HWDiamond Price

高嘉妤、柯堯珹、吳承恩、趙友誠

2024-11-25

Table of contents

Kaggle URL: https://www.kaggle.com/datasets/enashed/diamond-prices/discussion/547512

```
library(readr)
library(psych)
library(Hmisc)
library(DataExplorer)
library(ggplot2)
library(MASS)
library(car)
library(stargazer)
data <- data.table::fread("Diamonds Prices2022.csv")[,-1]
```

0. 資料簡介

Dimension of the Data : ***53943 samples x 10 columns***

Variables	Explanation	remark
carat	克拉 (重量)	連續變數 (公克)
cut	切工	類別變數, Fair,Good,Ideal,Premium,Very Good
color	顏色	類別變數, D,E,F,G,H,I,J 無色(D~F),近乎無色(G~J)
clarity	淨度	類別變數, IF: 內部無暇,VVS1: 極輕微瑕,VS1: 輕微內含物 1,VS2: 輕微內含物 2,SI1: 微內含物 1,SI2: 微內含物 2,I1: 內含物
depth	深度	連續變數 (mm)
table	檯面尺寸	連續變數
price	價格	連續變數
x	鑽石的長	連續變數 (mm)

Variables	Explanation	remark
y	鑽石的寬	連續變數 (mm)
z	鑽石的高	連續變數 (mm)

1.Data Preprocessing

```
latex(describe(data, title="Diamond Price Dataset"), title="", file="")
```

data

10 Variables 53943 Observations

carat

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
53943	0	273	0.999	0.7979	0.5122	0.30	0.31	0.40	0.70	1.04	1.51	1.70

```
lowest : 0.2  0.21 0.22 0.23 0.24, highest: 4    4.01 4.13 4.5  5.01
```

cut

n	missing	distinct
53943	0	5

Value	Fair	Good	Ideal	Premium	Very Good
Frequency	1610	4906	21551	13793	12083
Proportion	0.030	0.091	0.400	0.256	0.224

color

n	missing	distinct
53943	0	7

Value	D	E	F	G	H	I	J
Frequency	6775	9799	9543	11292	8304	5422	2808
Proportion	0.126	0.182	0.177	0.209	0.154	0.101	0.052

clarity

n	missing	distinct
53943	0	8

Value	I1	IF	SI1	SI2	VS1	VS2	VVS1	VVS2
Frequency	741	1790	13067	9194	8171	12259	3655	5066
Proportion	0.014	0.033	0.242	0.170	0.151	0.227	0.068	0.094

depth

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
53943	0	184	0.999	61.75	1.515	59.3	60.0	61.0	61.8	62.5	63.3	63.8

```
lowest : 43   44   50.8 51   52.2, highest: 72.2 72.9 73.6 78.2 79
```

table

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
53943	0	127	0.98	57.46	2.448	54	55	56	57	59	60	61

```
lowest : 43   44   49   50   50.1, highest: 71   73   76   79   95
```

price

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
53943	0	11602	1	3933	4012	544	646	950	2401	5324	9821	13107

```
lowest :   326   327   334   335   336, highest: 18803 18804 18806 18818 18823
```

x

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
53943	0	554	1	5.731	1.276	4.29	4.36	4.71	5.70	6.54	7.31	7.66

```
lowest : 0     3.73  3.74  3.76  3.77 , highest: 10.01 10.02 10.14 10.23 10.74
```

y

n	missing	distinct	Info	Mean	Gmd	.05	.10	.25	.50	.75	.90	.95
53943	0	552	1	5.735	1.269	4.30	4.36	4.72	5.71	6.54	7.30	7.65

```
lowest : 0     3.68  3.71  3.72  3.73 , highest: 10.1  10.16 10.54 31.8  58.9
```

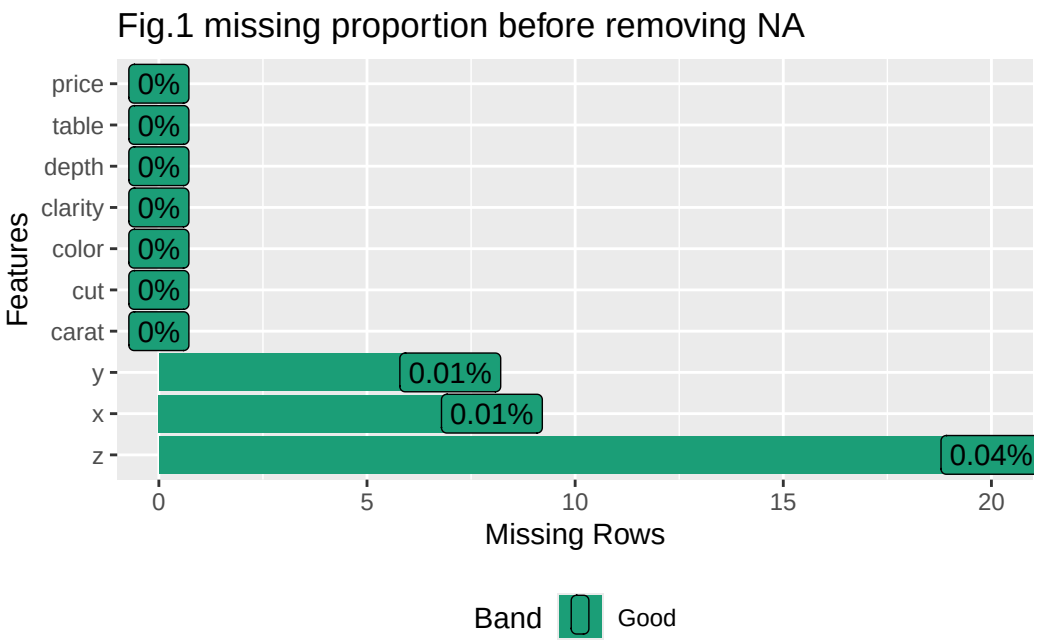
```
z
       n  missing distinct     Info     Mean      Gmd      .05      .10      .25      .50      .75      .90      .95
   53943        0      375        1    3.539   0.7901     2.65     2.69     2.91     3.53     4.04     4.52     4.73

lowest : 0    1.07 1.41 1.53 2.06, highest: 6.43 6.72 6.98 8.06 31.8
```

由 describe 可以發現 x,y,z(長、寬、高) 當中出現 0，這是不合理的量測數值，因此將其作為遺失值處理。

經檢查，此資料有 20 筆資料的長或寬或高為 0，與五萬多的樣本數相比算小，因此刪除。

此外還發現反應變數”價格”有離群值，在後續使用模型建立預測模式時需多加考量。

```
data$x[data$x==0] <- NA
data$y[data$y==0] <- NA
data$z[data$z==0] <- NA
DataExplorer::plot_missing(
  data,
  title = "Fig.1 missing proportion before removing NA")
```




```
data <- data[!(is.na(data$x) | is.na(data$y) | is.na(data$z)),]
```

我們認為僅僅觀察單一的長、寬、高並不具有任何意義，因此建立新變數體積 $= x \times y \times z$。

此外，一些類別型變數以 ordinal 的方式處理，賦予其從 1 開始的整數。

Cut(切工) 從 Fair 到 Very Good 分成 1-5 級

Color(鑽石顏色) 按照 D-J 排序為 1-7

Clarity(鑽石純淨程度) 按照 I1-IF 排序成 1-8

```
# 定義體積 = x * y * z
data$volume <- data$x * data$y * data$z
data$x <- NULL
data$y <- NULL
data$z <- NULL
# 發現體積中有離群值, 故將其刪除
data <- data[data$volume < 800]
# 定義類別順序
levelcut <- c("Fair", "Good", "Ideal", "Premium", "Very Good")
```

```
levelcolor <- c("D","E","F","G","H","I","J")
levelclarity <- c("I1","SI2","SI1","VS2","VS1","VVS2","VVS1","IF")

# 使用 match 進行編碼
data$cut <- match(data$cut,levelcut)
data$color <- match(data$color,levelcolor)
data$clarity <- match(data$clarity,levelclarity)
```

2.Data visualization for exploratory data analysis

從圖 2 的 Spearman 相關係數可以發現，鑽石的克拉數與體積幾乎與價格是正比，在後續分析的時候可以稍微注意，此外並無其他變數有特別的相關性。

```
corrplot::corrplot(cor(data, method = 'spearman'),
                   method = 'number',
                   mar = c(1, 0, 0, 0))
mtext("Fig.2 Spearman Correlation", side = 1, line = 4, cex = 1)
```

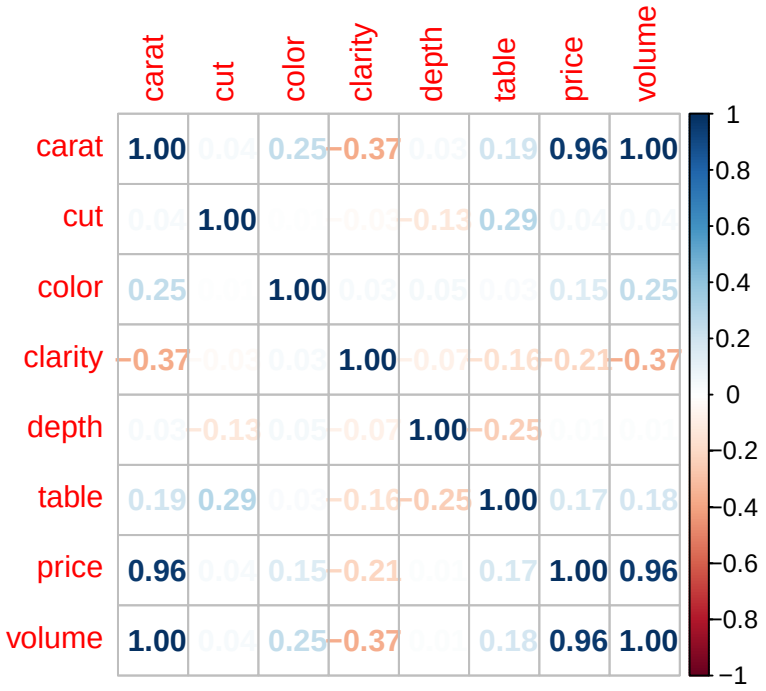
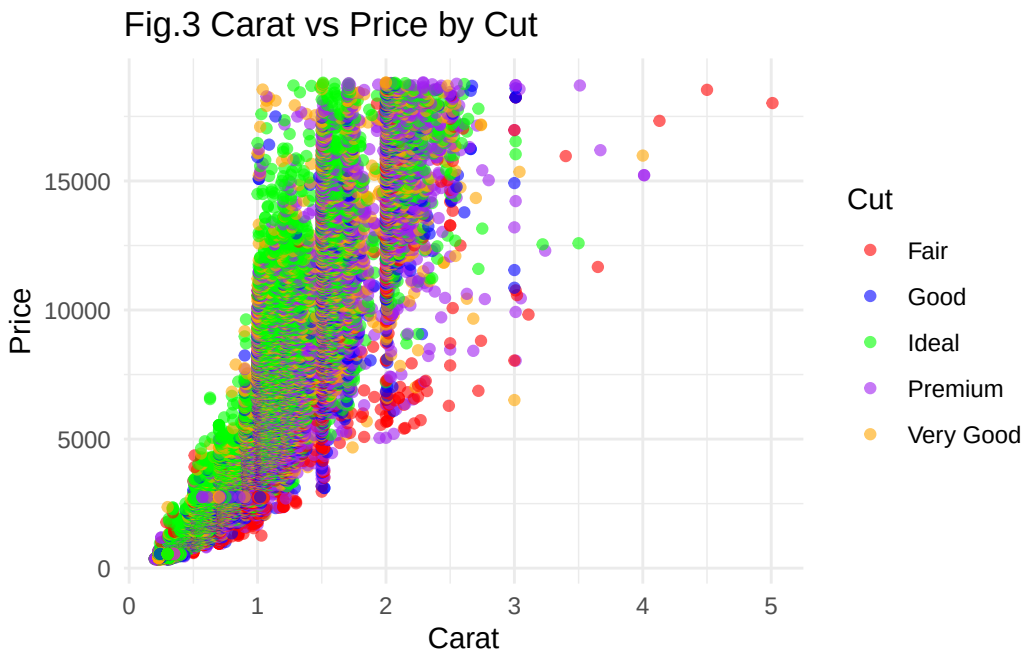


Fig.2 Spearman Correlation

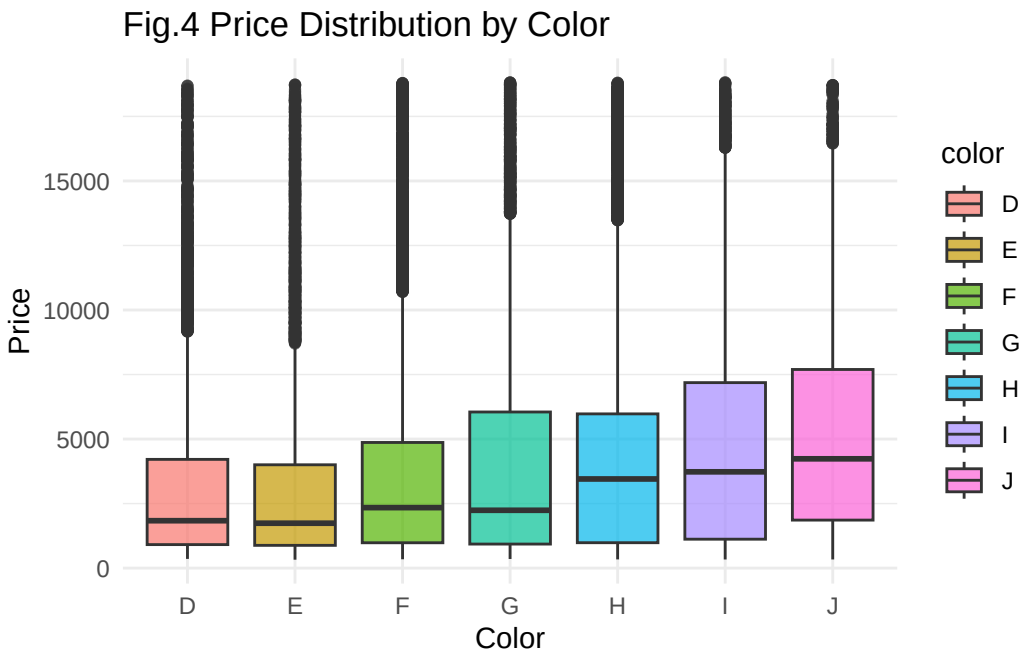
```
# 克拉對價格（加切工）
ggplot(data, aes(x = carat,
                 y = price,
                 color = factor(cut,
                                levels = c("1", "2", "3", "4", "5"),
                                labels = c("Fair", "Good", "Ideal", "Premium", "Very Good")))) +
  geom_point(alpha = 0.6) +
  labs(title = "Fig.3 Carat vs Price by Cut",
       x = "Carat",
       y = "Price",
       color = "Cut") +
  scale_color_manual(values = c( "Fair" = "red", "Good" = "blue", "Ideal" = "green", "Premium" = "purple
```

```
  theme_minimal()
```



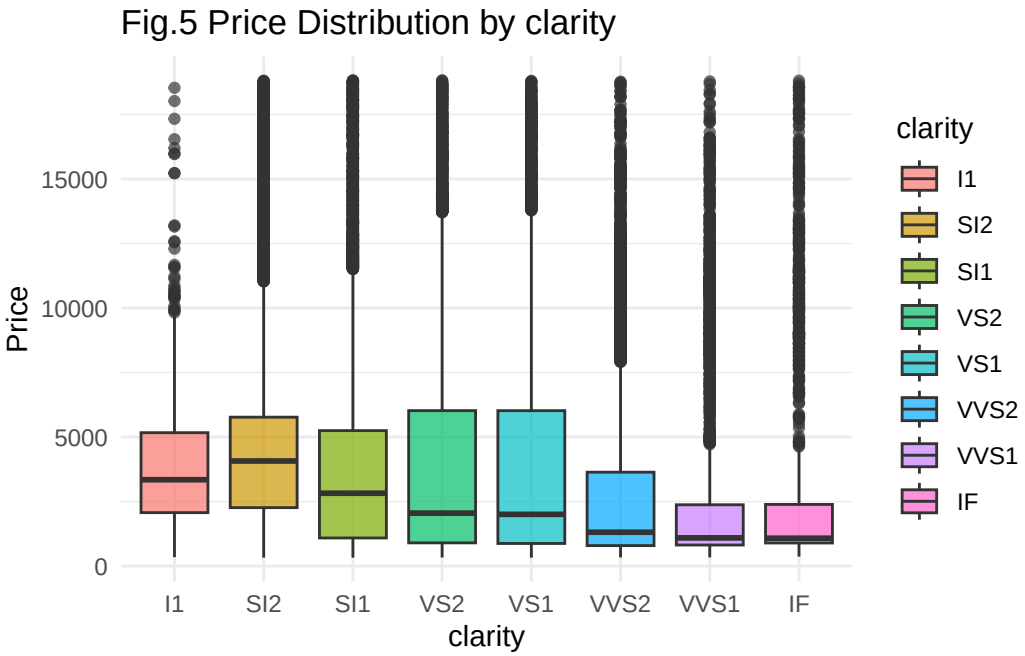
從圖 3 可發現大致上越重的鑽石價格越高。

```
# 顏色對價格圖
ggplot(data, aes(x = factor(color,
                        levels = c("1", "2", "3", "4", "5", "6", "7"),
                        labels = c("D", "E", "F", "G", "H", "I", "J")),
                 y = price,
                 fill = factor(color,
                        levels = c("1", "2", "3", "4", "5", "6", "7"),
                        labels = c("D", "E", "F", "G", "H", "I", "J")))) +
    geom_boxplot(alpha = 0.7) +
    labs(title = "Fig.4 Price Distribution by Color",
         x = "Color",
         y = "Price",
         fill = "color") +
    theme_minimal()
```



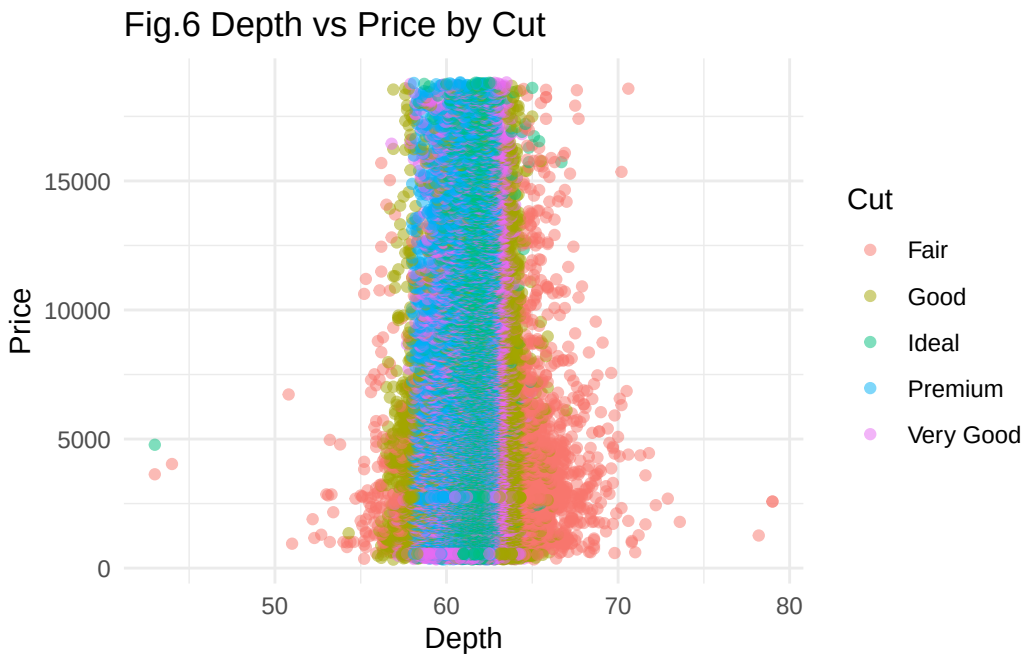
從圖 4 可發現當分類越靠近接近無色時價格越高。

```
# 淨度對價格
ggplot(data, aes(x = factor(data$clarity,
                        levels = c("1", "2", "3", "4", "5", "6", "7", "8"),
                        labels = c("I1","SI2","SI1","VS2","VS1","VVS2","VVS1","IF")),
                 y = price,
                 fill = factor(data$clarity,
                        levels = c("1", "2", "3", "4", "5", "6", "7", "8"),
                        labels = c("I1","SI2","SI1","VS2","VS1","VVS2","VVS1","IF")))) +
  geom_boxplot(alpha = 0.7) +
  labs(title = "Fig.5 Price Distribution by clarity",
       x = "clarity",
       y = "Price",
       fill = "clarity") +
  theme_minimal()
```



從圖 5 可發現單一淨度指標對價格並沒有直接關連, 高淨度的鑽石未必會有高價格。

```
# 深度對價格
ggplot(data, aes(x = depth,
                 y = price,
                 color = factor(data$cut,
                       levels = c("1", "2", "3", "4", "5"),
                       labels = c("Fair", "Good", "Ideal", "Premium", "Very Good")))) +
  geom_point(alpha = 0.5) +
  labs(title = "Fig.6 Depth vs Price by Cut",
       x = "Depth",
       y = "Price",
       color = "Cut") +
  theme_minimal()
```

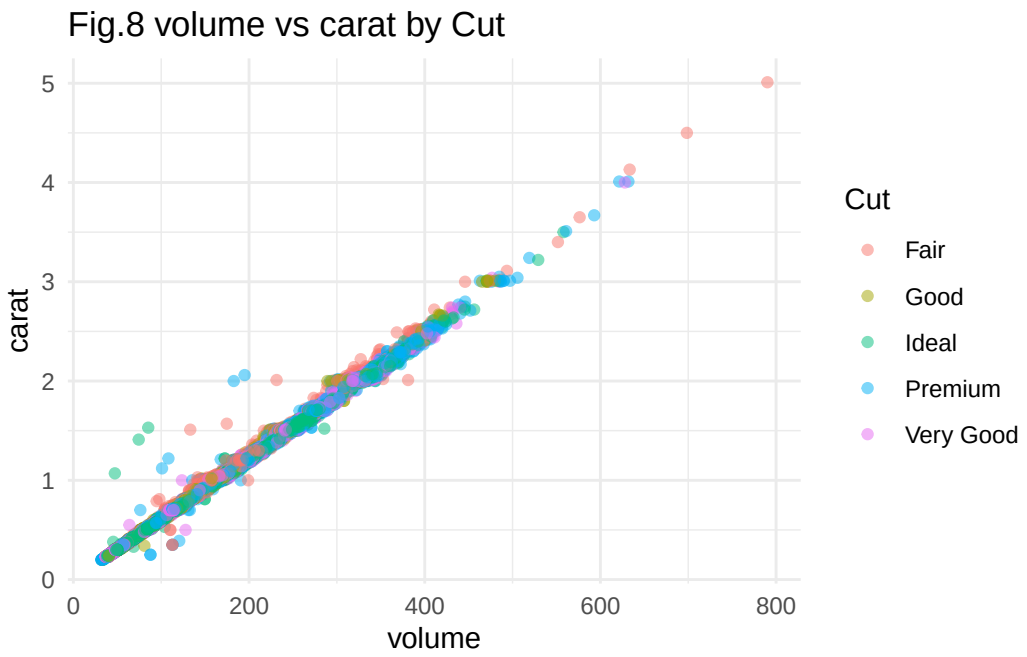


從圖 6 可發現深度和價格沒有相關, 且深度大多集中於 60 附近, 推測是因為深度比例在此區間能切割出最明亮的鑽石。

```
# 檯面尺寸對價格
ggplot(data, aes(x = table,
                 y = price,
                 color =factor(data$cut,
                       levels = c("1", "2", "3", "4", "5"),
                       labels = c("Fair", "Good", "Ideal", "Premium", "Very Good")))) +
  geom_point(alpha = 0.5) +
  labs(title = "Fig.7 table vs Price by Cut",
       x = "table",
       y = "Price",
       color = "Cut") +
  theme_minimal()
```

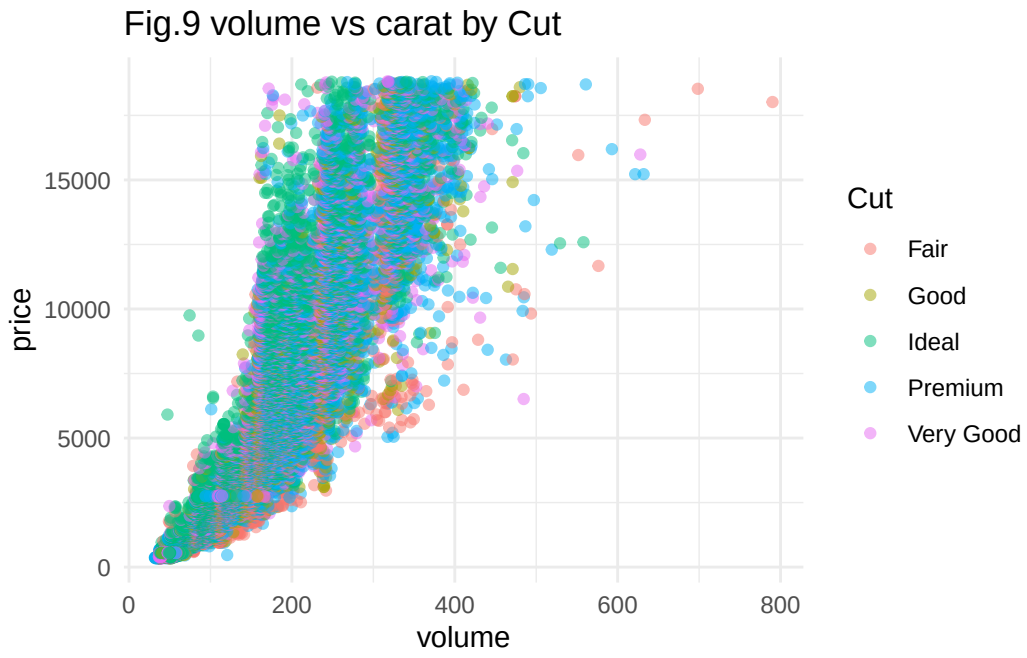
從圖 7 可發現檯面尺寸對價格沒有相關, 且檯面尺寸約集中在 56~62 之間, 推測也是在這個區間中能切割出最好的鑽石

```
# 體積對重量
ggplot(data,aes(x = volume,
                y = carat,
                color = factor(data$cut,
                       levels = c("1", "2", "3", "4", "5"),
                       labels = c("Fair", "Good", "Ideal", "Premium", "Very Good"))))+
  geom_point(alpha = 0.5)+
  labs(title ="Fig.8 volume vs carat by Cut",
      x ="volume",
      y ="carat",color ="Cut") +
  theme_minimal()
```



由圖 8 可發現體積和重量（克拉）有近乎正比關係

```
ggplot(data, aes(x = volume,
                 y = price,
                 color =factor(data$cut,
                       levels = c("1", "2", "3", "4", "5"),
                       labels = c("Fair", "Good", "Ideal", "Premium", "Very Good")))) +
  geom_point(alpha = 0.5) +
  labs(title = "Fig.9 volume vs carat by Cut",
       x = "volume",
       y = "price",
       color = "Cut") +
  theme_minimal()
```



由圖 9 可發現與圖 3 非常類似，可知道從重量和從體積對上價格能得到相同結果

典型相關分析 (CCA)

分析: 幾何特性 vs. 做工及價格之間的關係

```
# 選擇兩組變數
X <- data[, c("carat", "color", "clarity", "volume")]
Y <- data[, c("price", "cut", "depth", "table")]
cca <- cancor(X,Y)
print(cca)
```

```
$cor
[1] 0.95172554 0.52416801 0.07949970 0.01126333

$xcoef
                 [,1]          [,2]          [,3]          [,4]
carat   -3.613020e-03  0.1871277417  0.0571111515  0.0344228776
color    3.584621e-04  0.0002197235  0.0005444205 -0.0025843239
clarity -5.881181e-04 -0.0006368735  0.0025672725  0.0008290882
volume  -3.915824e-05 -0.0011613396 -0.0003465559 -0.0001993115

$ycoef
               [,1]          [,2]          [,3]          [,4]
price -1.074556e-06 -1.219673e-07  1.343038e-07  4.872832e-09
cut    8.470587e-05 -3.778534e-04 -4.155291e-04 -4.256938e-03
depth -4.714159e-05  2.760516e-03  1.327289e-03 -8.880334e-04
table -7.900934e-05  1.419096e-03 -1.454370e-03  2.112245e-04

$xcenter
      carat       color     clarity      volume
  0.7976816   3.5939911   4.0514651 129.8015976

$ycenter
      price         cut       depth       table
3930.845122    3.553097   61.749483   57.457007
```

```
cca$cor
```

```
[1] 0.95172554 0.52416801 0.07949970 0.01126333
```

```
cca$xcoef;cca$ycoef
```

```
                 [,1]          [,2]          [,3]          [,4]
carat   -3.613020e-03  0.1871277417  0.0571111515  0.0344228776
color    3.584621e-04  0.0002197235  0.0005444205 -0.0025843239
clarity -5.881181e-04 -0.0006368735  0.0025672725  0.0008290882
volume  -3.915824e-05 -0.0011613396 -0.0003465559 -0.0001993115
               [,1]          [,2]          [,3]          [,4]
price -1.074556e-06 -1.219673e-07  1.343038e-07  4.872832e-09
cut    8.470587e-05 -3.778534e-04 -4.155291e-04 -4.256938e-03
depth -4.714159e-05  2.760516e-03  1.327289e-03 -8.880334e-04
table -7.900934e-05  1.419096e-03 -1.454370e-03  2.112245e-04
```

發現:

第一典型相關變數:最大典型相關係數為0.9513,第一典型變數主要由carat和table貢獻組成

第二典型相關變數:最大典型相關係數為0.2112(相關性低)

```
X_loadinds <- cor(X,as.matrix(X) %*% cca$xcoef)
Y_loadinds <- cor(Y,as.matrix(Y) %*% cca$ycoef)
X_loadinds;Y_loadinds
```

```
              [,1]        [,2]       [,3]       [,4]
carat   -0.9712718  0.10329533 -0.1462553 -0.1567498
color   -0.1821301  0.08305418  0.2819089 -0.9383272
clarity  0.1613851 -0.31157473  0.9171247  0.1890985
volume  -0.9725448  0.05867045 -0.1556928 -0.1627085
```

```
              [,1]        [,2]        [,3]        [,4]
price -0.999094752 -0.03261789  0.01970337 -0.01890836
cut   -0.022582242 -0.16211424 -0.29373299 -0.94176958
depth  0.003096597  0.71910138  0.68255501 -0.13039280
table -0.159873694  0.43667075 -0.88418926  0.04436677
```

發現:

第一典型變數主要受carat(-),volume(-)和price(-)影響

第二典型變數主要受clarity(+),depth(-)和table(-)影響

3.Construct a predictive model for price

Data splitting

資料集以 7:3 比例切割成訓練集和測試集。

```
set.seed(123)
train_ratio <- 0.7
n <- nrow(data)
train_indices <- sample(1:n, size = floor(train_ratio * n))
train_data <- data[train_indices, ]
test_data <- data[-train_indices, ]
```

Linear regression model

我們採用線性回歸模型搭配 AIC 的逐步回歸進行變數選取。

```
model.1 <- stepAIC(
  lm(price ~ ., data = train_data),
  direction = "both",
  trace = 0)
summary(model.1)
```

```
Call:
lm(formula = price ~ carat + cut + color + clarity + depth +
    table + volume, data = train_data)

Residuals:
     Min       1Q   Median       3Q      Max
-14594.5   -692.2   -171.0    554.0   8902.5
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -240.830    433.162  -0.556    0.578
carat       2805.484    325.212   8.627  < 2e-16 ***
cut           75.969      6.287  12.084  < 2e-16 ***
color       -320.259      3.924 -81.608  < 2e-16 ***
clarity      526.283      4.175 126.043  < 2e-16 ***
depth        -28.831      5.204  -5.540 3.04e-08 ***
table        -40.811      3.239 -12.599  < 2e-16 ***
volume        37.058      2.008  18.454  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1224 on 37736 degrees of freedom
Multiple R-squared:  0.9055,    Adjusted R-squared:  0.9055
F-statistic: 5.167e+04 on 7 and 37736 DF,  p-value: < 2.2e-16
```

```
vif(model.1)
```

```
     carat        cut      color    clarity      depth      table     volume
594.410725   1.052485   1.115091   1.189357   1.413459   1.316505 590.073121
```

```
# remove carat
model.2 <- stepAIC(
  lm(price ~.-carat  , data = train_data),
  direction = "both",
  trace = 0)
summary(model.2)
```

```
Call:
lm(formula = price ~ (carat + cut + color + clarity + depth +
    table + volume) - carat, data = train_data)

Residuals:
     Min       1Q   Median       3Q      Max
-14394.9   -693.4   -171.1    554.5  10270.3

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.020e+03  3.813e+02  -5.297 1.18e-07 ***
cut          7.312e+01  6.284e+00  11.636  < 2e-16 ***
color       -3.197e+02  3.928e+00 -81.402  < 2e-16 ***
clarity      5.251e+02  4.177e+00 125.703  < 2e-16 ***
depth       -8.981e+00  4.672e+00  -1.922   0.0546 .
table       -3.110e+01  3.040e+00 -10.229  < 2e-16 ***
volume       5.436e+01  9.369e-02 580.220  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1226 on 37737 degrees of freedom
Multiple R-squared:  0.9053,    Adjusted R-squared:  0.9053
F-statistic: 6.016e+04 on 6 and 37737 DF,  p-value: < 2.2e-16
```

```
vif(model.2)
```

```
     cut    color  clarity    depth    table   volume
```

```
1.049590 1.114809 1.188033 1.137104 1.157381 1.282051
```

```r
oldformula <- model.2$formula

# remove depth
newformula <- update(oldformula, .~.-depth)
model.2 <- lm( newformula, data = train_data)
summary(model.2)
```

```
Call:
lm(formula = newformula, data = train_data)

Residuals:
     Min       1Q   Median       3Q      Max
-14427.1   -694.1   -170.9    555.7  10191.6

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.677e+03  1.683e+02  -15.91   <2e-16 ***
cut          7.505e+01  6.204e+00   12.10   <2e-16 ***
color       -3.201e+02  3.922e+00  -81.61   <2e-16 ***
clarity      5.260e+02  4.152e+00  126.68   <2e-16 ***
table       -2.946e+01  2.918e+00  -10.09   <2e-16 ***
volume       5.436e+01  9.369e-02  580.20   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1226 on 37738 degrees of freedom
Multiple R-squared:  0.9053,    Adjusted R-squared:  0.9053
F-statistic: 7.218e+04 on 5 and 37738 DF,  p-value: < 2.2e-16
```

檢查 VIF 可發現 carat 和 volume 值很大，故試將 carat 刪除建立 model.2

移除 carat 之後發現，depth 在顯著水準 0.05 之下變得不顯著，因此移除此變數，再配適一個模型。

而 model.2 的 R-squared = 0.9053，VIF 檢查後各變數也無異常。

因此以 model.2 作為預測模型。

以下將測試集資料輸入預測模型並評估表現:

```r
predictions <- predict(model.2, newdata = test_data)
# 計算 MSE 和 RMSE
rmse <- sqrt(mean((test_data$price - predictions)^2))
test_r2 <- 1 - sum((predictions - test_data$price)^2) /
               sum((test_data$price - mean(test_data$price))^2)
# 輸出結果
cat("Testing R^2:", test_r2, "\n")
```

```
Testing R^2: 0.904271
```

```r
cat("Root Mean Squared Error (RMSE):", rmse, "\n")
```

```
Root Mean Squared Error (RMSE): 1236.306
```

經過測試後，這個模型在測試集上的 R-square = 0.9043，RMSE = 1236.306

以線性回歸模型來說，已經是不錯的表現了。