Multiple Linear Regression and related analysis

End-semester presentation

Stat-Methods 2

Indian Statistical Institute, Kolkata

May 19,2022

Srijan Chattopadhyay(BS2126)
Rudrashis Bardhan(BS2118)
Agniv Chakraborty(BS2106)

Table of Contents

Introduction

Diamonds are one of the most valued products occurring naturally. Besides having their shine and lustre, diamonds are the hardest substance in nature. This leads to it being used both for jewellery as well as industrial purposes. In this presentation, we aim to predict the price of diamonds(in USD) based on several co-variates: Weight(in carats), Length(in mm), Width(in mm), Depth(in mm), Table(as percentage), Price per carat based on colour and cut(in USD per carat).

Physical Structure of a real diamond

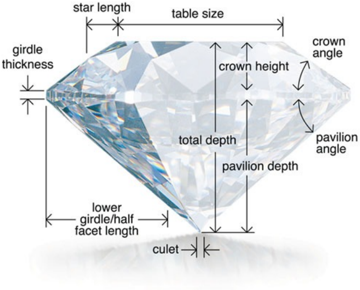

Figure: Labels of a diamond

Overview of the data

Multiple Linear Regression and related analysis

Introduction

Overview of the data

Multiple Linear Regression

Correlation

Model fitting and analysis

Further analysis

Simulation from the fitted model

Overall inference and conclusion

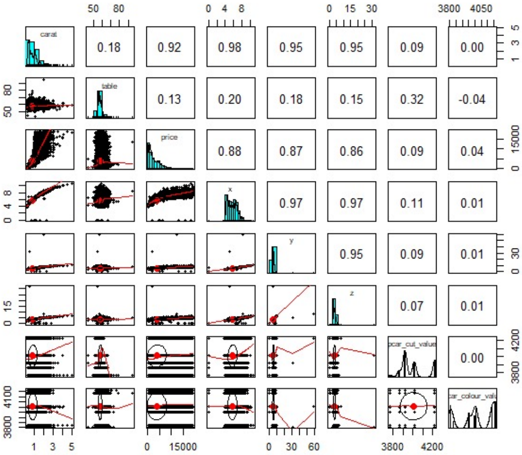

Figure: A total overview of the data

What is Multiple Linear Regression?

Multiple linear regression is a statistical technique which uses several explanatory variables to predict the outcome of a response variable. Suppose Y is the response variable and $\mathrm{X_j}$'s are the predictors $\forall j = 1(1)n$, then a linear equation consisting of $\mathrm{X_j}$'s is

$$Y_i = \beta_0 + \beta_1 X_{1\mathrm{i}} + ... + \beta_\mathrm{n} X_{\mathrm{ni}} + \epsilon_\mathrm{i}, \forall i = 1(1)k$$

or in a matrix form

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

$Y = [Y_1 \ Y_2 \ .... \ Y_\mathrm{k}]^\mathrm{T}, \epsilon = [\epsilon_1 \ \epsilon_2 \ .... \ \epsilon_\mathrm{k}]^\mathrm{T}, \beta = [\beta_0 \ \beta_1 \ ... \ \beta_\mathrm{n}]^\mathrm{T}$
and X is the following matrix

$$\begin{bmatrix} 1 & X_{11} \ ... \ X_{\mathrm{n}1} \\ 1 & X_{12} \ ... \ X_{\mathrm{n}2} \\ . & \\ 1 & X_{1\mathrm{k}} \ ... \ X_{\mathrm{nk}} \end{bmatrix}$$

Model assumptions and OLS solution

The model assumptions are $\epsilon \sim N(0, \sigma^2 I)$ where Y is a stochastic variable, while X_j's are non-stochastic. Hence, by properties of normal,

$$Y \sim N(X\beta, \sigma^2 I)$$

Now using ordinary least square method, the optimum solution of β is

$$\hat{\beta}_{\mathrm{OLS}} = (X^{\mathrm{T}}X)^{\text{-g}}X^{\mathrm{T}}Y$$

['-g' stands for the G-inverse]

Correlation

```
> cor(final_data)
                          carat       table      price          x           y           z ppcar_cut_values
carat               1.000000000  0.18161755 0.92159130 0.97509423 0.951722199 0.953387381      0.093597304
table               0.181617547  1.00000000 0.12713390 0.19534428 0.183760147 0.150928692      0.316183345
price               0.921591301  0.12713390 1.00000000 0.88443516 0.865420898 0.861249444      0.091071372
x                   0.975094227  0.19534428 0.88443516 1.00000000 0.974701480 0.970771799      0.105008713
y                   0.951722199  0.18376015 0.86542090 0.97470148 1.000000000 0.952005716      0.092238169
z                   0.953387381  0.15092869 0.86124944 0.97077180 0.952005716 1.000000000      0.065166998
ppcar_cut_values    0.093597304  0.31618335 0.09107137 0.10500871 0.092238169 0.065166998      1.000000000
ppcar_colour_values 0.002862426 -0.03519675 0.03762202 0.01069639 0.009803924 0.009271553     -0.001955349
                    ppcar_colour_values
carat                       0.002862426
table                      -0.035196749
price                       0.037622019
x                           0.010696389
y                           0.009803924
z                           0.009271553
ppcar_cut_values           -0.001955349
ppcar_colour_values         1.000000000
>
```

Figure: Correlation of the data set

Partial Correlation

Multiple Linear Regression and related analysis

Introduction

Overview of the data

Multiple Linear Regression

Correlation

Model fitting and analysis

Further analysis

Simulation from the fitted model

Overall inference and conclusion

```
> pcor(final_data)
$estimate
                           carat        table       price           x            y            z ppcar_cut_values
carat                1.000000000  0.050339811  0.57843898  0.50024143 -0.006618569  0.138781500     -0.038996507
table                0.050339811  1.000000000 -0.10980713  0.10034376 -0.001403681 -0.128436527      0.287649847
price                0.578438978 -0.109807135  1.00000000 -0.08554743  0.028071580 -0.073632039      0.046364083
x                    0.500241432  0.100343760 -0.08554743  1.00000000  0.559131851  0.454127129      0.099303601
y                   -0.006618569 -0.001403681  0.02807158  0.55913185  1.000000000  0.100565146     -0.027915495
z                    0.138781500 -0.128436527 -0.07363204  0.45412713  0.100565146  1.000000000     -0.101128627
ppcar_cut_values    -0.038996507  0.287649847  0.04636408  0.09930360 -0.027915495 -0.101128627      1.000000000
ppcar_colour_values -0.082294328 -0.029691104  0.09384164  0.03693039 -0.004810729  0.001290787      0.002693259
                    ppcar_colour_values
carat                      -0.082294328
table                      -0.029691104
price                       0.093841642
x                           0.036930389
y                          -0.004810729
z                           0.001290787
ppcar_cut_values            0.002693259
ppcar_colour_values         1.000000000
```

Figure: Partial Correlation of the data set

Semi-Partial Correlation


```
> spcor(final_data)
$estimate
                           carat         table        price           x            y            z ppcar_cut_values
carat                1.000000000  0.0090355526  0.127117574  0.10356455 -0.001186495  0.025121575     -0.006995977
table                0.046410521  1.0000000000 -0.101722879  0.09286280 -0.001292477 -0.119248897      0.276549190
price                0.267920363 -0.0417404420  1.000000000 -0.03244100  0.010610365 -0.027895852      0.017536433
x                    0.081921139  0.0143009506 -0.012175282  1.00000000  0.095630291  0.072278233      0.014151223
y                   -0.001469365 -0.0003116197  0.006234391  0.14971814  1.000000000  0.022439374     -0.006199699
z                    0.032482735 -0.0300191484 -0.017113752  0.11814855  0.023428940  1.000000000     -0.023561569
ppcar_cut_values    -0.036698113  0.2824268932  0.043645202  0.09384357 -0.026260451 -0.095585906      1.000000000
ppcar_colour_values -0.082089299 -0.0295296906  0.093703834  0.03673849 -0.004782522  0.001283205      0.002677446
                    ppcar_colour_values
carat                     -0.0148025888
table                     -0.0273509081
price                      0.0356130046
x                          0.0052403135
y                         -0.0010680016
z                          0.0002991939
ppcar_cut_values           0.0025326041
ppcar_colour_values        1.0000000000
```

Figure: Semi-Partial Correlation of the data set

Regression Model (R-Output)

The R-Output looks like the following

```
> summary(model)

Call:
lm(formula = price ~ carat + depth + table + length + width +
    ppcar_colour_values + ppcar_cut_values, data = final_data)

Residuals:
     Min       1Q   Median       3Q      Max
-18438.0   -800.6    -25.9    532.1  12526.0

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)         -2.377e+03  3.309e+02  -7.185 6.79e-13 ***
carat                7.827e+03  1.422e+01 550.327  < 2e-16 ***
depth               -5.270e+00  3.916e+01  -0.135    0.893
table               -8.319e+01  3.153e+00 -26.384  < 2e-16 ***
length              -3.699e+01  3.377e+01  -1.095    0.273
width               -3.106e+01  2.597e+01  -1.196    0.232
ppcar_colour_values  7.690e-01  5.419e-02  14.192  < 2e-16 ***
ppcar_cut_values     5.415e-01  5.092e-02  10.634  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1531 on 53932 degrees of freedom
Multiple R-squared:  0.8528,    Adjusted R-squared:  0.8528
F-statistic: 4.463e+04 on 7 and 53932 DF,  p-value: < 2.2e-16
```

Figure: R-Outputs of the fitted model

Details of the fitted model

As shown,the model has been fitted between response(Price) and predictors(Weight, Length, Width, Depth, Table, Price per carat based on colour and cut).The estimation of β is

$$\hat{\beta}_{\text{OLS}} = \begin{bmatrix} -2377 \\ 7827 \\ -5.27 \\ -83.19 \\ -36.99 \\ -31.06 \\ 0.769 \\ 0.5145 \end{bmatrix}$$

Details of the fitted model

Now, we have calculated the sample mean and sample variance of the errors. The sample mean is -0.05388688, which is very close to 0, as it should be because of the model assumption and as we increase the number of data points, the sample mean will converge to 0 almost surely. The sample variance of the errors is 5165.928^2, i.e. ϵ follows $N(0, 5165.928^2)$.

QQ Plot of predicted error(residuals) and distribution of actual error

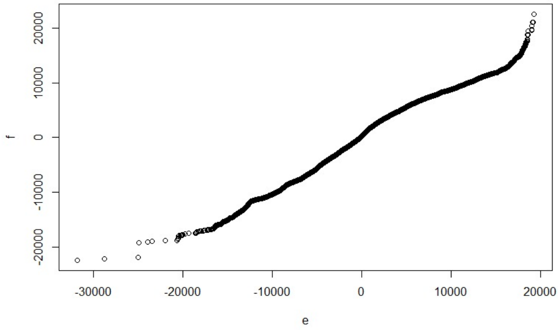

Figure: QQ Plot

Bi-Variate plot of the actual and fitted response

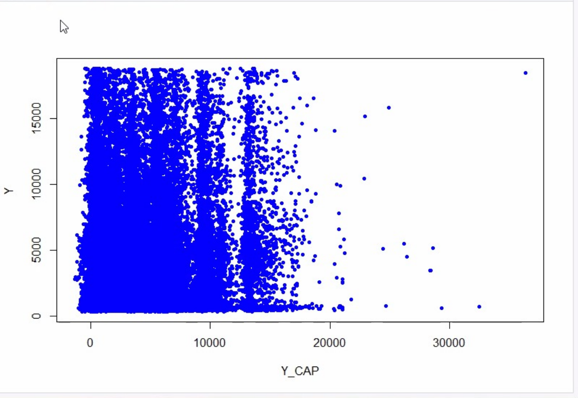

Figure: Bi-variate plot

Density plot of the residuals of the model

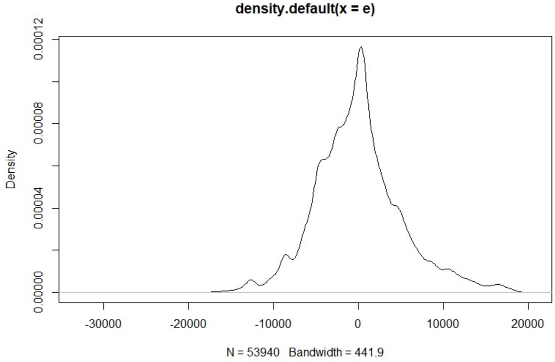

Figure: Density-plot of the residuals

Traversing through different permutations of the data set

Now we want to make clear that weight is the main game-changing predictor on which the price depends mostly, i.e., for same weight, length, width, depth and table has a negative role. That means, for same weight, dense diamond costs more. So, now we want to see how much only weight explain the model. And, then we want to see, when weight is not fixed, then what is the role of length? or maybe the table? Or, both length and table?

Weight(Carat) Only Model

```
Call:
lm(formula = price ~ carat, data = final_data)

Residuals:
     Min       1Q   Median       3Q      Max
-18585.3   -804.8    -18.9    537.4  12731.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -2256.36      13.06  -172.8   <2e-16 ***
carat        7756.43      14.07   551.4   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1549 on 53938 degrees of freedom
Multiple R-squared:  0.8493,	Adjusted R-squared:  0.8493
F-statistic: 3.041e+05 on 1 and 53938 DF,  p-value: < 2.2e-16
```

Figure: Weight-Price

Length only Model

```
Call:
lm(formula = price ~ length, data = final_data)

Residuals:
   Min     1Q Median     3Q    Max 
 -4150  -2934  -1486   1382  15167 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2905.70      89.31   32.53   <2e-16 ***
length        179.21      15.29   11.72   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3984 on 53938 degrees of freedom
Multiple R-squared:  0.002539,	Adjusted R-squared:  0.002521 
F-statistic: 137.3 on 1 and 53938 DF,  p-value: < 2.2e-16
```

Figure: Length-Price

Table only model

```
Call:
lm(formula = price ~ table, data = final_data)

Residuals:
   Min     1Q Median     3Q    Max
 -6522  -2751  -1490   1368  15746

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -9109.047    438.450  -20.78   <2e-16 ***
table         226.984      7.625   29.77   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3957 on 53938 degrees of freedom
Multiple R-squared:  0.01616,	Adjusted R-squared:  0.01614
F-statistic: 886.1 on 1 and 53938 DF,  p-value: < 2.2e-16
```

Figure: Table-Price

Table and Length Model

```
Call:
lm(formula = price ~ length + table, data = final_data)

Residuals:
   Min     1Q Median     3Q    Max 
 -6262  -2738  -1458   1366  15851 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -10087.031    445.916  -22.62   <2e-16 ***
length         176.325     15.170   11.62   <2e-16 ***
table          226.417      7.616   29.73   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3952 on 53937 degrees of freedom
Multiple R-squared:  0.01862,	Adjusted R-squared:  0.01858 
F-statistic: 511.7 on 2 and 53937 DF,  p-value: < 2.2e-16
```

Figure: Length and Table vs Price

Taking all but categorical ones


```
Call:
lm(formula = price ~ carat + depth + table + length + width,
    data = final_data)

Residuals:
     Min       1Q   Median       3Q      Max
-18763.9   -797.0    -29.3    538.8  12437.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2595.637    174.746  14.854   <2e-16 ***
carat       7842.937     14.236 550.905   <2e-16 ***
depth        -20.009     39.258  -0.510   0.6103
table        -74.819      3.008 -24.870   <2e-16 ***
length       -66.070     33.822  -1.953   0.0508 .
width        -30.107     26.049  -1.156   0.2478
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1535 on 53934 degrees of freedom
Multiple R-squared:  0.8519,	Adjusted R-squared:  0.8519
F-statistic: 6.206e+04 on 5 and 53934 DF,  p-value: < 2.2e-16
```

Figure: Removing the categorical predictors

Removing Length


```
Call:
lm(formula = price ~ carat + depth + table + width + ppcar_colour_values +
    ppcar_cut_values, data = final_data)

Residuals:
     Min       1Q   Median       3Q      Max
-18436.1   -800.8    -25.5    532.6  12527.2

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)
(Intercept)         -2.398e+03  3.303e+02  -7.259 3.96e-13 ***
carat                7.827e+03  1.422e+01 550.348  < 2e-16 ***
depth               -3.202e+01  3.061e+01  -1.046   0.2955
table               -8.319e+01  3.153e+00 -26.382  < 2e-16 ***
width               -5.059e+01  1.889e+01  -2.679   0.0074 **
ppcar_colour_values  7.722e-01  5.411e-02  14.270  < 2e-16 ***
ppcar_cut_values     5.421e-01  5.092e-02  10.646  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1531 on 53933 degrees of freedom
Multiple R-squared:  0.8528,    Adjusted R-squared:  0.8528
F-statistic: 5.207e+04 on 6 and 53933 DF,  p-value: < 2.2e-16
```

Figure: Removing the predictor Length

Simulation from the fitted model

Simulation is the process of generating more and more data without actually collecting data. We may now want to see the effects of only one or two predictors on the response. Then, we need to first simulate independently from the empirical cdf's (preferably) of those predictors and using the model, after finding the predicted Y and taking the average.

We know, that the cdf of any distribution follows Uniform(0,1). And,the quantile function is defined as

$$Q(p) = inf\{x : F(x) \geq p\}$$

Hence, we first simulated from Uniform(0,1) and applied the Quantile function upon those to get simulated values from our target distribution.

Simulation from the fitted model

As, 'weight(carat)' is the best predictor according to p-value comparison as declared before, hence, we may be interested in finding the value of expected price of diamond when 'weight' is fixed at some value.

Question 1:

What will be the approximate price of diamond when the weight is fixed at 0.5 carats and the table is fixed at 2 percent?

Answer: 1739 USD

Question 2:
What will be the approximate price of diamond when the weight is fixed at 2 carats and the length is fixed at 20 mm?

Answer: 9373.15 USD

Overall inference from the presentation

So from our findings, we have concluded that a higher trend of prices for diamonds having higher weight(carat) with smaller values of table(in percent),depth, length, width(in mm) implies higher density diamonds or diamonds with higher precision.