Pstat 126 final project

2025-12-06

Part 1

#1. Select a random sample

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.1     v stringr   1.5.2
## v ggplot2   4.0.0     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
data <- read_csv("Diamonds Prices2022.csv", show_col_types = FALSE)
```

```
## New names:
## * `` -> `...1`
```

```
set.seed(123)
sample <- sample_n(data, 1000)
```

#2. Describe all the variables

```
summary(sample)
```

```
##       ...1           carat            cut               color
##  Min.   :   31   Min.   :0.2300   Length:1000        Length:1000
##  1st Qu.:13223   1st Qu.:0.4000   Class :character   Class :character
##  Median :26435   Median :0.7100   Mode  :character   Mode  :character
##  Mean   :26658   Mean   :0.7962
##  3rd Qu.:40271   3rd Qu.:1.0400
##  Max.   :53912   Max.   :2.3200
##    clarity             depth           table           price
##  Length:1000        Min.   :57.50   Min.   :52.00   Min.   :  368.0
##  Class :character   1st Qu.:61.00   1st Qu.:56.00   1st Qu.:  956.8
##  Mode  :character   Median :61.90   Median :57.00   Median : 2522.0
##                     Mean   :61.79   Mean   :57.41   Mean   : 3935.9
##                     3rd Qu.:62.50   3rd Qu.:59.00   3rd Qu.: 5301.8
##                     Max.   :69.60   Max.   :66.00   Max.   :18706.0
##        x               y               z
##  Min.   :3.920   Min.   :3.960   Min.   :0.000
##  1st Qu.:4.707   1st Qu.:4.718   1st Qu.:2.900
##  Median :5.720   Median :5.720   Median :3.540
```

```
##  Mean   :5.733   Mean   :5.736   Mean   :3.539
##  3rd Qu.:6.530   3rd Qu.:6.513   3rd Qu.:4.030
##  Max.   :8.570   Max.   :8.520   Max.   :5.280
```

Categorical variables:

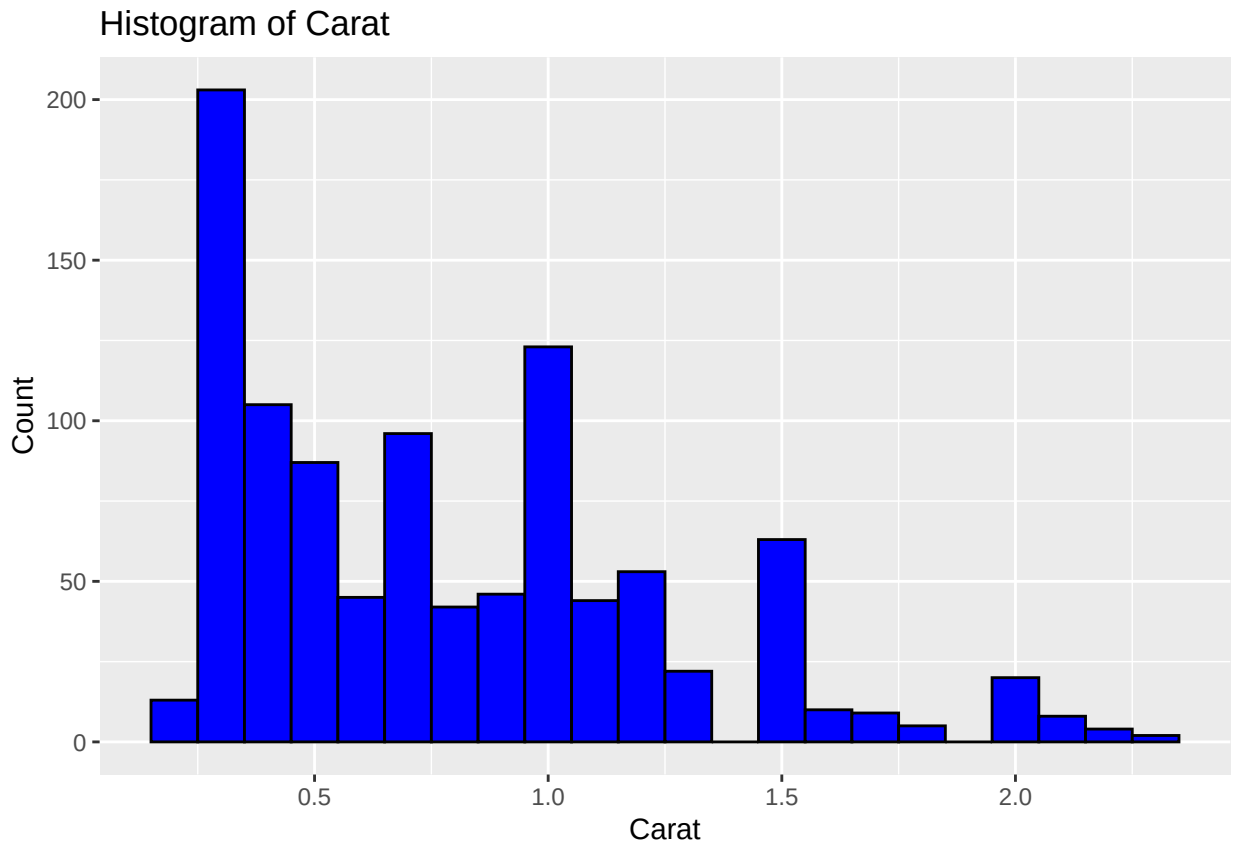
- Cut (character, categorical) – quality of the diamond cut; Categories: Fair, Good, Very Good, Premium, Ideal. Most diamonds are Ideal or Premium, fewer are Fair.
- color (character, categorical) – diamond color grade, Categories: D (best) to J (worst). Most diamonds are G, H, or E color; D and J are less common.
- clarity (character, categorical) – diamond clarity (IF, VVS1, VVS2, VS1, VS2, SI1, SI2, I1). Most diamonds are VS2, VS1, or SI1, fewer are IF or I1.

Continuous/numeric variables:

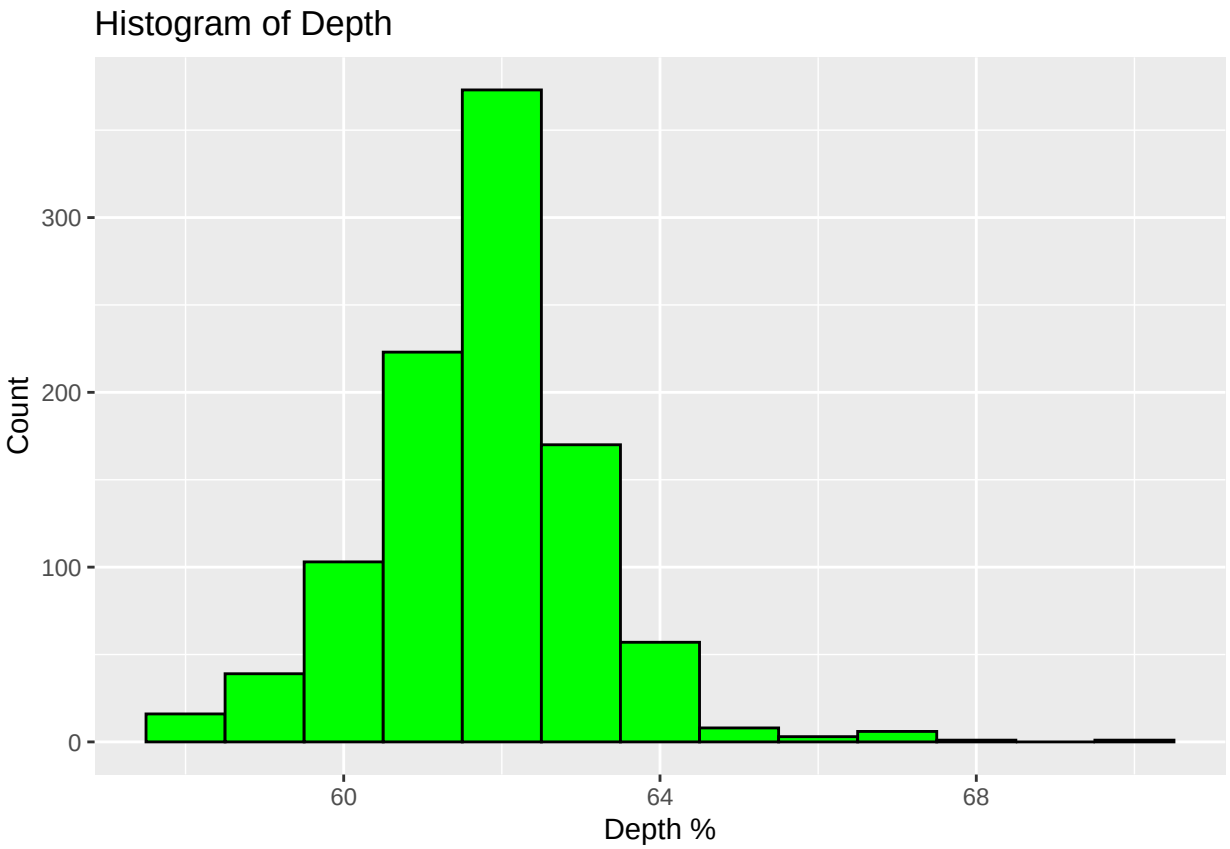
- Carat - Ranges from 0.23 to 2.32. The distribution is right-skewed, with a median of 0.71 and a mean of 0.7962. Most diamonds are smaller than 1 carat, with a few larger diamonds raising the mean.
- Depth - Ranges from 57.5% to 69.6%. The distribution is approximately symmetric, with a median of 61.9% and a mean of 61.79%. Most diamonds have depth between 61% and 63%, and the narrow IQR indicates low variation.
- Table - Ranges from 52% to 66%, slightly right-skewed, with a median of 57% and mean of 57.41%. Most values fall between 56% and 59%, showing limited variation.
- Price - Ranges from $368 to $18,706. Right-skewed, with a median of $2,522 and mean of $3,936, reflecting that a few very expensive diamonds pull the average up.
- X, Y, and Z (physical dimensions in mm) Show approximately normal-like distributions, with medians x = 5.72 mm, y = 5.72 mm, z = 3.54 mm. Some extreme maximum values (x = 8.57 mm, y = 8.52 mm, z = 5.28 mm) and zeros indicate potential outliers or data errors.

Overall, the numeric data shows a combination of right-skewed distributions for size/price and stable distributions for proportions and dimensions, which reflects typical characteristics of diamonds: most are small, reasonably priced, and cut to standard proportions.

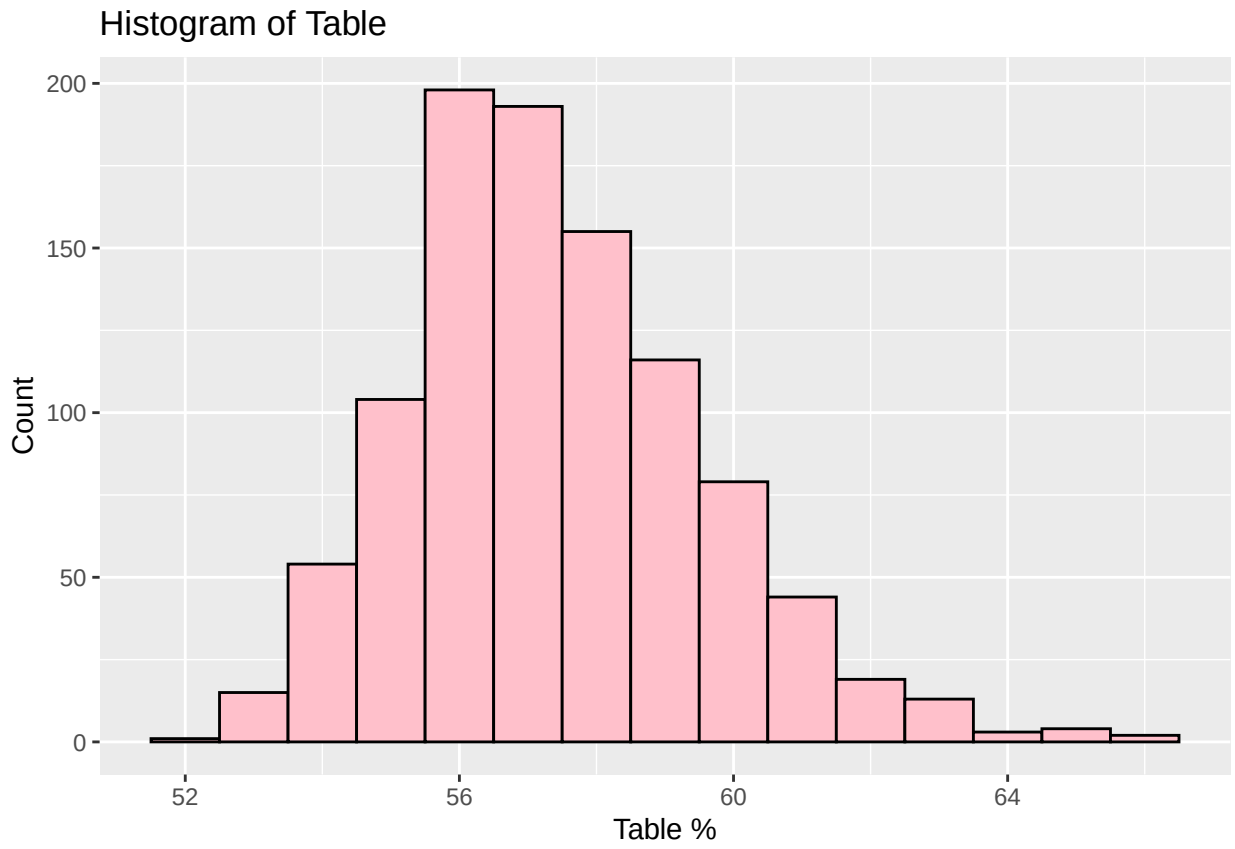
```r
# Carat
ggplot(sample, aes(x = carat)) +
  geom_histogram(binwidth = 0.1, fill = "blue", color = "black") +
  labs(title="Histogram of Carat", x="Carat", y="Count")
```



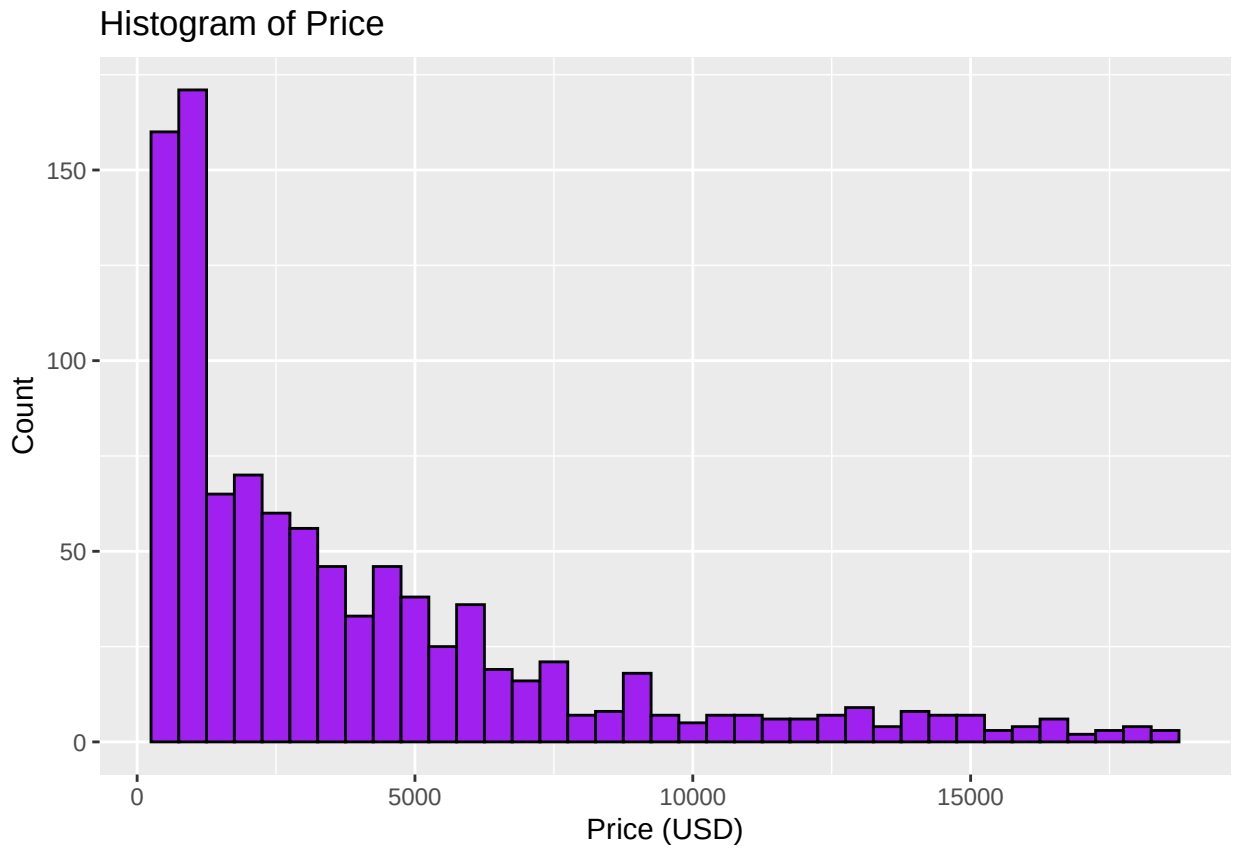
```r
# Depth
ggplot(sample, aes(x = depth)) +
  geom_histogram(binwidth = 1, fill = "green", color = "black") +
  labs(title="Histogram of Depth", x="Depth %", y="Count")
```



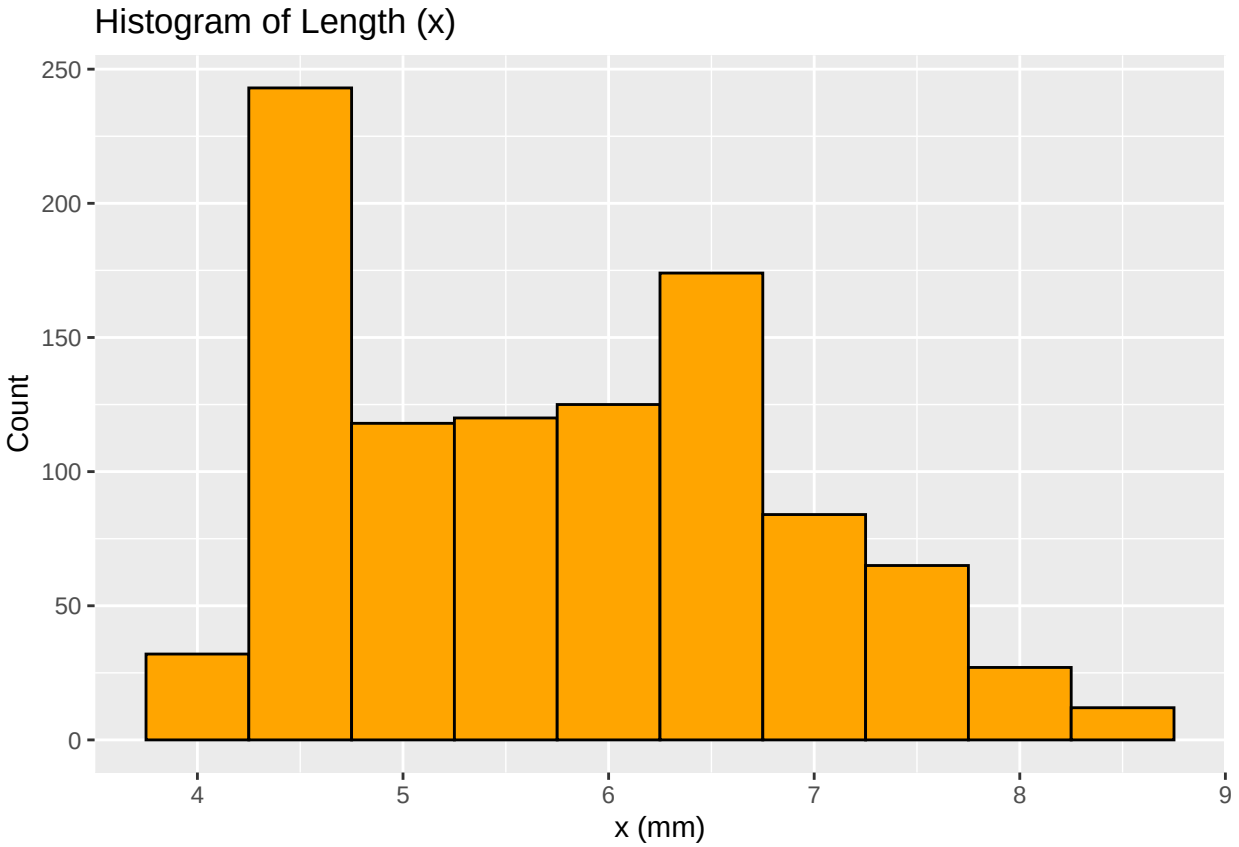
```r
# Table
ggplot(sample, aes(x = table)) +
  geom_histogram(binwidth = 1, fill = "pink", color = "black") +
  labs(title="Histogram of Table", x="Table %", y="Count")
```



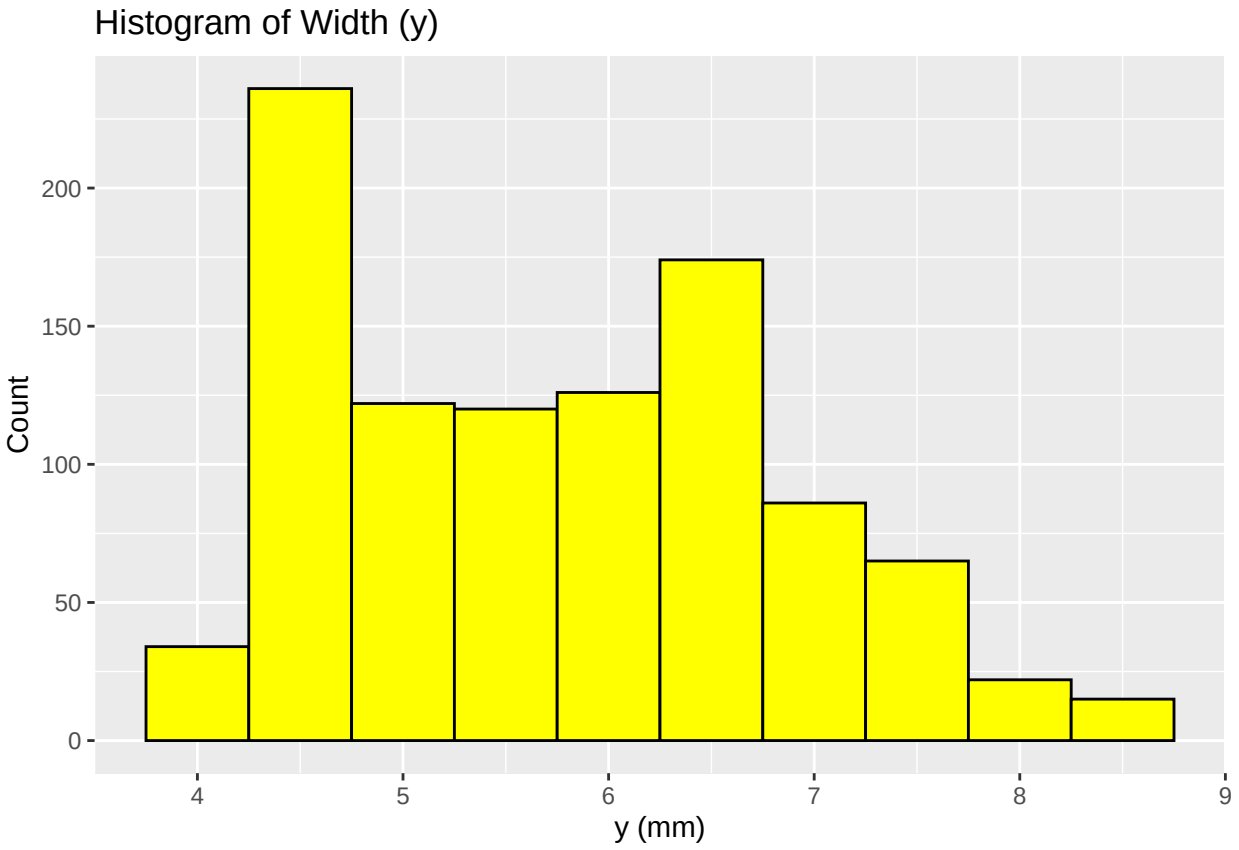
```r
# Price
ggplot(sample, aes(x = price)) +
  geom_histogram(binwidth = 500, fill = "purple", color = "black") +
  labs(title="Histogram of Price", x="Price (USD)", y="Count")
```



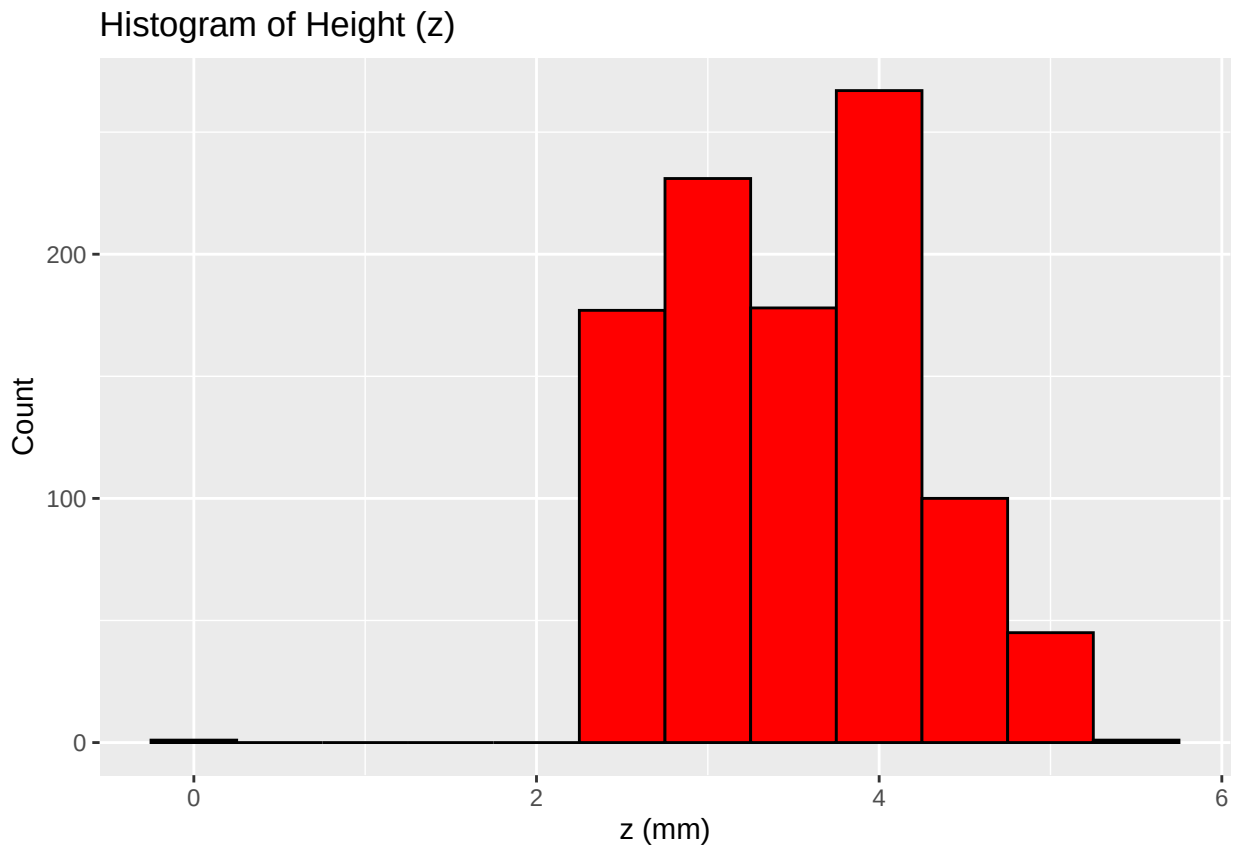
```r
# Dimensions x, y, z
ggplot(sample, aes(x = x)) +
  geom_histogram(binwidth = 0.5, fill = "orange", color = "black") +
  labs(title="Histogram of Length (x)", x="x (mm)", y="Count")
```



```r
ggplot(sample, aes(x = y)) +
  geom_histogram(binwidth = 0.5, fill = "yellow", color = "black") +
  labs(title="Histogram of Width (y)", x="y (mm)", y="Count")
```



```r
ggplot(sample, aes(x = z)) +
  geom_histogram(binwidth = 0.5, fill = "red", color = "black") +
  labs(title="Histogram of Height (z)", x="z (mm)", y="Count")
```

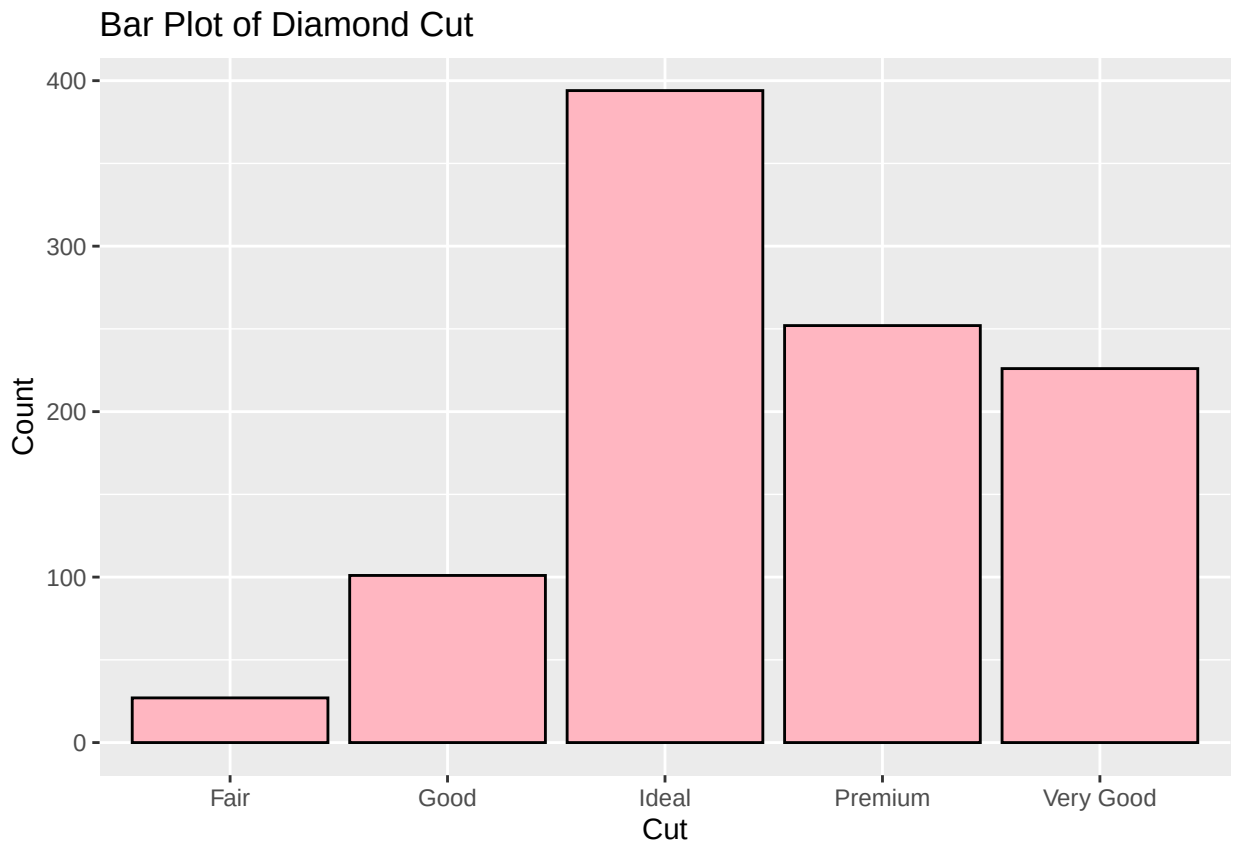


Observations of continuous random variable:

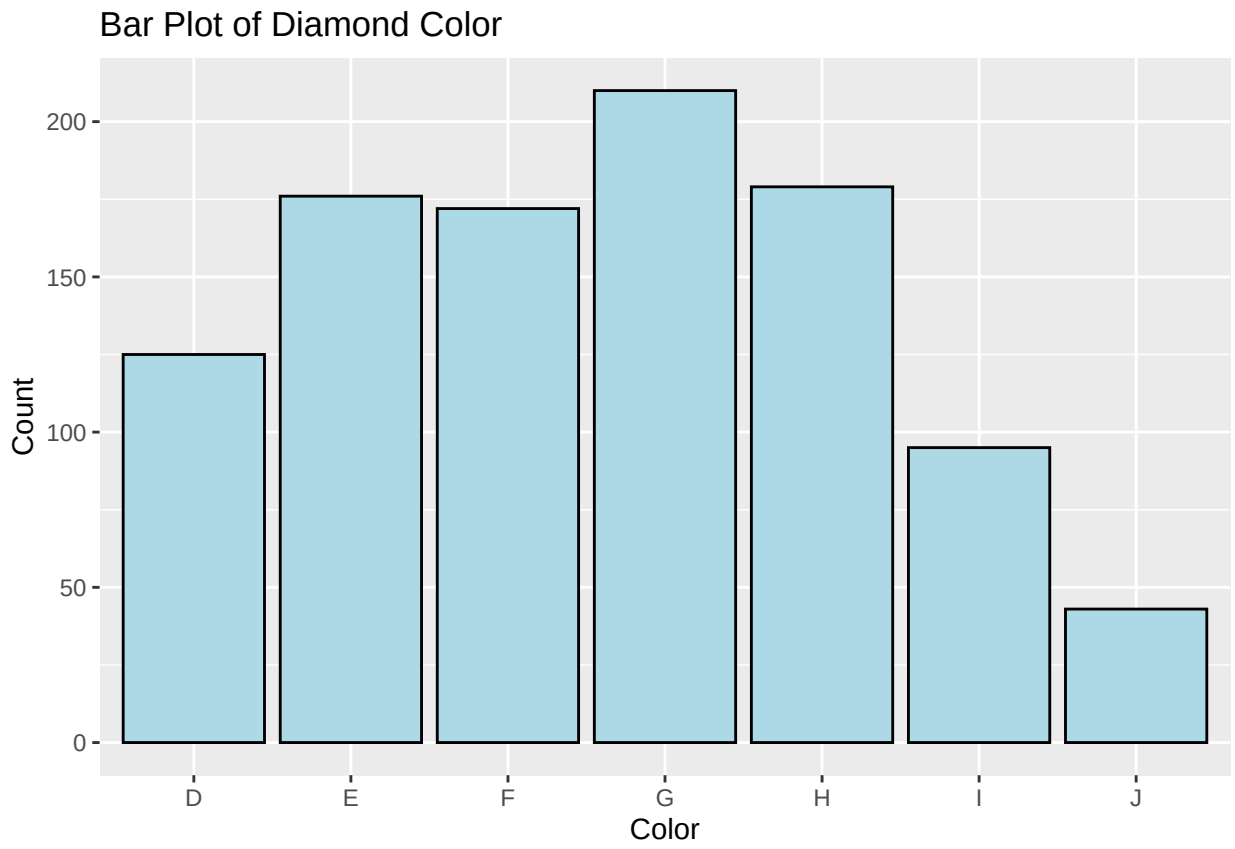
- Carat: The histogram shows most diamonds are small, they are mostly less than 1 carat, with a long tail to the right for larger diamonds. This is the right-skewed distribution.
- Depth: This variable has a fairly symmetric and narrow distribution, with most values between 61% and 63%, indicating low variation in cut depth.
- Table: The histogram of table is approximately symmetric, with most values cluster tightly around the mid-50s(between 56 and 59). The maximum (66) is not extremely far from the rest which create a slight right tail.
- Price: The histogram of price is right skew, with most of the observations fall in the lower price range (under 5000 USD), while a smaller number of expensive diamonds create the right tail.
- Dimensions (x, y, z): Length (x) and width (y) are approximately normal, with most diamonds around 5.7 mm. Height (z) is also roughly normal but shows some potential outliers, especially the highest values and some zeros, which may indicate data errors.

Overall, continuous variables are either symmetric or right-skewed.

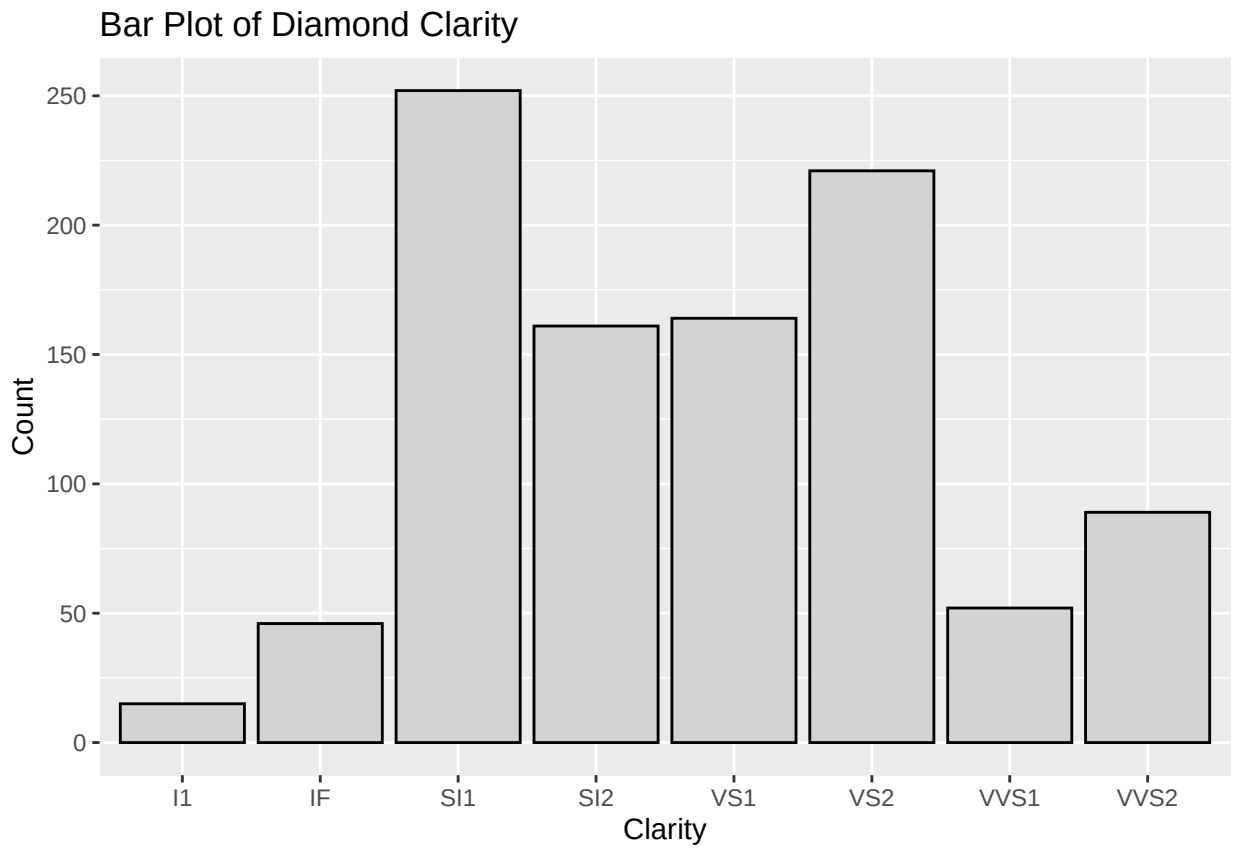
```r
ggplot(sample, aes(x = cut)) +
  geom_bar(fill = "lightpink", color = "black") +
  labs(title = "Bar Plot of Diamond Cut", x = "Cut", y = "Count")
```



```r
ggplot(sample, aes(x = color)) +
geom_bar(fill = "lightblue", color = "black") +
labs(title = "Bar Plot of Diamond Color", x = "Color", y = "Count")
```



```r
ggplot(sample, aes(x = clarity)) +
geom_bar(fill = "lightgrey", color = "black") +
labs(title = "Bar Plot of Diamond Clarity", x = "Clarity", y = "Count")
```



Observations of Categorical Variables:

- Cut: Most diamonds are Ideal or Premium, fewer are Fair or Good, which shows the distribution is stable.
- Color: Most diamonds fall in G, H, or E, with fewer at the extremes (D and J), which shows distribution is stable.
- Clarity: Most diamonds are VS2, VS1, or SI1, fewer are IF or I1, shows distribution is stable.

Overall, the categorical variables show stable distributions, indicating that the sample mostly contains mid-to-high quality diamonds, typical of a retail dataset.

#3. Determine correlation between variables

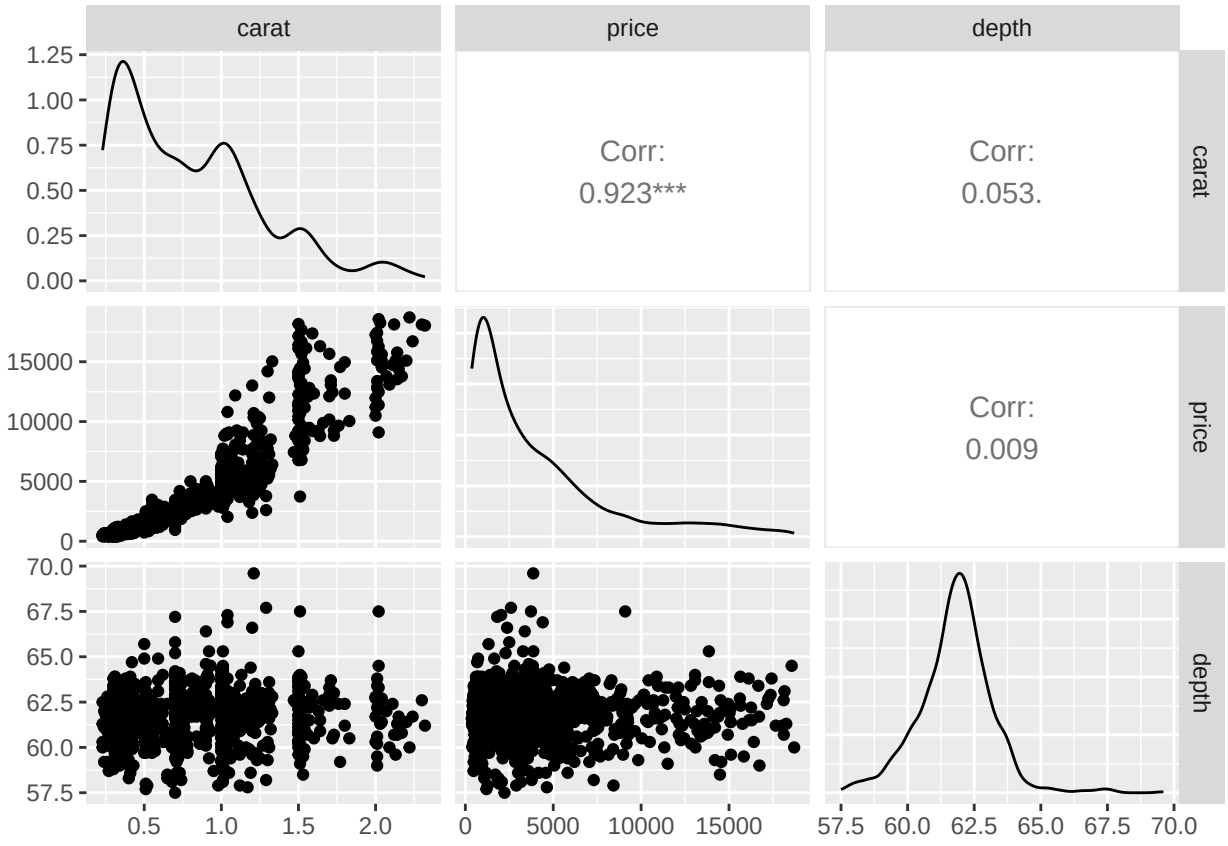
3 quantitative variables: price(dependent), carat(independent), depth(independent)

2 categorical variables: cut(independent), color(independent)

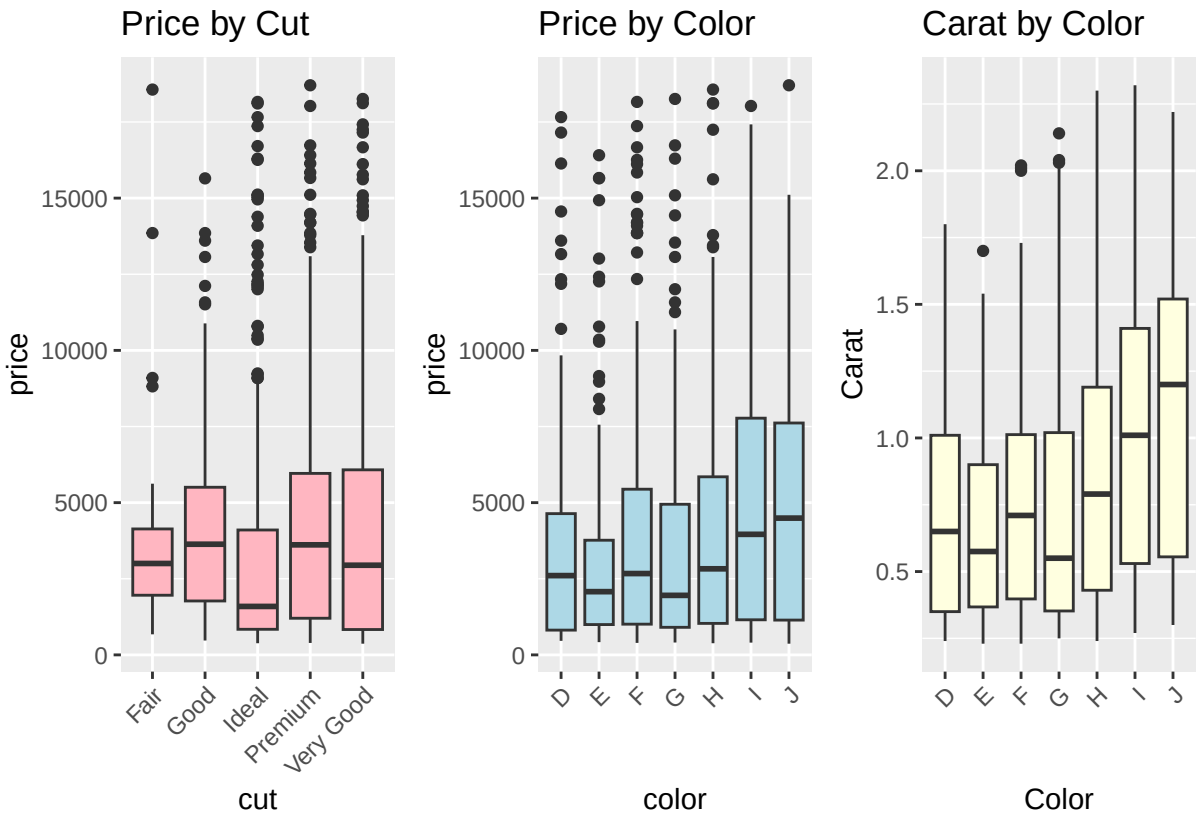
```
library(dplyr)
library(GGally)
library(ggplot2)
library(patchwork)

sample1 <- sample

num <- sample1 %>% select(carat, price, depth)
cor(num, use = "complete.obs")
```

```
##           carat       price       depth
## carat 1.0000000 0.923340431 0.052779405
## price 0.9233404 1.000000000 0.009338803
## depth 0.0527794 0.009338803 1.000000000
```

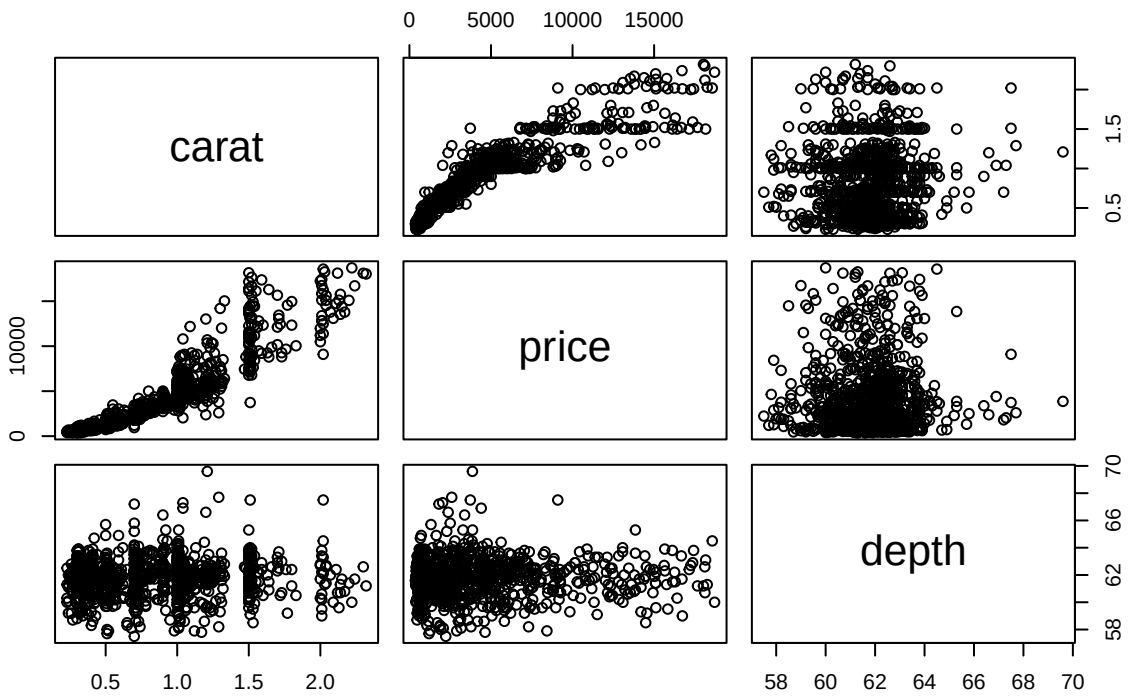
```
numeric_plot <- ggpairs(num)

box_cut_price <- ggplot(sample1, aes(x = cut, y = price)) +
  geom_boxplot(fill = "lightpink") +
  labs(title = "Price by Cut") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

box_color_price <- ggplot(sample1, aes(x = color, y = price)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Price by Color") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

box_color_carat <- ggplot(sample1, aes(x = color, y = carat)) +
  geom_boxplot(fill = "lightyellow") +
  labs(title = "Carat by Color", x = "Color", y = "Carat") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

numeric_plot
```




```
(box_cut_price | box_color_price | box_color_carat)
```




```
numvars <- dplyr::select(sample, carat, price, depth)

pairs(numvars, main = "Scatterplot Matrix for Numeric Variables")
```



Quantitative variables(scatterplot):

- Carat and Price: There is a strong correlation(0.9233) between carat and price, which suggest that larger diamond will likely to be more expensive.
- Depth and Price: There is a very weak correlation(0.0093) between depth and price, which suggest that depth does not significantly impact diamond price directly. Therefore, the variable Depth is not meaningful.
- Carat and Depth: There is a weak correlation(0.0528) between carat and depth, which suggest that the depth does not significantly impact the size of diamond.

Categorical variables(boxplot):

- Price by Cut: Shows how price varies across cut quality. Ideal and Premium cuts tend to have higher median prices.
- Price by Color: Shows that better colors (D, E, F) tend to have slightly higher prices, but there's wide overlap due to carat and cut.
- Carat by Color: Shows how diamond size varies with color. Some variation exists, but differences are smaller than price differences by cut.

Observation:

- Numeric variables: Carat and price are strongly positively correlated, depth shows weak correlation with both depth data may not be useful in the dataset.
- Categorical vs Numeric: Cut has a clear impact on price, with higher-quality cuts like Ideal and Premium typically costing more. Color has a smaller effect, slightly influencing carat, but the differences are less pronounced. Overall, categorical factors like cut and color help explain some variation in numeric variables, though carat and price remain primarily influenced by size.

- Categorical variables: No strong correlation, though certain combinations are more common.

#4. Run model and observe summary

```
model <- lm(price ~ carat + cut+ color + clarity + depth + table + price + x + y + z, data = sample)
```

```
## Warning in model.matrix.default(mt, mf, contrasts): the response appeared on
## the right-hand side and was dropped
```

```
## Warning in model.matrix.default(mt, mf, contrasts): problem with term 7 in
## model.matrix: no columns are assigned
```

```
summary(model)
```

```
##
## Call:
## lm(formula = price ~ carat + cut + color + clarity + depth +
##     table + price + x + y + z, data = sample)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4375.3  -563.2  -177.1   392.9  6463.4
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9101.25    3191.77   2.851 0.004443 **
## carat        14152.49     405.88  34.869  < 2e-16 ***
## cutGood        308.22     268.35   1.149 0.251000
## cutIdeal       565.84     272.97   2.073 0.038443 *
## cutPremium     354.90     261.74   1.356 0.175441
## cutVery Good   502.34     264.37   1.900 0.057711 .
## colorE        -250.14     126.67  -1.975 0.048575 *
## colorF        -240.34     127.57  -1.884 0.059862 .
## colorG        -543.08     124.05  -4.378 1.33e-05 ***
## colorH       -1057.16     128.37  -8.235 5.72e-16 ***
## colorI       -1660.84     150.62 -11.027  < 2e-16 ***
## colorJ       -2467.68     195.13 -12.646  < 2e-16 ***
## clarityIF     3889.34     359.43  10.821  < 2e-16 ***
## claritySI1    2756.88     321.59   8.573  < 2e-16 ***
## claritySI2    1804.18     326.61   5.524 4.25e-08 ***
## clarityVS1    3789.18     327.79  11.560  < 2e-16 ***
## clarityVS2    3529.03     325.62  10.838  < 2e-16 ***
## clarityVVS1   3939.78     355.25  11.090  < 2e-16 ***
## clarityVVS2   4012.24     339.65  11.813  < 2e-16 ***
## depth          -79.38      36.78  -2.158 0.031153 *
## table          -38.56      20.96  -1.840 0.066077 .
## x            -2845.46     809.25  -3.516 0.000458 ***
## y             1003.05     810.83   1.237 0.216363
## z             -476.83     288.48  -1.653 0.098675 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1070 on 976 degrees of freedom
## Multiple R-squared:  0.9294, Adjusted R-squared:  0.9278
## F-statistic:   559 on 23 and 976 DF,  p-value: < 2.2e-16
```

Observe from the summary:

- Multiple R-squared = 0.9294
- Adjusted R-squared = 0.9278

This means the model explains about 93% of the variability in diamond price, which indicates an extremely strong model fit. After adjusting for the number of predictors, the model still explains about 92.8%, suggesting that most variables meaningfully contribute to the model.

- F-statistic = 559, p < 2.2e–16 The overall model is highly statistically significant. At least one predictor reliably contributes to predicting price.
- Residual standard error: 1070 This means the model predictions are on average about 1070 dollars away from the true price, which is reasonable given the wide price range (from ~$300 to ~$18,000).

Interpretation of Significant Predictors:

- Carat(Estimate = 14152.49): This is the most powerful predictor in the model. It indicates that for each additional carat, diamond price increases by roughly $14,152, holding all else constant.
- Cut: the baseline is Fair, better cuts have higher coefficients compared to the baseline cut. Ideal cut shows a statistically significant price increase.
- Color: baseline is D, lower-quality colors have large negative coefficients compared to D. As color moves from D to J, price decreases sharply, which matches diamond industry grading.
- Clarity: baseline is I1, higher clarity corresponds to higher prices, and the effect is large and significant.
- x, y, z are highly correlated with carat (multicollinearity).

Overall, this regression model explains about 93% of the variance in diamond prices, with carat as the strongest predictor, and strong contributions from color and clarity, while dimensional variables (x, y, z) show instability due to multicollinearity.

#5. Comment on interest

- The data largely behaved as expected, carat is the strongest predictor of price, and higher clarity diamonds consistently cost more.
- The high R-squared (0.929) indicates that most of the variation in price is explained by these variables. However, it's interesting to see that depth may contribute little to the model's predictive power and might risk overfitting if included unnecessarily.
- Cut and color have smaller numerical effects compared to carat and clarity, though extreme color levels (I and J) reduce price noticeably.
- Other numeric dimensions (x, y, z) also have relatively small or inconsistent contributions, suggesting that physical measurements beyond carat are less important for predicting price in this dataset.

Carat is the strongest predictor of diamond price, with higher clarity also increasing price. Cut and color have smaller effects, though extreme colors lower price. Most other numeric variables, like depth and x,y,z, contribute little and may risk overfitting. Overall, the model explains about 93% of price variation, capturing the main factors influencing diamond pricing.

Part 2

#1. Start with the predictor 'carat' and the response 'price'

Simple linear regression: price ~ carat, price is the response variable (y), carat is the predictor (x)

price = B0 + B1(carat) + e

```
model <- lm(price ~ carat, data = sample)
summary(model)
```

```
##
## Call:
## lm(formula = price ~ carat, data = sample)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5932.7  -905.1     0.8   631.1  8572.6
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2456.41      97.05  -25.31   <2e-16 ***
## carat        8028.56     105.69   75.97   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1529 on 998 degrees of freedom
## Multiple R-squared:  0.8526, Adjusted R-squared:  0.8524
## F-statistic:  5771 on 1 and 998 DF,  p-value: < 2.2e-16
```

summary(model) will show the regression coefficients, R-squared, residual standard error, and significance of the predictor. Where B0 = intercept, and B1 = slope.

#2. Run the model and examine the summary statistics

Coefficients:

- Intercept (-2456.41): The model predicts a negative price of -2456.41 at 0 carats.
- Slope (8028.56): Each 1 unit additional carat increases the price by roughly $8,029 on average.

Both coefficients are highly significant because the p-value is very small

- R-squared = 0.8526: About 85% of the variation in diamond price is explained by carat alone.Indicating a very strong linear relationship.
- Adjusted R-squared = 0.8524: Very similar since there's only one predictor.
- (F-statistic: 5771 on 1 and 998 DF, p-value: $<$ 2.2e-16) The extremely high F-value and low p-value confirms that carat is a highly significant predictor of price.
- Residual standard error = 1529: On average, the predicted price is about $1,529 away from the actual price.

Therefore, it shows strong positive relationship, which price increases sharply with carat. The high R-squared shows carat alone explains most of the variation in price. The residuals range from -$5,933 to $8,573, showing some extreme deviations likely due to unusually expensive or large diamonds.

Hypothesis Test:

- Null hypothesis: B1 = 0 (there is no linear relationship between carat and price)

- Alternative hypothesis: B1 != 0 (there is a specific relationship between carat and price)

Since p-value < 2.2e-16 is extremely small and less than 0.05, we reject H0. There is enough evidence to conclude that carat is a highly significant predictor of price.

Confidence Interval:

```
confint(model)
```

```
##                 2.5 %    97.5 %
## (Intercept) -2646.851 -2265.968
## carat        7821.169  8235.958
```

We are 95% confident the true increase in price per carat is between $7821.169 and $8235.958.

Prediction Interval:

```
predict(model,interval = "prediction", newdata = data.frame(carat=1))
```

```
##        fit      lwr      upr
## 1 5572.153 2570.076 8574.231
```

```
predict(model,interval = "prediction", newdata = data.frame(carat=2))
```

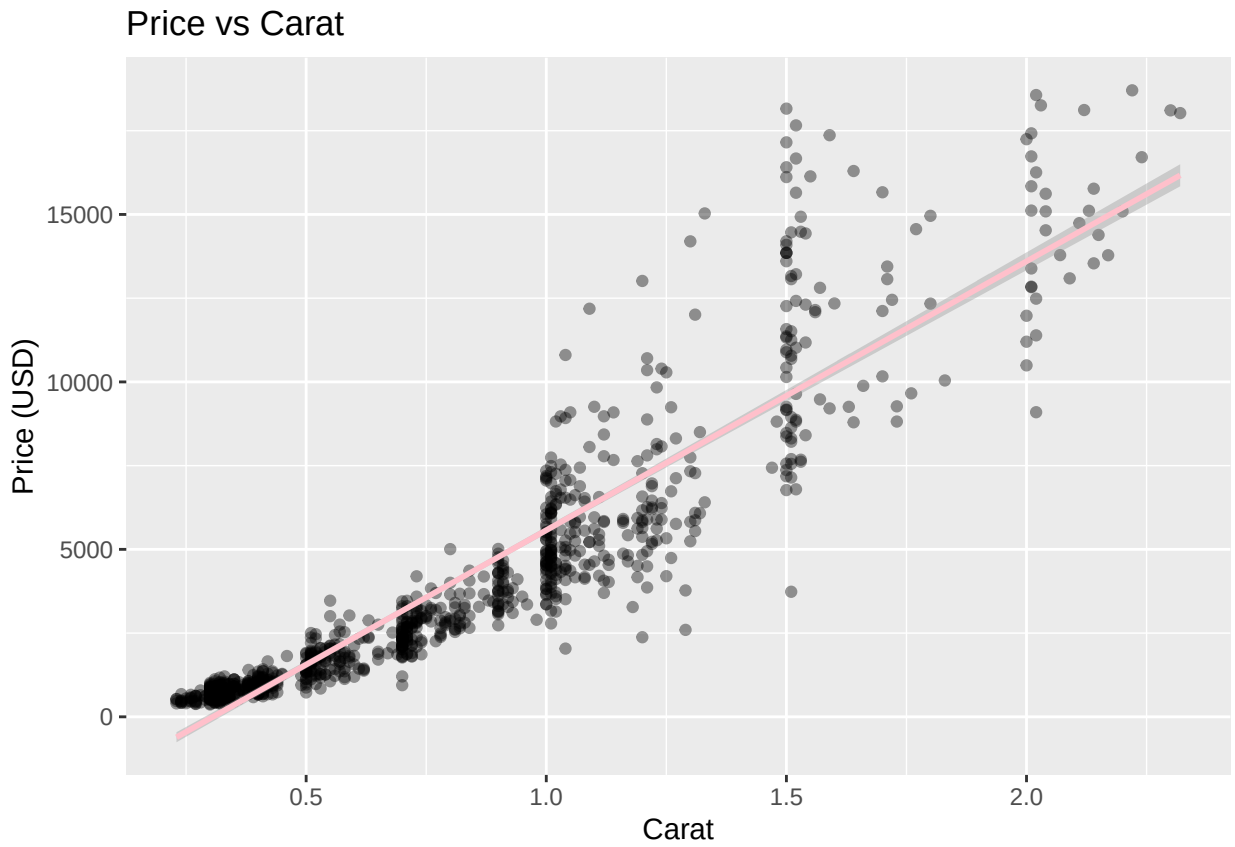
```
##        fit      lwr      upr
## 1 13600.72 10588.57 16612.86
```

We are 95% sure that the predicted price of a diamond whose carat = 1 will be between 2570.076 and 8574.231.

We are 95% sure that the predicted price of a diamond whose carat = 2 will be between 10588.57 and 8380.732.

```
ggplot(sample, aes(x = carat, y = price)) +
  geom_point(alpha = 0.4) +
  geom_smooth(method = "lm", color = "pink") +
  labs(title = "Price vs Carat", x = "Carat", y = "Price (USD)")
```

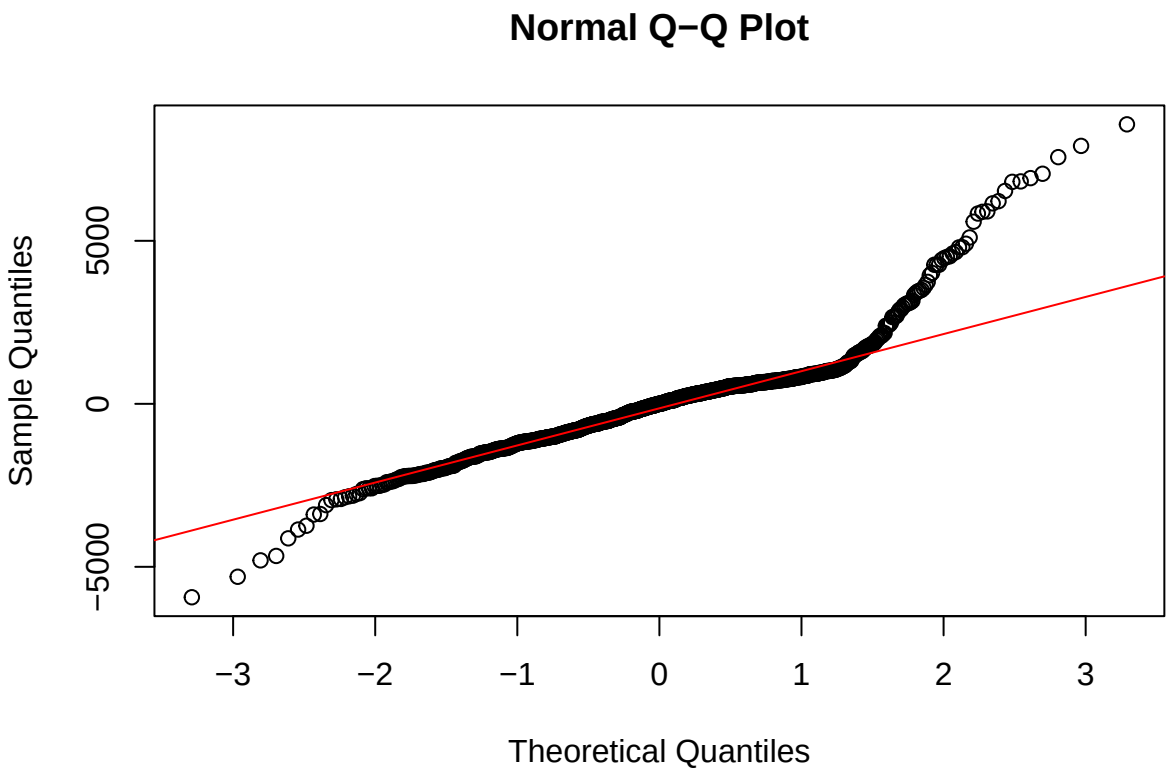
```
## `geom_smooth()` using formula = 'y ~ x'
```



The scatterplot displays a strong upward trend. As carat increases, price also increases. The regression line captures this pattern well, showing a clear positive linear relationship. Most points cluster closely around the line at lower carat values, but the spread becomes wider for larger diamonds. This indicates that price becomes more variable for big diamonds, even though the overall trend is still increasing. The smooth fitted line confirms that carat is a strong predictor of price.

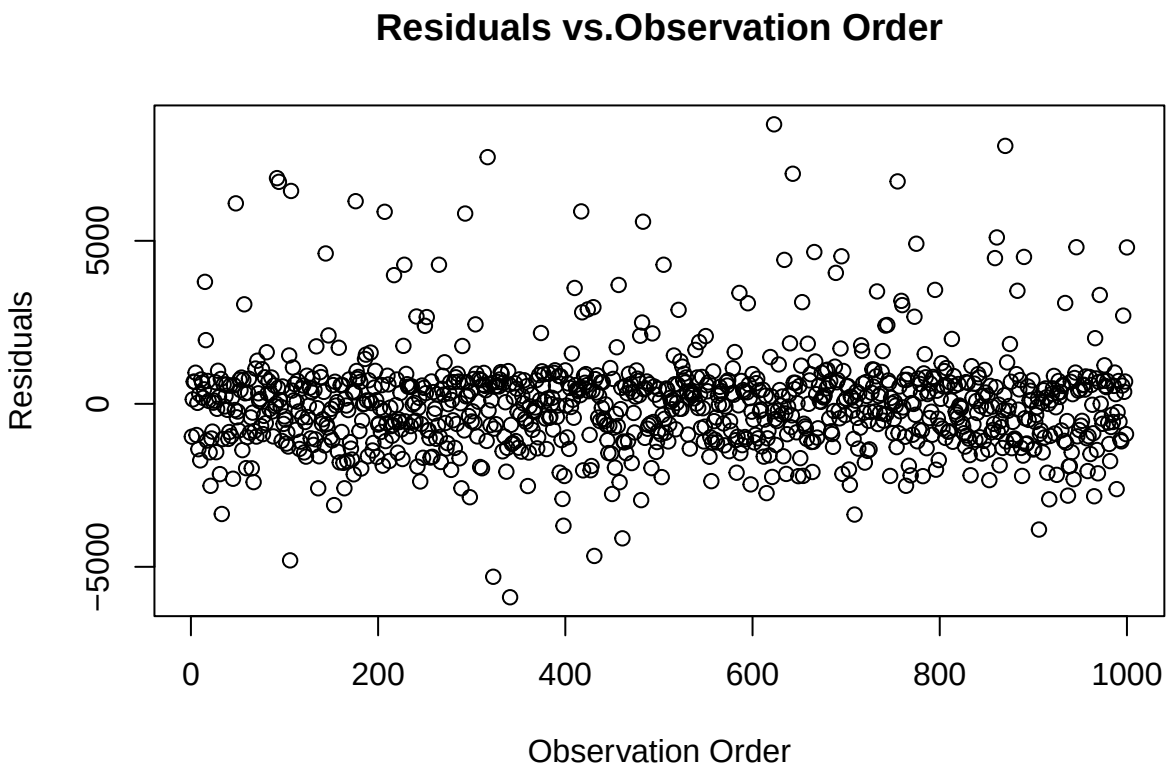
#3. Test the assumptions and apply transformation

```
x <- lm(price ~ carat, data = sample)
qqnorm(x$residuals)
qqline(x$residuals, col = "red")
```



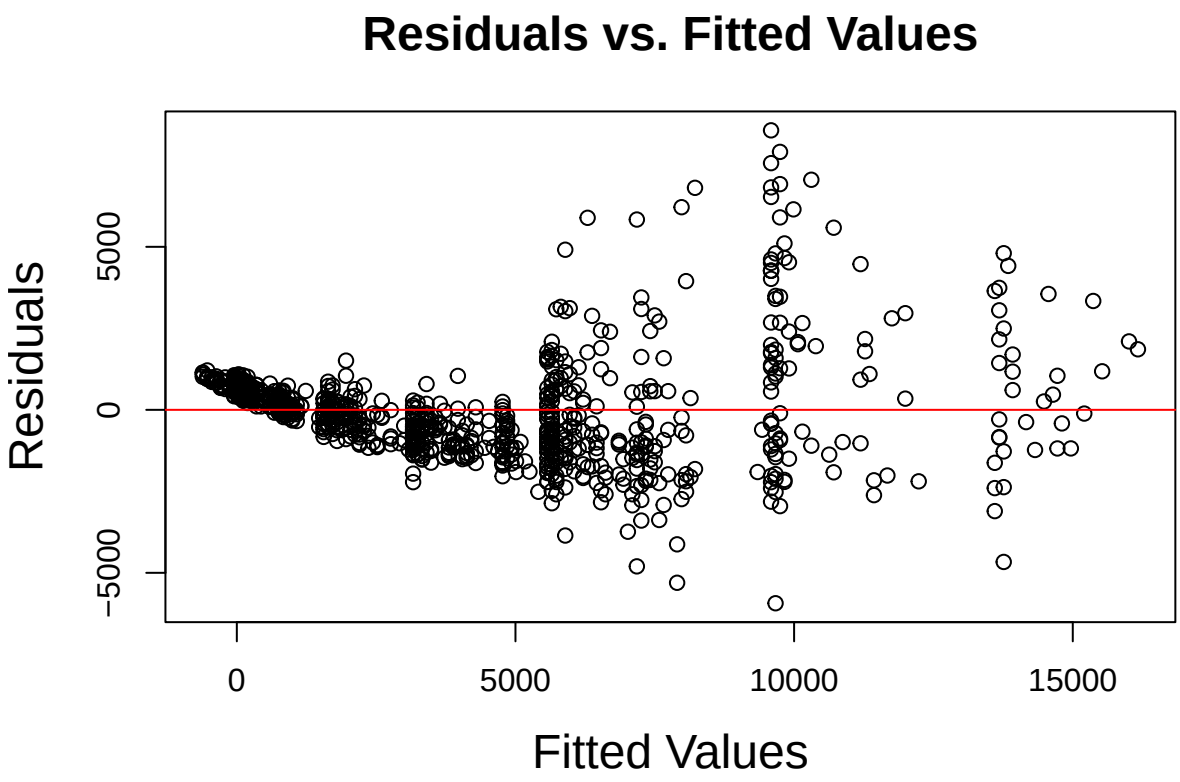
It is clear that the points deviate substantially from the red line at the tails. This indicates non-normal residuals and heavy-tailed error distribution, which means the normality assumption is violated.

```
a <-1:length(x$residuals)
 plot(x$residuals~a,
 ylab = "Residuals",xlab="Observation Order",
 main = "Residuals vs.Observation Order")
```



This plot shows that the residuals cluster close to zero, which might indicate non-constant variance or poor scaling. Even if no pattern exists, the model fit may not be great. Independence okay, but residual distribution might be compressed

```
plot(x$residuals ~ x$fitted.values,
   xlab = "Fitted Values", ylab = "Residuals",
   main = "Residuals vs. Fitted Values", cex.lab = 1.5, cex.main = 1.5)
abline(h = 0, col = "red")
```



The spread of residuals form a funnel shape: small residuals for small fitted values, large spread for large fitted values. This indicates heteroscedasticity: variance increases with fitted price. Also, a slight curvature suggests non-linearity.

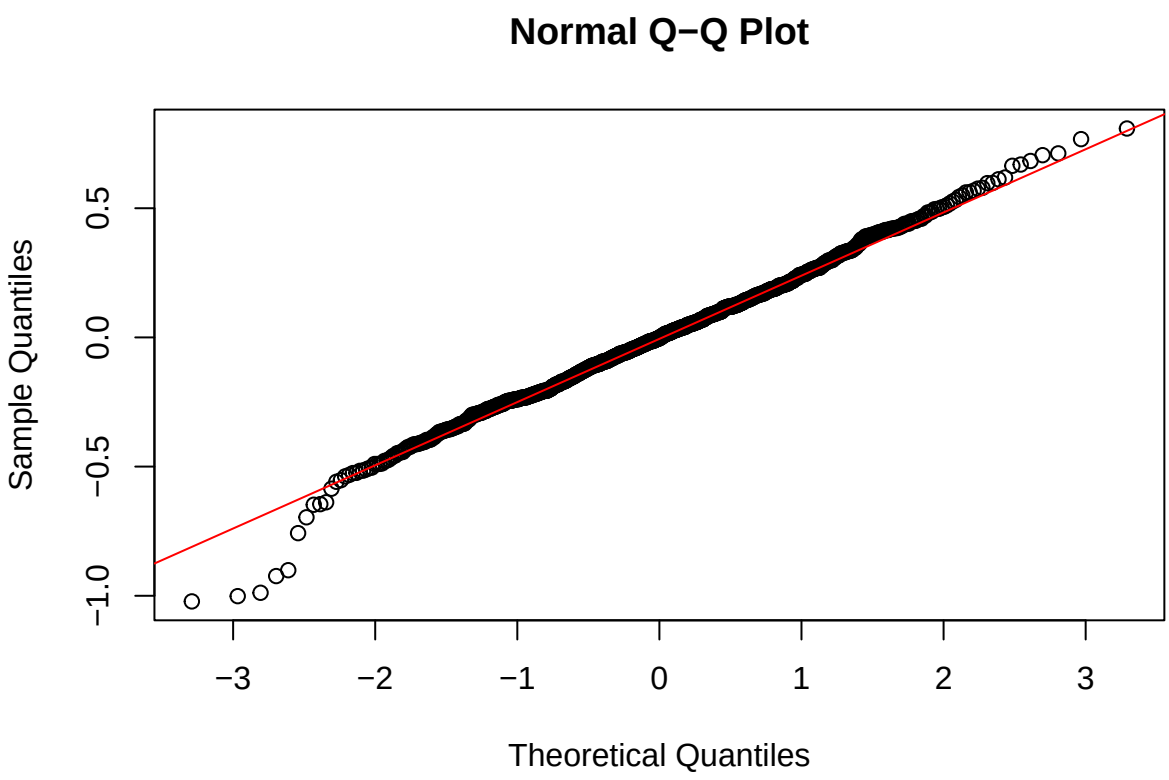
Now: Log–Log Transformed Model

```
y <- sample %>%
  mutate(logprice = log(price),
         logcarat = log(carat))
logmodel <- lm(logprice ~ logcarat, data = y)
summary(logmodel)
```

```
##
## Call:
## lm(formula = logprice ~ logcarat, data = y)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.02178 -0.17069 -0.00159  0.15935  0.80854
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 8.454101   0.009627   878.1   <2e-16 ***
## logcarat    1.686313   0.013727   122.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2523 on 998 degrees of freedom
```

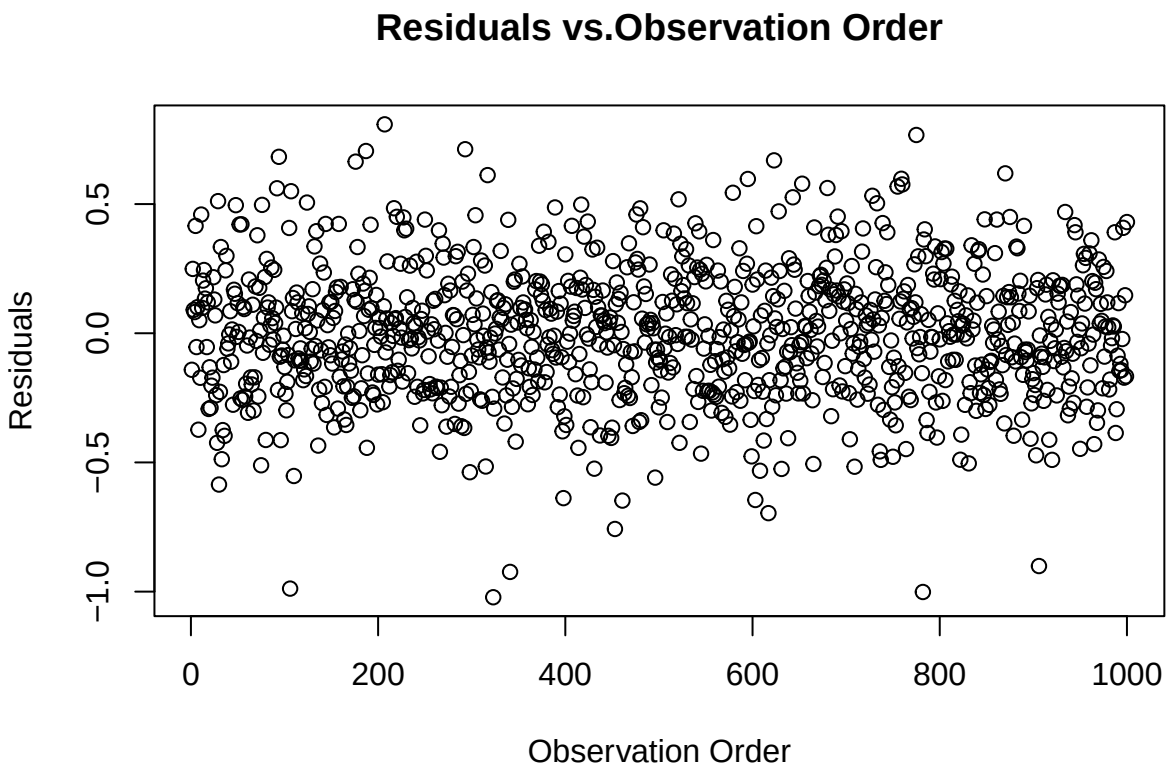
```
## Multiple R-squared:  0.938,  Adjusted R-squared:  0.9379
## F-statistic: 1.509e+04 on 1 and 998 DF,  p-value: < 2.2e-16
```

```
qqnorm(logmodel$residuals)
qqline(logmodel$residuals, col = "red")
```



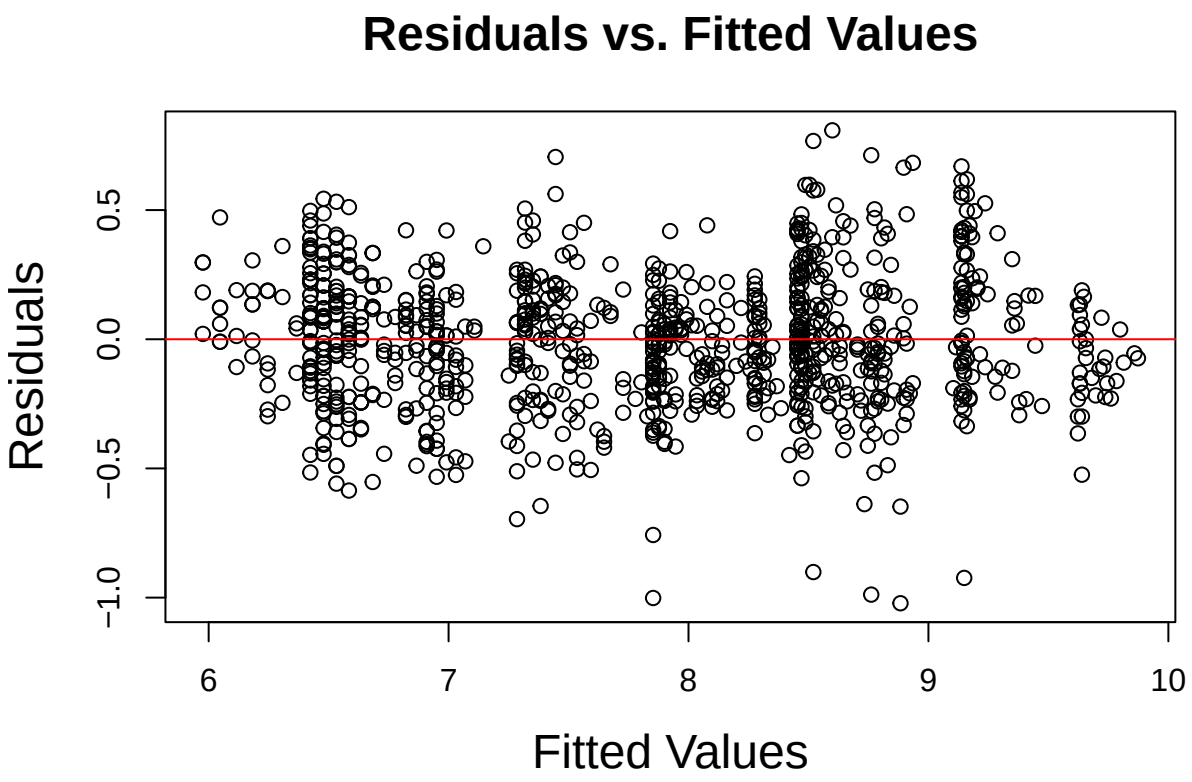
This plot shows that the points lie closer to the reference line. Only small deviations at extreme tails only. This indicates approximately normal residuals, which is a large improvement compared to the original model.

```
a <-1:length(logmodel$residuals)
 plot(logmodel$residuals~a,
 ylab = "Residuals",xlab="Observation Order",
 main = "Residuals vs.Observation Order")
```



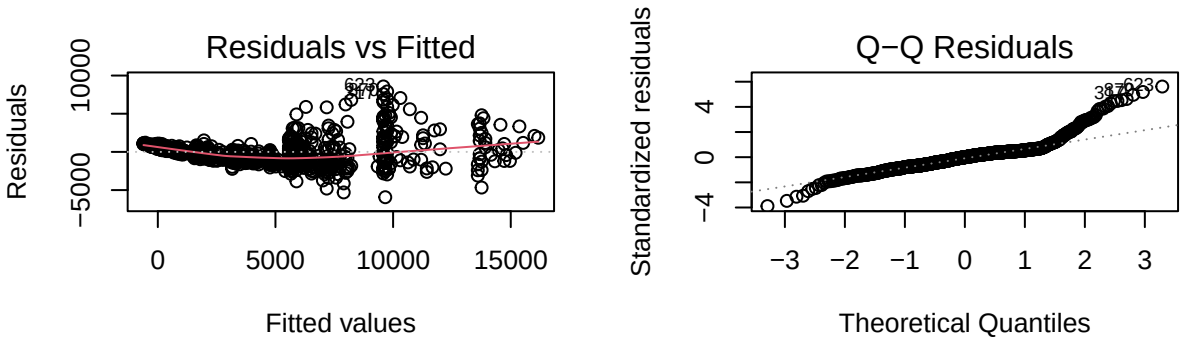
The plot show points randomly scattered with no obvious patterns. This indicates independence of errors is likely satisfied. No time-type structure or clustering exists (good).

```
plot(logmodel$residuals ~ logmodel$fitted.values,
   xlab = "Fitted Values", ylab = "Residuals",
   main = "Residuals vs. Fitted Values", cex.lab = 1.5, cex.main = 1.5)
abline(h = 0, col = "red")
```

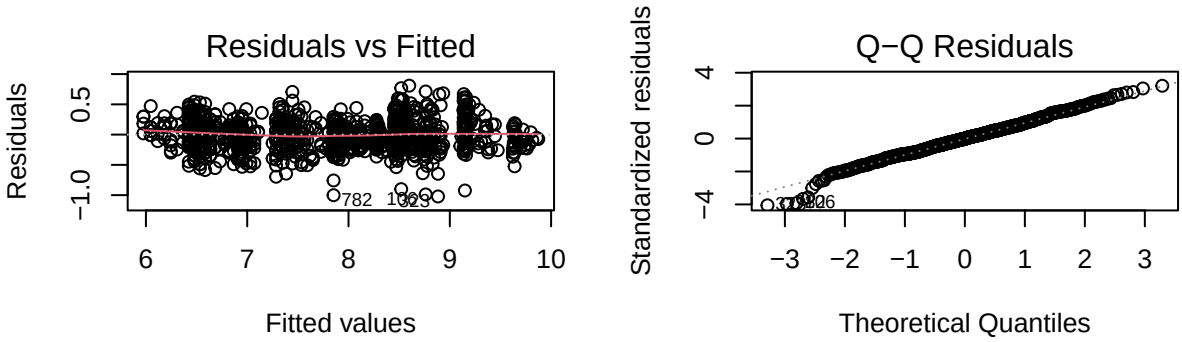


The funnel shape disappears. Residuals have constant variance, not increasing with fitted values. No curvature appears means linearity is good. This indicates the log transformation fixed the heteroscedasticity and non-linearity.

```r
model1 <- lm(price ~ carat, data = sample)
par(mfrow = c(2, 2))
plot(model1, which = c(1, 2))

model_log <- lm(log(price) ~ log(carat), data = sample)
par(mfrow = c(2, 2))
```




```r
plot(model_log, which = c(1, 2))
```



Here's a clear comparison between the original and the log-log transformed model.

#4. Summary, notes the change

```
sample <- sample %>%
  mutate(
    logprice = log(price),
    logcarat = log(carat)
  )

summary(sample[, c("price", "carat", "logprice", "logcarat")])
```

```
##      price            carat           logprice        logcarat
##  Min.   :  368.0   Min.   :0.2300   Min.   :5.908   Min.   :-1.46968
##  1st Qu.:  956.8   1st Qu.:0.4000   1st Qu.:6.864   1st Qu.:-0.91629
##  Median : 2522.0   Median :0.7100   Median :7.833   Median :-0.34249
##  Mean   : 3935.9   Mean   :0.7962   Mean   :7.792   Mean   :-0.39263
##  3rd Qu.: 5301.8   3rd Qu.:1.0400   3rd Qu.:8.576   3rd Qu.: 0.03922
##  Max.   :18706.0   Max.   :2.3200   Max.   :9.837   Max.   : 0.84157
```

```
summary(logmodel)
```

```
##
## Call:
## lm(formula = logprice ~ logcarat, data = y)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -1.02178 -0.17069 -0.00159  0.15935  0.80854
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 8.454101   0.009627   878.1   <2e-16 ***
## logcarat    1.686313   0.013727   122.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2523 on 998 degrees of freedom
## Multiple R-squared:  0.938,  Adjusted R-squared:  0.9379
## F-statistic: 1.509e+04 on 1 and 998 DF,  p-value: < 2.2e-16
```

Original Variable (price and carat):

- price: Very large spread(from min 368 to max 18706). The mean (3935) is much higher than the median (2522), which confirms strong right-skewness.
- carat: The range is from min 0.23 to max 2.32. The mean (0.7962) is also higher than the median (0.71), which also confirms that there it is right skiewed.

Transformed Variables (logprice and logcarat):

- logprice: The spread is much more compressed (from min 5.908 to max 9.837). The mean (7.792) is very close to the median (7.833), which shows that logprice is now much closer to symmetric.
- logcarat: The range is from -1.46968 to 0.84157. The mean (-0.39263) is also very close to the median (-0.34249), which shows that, again, logcarat is now much closer to symmetric.

Noticeable changes:

- Both price and carat initially exhibited right-skewed distributions, with many small diamonds and relatively few large or expensive ones.
- Applying the log transformation compressed extreme values, reducing the influence of outliers and high-leverage points, and stabilized the variance across observations.
- The IQR became more balanced and closer to the mean, indicating that the transformed variables are less affected by extreme values.
- Residuals are more normal. Histogram of residuals is now roughly symmetric. Q–Q plot points lie closer to the reference line. Residuals vs. fitted values plot no longer shows a funnel shape and variance of residuals is more constant across fitted values. Extreme values are handled better.
- Model fit improved, adjusted R-squared increased from 0.8524 to 0.9379, and residual standard error decreased.
- The transformation also linearized the relationship between carat and price, making it easier for the model to capture a consistent pattern.

Overall, the log transformation made the data more suitable for linear regression, improved interpretability, and reduced potential bias from extreme observations.

After applying the log transformation to both variables, the summary statistics show that the transformed variables have significantly reduced skewness. In the original scale, both price and carat were strongly right-skewed, with large ranges and means much greater than their medians. After transformation, logprice and logcarat have much narrower ranges, and their means and medians are nearly equal, indicating a much more symmetric distribution. This suggests that the log transformation successfully stabilizes variance and reduces skewness, making the variables more suitable for linear regression modeling and improving the validity of the regression assumptions.

#5. Add other variables to the model

After establishing that the log–log simple linear regression model (logprice ~logcarat) provided a substantially better fit than the original untransformed model, we next evaluated whether adding other predictors would improve model performance. Following the instructions, we added additional variables—both quantitative (depth, table, x, y, z) and categorical (cut, color, clarity) to examined whether each inclusion increased the adjusted R-squared and reduced residual standard error. If a variable improved the model fit, it was retained; if it decreased the adjusted R-squared or introduced new assumption violations, it was excluded.

We used the simple model: log(price)~log(carat) as our starting point. This initial model, with log(price) as the response and log(carat) as the only predictor, already showed a strong relationship, achieving an adjusted R-squared of 0.9379.

Starting with carat and price, the model had R-squared of 0.9373 and RSE of 0.2523. Adding variables sequentially:

- Depth: R-squared increased to 0.9392, RSE decreased to 0.2496. This shows depth adds useful information.
- Table: R-squared increased to 0.9404, RSE decreased to 0.2471, indicating table contributes slightly to the model.
- Cut: R-squared increased to 0.9426, RSE decreased to 0.2425, showing cut provides additional predictive value.
- Color: R-squared increased to 0.9531, RSE decreased to 0.2191, reflecting a substantial improvement.
- Clarity: R-squared increased significantly to 0.9838, RSE decreased to 0.1289, indicating clarity has a major impact.
- X: R-squared increased slightly to 0.9844, RSE decreased to 0.1266.
- Y: R-squared stayed the same at 0.9844, RSE decreased slightly to 0.1265.
- Z: R-squared increased to 0.9845, RSE decreased slightly to 0.1262.

Overall, each variable either increased R-squared or maintained it while reducing RSE, with clarity contributing the largest improvement to the model.

Summary:

During this process, cut, color, and clarity all significantly improved the model. Their addition increased the adjusted R-squared, reduced RSE, and added meaningful explanatory power because these categorical grading variables capture quality differences not explained by carat alone.

Including the quantitative variables depth and table in the model resulted in an improvement in adjusted R-squared, indicating that both variables provide additional explanatory power beyond carat and the grading categories. These two measurements describe the proportions of a diamond, so their contribution suggests that overall cut proportions influence price even after accounting for weight, color, clarity, and cut category.

The variables x (length) and z (total depth) also produced a slight increase in adjusted R-squared, but the improvement is small. Other variables such as y did not improve the model. Since there is not much difference in the fit, we would prefer the smaller model on the principle that simpler explanations are preferred. Therefore, x, y, and z are excluded.

Through this stepwise evaluation, the final model that produced the best balance of predictive power, interpretability, and assumption satisfaction was:

log(price) ~ log(carat) + cut + color + clarity + depth + table

---

#6. Comment on Interest

One of the most interesting observations during this part of the analysis was how dramatically the addition of categorical grading variables—cut, color, and clarity—improved model performance. While the simple log–log model with only log(carat) already had a high adjusted R-squared (~0.938), including these categorical variables increased it to ~0.984 and substantially reduced the residual standard error. This clearly illustrates that diamond quality, not just size, plays a major role in pricing, which aligns with real-world expectations.

It was also interesting to see that some quantitative variables, like depth and table, contributed modest improvements, while others, like x, y, and z, offered little to no improvement. This highlighted the principle of parsimony: including variables that add minimal explanatory power may not be worth the added complexity.

Another point of interest was the stepwise evaluation process itself. Iteratively adding variables and monitoring adjusted R-squared and RSE reinforced the importance of both statistical significance and practical significance. For example, even if a variable slightly improved R-squared, if it didn't improve interpretability or violated assumptions, it was reasonable to exclude it.

Finally, it was striking how the log transformation of the response and predictor laid the groundwork for a more linear and homoscedastic relationship. Without this transformation, residual diagnostics would have shown strong heteroscedasticity and non-normality, making multivariable modeling much less reliable. Overall, this part demonstrated the interplay between variable selection, transformations, and regression assumptions, emphasizing that careful data exploration and stepwise evaluation are critical for building robust predictive models.

Part 3

#1. Analyze the summary function

```
finalmodel <- lm(log(price) ~ log(carat) + cut + color + clarity + depth + table, data = sample)
summary(finalmodel)
```

```
##
## Call:
## lm(formula = log(price) ~ log(carat) + cut + color + clarity +
##     depth + table, data = sample)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38566 -0.08705 -0.00129  0.08784  0.37743
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   7.9508574  0.3280562  24.236  < 2e-16 ***
## log(carat)    1.8889556  0.0081839 230.813  < 2e-16 ***
## cutGood       0.1146925  0.0316966   3.618 0.000312 ***
## cutIdeal      0.2035390  0.0324598   6.271 5.39e-10 ***
## cutPremium    0.1512802  0.0314132   4.816 1.70e-06 ***
## cutVery Good  0.1637435  0.0309277   5.294 1.47e-07 ***
## colorE       -0.0431550  0.0152658  -2.827 0.004796 **
## colorF       -0.0855189  0.0153753  -5.562 3.44e-08 ***
## colorG       -0.1326346  0.0149566  -8.868  < 2e-16 ***
## colorH       -0.2375272  0.0154400 -15.384  < 2e-16 ***
## colorI       -0.3694578  0.0180532 -20.465  < 2e-16 ***
## colorJ       -0.5122795  0.0233698 -21.921  < 2e-16 ***
## clarityIF     1.0243944  0.0416884  24.573  < 2e-16 ***
## claritySI1    0.5789271  0.0371596  15.579  < 2e-16 ***
## claritySI2    0.4017028  0.0378031  10.626  < 2e-16 ***
## clarityVS1    0.8002427  0.0377771  21.183  < 2e-16 ***
## clarityVS2    0.7246030  0.0375752  19.284  < 2e-16 ***
## clarityVVS1   0.9874107  0.0411373  24.003  < 2e-16 ***
## clarityVVS2   0.9057799  0.0392778  23.061  < 2e-16 ***
## depth        -0.0015523  0.0036563  -0.425 0.671255
## table        -0.0003379  0.0025295  -0.134 0.893758
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1289 on 979 degrees of freedom
## Multiple R-squared:  0.9841, Adjusted R-squared:  0.9838
## F-statistic:  3030 on 20 and 979 DF,  p-value: < 2.2e-16
```

The Model is log(price) = B0 + B1(log(carat)) + B2(depth) + B3(table) + B4(cut) + +B5(color) + B6(clarity) + e

- log(price): Natural log of the diamond price, which helps stabilize variance and makes relationships multiplicative.
- log(carat): Log of diamond weight, which captures the exponential effect of size on price.
- cut, color, clarity: Categorical variables (factors) that affect quality and thus price.
- depth, table: Numeric factors describing diamond proportions.

- R automatically chooses baseline/reference levels: Cut = "Fair", Color = "D" (best), Clarity = "I1" (lowest).
- All other levels are interpreted relative to these baselines.

Residuals = observed log(price) - predicted log(price)

- Symmetry around 0 which is good, it shows no major bias. Means model fits well, no extreme skew in residuals.
- 1st and 3rd quartiles show most residuals are within +-0.09, meaning predictions are very close to actual values in log-scale.
- Min and Max indicate a few points are farther off, which are likely due to large diamonds.

Estimator:

- Intercept - 7.9508574: Expected log(price) for baseline levels (cut = Fair, color = D, clarity = I1, log(carat)=0).
- log(carat) - 1.8889556: For 1% increase in carat, log(price) increases by 1.8889556%.
- cutGood - 0.1146925: "Good" cut increases log(price) by 0.1146925 compared to "Fair".
- cutIdeal - 0.2035390: "Ideal" cut adds 0.2035390 to log(price) compared to "Fair".
- cutPremium - 0.1512802: "Premium" cut adds 0.1512802 to log(price) compared to "Fair".
- cutVery Good - 0.1637435: "Very Goos" cut adds 0.1637435 to log(price) compared to "Fair".
- colorE-J: All negative, worse color lowers log(price). As color worsens (closer to J), price decreases.
- clarityIF-VVS2: All positive, measures internal flaws and surface imperfections of a diamond. Using I1 as the baseline, higher clarity grades (SI1–IF, VVS1–VVS2) all significantly increase log(price) (all $p < 0.05$). For example, IF and VVS1 diamonds have the largest positive effects, showing that better clarity consistently increases diamond price, as expected.

We can say log(carat) dominates, it has largest effect on price. Where cut, color, clarity also have meaningful effects. Coefficients match our expectations and better diamonds cost more, worse diamonds cost less.

- Significance p-values: Notice that both p-values of depth and table are much larger than 0.05, indicating that they are not statistically significant predictors of log(price) when controlling for carat, cut, color, and clarity. This shows that the model is not the best model, we should reconsider the model.
- Residual standard error (RSE): 0.1289, average deviation of predicted log(price) from actual log(price). It is small, a sign of good fit.
- R-squared (0.9841) and Adjusted R-squared (0.9838): 98% of variation in log(price) explained by the model, extremely high which is good.
- Adjusted R-squared accounts for number of predictors and penalizes not useful facts, it barely lower than R-squared, showing no over fitting concern which is good.
- F-statistic: 3030, p < 2.2e-16, signs of highly significant, which shows model is meaningful.

Since we find that this is still not the best model, we want to use aic to find the best model.

```
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:patchwork':
##
##     area
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
fullmodel <- lm(log(price) ~ log(carat) + cut + color + clarity + depth + table + x + y + z, data = samp
aicmodel <- stepAIC(fullmodel, direction = "both", trace = 0)
summary(aicmodel)
```

```
##
## Call:
## lm(formula = log(price) ~ log(carat) + cut + color + clarity +
##     depth + x + z, data = sample)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.42579 -0.08349  0.00105  0.08756  0.37765
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5.485144   0.424445  12.923  < 2e-16 ***
## log(carat)    1.460119   0.070800  20.623  < 2e-16 ***
## cutGood       0.110302   0.030811   3.580 0.000361 ***
## cutIdeal      0.188415   0.029736   6.336 3.59e-10 ***
## cutPremium    0.133207   0.030388   4.384 1.29e-05 ***
## cutVery Good  0.152826   0.029679   5.149 3.16e-07 ***
## colorE       -0.038067   0.014945  -2.547 0.011014 *
## colorF       -0.084329   0.015049  -5.604 2.73e-08 ***
## colorG       -0.130351   0.014638  -8.905  < 2e-16 ***
## colorH       -0.243510   0.015133 -16.091  < 2e-16 ***
## colorI       -0.380094   0.017757 -21.405  < 2e-16 ***
## colorJ       -0.527176   0.022982 -22.939  < 2e-16 ***
## clarityIF     1.063154   0.042289  25.140  < 2e-16 ***
## claritySI1    0.622688   0.038000  16.386  < 2e-16 ***
## claritySI2    0.442963   0.038575  11.483  < 2e-16 ***
## clarityVS1    0.840493   0.038609  21.769  < 2e-16 ***
## clarityVS2    0.766592   0.038409  19.959  < 2e-16 ***
## clarityVVS1   1.023208   0.041754  24.505  < 2e-16 ***
## clarityVVS2   0.943078   0.039993  23.581  < 2e-16 ***
## depth         0.013817   0.004245   3.255 0.001173 **
## x             0.280532   0.042584   6.588 7.28e-11 ***
## z            -0.086433   0.033779  -2.559 0.010652 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1262 on 978 degrees of freedom
## Multiple R-squared:  0.9848, Adjusted R-squared:  0.9845
## F-statistic:  3015 on 21 and 978 DF,  p-value: < 2.2e-16
```

After using aic, final model is: log(price)~log(carat) + cut + color + clarity + depth + x + z

- Response variable: log(price)
- Predictors included: log(carat): log-transformed weight of the diamond, cut: categorical, quality of cut, color: categorical, color grade, clarity: categorical, clarity grade, depth: numeric, depth percentage, x: numeric, length in mm, z: numeric, height in mm

Note: table and y were dropped by the stepwise AIC process, likely because they didn't improve the model

enough relative to the penalty for adding parameters.

The residuals are roughly symmetric around 0. The small interquartile range indicates a good fit.

- R-squared = 0.9848. The model explains about 98.5% of the variation in log(price), which is excellent.
- Adjusted R-squared = 0.9845. Adjusts for the number of predictors; still very high, indicating the model is not overfitting.
- Residual standard error (0.1262). On the log scale, residuals are small.
- All included predictors and the model are statistically significant ($p < 0.05$)

Observations from the model:

- Most influential predictors: log(carat) and clarity/cut/color have the largest effects.
- table and y were removed. This suggests their contribution is not significant when other variables are included.
- Model adequacy: High R-squared and significant predictors suggest a strong predictive model for diamond prices.

#2. Detect multicollinearity (use of VIF)

```
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'
```

```
## The following object is masked from 'package:dplyr':
##
##     recode
```

```
## The following object is masked from 'package:purrr':
##
##     some
```

```
bestmodel = lm(formula = log(price) ~ log(carat) + cut + color + clarity + depth + x + z, data = sample)
vif(bestmodel)
```

```
##                  GVIF Df GVIF^(1/(2*Df))
## log(carat) 106.294909  1       10.309942
## cut          1.729001  4        1.070840
## color        1.271861  6        1.020242
## clarity      1.757165  7        1.041086
## depth        2.138922  1        1.462505
## x          138.063205  1       11.750030
## z           34.330505  1        5.859224
```

Observation:

- log(carat): GVIF = 106.294909 and GVIF^(1/(2*Df)) = 10.309942, which are very high. Suggests strong collinearity with other predictors. Possibly with x and z, since dimensions are strongly related to carat.
- cut: GVIF = 1.729001 and GVIF^(1/(2*Df)) = 1.070840, which are all smaller than 5. This shows that it has very low correlation with other predictors, which means no multicollinearity concern.

- color: GVIF = 1.271861 and GVIF^(1/(2*Df)) = 1.020242, which are all smaller than 5. This shows that it has very low correlation with other predictors, which means no multicollinearity concern.
- clarity: GVIF = 1.757165 and GVIF^(1/(2*Df)) = 1.041086, which are all smaller than 5. This shows that it has very low correlation with other predictors, which means no multicollinearity concern.
- x: GVIF = 138.063205 and GVIF^(1/(2*Df)) = 11.750030, which are very high. Strong collinearity. Likely because x (length) is strongly correlated with carat (size).
- z: GVIF = 34.330505 and GVIF^(1/(2*Df)) = 5.859224, which are moderate-to-high. Shows some collinearity with other size-related variables (carat, x).

Summary:

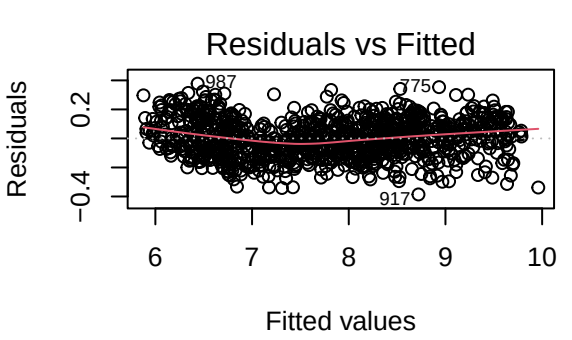
- High collinearity among size-related variables: log(carat), x, and z all have high VIFs. These variables are highly correlated by nature, so including them together inflates coefficients' standard errors. We need to remove x and z to reduce multicollinearity. The model will still capture size through log(carat), which is the most important predictor.
- Cut, color, and clarity have very low VIFs. No multicollinearity concerns for these grading variables.

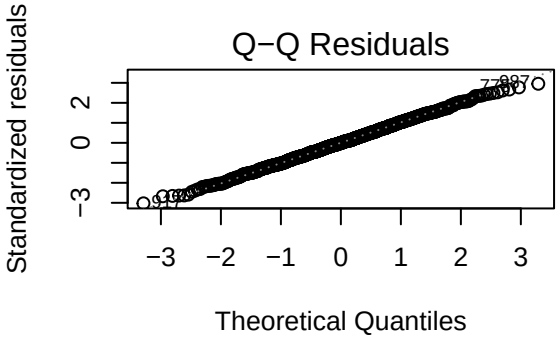
A cleaner, interpretable model could be: lm(log(price) ~ log(carat) + cut + color + clarity + depth, data = sample)

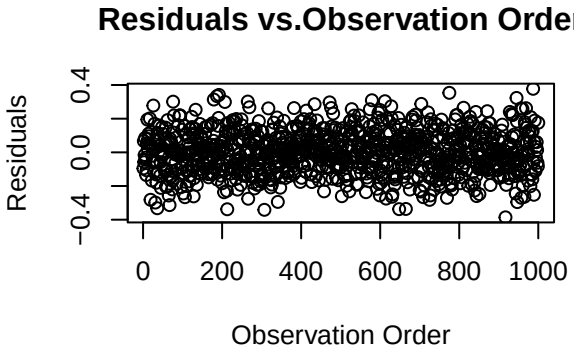
This keeps the important predictors and avoids collinearity issues from x and z.

```
# Model Assumption for final model

final <- lm(log(price) ~ log(carat) + cut + color + clarity + depth, data = sample)
par(mfrow = c(2, 2))
plot(final, which = c(1, 2))

a <-1:length(final$residuals)
 plot(final$residuals~a,
 ylab = "Residuals",xlab="Observation Order",
 main = "Residuals vs.Observation Order")

library(car)
vif(final)
```

```
##                GVIF Df GVIF^(1/(2*Df))
## log(carat) 1.341983  1        1.158440
## cut        1.664639  4        1.065774
## color      1.213431  6        1.016252
## clarity    1.589432  7        1.033652
## depth      1.271090  1        1.127426
```







Model Assumption Conclusion:

Based on the plots for the model lm(log(price) ~ log(carat) + cut + color + clarity + depth, data = sample):

Linearity: The residuals vs. fitted plot shows no obvious pattern, suggesting that the relationship between the predictors and log(price) is approximately linear.

Homoscedasticity: Using the residuals vs. fitted plot, it shows the spread of residuals appears fairly constant across fitted values, indicating that the assumption of constant variance is reasonably satisfied.

Normality of Residuals: The Q-Q plot of residuals is approximately linear, suggesting that residuals are roughly normally distributed.

Independence: Residuals plotted against observation order show no systematic pattern, supporting the independence assumption.

There is very little to no multicollinearity among our predictors, all VIFs are around 1. Each predictor contributes unique information to the model, and our coefficient estimates are stable.

The final model satisfies the key assumptions: the relationship between predictors and log(price) is approximately linear, residuals show constant variance and are roughly normally distributed, and there is no apparent dependence among observations. Multicollinearity is negligible, with all VIFs around 1, indicating stable and reliable coefficient estimates.

#3. CIs and PIs

```
library(knitr)
library(dplyr)

modelfinal <- lm(formula = log(price) ~ log(carat) + cut + color + clarity + depth, data = sample)
```

```r
set.seed(666)

num1 <- sample_n(sample, 1)
kable(num1)
```

...1	carat	cut	color	clarity	depth	table	price	x	y	z	logprice	logcarat
28613	0.3	Very Good	E	SI1	63.2	57	675	4.3	4.25	2.7	6.514713	-1.203973

```r
# Predict log(price) with confidence interval
predict(modelfinal, newdata = num1, interval = "confidence", level = 0.95)
```

```
##        fit      lwr     upr
## 1 6.258623 6.227526 6.28972
```

```r
# Predict log(price) with prediction interval
predict(modelfinal, newdata = num1, interval = "prediction", level = 0.95)
```

```
##        fit     lwr      upr
## 1 6.258623 6.00381 6.513436
```

We randomly select one observation from the dataset using. This gives a combination of predictors (log(carat), cut, color, clarity, depth).

CI Interpretation:

- fit = 6.2586: the predicted log(price) for this combination.
- lwr = 6.2275, upr = 6.2897: 95% confidence interval for the mean log(price) of all diamonds with these characteristics.
- The mean price is estimated between $506(exp(6.227526)) and $541(exp(6.28972)) for diamonds with these characteristics.

PI Interpretation: It is wider because it accounts for both model uncertainty and the natural variability of individual diamond prices.

- fit = 6.2586: predicted log(price) for a single future diamond with these characteristics.
- lwr = 6.0038, upr = 6.5134: 95% prediction interval for a single future diamond.
- The predicted price of a single future diamond is between $403 and $671.

Conclusion: We are 95% confident that the average log(price) for diamonds with carat = 0.3, cut = Very Good, color = E, clarity = SI1, and depth = 63.2 lies between 6.228 and 6.290. For prediction, we are 95% confident that the price of a single new diamond with these characteristics will fall between log(price) 6.004 and 6.513. This shows that while the average price is fairly precise, individual diamond prices can vary more widely due to natural variation and other unobserved factors.

#4. Summarize report

This analysis investigates the relationship between diamond prices and their physical characteristics, with the goal of building a predictive model that is both accurate and interpretable. The dataset contains 1,000 observations and includes quantitative variables such as carat, depth, table, and the diamond dimensions (x, y, z), as well as categorical grading variables cut, color, and clarity. Initial exploratory analysis revealed that both price and carat are strongly right-skewed, with means substantially higher than medians, suggesting

a non-normal distribution. Scatterplots and correlation analysis indicated a strong positive relationship between carat and price, and high correlations among size-related variables (x, y, z).

We first fitted a simple linear regression model with price as a function of carat. This model showed a strong positive relationship, with carat explaining approximately 85% of the variance in price. However, residual diagnostics revealed violations of regression assumptions, including non-normality and heteroscedasticity, as evidenced by a funnel shape in residuals versus fitted values and heavy-tailed residuals in Q-Q plots. To address these issues, we applied a log transformation to both price and carat, creating a log–log model. This transformation substantially improved the model: residuals became more symmetric, the funnel shape disappeared, variance stabilized, and the adjusted R-squared increased from 0.8524 to 0.9379. The transformation also linearized the relationship, reduced the influence of outliers, and improved interpretability.

Building on the log–log model, we then evaluated whether including additional predictors would improve performance. We first fitted a model including log(carat), cut, color, clarity, depth, and table. Stepwise model selection using AIC suggested an alternative model including log(carat), cut, color, clarity, depth, x, and z. However, variance inflation factor (VIF) analysis revealed severe multicollinearity for x and z, as well as for log(carat), with GVIF values exceeding acceptable thresholds. The categorical grading variables (cut, color, clarity) and depth had low VIFs and no collinearity concerns. Based on this, x and z were removed, resulting in a final, well-behaved model:

$$\log(\text{price}) \sim \log(\text{carat}) + \text{cut} + \text{color} + \text{clarity} + \text{depth}$$

This final model balances predictive power, interpretability, and compliance with regression assumptions.

Using the final model, we demonstrated prediction for a specific combination of predictors. For a diamond with carat = 0.3, cut = Very Good, color = E, clarity = SI1, and depth = 63.2, the predicted log(price) was 6.259. The 95% confidence interval for the mean log(price) was 6.228–6.290, corresponding to an estimated mean price between approximately \$506 and \$541. The 95% prediction interval for a single diamond was 6.004–6.513, corresponding to a predicted price between \$403 and \$671. This highlights the difference between estimating the mean price for diamonds with these characteristics and predicting the price of an individual diamond, with the prediction interval being wider due to additional variability.

Overall, the analysis confirms that carat is the strongest predictor of diamond price, but grading variables (cut, color, clarity) and depth provide significant additional explanatory power. The log transformation was crucial for satisfying regression assumptions, improving residual behavior, and stabilizing variance. Stepwise evaluation, AIC, and VIF diagnostics allowed us to construct a final model that is robust, interpretable, and avoids multicollinearity issues, making it suitable for predicting diamond prices based on physical characteristics and quality metrics.