x <-data.frame(floors=2, condition=3, view=4, sqft_living=10, sqft_above=10,
sqft_basement=10)
predict(HoiQuy,newdata=x)
```

5.2 Kết quả thực nghiệm

```
> x <-data.frame(floors=2,condition=3,view=4,sqft_living=10,sqft_above=10,sqft_basement=10)
> predict(HoiQuy,newdata=x)
       1
15.70464
```

Vậy giá nhà ở quận King trung bình được dự báo khi floors=2, condition=3, view=4, sqft_living=10,sqft_above=10, sqft_basement=10 là 15.70464.

Phần C
HOẠT ĐỘNG 2

Tập tin "League of Legends 2021 World Championship Play-In Groups Statistics - Raw Data.csv" thể hiện thông tin của các thành viên thi đấu trong giải Liên Minh Thế Giới.
Để dễ đọc vào hơn ta sẽ đổi tên dữ liệu thành "ChampionLOLdata.csv".
Dữ liệu gốc được cung cấp tại: https://www.kaggle.com/datasets/braydenrogowski/league-of-legends-worlds-2021-playin-group-stats.
Các biến chính trong bộ dữ liệu:

- Gold.Earned: Chỉ số vàng
- Kills: Chỉ số giết team địch
- Assists: Chỉ số hỗ trợ
- Position: Vị trí người chơi.
- Creep.Score: Chỉ số giết lính,
- Wards.Destroyed: Số đèn tầm nhìn bị hủy.

1 Đọc dữ liệu (Import)

1.1 Lời giải R

Dùng lênh read.csv() để đọc tệp tin.

```
data1 <- read.csv("ChampionLOLdata.csv")
```

1.2 Kết quả thực nghiệm

Sau khi chạy câu lệnh R để đọc dữ liệu, ta thu được bảng sau trong R:

	Team	Player	Opponent	Position	Champion	Kills	Deaths	Assists	Creep.Score	Gold.Earned	Champion.Damage.Share
1	UOL	Boss	GS	Top	Camille	4	5	3	188	11107	0.17
2	GS	Crazy	UOL	Top	Gwen	3	1	9	217	12201	0.20
3	UOL	Ahahacik	GS	Jungle	Trundle	2	4	5	156	9048	0.15
4	GS	Mojito	UOL	Jungle	Talon	5	4	10	194	11234	0.23
5	UOL	Nomanz	GS	Mid	Leblanc	1	3	4	216	9245	0.29
6	GS	Bolulu	UOL	Mid	Twisted Fate	5	0	13	205	12737	0.25
7	UOL	Argonavt	GS	Adc	Ezreal	1	5	3	202	9539	0.32
8	GS	Alive	UOL	Adc	Miss Fortune	9	2	5	273	15085	0.27
9	UOL	Santas	GS	Support	Amumu	1	6	4	41	6328	0.07
10	GS	Zergsting	UOL	Support	Rakan	0	2	16	42	7395	0.05
11	DFM	Evi	C9	Top	Gnar	1	4	2	191	8931	0.23
12	C9	Fudge	DFM	Top	Irelia	1	2	4	263	11834	0.18
13	DFM	Steal	C9	Jungle	Xin Zhao	1	3	1	156	8420	0.19
14	C9	Blaber	DFM	Jungle	Qiyana	0	1	5	182	9464	0.10
15	DFM	Aria	C9	Mid	Leblanc	1	2	0	219	9378	0.27

2 Làm sạch dữ liệu (Data cleaning)

2.1 Trích dữ liệu

Trích ra một dữ liệu con đặt tên là new_DF1 chỉ bao gồm các biến chính mà ta quan tâm như đã trình bày trong phần giới thiệu dữ liệu. Từ câu hỏi này về sau, mọi yêu cầu xử lý đều dựa trên tập dữ liệu con new_DF1 này.

2.1.1 Lời giải R

```
new_DF1 <- subset(data1, select=c("Gold.Earned", "Kills","Deaths",
                                  "Assists", "Creep.Score",
                                  "Wards.Destroyed","Kill.Participation",
                                  "Position"))
```

2.1.2 Kết quả thực nghiệm

Sau khi chạy câu lệnh R, ta được bảng new_DF1 chỉ gồm những biến ta quan tâm:

	Gold.Earned	Kills	Assists	Creep.Score	Wards.Destroyed	Position
1	11107	4	3	188	8	Top
2	12201	3	9	217	7	Top
3	9048	2	5	156	14	Jungle
4	11234	5	10	194	8	Jungle
5	9245	1	4	216	9	Mid
6	12737	5	13	205	1	Mid
7	9539	1	3	202	2	Adc
8	15085	9	5	273	5	Adc
9	6328	1	4	41	10	Support
10	7395	0	16	42	6	Support
11	8931	1	2	191	4	Top
12	11834	1	4	263	4	Top
13	8420	1	1	156	10	Jungle
14	9464	0	5	182	14	Jungle
15	9378	1	0	219	6	Mid

2.2 Thay thế dữ liệu bị khuyết

Kiểm tra các dữ liệu bị khuyết trong tập tin. Nếu có dữ liệu bị khuyết, đề xuất phương pháp thay thế cho những dữ liệu bị khuyết này.

2.2.1 Tìm giá trị bị khuyết

- Lời giải R:

```
apply(is.na(new_DF), 2, sum)
```

- Sau khi chạy R ra được kết quả:

```
Gold.Earned           Kills         Assists    Creep.Score Wards.Destroyed
          0               0               0               0               0
   Position
          0
```

Nhận xét
Trong bảng dữ liệu trích lọc new_DF1 không có giá trị NA vì vậy không cần xóa.

3 Làm rõ dữ liệu (Data visualization)

3.1 Chuyển đổi biến

Chuyển dữ liệu liệu ở biến Position từ [Jungle,Support, Top, Mid, Adc] thành [1,3,2,4,5].

3.1.1 Lời giải R

```
for (i in 1:nrow(new_DF1)) {
  if (new_DF1$Position[i] == "Jungle"){
    new_DF1$Position[i] = "1"
  }
  if (new_DF1$Position[i] == "Support"){
    new_DF1$Position[i] = "2"
  }
  if (new_DF1$Position[i] == "Top"){
    new_DF1$Position[i] = "3"
  }
  if (new_DF1$Position[i] == "Mid"){
    new_DF1$Position[i] = "4"
  }
  if (new_DF1$Position[i] == "Adc"){
    new_DF1$Position[i] = "5"
  }
}
new_DF1$Position = as.numeric(new_DF1$Position)
```

3.1.2 Kết quả thực nghiệm

Sau khi chạy các lệnh trên, ta thu được bảng new_DF1 mới:

	Gold.Earned	Kills	Assists	Creep.Score	Wards.Destroyed	Position
1	11107	4	3	188	8	3
2	12201	3	9	217	7	3
3	9048	2	5	156	14	1
4	11234	5	10	194	8	1
5	9245	1	4	216	9	4
6	12737	5	13	205	1	4
7	9539	1	3	202	2	5
8	15085	9	5	273	5	5
9	6328	1	4	41	10	2
10	7395	0	16	42	6	2
11	8931	1	2	191	4	3
12	11834	1	4	263	4	3
13	8420	1	1	156	10	1
14	9464	0	5	182	14	1
15	9378	1	0	219	6	4

3.2 Thống kê mô tả

3.2.1 Tính các giá trị thống kê mô tả của biến liên tục

Đối với các biến liên tục, tính các giá trị thống kê mô tả bao gồm: trung bình, trung vị, độ lệch chuẩn, giá trị lớn nhất và giá trị nhỏ nhất.

3.2.1.a Lời giải R

```
thongke1<-as.data.frame (
  rbind(
    apply(new_DF[1:6],MARGIN=2,mean),
    apply(new_DF[1:6],MARGIN=2,median),
    apply(new_DF[1:6],MARGIN=2,sd),
    apply(new_DF[1:6],MARGIN=2,min),
    apply(new_DF[1:6],MARGIN=2,max)
  ),
  row.names=c("Trung binh","Trung vi","Do lech chuan","GTNN","GTLN")
)
```

3.2.1.b Kết quả thực nghiệm

Sau khi chạy câu lệnh R, ta được bảng các giá trị thống kê của các biến liên tục như sau:

	Gold.Earned	Kills	Assists	Creep.Score	Wards.Destroyed
Trung binh	11008.159	2.709091	5.668182	200.3409	8.704545
Trung vi	10454.500	2.000000	5.000000	210.0000	7.000000
Do lech chuan	3198.806	2.579673	3.888149	101.3282	5.101614
GTNN	4714.000	0.000000	0.000000	14.0000	1.000000
GTLN	20546.000	13.000000	19.000000	419.0000	30.000000

3.2.2 Lập bảng thống kê số lượng cho biến phân loại

3.2.2.a Lời giải R

```
Pos_table <- table(new_DF1$Pos, dnn = "Pos")
View(Pos_table)
```

3.2.2.b Kết quả thực nghiệm

- Đối với biến Position

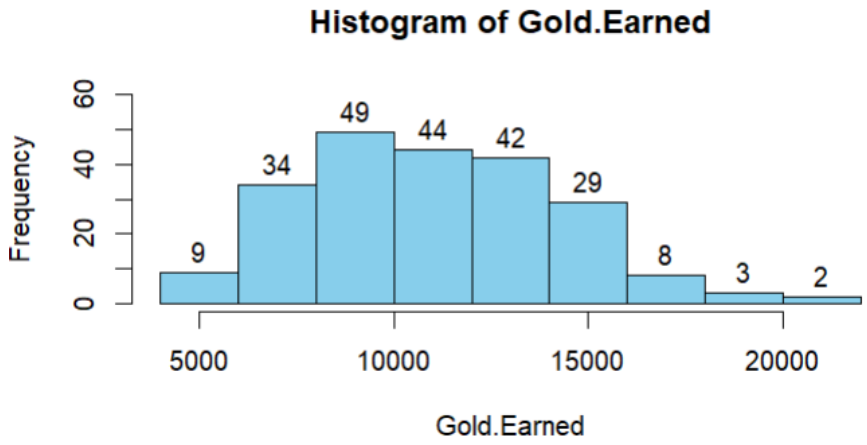
	Position	Freq
1	1	44
2	2	44
3	3	44
4	4	44
5	5	44

3.2.3 Đồ thị phân phối của biến price

3.2.3.a Lời giải R

```
hist(new_DF1$Gold.Earned, main="Histogram of Gold.Earned",xlab="Gold.Earned",
 ylab="Frequency",ylim=c(0,45), labels = T, col = "skyblue")
```

3.2.3.b Kết quả thực nghiệm



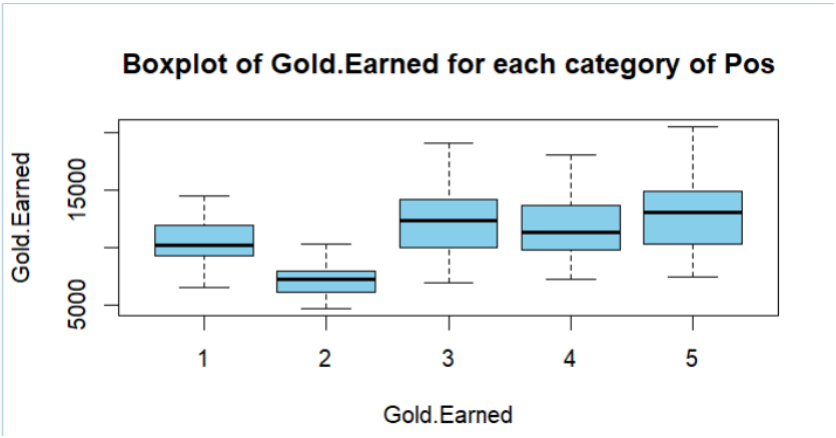
3.2.4 Đồ thị phân phối của biến price cho từng biến phân loại

3.2.4.a Lời giải R

```
boxplot(EGPM~Pos, new_DF, main="Boxplot of EGPM for each category of pos",
xlab="Pos", ylab="EGPM", col = "skyblue")
```

3.2.4.b Kết quả thực nghiệm

- Đối với biến Positon



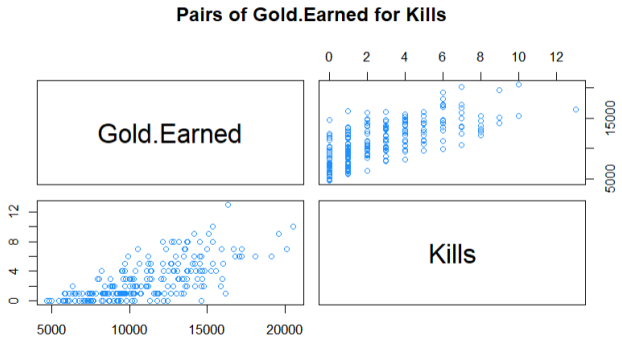
3.2.5 Đồ thị phân phối của biến price theo từng biến liên tục

3.2.5.a Lời giải R

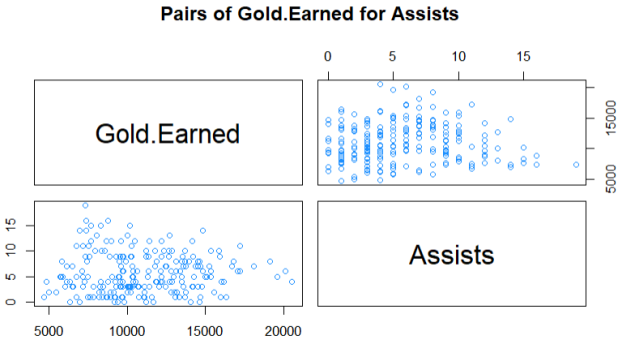
```
pairs (Gold.Earned~Kills, main ="Pairs of Gold.Earned for Kills",
       new_DF1, col="dodgerblue")
pairs (Gold.Earned~Assists , main ="Pairs of Gold.Earned for Assists",
       new_DF1, col="dodgerblue")
pairs (Gold.Earned~Wards.Destroyed, main ="Pairs of Gold.Earned for
       Wards.Destroyed", new_DF1, col="dodgerblue")
pairs (Gold.Earned~Creep.Score, main ="Pairs of Gold.Earned for Creep.Score",
       new_DF1, col="dodgerblue")
```

3.2.5.b Kết quả thực nghiệm

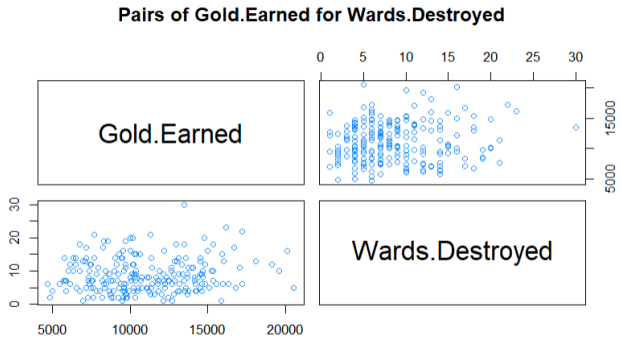
- Đối với biến Kills



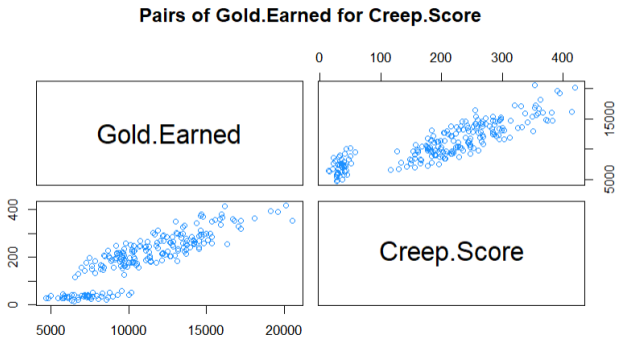
- Đối với biến Assists



- Đối với biến Wards.Destroyed



- Đối với biến Creep.Score



4 Xây dựng các mô hình hồi quy tuyến tính (Fitting linear regression models)

Xét mô hình hồi quy tuyến tính bao gồm biến Gold.Earned là một biến phụ thuộc và tất cả các biến còn lại đều là biến độc lập. Dùng lệnh lm() để thực thi mô hình hồi quy tuyến tính bội.

4.1 Lời giải R

```
HoiQuy1<-lm(Gold.Earned~Assists+Kills+Creep.Score+Wards.Destroyed+Position,
            data=new_DF1)
summary(HoiQuy1)
```

4.2 Kết quả thực nghiệm

```
Call:
lm(formula = Gold.Earned ~ Assists + Kills + Creep.Score + Wards.Destroyed +
    Position, data = new_DF1)

Residuals:
    Min      1Q  Median      3Q     Max
-2258.8  -609.2   -46.1   608.1  3458.4

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)     3512.4690   228.5375  15.369  < 2e-16 ***
Assists          166.4239    17.0896   9.738  < 2e-16 ***
Kills            400.7228    27.5683  14.536  < 2e-16 ***
Creep.Score       25.1155     0.8604  29.190  < 2e-16 ***
Wards.Destroyed   72.4428    12.5413   5.776 2.67e-08 ***
Position         -65.1532    54.2190  -1.202    0.231
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 904.5 on 214 degrees of freedom
Multiple R-squared:  0.9219,    Adjusted R-squared:  0.9201
F-statistic: 505.1 on 5 and 214 DF,  p-value: < 2.2e-16
```

Nhận xét:

- Từ cột **Estimate** ta có phương trình hồi quy:
  Gold.Earned = 3512.4690+166.4239*Assists+400.7228*Kills+25.1155*Creep.Score+ 72.4428*Warsd.Destroyed -65.1532*Position
- Ta nhận thấy hệ số của biến Warsd.Destroyed là số âm có nghĩa là giá trị của biến Warsd.Destroyed tỉ lệ nghịch với chỉ số vàng, các biến còn lại hệ số dương nên tỉ lệ thuận với chỉ số vàng.
- Để đánh giá sự tác động của các biến lên chỉ số vàng, ta quan tâm các hệ số hồi quy P_{value} tương ứng. Ta thấy P_{value} của các biến Assists, Kills, Creep.Score đều $<$ 2e-16, điều này nói lên rằng ảnh hưởng của các biến này có ý nghĩa rất lớn đến chỉ số vàng. Ta còn nhận thấy sự ảnh hưởng của biến Wards.Destroyed ít hơn các biến trên nhưng vẫn có ảnh hưởng đến chỉ số vàng. Còn biến Position không có ảnh hưởng đến chỉ số vàng.

Theo mô hình hồi quy tuyến tính HoiQuy1, do biến Position không ảnh hưởng nhiều lên Gold.Earned nên đề xuất mô hình hồi quy tuyến tính HoiQuy2 với Gold.Earned phụ thuộc vào các biến còn lại ngoại trừ Position:

4.3 Lời giải R

```
HoiQuy2<-lm(Gold.Earned~Assists+Kills+Creep.Score+Wards.Destroyed,
            data=new_DF1)
summary(HoiQuy2)
```

4.4 Kết quả thực nghiệm

```
Call:
lm(formula = Gold.Earned ~ Assists + Kills + Creep.Score + Wards.Destroyed,
    data = new_DF1)

Residuals:
    Min      1Q  Median      3Q     Max
-2387.5  -647.9   -50.9   619.5  3405.1

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     3397.918    207.921  16.342  < 2e-16 ***
Assists          165.565     17.092   9.687  < 2e-16 ***
Kills            403.168     27.521  14.649  < 2e-16 ***
Creep.Score       24.563      0.728  33.742  < 2e-16 ***
Wards.Destroyed   75.664     12.264   6.170 3.36e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 905.4 on 215 degrees of freedom
Multiple R-squared:  0.9214,    Adjusted R-squared:  0.9199
F-statistic: 629.7 on 4 and 215 DF,  p-value: < 2.2e-16
```

Nhận xét:

- Từ cột **Estimate** ta có phương trình hồi quy:
  Gold.Earned = 3397.918+165.565*Assists+403.168*Kills+24.563*Creep.Score+
  75.664*Warsd.Destroyed
- Ta nhận thấy hệ số của tất cả các biến đều dương có nghĩa là giá trị của các biến tỉ lệ thuận với chỉ số vàng.
- Để đánh giá sự tác động của các biến lên chỉ số vàng, ta quan tâm các hệ số hồi quy P_{value} tương ứng. Ta thấy P_{value} của các biến Assists, Kills, Creep.Score đều $<$ 2e-16, điều này nói lên rằng ảnh hưởng của các biến này có ý nghĩa rất lớn đến chỉ số vàng. Ta còn nhận thấy sự ảnh hưởng của biến Wards.Destroyed ít hơn các biến trên nhưng vẫn có ảnh hưởng đến chỉ số vàng.

5 Dự báo (Predictions)

Sử dụng mô hình hồi quy tuyến tính HoiQuy2 để dự đoán lượng Gold.Earned khi Assists = 3, Kills = 7, Creep.Score = 4, Wards. Destroyed = 20.

5.1 Lời giải R

```
x <-data.frame(Assists=3, Kills=7, Creep.Score=4, Wards.Destroyed =20)
predict (HoiQuy2, newdata=x, interval = "confidence")
```

5.2 Kết quả thực nghiệm

```
> predict(HoiQuy2,newdata=x, interval = "confidence")
       fit      lwr      upr
1 8328.331 7755.691 8900.972
```

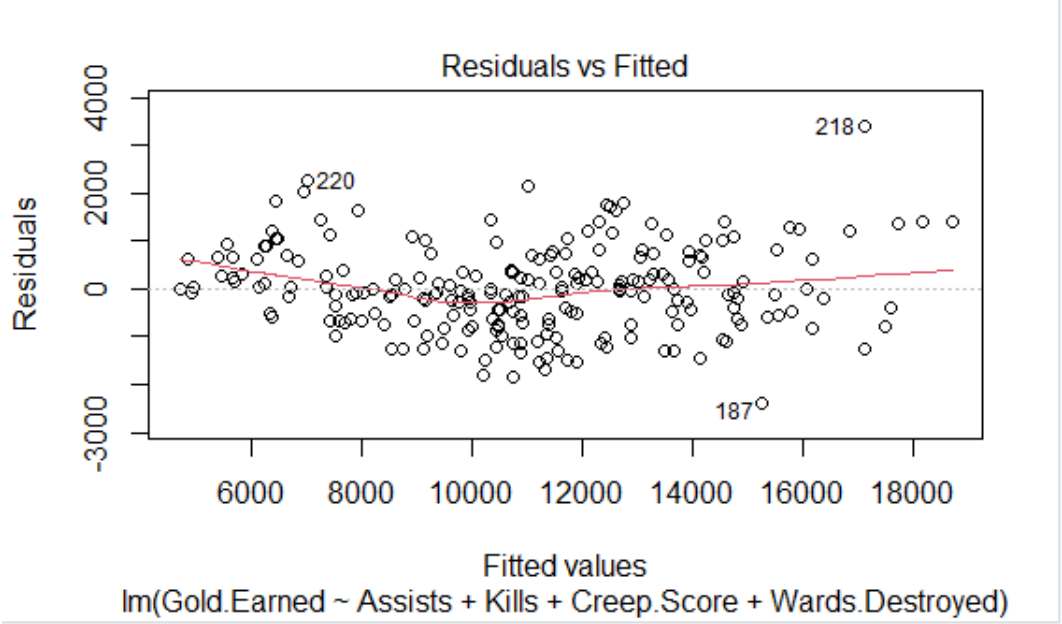
Vậy chỉ số vàng trung bình được dự báo khi Assists = 3, Kills = 7, Creep.Score = 4, Wards. Destroyed = 20 là 8328.331

5.3 Đồ thị dự báo

Vẽ đồ thị thể hiện giá trị dự đoán và giá trị thặng dư theo mô hình hồi quy tuyến tính HoiQuy2:

```
plot(HoiQuy2, which = 1)
```

Kết quả:



Nhận xét: Mô hình dự báo cho kết quả tương đối tốt: đường hồi quy (đường màu đỏ) gần như gần sát với đường Residuals = 0, giá trị của các quan sát tập trung xung quanh đường hồi quy. Điều này đã chứng tỏ rằng mô hình HoiQuy2 là mô hình tốt

Tài liệu tham khảo

[1] Nguyễn Đình Huy, 2018. Giáo trình xác suất thống kê, lần 9, NXB Đại học Quốc gia TP.HCM.

[2] Nguyễn Kiều Dung, Slide bài giảng trên lớp.

[3] Nguyễn Tiến Dũng, Đỗ Đức Thái, Nhập môn hiện đại về Xác Suất Thống Kê, tủ sách Sputnik, sách điện tử SE001, 201.

[4] Documentation for R, rdocumentation.org