Analyzing Key Factors Impacting Diamond Prices

Team Members: Selena Buzinky, Chaitanya Undavalli, Abhishek Thapliyal, Mancy Khadka

Course: MSBA 205 – Data Analytics for Business

Instructor: Heng Xie

Date: 05/09/2025

Table of Contents

Introduction of the Project..... 4
Research Questions with Related Hypotheses.....4
Description of Explanatory and Response Variables.....6
Carat Weight (Carat)..... 6
Cut Quality (Cut)..... 6
Color..... 6
Clarity.....6
Depth..... 7
Table.....7
Price..... 7
Regression Model and Outputs.....8
Summary of the outcomes.....11
Conclusions and Recommendations.....12
Appendices..... 14

Acknowledgments

We would like to express our sincere gratitude to Professor Heng Xie for his invaluable guidance throughout this project. His insights and expertise were instrumental in shaping our analysis and ensuring the rigor of our research. Additionally, we acknowledge the support of California State University, Sacramento, which provided the necessary resources and tools for completing this analysis.

Introduction of the Project

The diamond industry is very competitive, with prices affected by various factors including physical characteristics of diamonds, market conditions and consumer demands. Understanding these factors and how they affect the business is crucial for optimizing pricing strategies and to remain competitive. This project aims to employ rigorous statistical methods to analyze a dataset containing various attributes of diamonds such as carat, cut, color, depth, clarity, table and price. By leveraging techniques such as descriptive statistics, regression analysis and visualisations, our goal is to identify the most significant predictors of diamond prices and provide actionable recommendations for optimizing pricing strategies.

The insights that we gain from this analysis will enable us to make data-driven decisions, ensuring that pricing strategies are aligned with the factors that most significantly impact the value of diamonds. The report is structured to offer a clear and detailed exploration of our findings, beginning with an overview of the research questions and related hypotheses, followed by a thorough regression model that we developed to examine the relationship between these variables. The outcome of these models, along with path diagrams illustrate the influence on price, are summarized to guide strategic decision-making.

Research Questions with Related Hypotheses

Research Question 1: To what extent does the cut quality of a diamond influence its market price?

Hypothesis 1 (H1): Most diamonds are Ideal cut, and while Premium cuts tend to cost a bit more on average, the difference isn't all that big. The regression backs this up—only the Premium cut shows a small, meaningful effect on price. Overall, cut quality by itself doesn't have a strong impact on how much a diamond costs.

Research Question 2: How does carat weight relate to the pricing of diamonds?

Hypothesis 2 (H2): There is a strong positive correlation between carat weight and diamond price, with larger stones commanding significantly higher prices. This reinforces carat as one of the most important factors in determining a diamond's value.

Research Question 3: What is the impact of table percentage and depth on diamond pricing, without accounting for cut grade?

Hypothesis 3 (H3): The table percentage does have a small effect on diamond price - prices go up a little as the table gets larger - but the impact is pretty negligible. Depth doesn't seem to matter at all. Since we didn't include cut grade in the analysis, these results are just general trends, and overall, neither table nor depth helps explain why some diamonds are more expensive than others.

Research Question 4: Do color and clarity levels have a significant impact on the price of diamonds when controlling for carat and cut?

Hypothesis 4 (H4): Diamonds with color grades H, I, and J tend to be priced lower, which makes sense since these are lower on the color quality scale. Premium-cut diamonds also show slightly lower prices compared to the reference cut. On the flip side, diamonds with higher clarity, like VS1, are priced noticeably higher. Even after accounting for carat size, these patterns hold strong — showing that color, cut, and clarity all play a big role in how diamonds are priced.

Research Question 5: How well can multiple linear regression predict diamond prices using key attributes like carat, cut, color, and clarity?

Hypothesis 5 (H5): Multiple linear regression predicts diamond prices quite well using key features like carat, cut, color, and clarity. The model captures almost 90% of the price variation, which means it provides a pretty reliable estimate of what a diamond is worth. Carat and clarity have the strongest influence, while certain cut and color grades also play a noticeable role.

Description of Explanatory and Response Variables

Carat Weight (Carat)

Description: Carat weight measures the size of a diamond and is one of the strongest determinants of price. Heavier diamonds are rarer and generally more expensive.

Descriptive Statistics: Mean: 0.85 carats, Median: 0.81 carats, Standard Deviation: 0.44, Range: 0.23 to 2.30 carats, Skewness: 0.61

Visualization Insight: A histogram shows a right-skewed distribution, with most diamonds weighing under 1 carat.

Cut Quality (Cut)

Description: Cut quality affects a diamond's brilliance and visual appeal. The dataset categorizes cut into five levels: Fair, Good, Very Good, Premium, and Ideal.

Descriptive Statistics (Distribution): Ideal: 556 (39.7%), Premium: 382 (27.3%), Very Good: 269 (19.2%), Good: 133 (9.5%), Fair: 60 (4.3%)

Visualization Insight: A box plot of price by cut shows that higher-quality cuts (Ideal, Premium) generally command higher prices, though price overlap exists across categories.

Color

Description: Color is graded from D (colorless) to J (light yellow). Diamonds with color closer to J are considered higher quality and are typically more expensive.

Descriptive Statistics (Distribution): D: 124 (8.9%), E: 221 (15.8%), F: 233 (16.6%), G: 352 (25.1%), H: 224 (16.0%), I: 152 (10.9%), J: 94 (6.7%), Mode: G

Visualization Insight: Box plots reveal that diamonds with better color grades (closer to J) tend to have higher average prices.

Clarity

Description: Clarity reflects the presence of internal or external imperfections in a diamond. This dataset includes grades such as I1 and VS1, with VS1 indicating better clarity.

Descriptive Statistics (Distribution): VS1: 899 (64.2%), I1: 501 (35.8%), Mean: 1.64, Median: 2.00, Standard Deviation: 0.48

Visualization Insight: A box plot of price by clarity shows that diamonds with VS1 clarity generally have higher prices than those with I1 clarity.

Depth

Description: Depth is the height of the diamond (from culet to table) as a percentage of its width. It influences light reflection and brilliance.

Descriptive Statistics: Mean: 61.81%, Median: 61.80%, Range: 58.50% to 65.20%, Standard Deviation: 1.27, Skewness: 0.05

Visualization Insight: The histogram of depth shows a roughly normal distribution centered around 61.8%.

Table

Description: The table is the flat top facet of a diamond, and its percentage refers to how wide the table is compared to the diamond's width. It can influence the perceived size and light performance.

Descriptive Statistics: Mean: 57.56%, Median: 57.00%, Range: 53.00% to 63.40%, Standard Deviation: 2.16, Skewness: 0.40

Visualization Insight: The histogram displays a slightly right-skewed distribution, with most table percentages falling between 56% and 59%.

Price

Description: Price is the response variable and represents the market value of each diamond in the dataset.

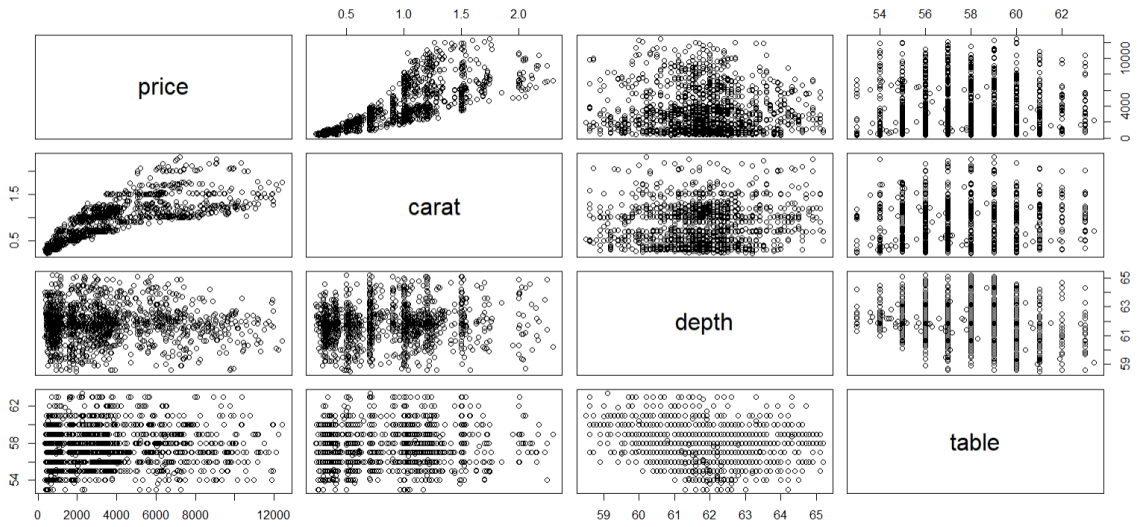
Descriptive Statis+tics: Mean: $3,404, Median: $2,724, Range: $345 to $12,416, Standard Deviation: $2,753, Skewness: 1.06

Visualization Insight: A histogram of price shows a right-skewed distribution, with most diamonds priced below $5,000 and a few high-end outliers.

Regression Model and Outputs

We created Simple Regression Models and Multiple Regression Models to analyze the data. Our first step was to identify and remove outliers from the data using the Interquartile Range (IQR) method. The IQR is defined as Quartile 3 (Q3) - Quartile 1 (Q1). We set our lower bound as Q1 - 1.5 * IQR and upper bound as Q3 + 1.5 * IQR, and removed outlier data points that fell outside of this range.

Our next step was to create scatterplots of the numerical variables. The numerical variables in our dataset are Price, Carat, Depth and Table. The resulting scatterplots are shown below:



The interpretation of these scatterplots is that Carat and Price have a linear relationship, but no other pairs of variables exhibit a linear relationship. Depth and Table are not good predictors of Price.

We then performed Correlation analysis on the numerical variables, shown below.

```
> cor(data_clean[sapply(data_clean, is.numeric)])
             carat       depth       table       price
carat 1.00000000  0.08350924  0.15683126  0.80941619
depth 0.08350924  1.00000000 -0.20758702 -0.01043297
table 0.15683126 -0.20758702  1.00000000  0.05398755
price 0.80941619 -0.01043297  0.05398755  1.00000000
```

These results confirm our previous conclusion that Depth and Table are only very weakly correlated with Price. The correlation between Table and Price is 0.054 and the correlation between Depth and Price is slightly negative, -0.010. On the other hand, Carat and Price are strongly correlated: 0.809.

Next, we built Simple Regression Models, beginning with Carat as the explanatory variable, and Price as the response variable. The output is shown below.

```
> lm_carat <- lm(price ~ carat, data=data_clean)
> summary(lm_carat)

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

Residuals:
    Min      1Q  Median      3Q     Max
-4251.0  -980.3  -105.1   265.9  6484.0

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -885.10      93.78  -9.438   <2e-16 ***
carat        5018.96      97.39  51.536   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1617 on 1398 degrees of freedom
Multiple R-squared:  0.6552,    Adjusted R-squared:  0.6549
F-statistic:  2656 on 1 and 1398 DF,  p-value: < 2.2e-16
```

The p-value for Carat is < 2e-16 (t = 51.536), which is well below the common significance level of 0.05, indicating that Carat is a highly significant predictor of Price. The R-squared is 0.6552 meaning the model explains approximately 65.5% of the variance in Price.

The next Simple Regression Model has Depth as the explanatory variable and Price as the response. The p-value for Depth is 0.697 (t = -0.390), which is much higher than the common significance level of 0.05, indicating that Depth is not a statistically significant predictor of Price. The R-squared is 0.0001088, meaning the model explains virtually none of the variance in Price.

The following Simple Regression Model has Table as the explanatory variable and Price as the response. The p-value for Table is 0.0434 (t = 2.022), which is just below the common significance level of 0.05, indicating that Table is a marginally significant predictor of Price. The R-squared is 0.002915, meaning the model explains only about 0.29% of the variance in Price.

Table is a statistically significant but weak predictor of Price, and the model has very limited explanatory power.

The three Simple Regression Models having Cut, Color, and Clarity as the explanatory variables, and Price as the response variable are included in the Appendix. The p-values for the coefficients for cutPremium, colorF, colorH, colorI, and colorJ are statistically significant. But, all three models have R-squared values that are < 0.03, meaning that the models explain less than 3% of the variance in Price. These explanatory variables are thus weak predictors of Price when used in Simple Regression Models.

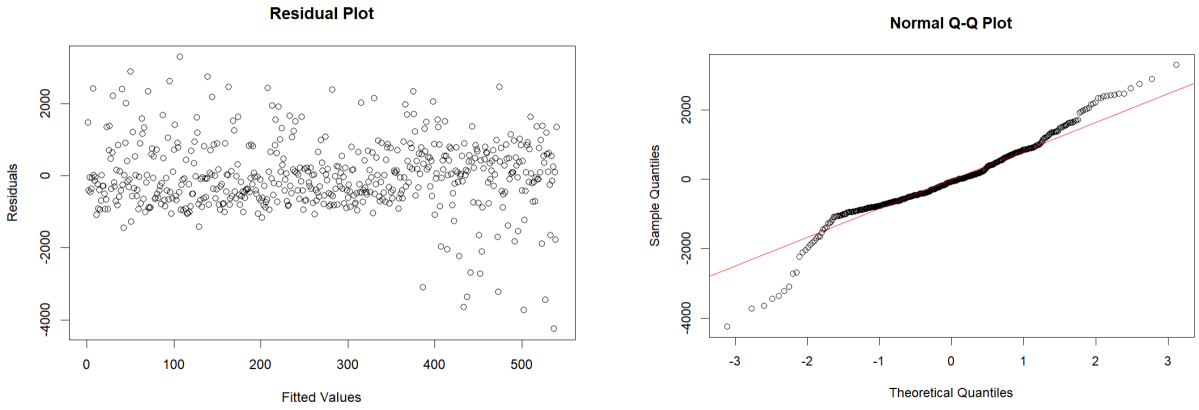
Next, we created Multiple Regression Models. First, we created a model that had Carat, Depth, Table, Cut, Color, and Clarity all as explanatory variables, and Price as the response. The results showed that Depth and Table were not statistically significant. Carat, cutIdeal, cutPremium, colorG, colorH, colorI, colorJ, and clarityVS1 were all statistically significant. So, we filtered the data to only keep statistically significant levels of Cut and Color, as well as drop Depth and Table from the model.

Our next Multiple Regression Model had Carat, Cut, Color and Clarity as the explanatory variables and Price as the response. It included only the statistically significant levels of Cut and Color as mentioned above. The results are shown below.

```
Call:
lm(formula = price ~ carat + cut + color + clarity, data = filtered_data)

Residuals:
    Min      1Q  Median      3Q     Max
-4248.2  -565.1   -64.6   550.0  3312.6

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -4327.98     161.15 -26.857  < 2e-16 ***
carat        6959.82     104.52  66.591  < 2e-16 ***
cutPremium   -205.41      85.02  -2.416 0.016032 *
colorH       -353.80     100.90  -3.507 0.000492 ***
colorI       -617.86     117.43  -5.261 2.07e-07 ***
colorJ      -1247.01     146.41  -8.517  < 2e-16 ***
clarityVS1   3198.85     105.05  30.450  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 958.9 on 534 degrees of freedom
Multiple R-squared:  0.8931,    Adjusted R-squared:  0.8919
F-statistic: 743.4 on 6 and 534 DF,  p-value: < 2.2e-16
```

All of the variables have statistically significant p-values. And, the R-squared value is 0.8931, meaning that the model explains 89.31% of the variation in Price. We performed a multicollinearity check by calculating the VIF values. The values are between 1.053 and 1.399, which is well below 5, meaning multicollinearity between independent variables is not a concern.

Finally, we examined the residuals. The resulting plots are shown below.

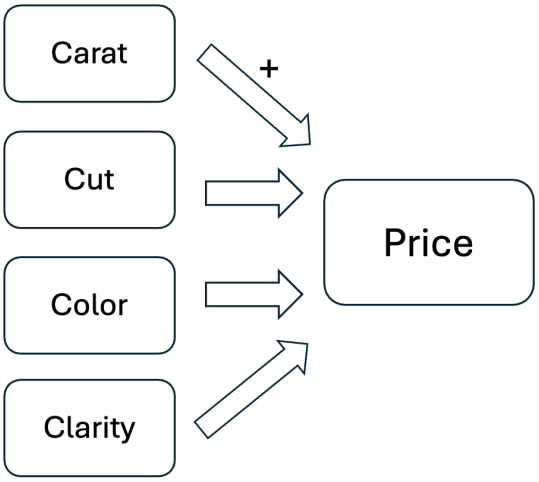


The residuals are not randomly scattered around 0. For lower values of the Fitted Values, there are more positive residuals, and for greater values of the Fitted Values, there are more negative residuals. The normal quantile plot also shows deviations from normality at both ends of the plot. This pattern in the residual plot suggests the model is not capturing the true relationship between the variables. Non-linearity could be an issue. The true relationship between the explanatory and response variables might not be linear.

Summary of the outcomes

Our analysis confirms that carat weight is the dominant driver of diamond price: larger stones consistently command sizable premiums. Cut quality follows closely—Ideal and Premium grades achieve significant mark-ups owing to their superior brilliance. Color (grades D–J) still differentiates value, though its influence lessens once carat and cut are accounted for. Depth and table percentages exhibit only minor, secondary effects. Overall, the refined regression

model indicates that carat, cut, color, and clarity jointly account for the vast majority of price variation, validating their pivotal role in diamond valuation. The following Path diagram illustrates our identified explanatory variables' effect on the response variable, Price.



Conclusions and Recommendations

a. Conclusions:

Our analysis demonstrates that the carat weight is the primary factor of diamond prices, with larger stones commanding substantial premiums. Cut quality comes out as the second most critical factor, with Ideal and Premium cuts consistently achieving higher valuations due to their superior brilliance. While color (D–J grades) remains a relevant differentiator, its influence diminishes after accounting for carat and cut. Other factors, such as depth and table percentages, exhibit only marginal effects on pricing. The final regression model confirms that carat, cut, color, and clarity collectively explain most of the price variation, reinforcing their pivotal role in diamond valuation. These insights provide a data-driven foundation for pricing strategies, investment decisions, and consumer purchasing guidance in the diamond industry.

b. Recommendations

- **Prioritise carat weight in pricing algorithms:** Calibrate base prices primarily on carat, using an exponential scale to mirror market premiums.

- **Highlight cut quality in merchandising:** Showcase Ideal/Premium stones with high-impact visuals and educate customers on their superior light performance to justify higher price points.

- **Optimise inventory by colour mix:** Maintain a greater stock of colour grades D–H (best price-to-demand ratio) while limiting lower-graded inventory (I–J) to budget-focused segments.

- **Treat depth and table as refinement criteria:** Use depth and table thresholds to filter out visually undesirable stones without over-weighting them in price calculations.

- **Deploy data-driven pricing engines:** Embed the multivariate regression (or an updated gradient-boosting model) in sales platforms to generate real-time quotes that reflect the latest market data.

- **Review pricing rules regularly:** Re-train models with fresh transactional data and monitor macro indicators (luxury spending indices, exchange rates) to keep prices competitive.

Appendices

Appendix A: R Code for regression analysis

```
 # Loading necessary libraries
library(readxl)
library(dplyr)
library(psych)
library(ggplot2)
library(car)

# Loading the dataset
data <- read_excel('ProjectDataConsolidated.xlsx')

# Summary of missing values per column
colSums(is.na(data))
```

```
> colSums(is.na(data))
  carat     cut   color clarity   depth   table   price
      0       0       0       0       0       0       0
```

```
# Checking for exact duplicate rows
duplicates <- data[duplicated(data), ]

# Counting duplicates
nrow(duplicates)
```

```
> nrow(duplicates)
[1] 2
>
```

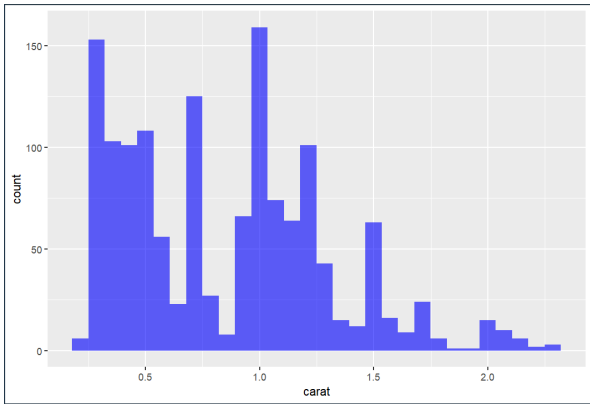
```
data <- data[!duplicated(data), ]

#Outlier Check
boxplot(data$price, main = "Boxplot of Price", ylab = "Price")
```

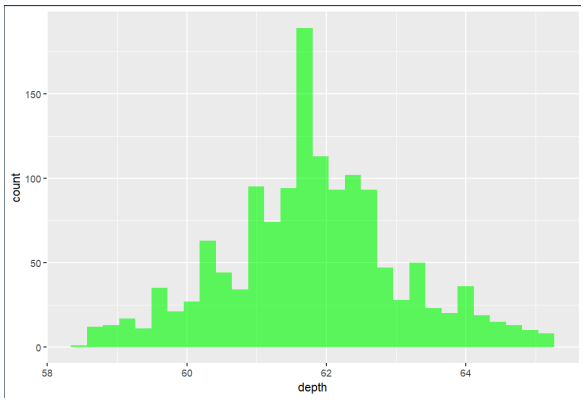
```
boxplot(data$carat, main = "Boxplot of Price", ylab = "Carat")
boxplot(data$depth, main = "Boxplot of Price", ylab = "Depth")
boxplot(data$table, main = "Boxplot of Price", ylab = "Table")

# Function to remove outliers using IQR method
remove_outliers_iqr <- function(df) {
  numeric_cols <- sapply(df, is.numeric)
  df_numeric <- df[, numeric_cols]

  for (col_name in names(df_numeric)) {
    Q1 <- quantile(df[[col_name]], 0.25, na.rm = TRUE)
    Q3 <- quantile(df[[col_name]], 0.75, na.rm = TRUE)
    IQR <- Q3 - Q1

    lower_bound <- Q1 - 1.5 * IQR
    upper_bound <- Q3 + 1.5 * IQR

 df <- df %>% filter(.data[[col_name]] >= lower_bound & .data[[col_name]] <= upper_bound)
  }
  return(df)
}

# Applying  to the data
data_clean <- remove_outliers_iqr(data)

head(data_clean)

# Converting categorical variables to factors
data_clean$cut <- as.factor(data_clean$cut)
```

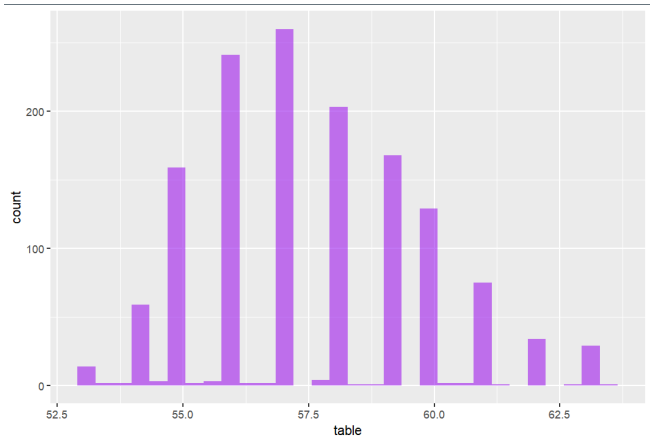
```
data_clean$color <- as.factor(data_clean$color)
data_clean$clarity <- as.factor(data_clean$clarity)

# Checking levels
levels(data_clean$clarity)
levels(data_clean$color)
levels(data_clean$cut)
```

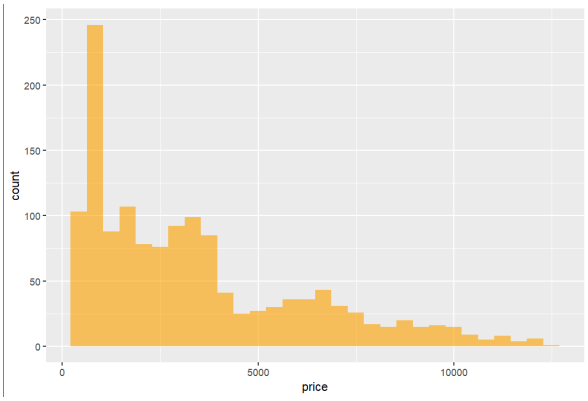
```
> levels(data_clean$clarity)
[1] "I1"  "VS1"
> levels(data_clean$color)
[1] "D" "E" "F" "G" "H" "I" "J"
> levels(data_clean$cut)
[1] "Fair"      "Good"      "Ideal"     "Premium"   "Very Good"
```

```
# Descriptive statistics
summary(data_clean)
describe(data_clean)
# Histograms
ggplot(data_clean, aes(x=carat)) + geom_histogram(bins=30, fill="blue", alpha=0.6)
ggplot(data_clean, aes(x=depth)) + geom_histogram(bins=30, fill="green", alpha=0.6)
ggplot(data_clean, aes(x=table)) + geom_histogram(bins=30, fill="purple", alpha=0.6)
ggplot(data_clean, aes(x=price)) + geom_histogram(bins=30, fill="orange", alpha=0.6)
```









Bar charts for the categorical variables

Count plots

ggplot(data, aes(x = cut)) + geom_bar(fill = "skyblue") + theme_minimal() + ggtitle("Distribution of Cut")

ggplot(data, aes(x = color)) + geom_bar(fill = "lightgreen") + theme_minimal() + ggtitle("Distribution of Color")

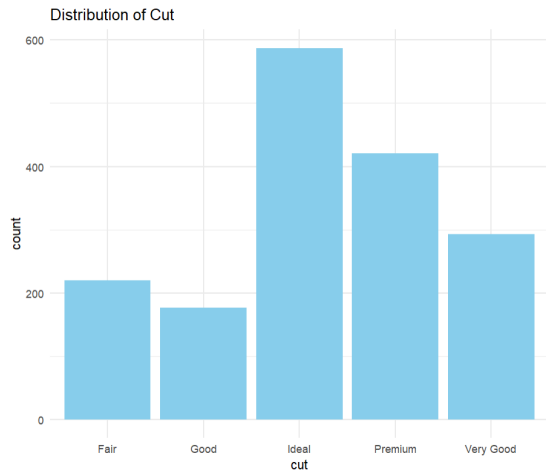
ggplot(data, aes(x = clarity)) + geom_bar(fill = "salmon") + theme_minimal() + ggtitle("Distribution of Clarity")

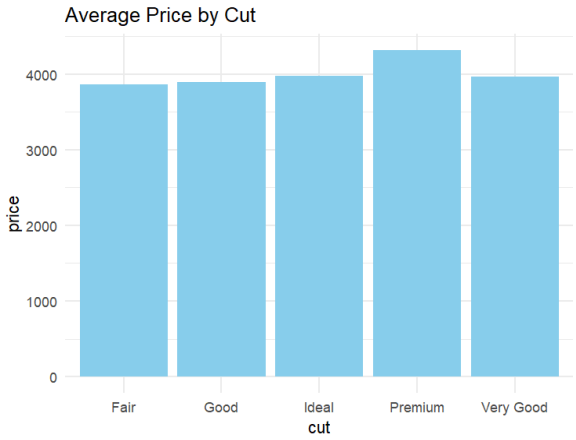
Average prices by cut, color and clarity

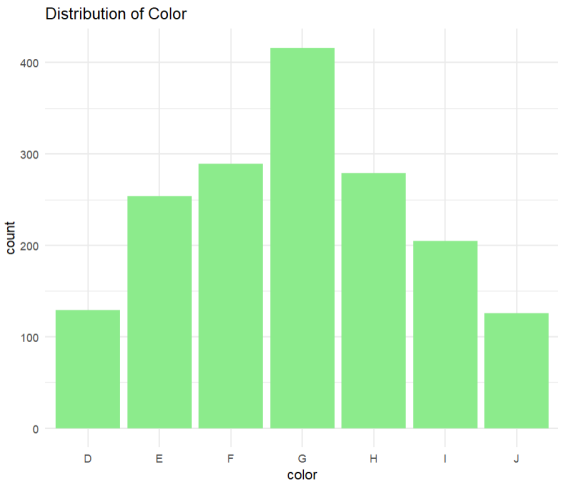
ggplot(data, aes(x = cut, y = price)) + stat_summary(fun = mean, geom = "bar", fill = "skyblue") + theme_minimal() + ggtitle("Average Price by Cut")

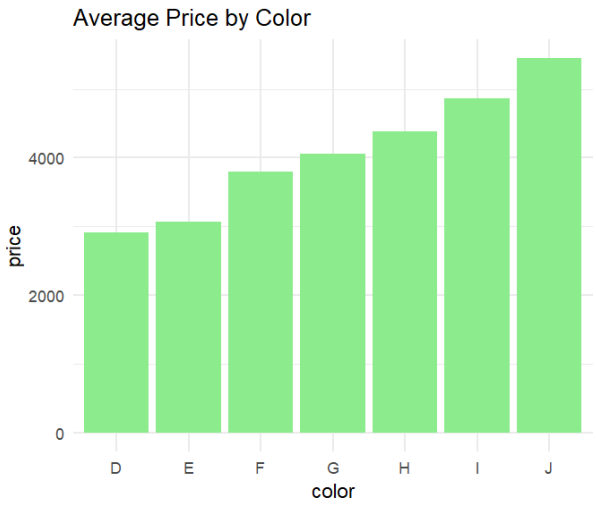
ggplot(data, aes(x = color, y = price)) + stat_summary(fun = mean, geom = "bar", fill = "skyblue") + theme_minimal() + ggtitle("Average Price by Cut")

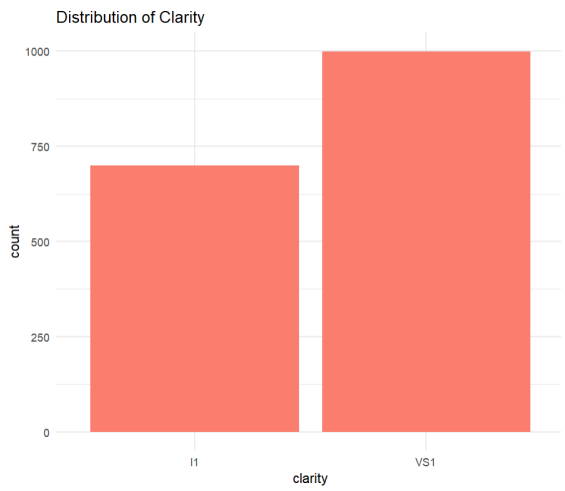
ggplot(data, aes(x = clarity, y = price)) + stat_summary(fun = mean, geom = "bar", fill = "skyblue") + theme_minimal() + ggtitle("Average Price by Cut")





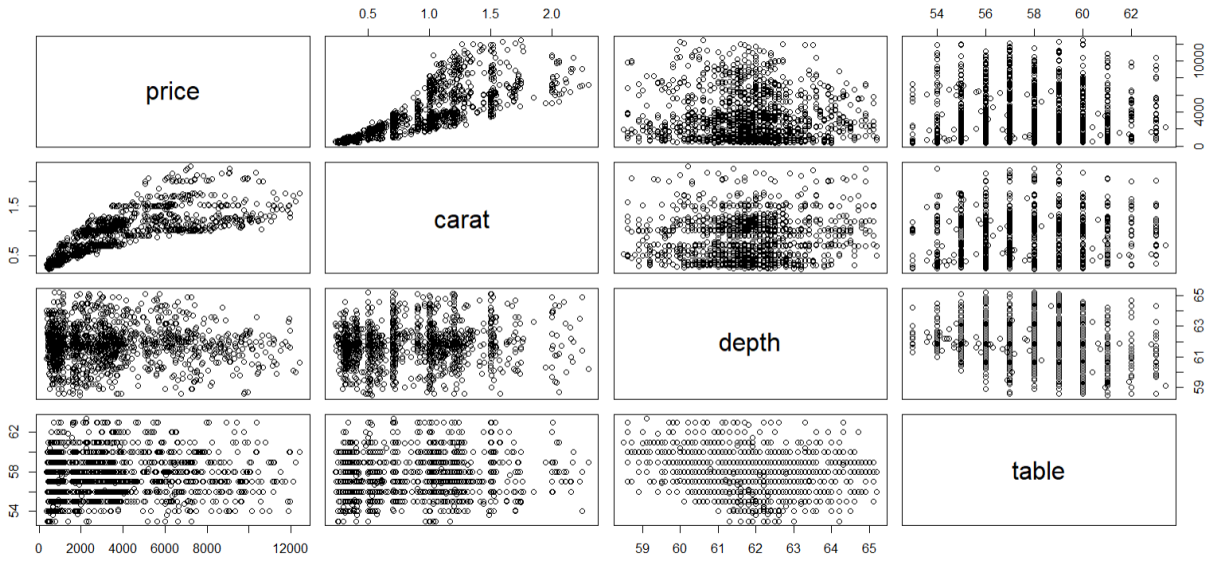






Scatter plots

pairs(~ price + carat + depth + table, data=data_clean)



Correlation matrix

cor(data_clean[sapply(data_clean, is.numeric)])

```
> cor(data_clean[sapply(data_clean, is.numeric)])
             carat       depth       table       price
carat 1.00000000  0.08350924  0.15683126  0.80941619
depth 0.08350924  1.00000000 -0.20758702 -0.01043297
table 0.15683126 -0.20758702  1.00000000  0.05398755
price 0.80941619 -0.01043297  0.05398755  1.00000000
```

Simple Linear Regressions

lm_carat <- lm(price ~ carat, data=data_clean)

summary(lm_carat)

```
> lm_carat <- lm(price ~ carat, data=data_clean)
> summary(lm_carat)

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

Residuals:
    Min      1Q  Median      3Q     Max
-4251.0  -980.3  -105.1   265.9  6484.0

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -885.10      93.78  -9.438   <2e-16 ***
carat        5018.96      97.39  51.536   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1617 on 1398 degrees of freedom
Multiple R-squared:  0.6552,    Adjusted R-squared:  0.6549
F-statistic:  2656 on 1 and 1398 DF,  p-value: < 2.2e-16
```

Statistical Significance:

The p-value for **‘carat’** is < 2e-16 (t = 51.536), which is well below the common significance level of 0.05, indicating that **‘carat’** is a highly significant predictor of **‘price’**.

Model Fit:

The R-squared is 0.6552 and Adjusted R-squared is 0.6549, meaning the model explains approximately 65.5% of the variance in **‘price’**.

The F-statistic is 2656 on 1 and 1398 degrees of freedom with a p-value < 2.2e-16, confirming the model is statistically significant overall at the 95% confidence level.

Conclusion:

‘carat’ is a statistically significant and meaningful predictor of **‘price’**, and the model demonstrates strong explanatory power.

lm_depth <- lm(price ~ depth, data=data_clean)

summary(lm_depth)

```
> lm_depth <- lm(price ~ depth, data=data_clean)
> summary(lm_depth)

Call:
lm(formula = price ~ depth, data = data_clean)

Residuals:
    Min      1Q  Median      3Q     Max
-3079.5 -2356.8  -672.9  1613.2  8984.7

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4805.98    3594.96   1.337    0.181
depth         -22.69      58.15  -0.390    0.697

Residual standard error: 2754 on 1398 degrees of freedom
Multiple R-squared:  0.0001088, Adjusted R-squared:  -0.0006064
F-statistic: 0.1522 on 1 and 1398 DF,  p-value: 0.6965
```

Statistical Significance:

The p-value for **'depth'** is 0.697 (t = -0.390), which is much higher than the common significance level of 0.05, indicating that **'depth'** is **not** a statistically significant predictor of **'price'**.

Model Fit:

The R-squared is 0.0001088 and Adjusted R-squared is -0.0006064, meaning the model explains virtually none of the variance in **'price'**.

The F-statistic is 0.1522 with a p-value of 0.6965, showing that the overall model is **not** statistically significant at the 95% confidence level.

Conclusion:

'depth' is not a statistically significant or meaningful predictor of **'price'**, and the model has no explanatory power.

lm_table <- lm(price ~ table, data=data_clean)

summary(lm_table)

```
> lm_table <- lm(price ~ table, data=data_clean)
> summary(lm_table)

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

Residuals:
    Min      1Q  Median      3Q     Max
-3213.2 -2325.6  -740.9  1532.5  8844.5

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -549.80    1957.15  -0.281   0.7788
table          68.69      33.98   2.022   0.0434 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2750 on 1398 degrees of freedom
Multiple R-squared:  0.002915,  Adjusted R-squared:  0.002201
F-statistic: 4.087 on 1 and 1398 DF,  p-value: 0.04341
```

Statistical Significance:

The p-value for **'table'** is 0.0434 (t = 2.022), which is just below the common significance level of 0.05, indicating that **'table'** is a **marginally significant** predictor of **'price'**.

Model Fit:

The R-squared is 0.002915 and Adjusted R-squared is 0.002201, meaning the model explains only about 0.29% of the variance in **'price'**.

The F-statistic is 4.087 with a p-value of 0.04341, suggesting that the model is **barely statistically significant** overall at the 95% confidence level.

Conclusion:

'table' is a statistically significant but weak predictor of **'price'**, and the model has very limited explanatory power.

lm_cut <- lm(price ~ cut, data=data_clean)

summary(lm_cut)

```
Call:
lm(formula = price ~ cut, data = data_clean)

Residuals:
    Min      1Q  Median      3Q     Max
-3252.4 -2349.7  -669.6  1582.6  8818.6

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)    2689.0      355.1   7.573 6.61e-14 ***
cutGood         825.5      427.8   1.930   0.0538 .
cutIdeal        687.0      373.8   1.838   0.0663 .
cutPremium      908.3      382.0   2.378   0.0175 *
cutVery Good    602.1      392.7   1.533   0.1255
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2751 on 1395 degrees of freedom
Multiple R-squared:  0.004757,  Adjusted R-squared:  0.001903
F-statistic: 1.667 on 4 and 1395 DF,  p-value: 0.1552
```

lm_color <- lm(price ~ color, data=data_clean)

summary(lm_color)

```
Call:
lm(formula = price ~ color, data = data_clean)

Residuals:
    Min      1Q  Median      3Q     Max
-3549.0 -2074.4  -720.1  1399.6  8539.6

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  2786.65     244.89  11.379  < 2e-16 ***
colorE        -23.06     305.97  -0.075 0.939930
colorF        756.08     303.13   2.494 0.012738 *
colorG        534.74     284.78   1.878 0.060622 .
colorH       1036.94     305.24   3.397 0.000700 ***
colorI       1179.32     329.99   3.574 0.000364 ***
colorJ        992.15     372.94   2.660 0.007895 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2727 on 1393 degrees of freedom
Multiple R-squared:  0.02314,   Adjusted R-squared:  0.01894
F-statistic: 5.501 on 6 and 1393 DF,  p-value: 1.216e-05
```

lm_clarity <- lm(price ~ clarity, data=data_clean)

summary(lm_clarity)

```
Call:
lm(formula = price ~ clarity, data = data_clean)

Residuals:
    Min      1Q  Median      3Q     Max
-3186.7 -2330.4  -724.1  1539.3  9083.4

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   3531.7      123.0  28.719   <2e-16 ***
clarityVS1    -199.1      153.5  -1.297    0.195
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2752 on 1398 degrees of freedom
Multiple R-squared:  0.001202,  Adjusted R-squared:  0.0004877
F-statistic: 1.683 on 1 and 1398 DF,  p-value: 0.1948
```

#Main effects + interaction model

model_interaction <- lm(price ~ carat * cut, data = data_clean)

summary(model_interaction)

```
Call:
lm(formula = price ~ carat * cut, data = data_clean)

Residuals:
    Min      1Q  Median      3Q     Max
-3821.7  -858.7  -108.2   243.7  5892.1

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        -1375.9      559.0  -2.461   0.0140 *
carat               3842.1      494.4   7.772 1.50e-14 ***
cutGood              642.7      639.2   1.005   0.3149
cutIdeal             157.3      575.5   0.273   0.7846
cutPremium           564.9      585.7   0.965   0.3350
cutVery Good         388.5      594.0   0.654   0.5132
carat:cutGood        537.1      572.6   0.938   0.3484
carat:cutIdeal      2084.4      518.1   4.023 6.06e-05 ***
carat:cutPremium     932.5      522.5   1.785   0.0745 .
carat:cutVery Good  1386.3      540.0   2.567   0.0104 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1529 on 1390 degrees of freedom
Multiple R-squared:  0.6936,    Adjusted R-squared:  0.6916
F-statistic: 349.7 on 9 and 1390 DF,  p-value: < 2.2e-16
```

#Interaction only model

model_interaction2 <- lm(price ~ carat : cut, data = data_clean)

summary(model_interaction2)

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

Residuals:
    Min      1Q  Median      3Q     Max
-3992.6  -845.3  -124.4   273.4  5842.0

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       -1029.04      89.56  -11.49   <2e-16 ***
carat:cutFair      3555.13     189.66   18.75   <2e-16 ***
carat:cutGood      4628.35     144.78   31.97   <2e-16 ***
carat:cutIdeal     5737.07     115.91   49.50   <2e-16 ***
carat:cutPremium   4963.41     108.42   45.78   <2e-16 ***
carat:cutVery Good 5268.31     132.48   39.77   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1529 on 1394 degrees of freedom
Multiple R-squared:  0.6926,    Adjusted R-squared:  0.6915
F-statistic: 628.1 on 5 and 1394 DF,  p-value: < 2.2e-16
```

Multiple regression model

model1 <- lm(price ~ carat + depth + table + cut + color + clarity, data = data_clean)

Model summary

summary(model1)

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

Residuals:
    Min      1Q  Median      3Q     Max
-4298.1  -594.8  -172.4   426.7  4422.2

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -6715.82    2325.16  -2.888 0.003933 **
carat         6833.73      74.90  91.235  < 2e-16 ***
depth           18.65      27.54   0.677 0.498275
table           21.25      17.36   1.224 0.221210
cutGood        126.66     164.55   0.770 0.441570
cutIdeal       581.96     167.75   3.469 0.000538 ***
cutPremium     394.31     162.51   2.426 0.015377 *
cutVery Good   301.98     163.89   1.843 0.065608 .
colorE        -174.82     115.57  -1.513 0.130580
colorF          47.78     115.06   0.415 0.677999
colorG        -248.62     108.33  -2.295 0.021882 *
colorH        -669.25     117.07  -5.717 1.33e-08 ***
colorI        -904.87     126.76  -7.138 1.52e-12 ***
colorJ       -1354.77     143.63  -9.432  < 2e-16 ***
clarityVS1    2917.84      70.56  41.353  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1028 on 1386 degrees of freedom
Multiple R-squared:  0.862,     Adjusted R-squared:  0.8606
F-statistic: 618.5 on 14 and 1386 DF,  p-value: < 2.2e-16
```

Multicollinearity check

vif_values <- vif(model1)

print(vif_values)

Filter to keep only significant levels of color and cut

filtered_data <- data_clean %>%

```
  filter(color %in% c("G", "H", "I", "J"),
         cut %in% c("Ideal", "Premium"))

# Drop unused levels
filtered_data$color <- droplevels(filtered_data$color)
filtered_data$cut <- droplevels(filtered_data$cut)

# Run the refined model
model_final <- lm(price ~ carat + cut + color + clarity, data = filtered_data)
summary(model_final)
```

```
Call:
lm(formula = price ~ carat + cut + color + clarity, data = filtered_data)

Residuals:
    Min      1Q  Median      3Q     Max
-4248.2  -565.1   -64.6   550.0  3312.6

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -4327.98     161.15 -26.857  < 2e-16 ***
carat        6959.82     104.52  66.591  < 2e-16 ***
cutPremium   -205.41      85.02  -2.416 0.016032 *
colorH       -353.80     100.90  -3.507 0.000492 ***
colorI       -617.86     117.43  -5.261 2.07e-07 ***
colorJ      -1247.01     146.41  -8.517  < 2e-16 ***
clarityVS1   3198.85     105.05  30.450  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 958.9 on 534 degrees of freedom
Multiple R-squared:  0.8931,    Adjusted R-squared:  0.8919
F-statistic: 743.4 on 6 and 534 DF,  p-value: < 2.2e-16
```

```
# Multicollinearity check
vif_final<- vif(model_final)
print(vif_final)
```

```
> vif_final <- vif(model_final)
> print(vif_final)
            GVIF Df GVIF^(1/(2*Df))
carat    1.366516  1        1.168981
cut      1.053069  1        1.026192
color    1.083698  3        1.013487
clarity 1.399088  1        1.182830
```

```
# Residual analysis
plot(model_final$residuals, main = "Residual Plot", ylab = "Residuals", xlab = "Fitted Values")
qqnorm(model_final$residuals)
qqline(model_final$residuals, col = "red")
```