STAT 306 2024W1 — Assignment 2

Total marks: 37
Due date: Monday, December 2 at 11:59pm

Instructions

- To work on this assignment, edit this `.qmd` file directly.
  Fill in the code sections that are tagged with `### START SOLUTION` and `### END SOLUTION`.
  Fill in the text sections that are tagged with `<!-- START SOLUTION -->` and `<!-- END SOLUTION -->` .
- To submit this assignment, render this `.qmd` as a PDF and submit the PDF file to Gradescope.

Exercise 1: Price of diamonds

Total marks: 13

The price of a diamond depends on several variables:

- Its *caratage* (`Car`), or weight, with one carat being equivalent to 0.2 g.
- Its *colour purity* (`Col`), with the top colour purity being `D` and decreasing in grade with `E`, `F`, `G`, ..., etc.
- Its *clarity* (`Cla`), which categorizes its *inclusion* (non-diamond materials trapped in the diamond that are visible under a jeweler's magnifying glass) into one of five categories (in order of descending grade, with `VS2` as the lowest grade):

  1. `IF`: internally flawless
  2. `VVS1` and `VVS2`: very very slightly imperfect
  3. `VS1` and `VS2`: very slightly imperfect

- The *certification body* (`Cer`) that verified the quality of the diamond:
  - `GIA`: Gemological Institute of America
  - `IGI`: International Gemological Institute
  - `HRD`: Hoge Raad Voor Diamant

Other factors that may affect the price include its shape (with "round" being the most popular) and cut (how the trajectory of light falls within the diamond).

The data file `Diamonds.txt` contains the price (`P`, in dollars) of 308 round diamonds, along with their caratage, colour purity rating, clarity rating, and certification body. Our goal will be to produce a model for predicting the price of the diamond based on its features.

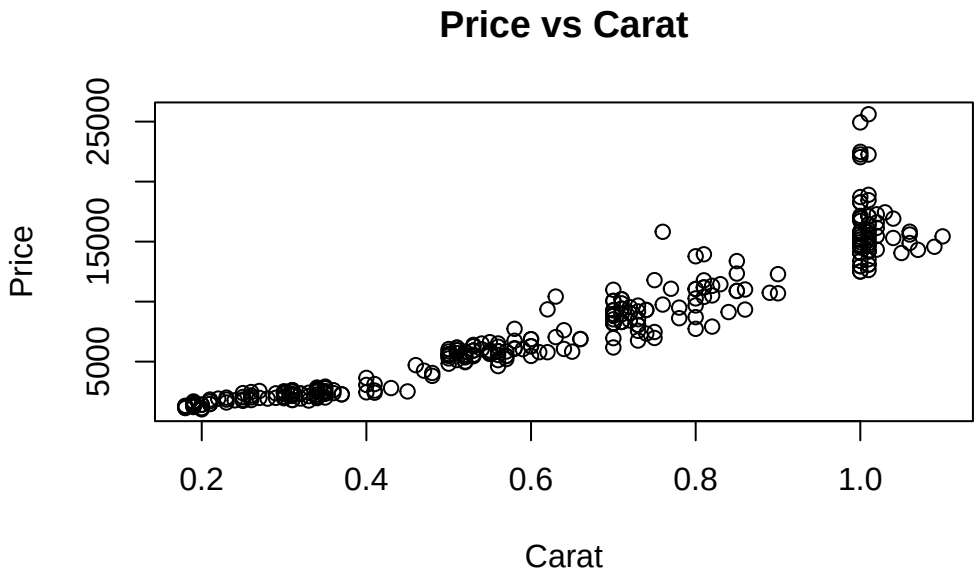
A. Explore the relationship between price and caratage

[2 marks]

Download the data file `Diamonds.txt` from Canvas. Create a plot that shows the relationship between diamond price and caratage.

Make sure that the axes are clearly labeled.

```
# Load the data
diamonds = read.table("Diamonds.txt", header=T)

### START SOLUTION
### --------------

# Create the scatterplot
# TODO

plot(diamonds$Car,diamonds$P,xlab = "Carat",
     ylab = "Price",
     main = "Price vs Carat")
```




```
### --------------
### END SOLUTION
```

Consider a linear model where price is the response and caratage is a covariate. Based on the plot, identify one assumption of the model that is unlikely to hold and explain why.

Solution

The linearity assumption may be unlikely to hold. The relationship between price and caratage may not be linear.

B. Explore the relationship between price and the categorical covariates

[3 marks]

Create the following plots to better understand the relationships between diamond price and the categorical covariates:

1. A plot that shows the relationship between price and colour purity.
2. A plot that shows the relationship between price and clarity.
3. A plot that shows the relationship between price and certification body.

Make sure that the axes are clearly labeled.

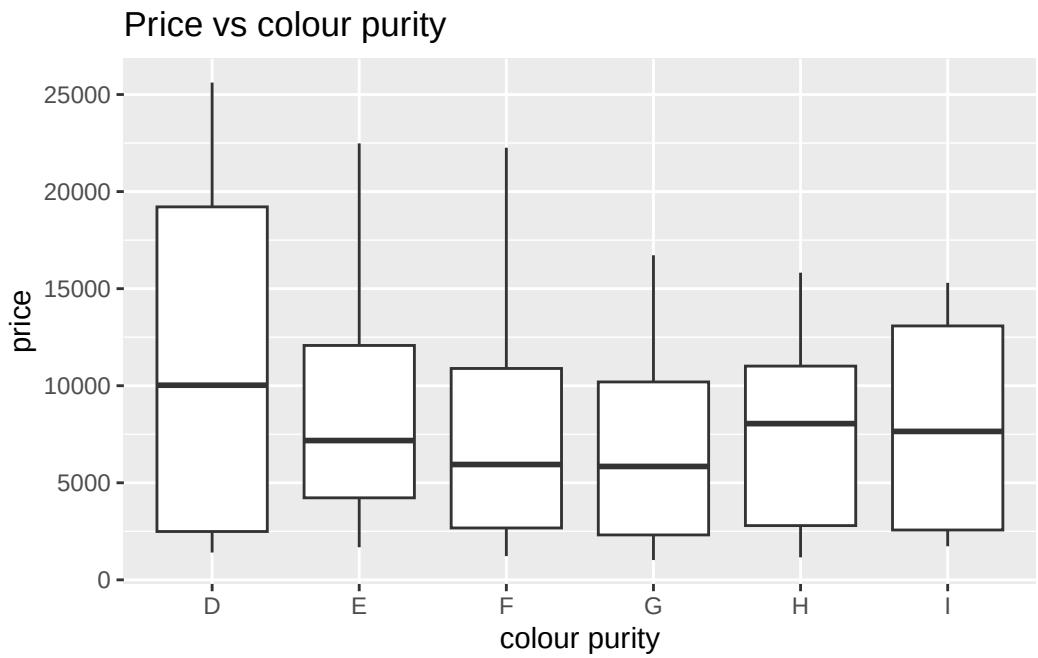
```
### START SOLUTION
### --------------
library(ggplot2)
```

```
Attaching package: 'ggplot2'
```

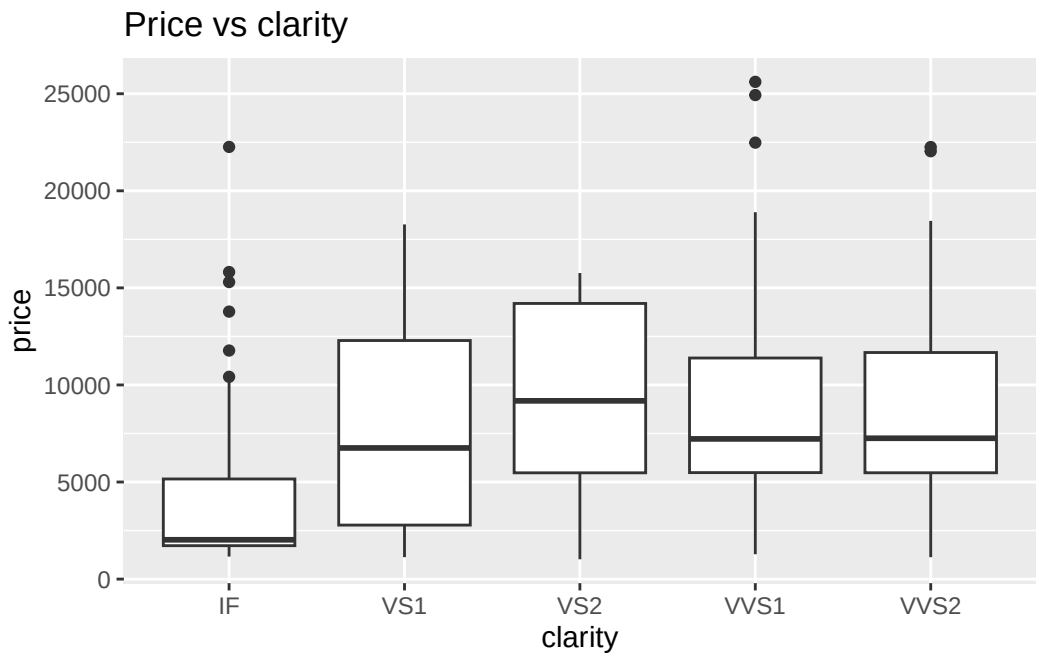
```
The following object is masked _by_ '.GlobalEnv':
```

```
    diamonds
```

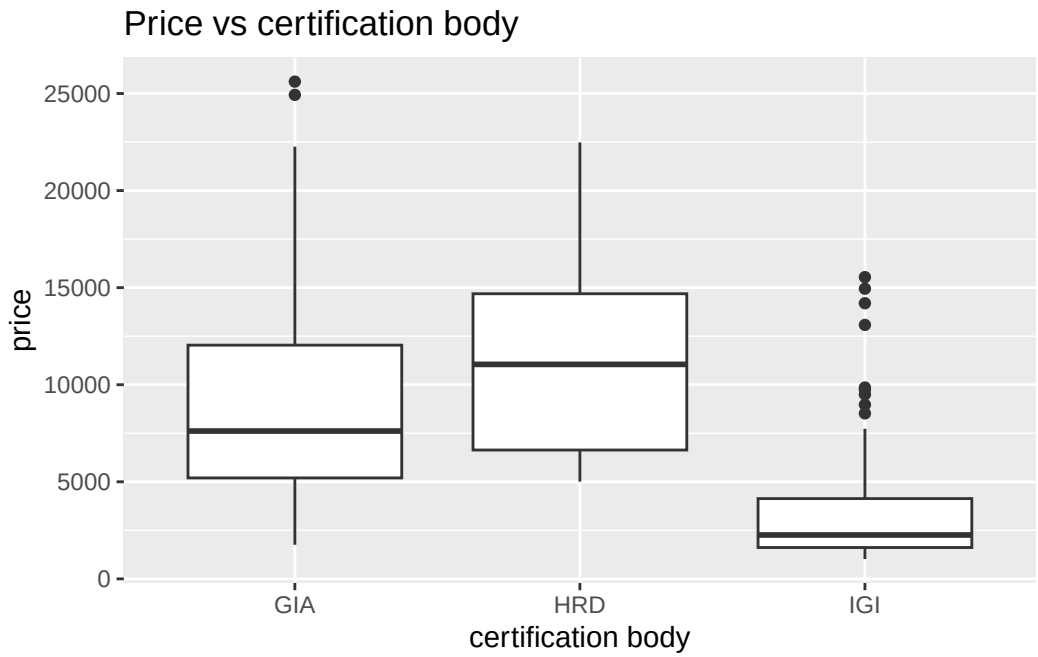
```
# Create the plots
# TODO
ggplot(diamonds, aes(x=Col, y=P)) +
  geom_boxplot() +
  xlab("colour purity")+
  ylab("price")+
  ggtitle("Price vs colour purity")
```



```
ggplot(diamonds, aes(x=Cla, y=P)) +
  geom_boxplot() +
  xlab("clarity")+
  ylab("price")+
  ggtitle("Price vs clarity")
```




```
ggplot(diamonds, aes(x=Cer, y=P)) +
  geom_boxplot() +
  xlab("certification body")+
  ylab("price")+
  ggtitle("Price vs certification body")
```



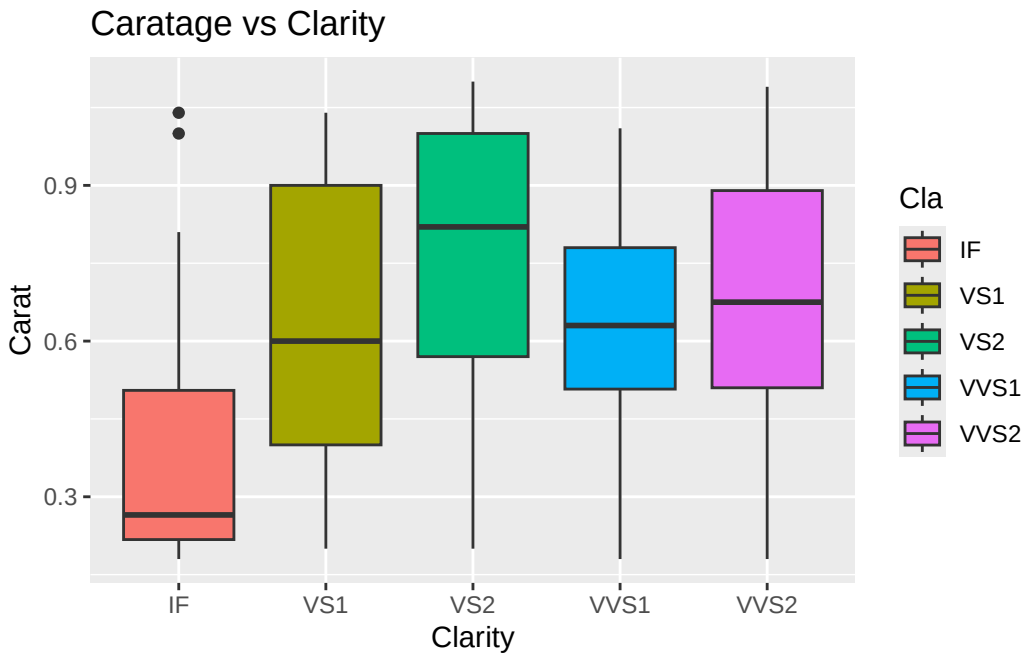

```
### --------------
### END SOLUTION
```

Observe that in the plot of price versus clarity, the prices of internally flawless diamonds tend to be lower than that of other clarity grades. This seems to go against intuition. Further explore the data to explain why this is the case.

Solution

From our price vs caratage plot, we observe a positive relationship, as the carat increases the price increases. Let us examine the relationship between clarity and carat.

```
ggplot(diamonds, aes(x = Cla, y = Car, fill = Cla)) +
  geom_boxplot() +
  labs(title = "Caratage vs Clarity", x = "Clarity", y = "Carat")
```



We observe that internally flawless diamonds tend to have lower caratage, which could explain the lower price for it compared to other clarity grades.

C. Fit the linear model

[2 marks]

The issue identified in the plot in part **A** may be alleviated by applying a log transformation to price.

Fit the linear model that has log price as the response and caratage, colour purity, clarity, and certification body as the covariates (without interactions). For colour purity and clarity, take the lowest grade as the baseline. For certification body, take GIA as the baseline.

```
### START SOLUTION
### --------------

# TODO

# Fit the linear model
diamonds$Col <- as.factor(diamonds$Col)
diamonds$Cla <- as.factor(diamonds$Cla)
diamonds$Cer <- as.factor(diamonds$Cer)
diamonds$Cla <- relevel(diamonds$Cla, ref = "VS2")
```

```r
diamonds$Col <- relevel(diamonds$Col, ref = "I")
diamonds$Cer <- relevel(diamonds$Cer, ref = "GIA")
reg1 = lm(log(diamonds$P)~diamonds$Car+diamonds$Col+diamonds$Cla+diamonds$Cer)
summary(reg1)
```

```
Call:
lm(formula = log(diamonds$P) ~ diamonds$Car + diamonds$Col +
    diamonds$Cla + diamonds$Cer)

Residuals:
     Min       1Q   Median       3Q      Max
-0.31224 -0.11528  0.01618  0.10833  0.36338

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       6.556125   0.041874 156.568  < 2e-16 ***
diamonds$Car      2.854995   0.036968  77.228  < 2e-16 ***
diamonds$ColD     0.416517   0.041383  10.065  < 2e-16 ***
diamonds$ColE     0.387041   0.030825  12.556  < 2e-16 ***
diamonds$ColF     0.310197   0.027480  11.288  < 2e-16 ***
diamonds$ColG     0.210190   0.028360   7.412 1.33e-12 ***
diamonds$ColH     0.128681   0.028524   4.511 9.31e-06 ***
diamonds$ClaIF    0.298514   0.033303   8.963  < 2e-16 ***
diamonds$ClaVS1   0.096618   0.024919   3.877  0.00013 ***
diamonds$ClaVVS1  0.297837   0.028103  10.598  < 2e-16 ***
diamonds$ClaVVS2  0.201921   0.025344   7.967 3.57e-14 ***
diamonds$CerHRD  -0.008854   0.020865  -0.424  0.67161
diamonds$CerIGI  -0.182706   0.024952  -7.322 2.33e-12 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1382 on 295 degrees of freedom
Multiple R-squared:  0.9723,    Adjusted R-squared:  0.9712
F-statistic: 863.6 on 12 and 295 DF,  p-value: < 2.2e-16
```

```r
### --------------
### END SOLUTION
```

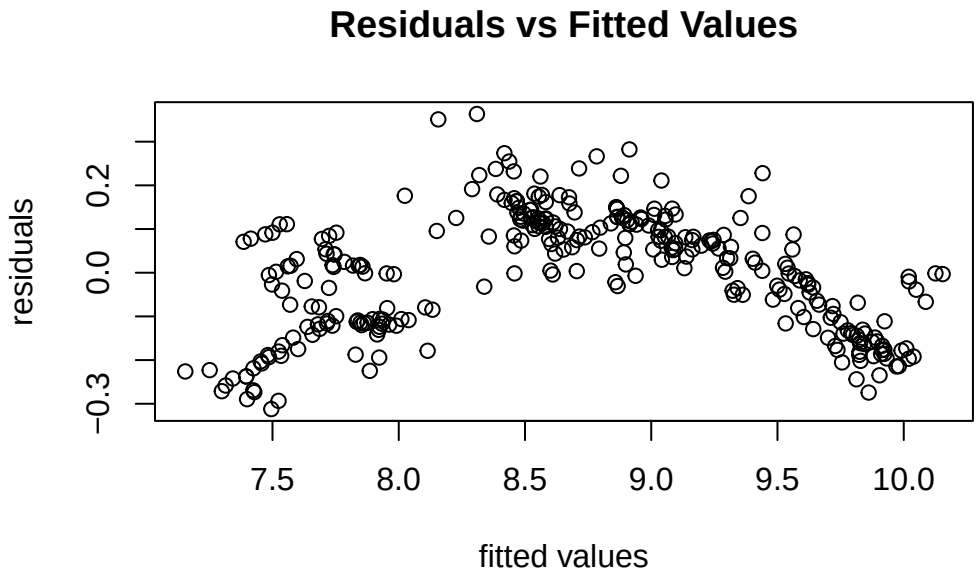
D. Examine the model fit

[2 marks]

Plot the residuals against the fitted values to examine the model fit.

Make sure that the axes are clearly labeled.

```
### START SOLUTION
### --------------

# Create the residual plot
# TODO
plot(reg1$fitted.values,reg1$residuals,xlab = "fitted values",ylab = 'residuals',title('Resi
```




```
### --------------
### END SOLUTION
```

Comment on the plot and what it says about the model fit.

Solution

The distribution does not appear to be randomly scattered. There appears to be a pattern, roughly like an upside parabola. This can suggest that the relationship isn't linear. The

residuals increases then decreases as the fitted values increase. This suggests a quadratic model could be needed.

E. Fit a quadratic model

[2 marks]

The issue identified in the plot in part **D** may be resolved by adding squared caratage to the model in part **C**.

Fit this new model and plot its residuals against its fitted values.

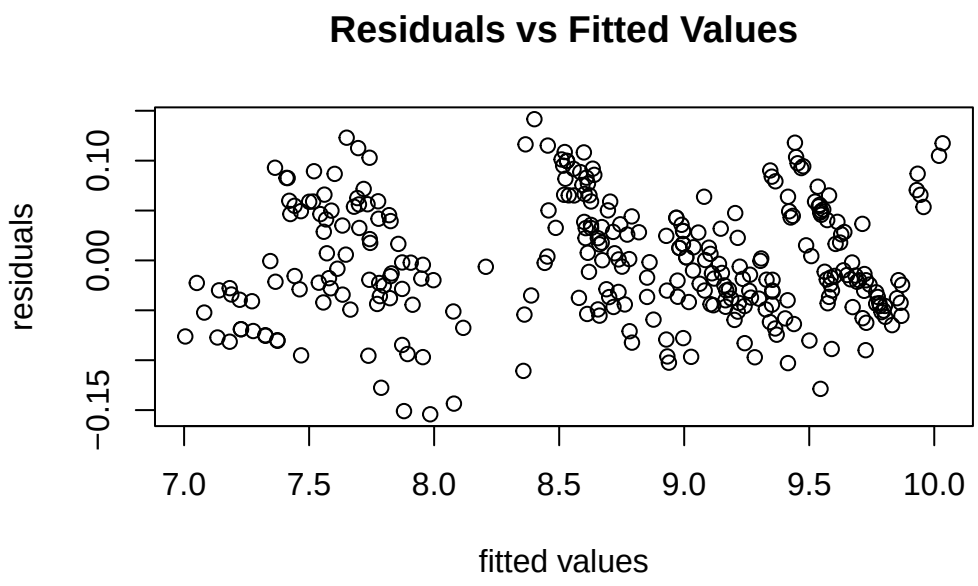
Make sure that the axes are clearly labeled.

```
### START SOLUTION
### --------------

# Fit the new model
reg2 = lm(log(diamonds$P)~diamonds$Car+diamonds$Col+diamonds$Cla+diamonds$Cer+I((diamonds$Ca
summary(reg2)
```

```
Call:
lm(formula = log(diamonds$P) ~ diamonds$Car + diamonds$Col +
    diamonds$Cla + diamonds$Cer + I((diamonds$Car)^2))

Residuals:
      Min        1Q    Median        3Q       Max
-0.154214 -0.041108 -0.009123  0.045467  0.141564

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)         5.782580   0.027881 207.404  < 2e-16 ***
diamonds$Car        5.670644   0.079288  71.520  < 2e-16 ***
diamonds$ColD       0.442567   0.017742  24.944  < 2e-16 ***
diamonds$ColE       0.363353   0.013221  27.483  < 2e-16 ***
diamonds$ColF       0.286613   0.011790  24.310  < 2e-16 ***
diamonds$ColG       0.197556   0.012154  16.254  < 2e-16 ***
diamonds$ColH       0.103508   0.012239   8.457 1.30e-15 ***
diamonds$ClaIF      0.320156   0.014279  22.421  < 2e-16 ***
diamonds$ClaVS1     0.075721   0.010691   7.083 1.04e-11 ***
diamonds$ClaVVS1    0.226175   0.012200  18.539  < 2e-16 ***
diamonds$ClaVVS2    0.143478   0.010976  13.072  < 2e-16 ***
```

```
diamonds$CerHRD     -0.006221   0.008938  -0.696   0.4870
diamonds$CerIGI     -0.025405   0.011537  -2.202   0.0284 *
I((diamonds$Car)^2) -2.102956   0.058025 -36.242  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.0592 on 294 degrees of freedom
Multiple R-squared:  0.9949,    Adjusted R-squared:  0.9947
F-statistic:  4445 on 13 and 294 DF,  p-value: < 2.2e-16
```

```
# Create the residual plot
# TODO
plot(reg2$fitted.values,reg2$residuals,xlab = "fitted values",ylab = 'residuals',title('Resi
```




```
### --------------
### END SOLUTION
```

Comment on the plot and what it says about the model fit.

Solution

The distribution appears to be randomly scattered. There doesn't appear to be any clear pattern. This suggests that the model covers the relationship between the covariates and the

response variable well. Most covariates p values remain significant, suggesting multicollinearity may not be an issue.

F. Compare two models

[2 marks]

Fit the same model from part **E** but without certification body as a covariate.

```
### START SOLUTION
### --------------

# Fit the model without certification body
reg3 = lm(log(diamonds$P)~diamonds$Car+diamonds$Col+diamonds$Cla+I((diamonds$Car)^2))
summary(reg3)
```

```
Call:
lm(formula = log(diamonds$P) ~ diamonds$Car + diamonds$Col +
    diamonds$Cla + I((diamonds$Car)^2))

Residuals:
     Min       1Q   Median       3Q      Max
-0.15028 -0.04057 -0.00799  0.04519  0.14464

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          5.75460    0.02439 235.947  < 2e-16 ***
diamonds$Car         5.74407    0.07087  81.053  < 2e-16 ***
diamonds$ColD        0.44279    0.01783  24.836  < 2e-16 ***
diamonds$ColE        0.36333    0.01326  27.393  < 2e-16 ***
diamonds$ColF        0.28565    0.01182  24.169  < 2e-16 ***
diamonds$ColG        0.19673    0.01217  16.165  < 2e-16 ***
diamonds$ColH        0.10435    0.01221   8.547 6.80e-16 ***
diamonds$ClaIF       0.30979    0.01355  22.855  < 2e-16 ***
diamonds$ClaVS1      0.07626    0.01073   7.105 9.01e-12 ***
diamonds$ClaVVS1     0.21994    0.01181  18.623  < 2e-16 ***
diamonds$ClaVVS2     0.13804    0.01074  12.852  < 2e-16 ***
I((diamonds$Car)^2) -2.15003    0.05385 -39.923  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.05949 on 296 degrees of freedom
Multiple R-squared:  0.9949,    Adjusted R-squared:  0.9947
F-statistic:  5202 on 11 and 296 DF,  p-value: < 2.2e-16
```

```
### --------------
### END SOLUTION
```

Between this fitted model and the fitted model from part **E**, explain which model you would choose for our purposes and why.

Solution

In the fitted model in part E there are some insignificant terms(the Cer terms). CerHRD isn't statistically significant and CerIGI is slightly statistically significant. So after removing certification body as a covariate, the model (in part F) then has all statistically significant terms. The R^2 and adjusted R^2 are about the same in both the models. So based on principle of parsimony, I would choose the model in part F as it has one fewer covariate.

Exercise 2: The Functional Dexterity Test

Total marks: 13

The Functional Dexterity Test (FDT) is an instrument for assessing the manual dexterity of an individual. The test involves the volunteer flipping sixteen pegs placed in holes on a wooden board as quickly as possible, using only one hand.

Gogola et al. (2013) used the FDT to explore how dexterity in children may depend on age, sex, and handedness. In total, 175 children were included in the study, each performing the FDT with each hand in an order assigned at random. The data from the second attempt by 174 of the children can be found in the file `Dexterity.txt` , which includes the following variables:

- `Time`: time taken to flip all sixteen pegs (in seconds)
- `Age`: age (in years)
- `Dominant`: whether the dominant hand was used (`1` if yes, `0` if no)
- `Sex`: biological sex (`1` for male, `0` for female)
- `HD`: preferred hand (`R` for right, `L` for left)
- `HU`: hand used on trial recorded (`R` for right, `L` for left)

Read the following article before attempting the questions below (free to access through the online UBC library):

Gloria R. Gogola, Paul F. Velleman, Shuai Xu, Adrianne M. Morse, Barbara Lacy, Dorit Aaron (2013): Hand Dexterity in Children: Administration and Normative Values of the

Functional Dexterity Test. *The Journal of Hand Surgery*, 38(12), 2426-2431, ISSN 0363-5023, https://doi.org/10.1016/j.jhsa.2013.08.123

A. Consider the model assumptions

[1 mark]

Only one observation is recorded per child in our data. If both observations per child were included in an analysis, which assumption(s), if any, of the linear model would likely be invalid? Explain your answer.

Solution

If both observations were included, then they wouldn't be independent. If there are multiple observations from the same child, the observations in the data won't be independent.

B. Explore the relationship between FDT speed and the variables

[2 marks]

As suggested by Gogola et al., take "speed" (in pegs/second) as the response variable.

Create the following plots:

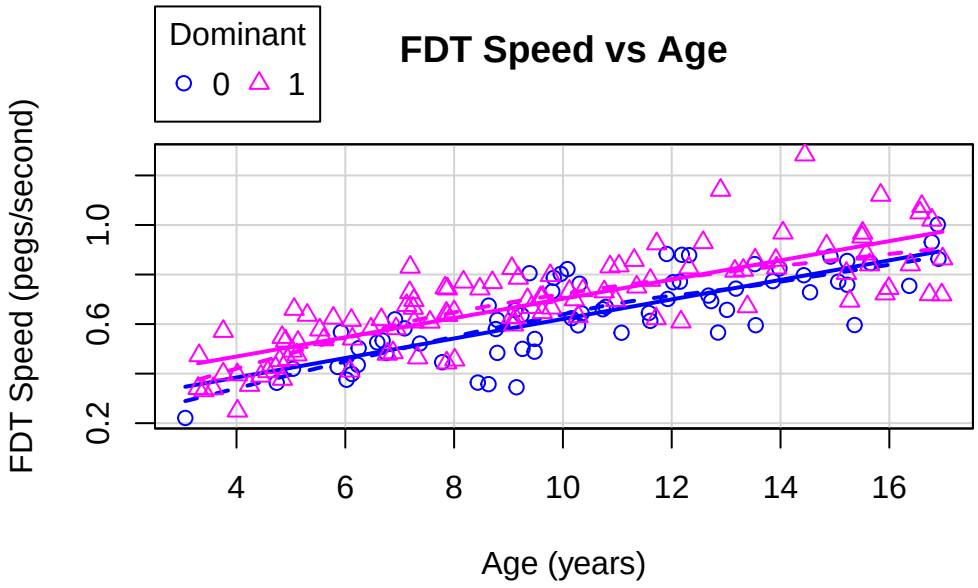
1. A plot that explores how FDT speed depends on age and whether the dominant hand was used.
2. A plot that explores how FDT speed depends on age and sex.

Make sure that the axes are clearly labeled. You may find the `scatterplot()` *function in the* `car` *package useful.*

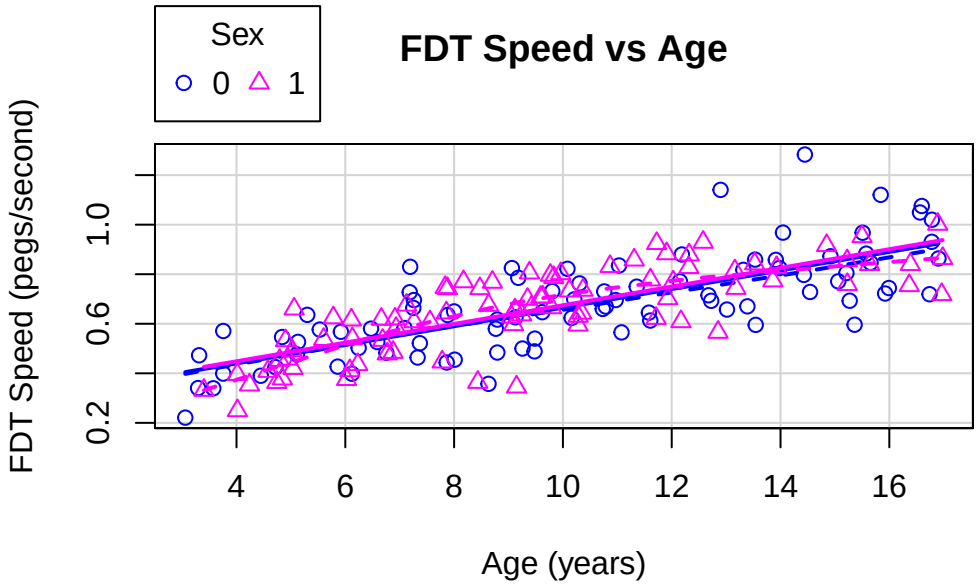
```
# Read the data
dex = read.table("Dexterity.txt", header=TRUE)

### START SOLUTION
### --------------

# Create the speed variable
# TODO
speed = 16/dex$Time
dex$Speed = speed

# Create the plots
# TODO
library(car)
```

```
Loading required package: carData
```

```
scatterplot(Speed ~ Age | Dominant,
            data = dex,
            xlab = "Age (years)",
            ylab = "FDT Speed (pegs/second)",
            main = "FDT Speed vs Age" )
```




```
scatterplot(Speed ~ Age | Sex,
            data = dex,
            xlab = "Age (years)",
            ylab = "FDT Speed (pegs/second)",
            main = "FDT Speed vs Age")
```




```
### --------------
### END SOLUTION
```

Comment on what the plots tell us about how manual dexterity in children varies by age, sex, and whether the dominant hand was used.

Solution

Looking at the first plot the lines are parallel, with dominant hand higher than non- dominant hand. The rate at which FDT speed increases with age is same whether the child uses their dominant hand or not. However, those where the dominant hand was used the speed stays higher, they start off with a higher speed. Looking at the second plot, we observe the line to be the same for both the sexes. This suggests that the sex doesn't affect the rate of change of FDT speed as age increases. FDT speed improves as age increases for both sexes.

C. Fit the linear model via backward selection

[3 marks]

Fit the model that includes all covariates and also the interaction between age and whether the dominant hand was used. We'll refer to this model as the full model (*though it would be possible to fit models with more parameters by including more interaction terms*).

Starting with the full model, perform backward selection using a critical value of $\alpha_c = 0.05$. List the term that is dropped in each iteration and the p-value of their corresponding test.

You do not need to show the summaries of the intermediate models. Save all the models that you fit as you will need them for the next question.

```
# Create a list of models

models = list()

### START SOLUTION
### --------------

# Perform backward selection

# Fit the full model
models[[1]] = lm(Speed ~ Age*Dominant+Sex+HD+HU,data = dex)


# Fit the next model in the backward selection procedure
models[[2]] = lm(Speed ~ Age*Dominant+HD+HU,data = dex)

# Continue...
# TODO
models[[3]] = lm(Speed ~ Age+Dominant+HD+HU,data = dex)

models[[4]] = lm(Speed ~ Age+Dominant+HU,data = dex)

models[[5]] = lm(Speed ~ Age+Dominant,data = dex)

# Print the final model
summary(models[[5]])
```

```
Call:
lm(formula = Speed ~ Age + Dominant, data = dex)

Residuals:
     Min       1Q   Median       3Q      Max
-0.25547 -0.07124  0.00343  0.06926  0.40760

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
```

```
(Intercept) 0.230933   0.027270   8.468 1.09e-14 ***
Age         0.038997   0.002241  17.400  < 2e-16 ***
Dominant    0.081137   0.017431   4.655 6.48e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1112 on 171 degrees of freedom
Multiple R-squared:  0.6424,    Adjusted R-squared:  0.6382
F-statistic: 153.6 on 2 and 171 DF,  p-value: < 2.2e-16
```

```
### --------------
### END SOLUTION
```

Solution

The terms that are dropped in the backward selection (in order of being dropped) and their corresponding p-values are:

Step	Variable	p-value
1	Sex	0.992
2	Age:Dominant	0.906
3	HD	0.423
4	HU	0.3825

Explain whether the model found using backward selection agrees with your findings made from the plots in part **B**.

Solution

Yes it does. From part B we observed both sex and whether using the dominant hand doesn't seem to affect the rate of change of FDT speed with age. However, using those using their dominant hand start off with a higher speed than those who do not. So, using dominant hand affects the average FDT speed. Hence, it matches with the model we obtained using backward selection, where only Age and Dominant (with their interaction) are the relevant covariates, whereas, sex was dropped in step 1 in backward selection.

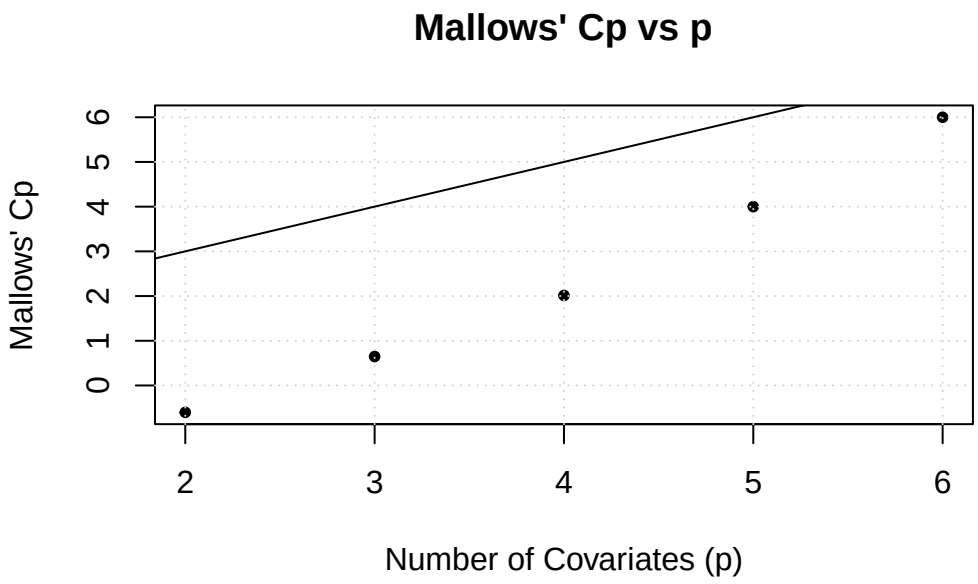
D. Measure model fit via Mallows' C_p

[3 marks]

For each of the models fitted in part **C** (including the full model), compute Mallows' C_p. Present your results in a C_p plot that shows C_p against number of covariates p in the model.

Make sure that the axes are clearly labeled.

```
### START SOLUTION
### --------------

# Compute Mallows' Cp
# TODO

cp <- numeric(5)
p <- numeric(5)
q <- length(coef(models[[1]]))
n <- nrow(dex)
msRes_q <- sum(residuals(models[[1]])^2) / (n - q - 1)

for (i in 1:5) {
  model <- models[[i]]
  rss_p <- sum(residuals(model)^2)
  p_curr <- length(coef(model)) - 1
  p[i] <- p_curr
 cp[i] <- (rss_p / msRes_q) - (n - 2 * (p_curr + 1))
}
# Create the Cp plot
# TODO
plot(p, cp, xlab = "Number of Covariates (p)", ylab = "Mallows' Cp",
     main = "Mallows' Cp vs p",pch=20)
grid()
abline(a = 1, b = 1)
```




```
### --------------
### END SOLUTION
```

Based on the C_p plot, which model do you think is best? Justify your choice.

Solution

Cp for the full model has the perfect Cp = p+1 but as Cp is calculated relative to the full model, it cannot be considered based on Cp. looking at the plot, model with 5 covariates appears to be the best model in terms of Cp as its the closest to p+1(6).

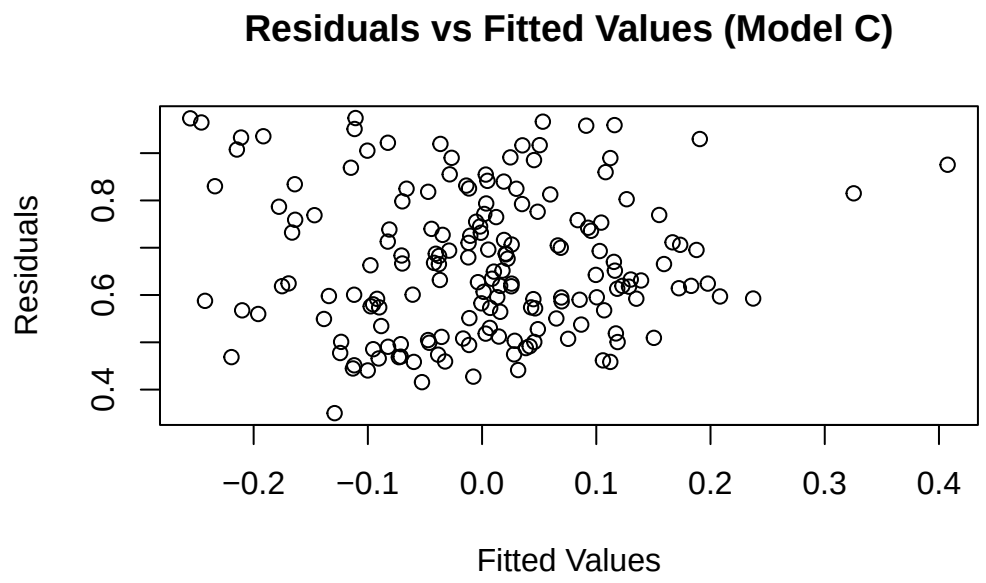
E. Examine the model fit

[2 marks]

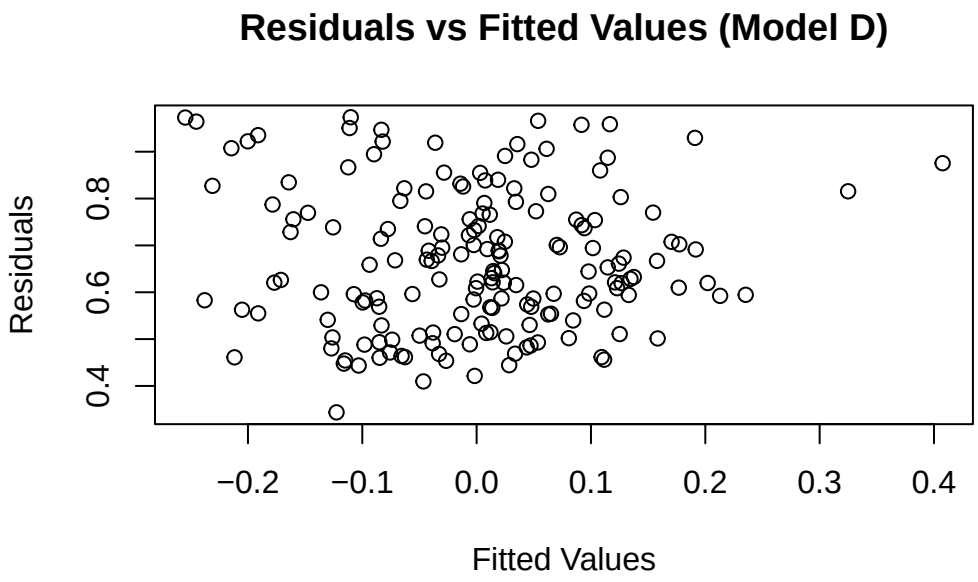
Create residual plots to assess the fit of the two models chosen in parts **C** and **D**. Comment on the plots.

Make sure that the axes are clearly labeled and that it is clear which model the plot corresponds to.

```
### START SOLUTION
### --------------

# Create the residual plots
# TODO
modelC = models[[5]]
modelD= lm(Speed ~ Age*Dominant+HD+HU,data = dex)

plot(modelC$residuals, modelC$fitted.values, main="Residuals vs Fitted Values (Model C)",
     xlab="Fitted Values", ylab="Residuals")
```




```
plot(modelD$residuals, modelD$fitted.values, main="Residuals vs Fitted Values (Model D)",
     xlab="Fitted Values", ylab="Residuals")
```




```
### --------------
### END SOLUTION
```

Solution

The distribution appears to be randomly scattered. There doesn't appear to be any clear pattern. This suggests that both model covers the relationship between the covariates and the response variable well.

F. Compare with using FDT completion time as the response

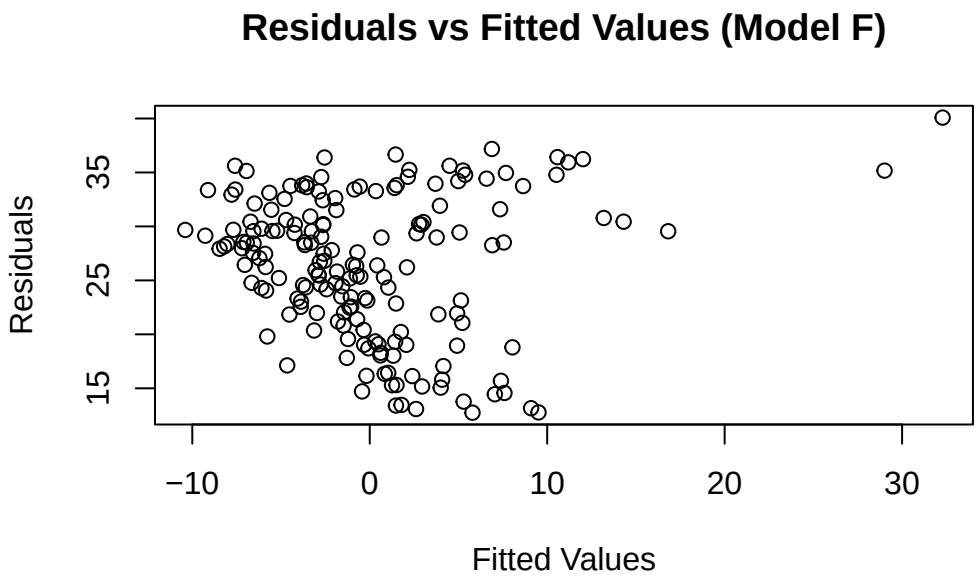
[2 marks]

Gogola et al. (2013) discuss problems that could arise with using FDT completion time as the response variable in the regression analysis for these data.

Fit the model that has the same covariates found in part **C** but with FDT completion time as the response. Create a residual plot for this new model.

```
### START SOLUTION
### --------------

# Fit the model with completion time as the response
```

```r
mod6 = lm(Time ~ Age+Dominant,data = dex)

# Create the residual plot
# TODO
plot(mod6$residuals,mod6$fitted.values, main="Residuals vs Fitted Values (Model F)",
     xlab="Fitted Values", ylab="Residuals")
```




```r
### --------------
### END SOLUTION
```

Do you see any of the issues raised by the authors? If so, which one(s)?

Solution

The author didn't use time as the response variable as it tend to violate the linearity and equal variance assumption for linear regression. Looking at the residuals vs fitted plot we observe that the plot looks funnel shaped, suggesting heteroscedasticity, violating the equal variance assumption.

Exercise 3: Recommended versus Amazon price of Lego sets

Total marks: 11

Lego is popular brand of toy products that involve sets of building bricks. The sets differ in terms of themes and number of pieces. Each set comes with a recommended price, which may differ from the price at which the set is sold for on Amazon. There is interest in a predictive model that determines when a set is sold on Amazon at a price higher than that of the recommended price.

The file `lego.csv` includes data for 337 Lego sets sold between 2018 and 2020. The variables recorded for each set include:

- `Price`: the recommended price (in USD)
- `Amazon_Price`: the price (in USD) at which the set is sold for on `Amazon.com`
- `Theme`: the theme of the set; sets from four themes are included in the data: `Friends`, `City`, `NINJAGO`, and `Star Wars`
- `Pieces`: the number of pieces that the set contains
- `Unique_Pieces`: the number of unique pieces that the set contains
- `Year`: the year that the set was released

A. Construct the response variable

[1 mark]

Construct the new variable `Amazon_Higher` that takes the value `TRUE` when the Amazon price is greater than the recommended price, and `FALSE` otherwise (i.e., when the Amazon price is lower or equal to the recommended price).

Count how many sets have a higher Amazon price and how many do not using the `table()` function.

```
# Read the data
lego = read.csv("lego.csv", header=TRUE)

### START SOLUTION
### --------------

lego$Amazon_Higher = lego$Amazon_Price > lego$Price

# Use table() to summarize the data
# TODO
table(lego$Amazon_Higher)
```

```
FALSE  TRUE
  163   174
```

```
### --------------
### END SOLUTION
```

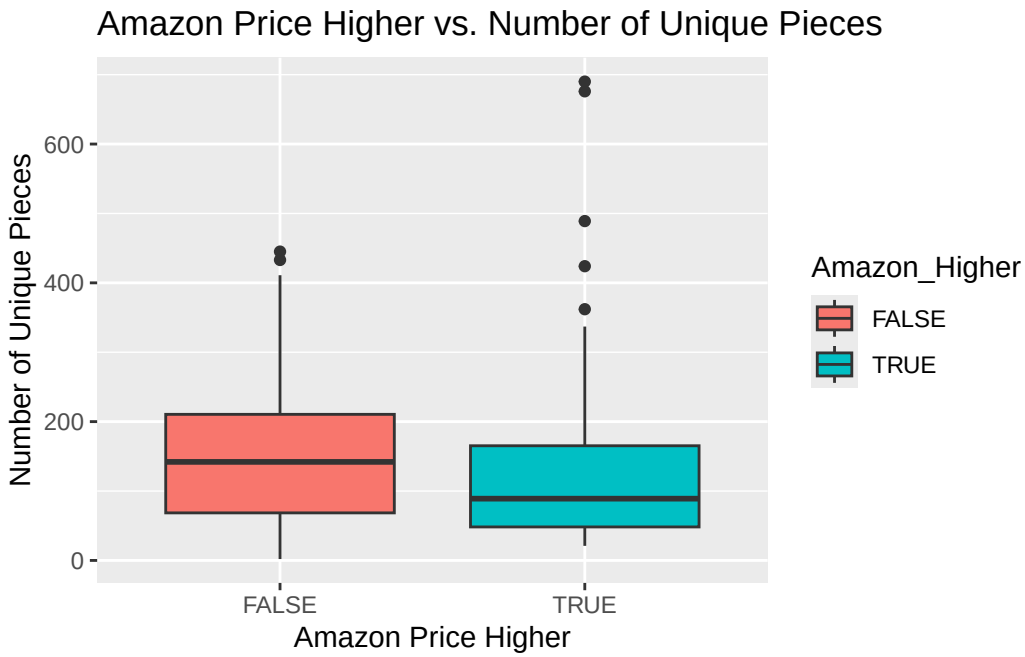
B. Explore the relationship between price and other variables

[2 marks]

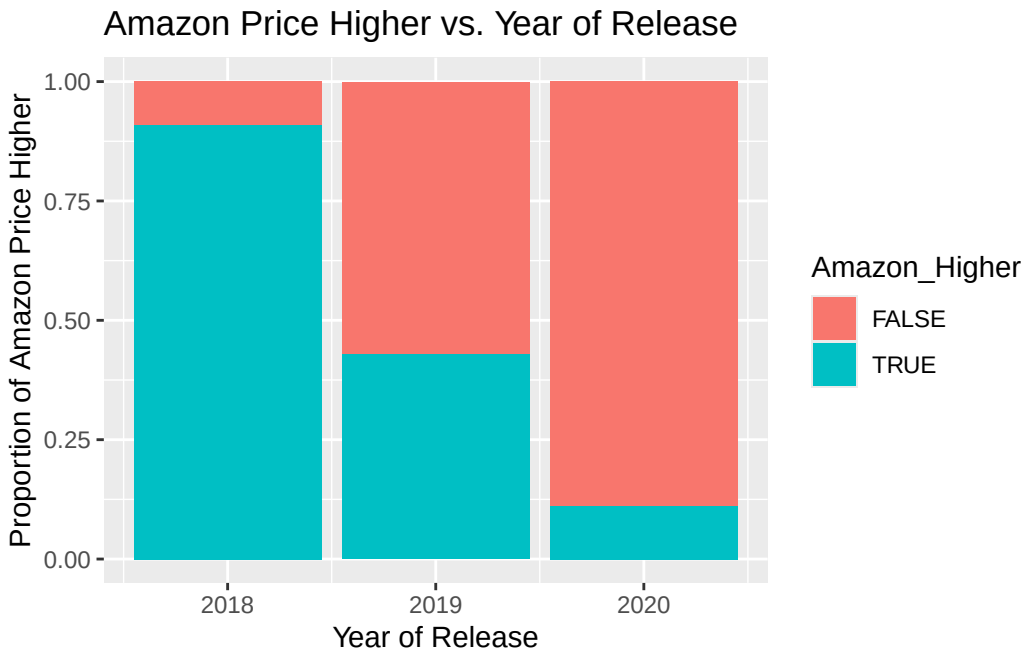
Create a plot or table that explores the relationship between `Amazon_Higher` and each of number of pieces, number of unique pieces, year, and set theme.

```
### START SOLUTION
### --------------

# Create the plots and tables
# TODO

ggplot(lego, aes(x = Amazon_Higher, y = Pieces, fill = Amazon_Higher)) +
  geom_boxplot() +
  labs(title = "Amazon Price Higher vs. Number of Pieces",
       x = "Amazon Price Higher", y = "Number of Pieces")
```

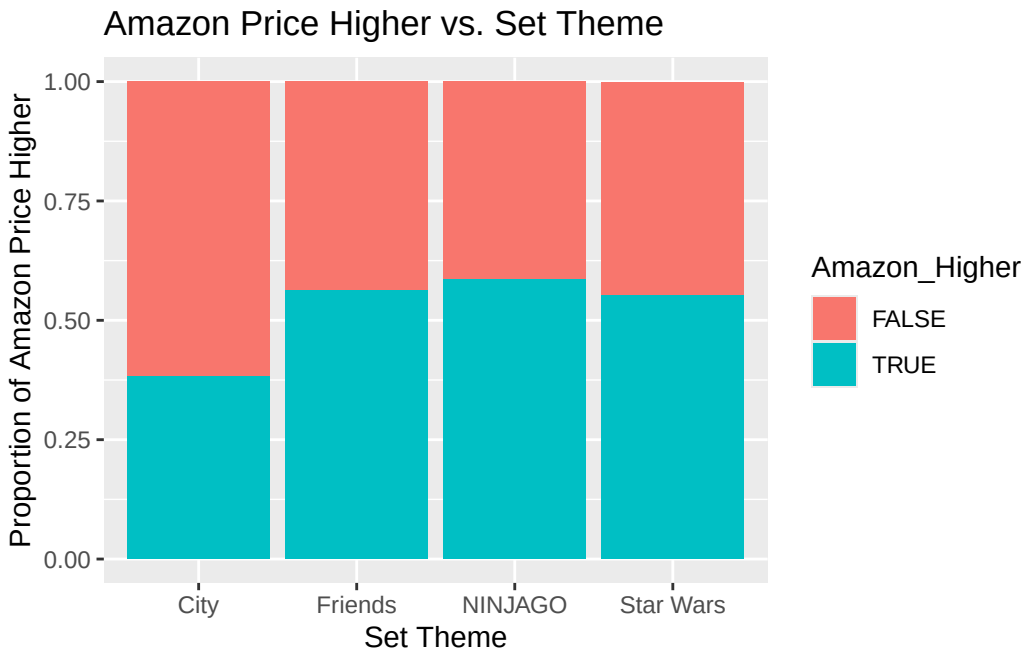

```
ggplot(lego, aes(x = Amazon_Higher, y = Unique_Pieces, fill = Amazon_Higher)) +
  geom_boxplot() +
  labs(title = "Amazon Price Higher vs. Number of Unique Pieces",
       x = "Amazon Price Higher", y = "Number of Unique Pieces")
```




```
ggplot(lego, aes(x = Year, fill = Amazon_Higher)) +
  geom_bar(position = "fill") +
  labs(title = "Amazon Price Higher vs. Year of Release",
       x = "Year of Release", y = "Proportion of Amazon Price Higher")
```




```
ggplot(lego, aes(x = Theme, fill = Amazon_Higher)) +
  geom_bar(position = "fill") +
  labs(title = "Amazon Price Higher vs. Set Theme",
       x = "Set Theme", y = "Proportion of Amazon Price Higher")
```




```
### --------------
### END SOLUTION
```

Comment on any notable observations you observe in your exploration.

Solution

Lego sets released in 2018 have the highest proportion of sets sold on Amazon at a price higher than the recommended price, followed by those released in 2019, with sets from 2020 having the lowest proportion of higher priced sets on Amazon. Newer sets could be more easily available than the older sets. Sets with fewer unique pieces appear to be higher priced on Amazon. Ninjago has the highest proportion of sets being sold at a higher price on Amazon and City the lowest compared to the other set themes.

C. Fit the logistic model

[2 marks]

Fit the logistic model for the response `Amazon_Higher` (where π is the probability that the Amazon price is greater than the recommended price) that includes all of the variables explored in part **B** as covariates and no interactions. For the theme variable, take `City` as the baseline level.

```
### START SOLUTION
### --------------


lego$Theme <- relevel(factor(lego$Theme), ref = "City")

lreg = glm(Amazon_Higher ~ Pieces + Unique_Pieces + Year + Theme, family = binomial, data = l

summary(lreg)
```

```
Call:
glm(formula = Amazon_Higher ~ Pieces + Unique_Pieces + Year +
    Theme, family = binomial, data = lego)

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)     5.465e+03  5.694e+02   9.598  < 2e-16 ***
Pieces          3.962e-04  5.649e-04   0.701  0.48312
Unique_Pieces  -8.249e-03  2.697e-03  -3.059  0.00222 **
Year           -2.707e+00  2.820e-01  -9.598  < 2e-16 ***
ThemeFriends    1.765e+00  4.529e-01   3.896 9.77e-05 ***
ThemeNINJAGO    1.387e+00  4.714e-01   2.942  0.00326 **
ThemeStar Wars  9.816e-01  4.352e-01   2.256  0.02409 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 466.82  on 336  degrees of freedom
Residual deviance: 267.39  on 330  degrees of freedom
AIC: 281.39

Number of Fisher Scoring iterations: 5
```

```
exp(lreg$coefficients)
```

```
   (Intercept)         Pieces  Unique_Pieces           Year   ThemeFriends
           Inf      1.0003963      0.9917846      0.0667336      5.8395720
  ThemeNINJAGO ThemeStar Wars
     4.0032567      2.6686903
```

```
1/0.9917846
```

```
[1] 1.008283
```

```
### --------------
### END SOLUTION
```

Interpret the fitted parameter for `Unique_Pieces` in the context of the data.

Solution

The odds of the set being sold on amazon at a lower price than the recommended price is 0.8283% higher while keeping the rest of the covariates constant.

D. Predict a probability

[1 mark]

Using the model fitted in part **C**, estimate the probability that the Amazon price is higher than the recommended price for a 2018 Star Wars set that contains 1000 pieces and 500 unique pieces.

```
### START SOLUTION
### --------------

# Predict the probability for the new set
# TODO

p = 1/(1+exp(-(5.465e+03+3.962e-04*1000-8.249e-03*500-2.707e+00*2018+9.816e-01)))
p
```

```
[1] 0.3839774
```

```
### --------------
### END SOLUTION
```

Solution

probability =

```
0.3839774
```

E. Select a model via train/test sets

[3 marks]

Intuitively, the number of pieces and the number of unique pieces that a set contains should be closely correlated. We can verify this:

```
# Compute the correlation between number of pieces and unique pieces
cor(lego$Pieces, lego$Unique_Pieces)
```

```
[1] 0.8591628
```

It's unclear if including both in a model would be useful for prediction.

Consider the logistic model in part **C** (including the covariates theme and year) but with either:

1. only number of pieces,
2. only number of unique pieces, or
3. both number of pieces and unique pieces.

We will choose the best model among these three models using a training set and a test set. We first randomly split our data so that the test set `lego_te` contains 100 data points and the training set `lego_tr` contains all other data points.

Change the seed below to the last four digits of your student number.

```
# Set the seed for reproducibility
set.seed(306)  # TODO: change the seed

# Split the data
n = nrow(lego)
test_inds = sample(n, size=100, replace=FALSE)
lego_te = lego[test_inds,]
lego_tr = lego[-test_inds,]
```

Next, fit the three models above on the training set.

```
### START SOLUTION
### --------------

# Fit models on the training set
# TODO
```

```
lreg1 = glm(Amazon_Higher ~ Pieces + Year + Theme, family = binomial, data = lego_tr)

lreg2 = glm(Amazon_Higher ~ Unique_Pieces + Year + Theme, family = binomial, data = lego_tr)

lreg3 = glm(Amazon_Higher ~ Pieces + Unique_Pieces + Year + Theme, family = binomial, data =

### --------------
### END SOLUTION
```

We now compare these three models on the test set. We'll use **prediction accuracy** (i.e., the proportion of test points that were correctly predicted) as the evaluation metric. To predict `Amazon_Higher`, we use the fitted model to predict the probability $\hat{\pi}$, and then predict `TRUE` if $\hat{\pi} > 0.5$ and `FALSE` otherwise.

```
### START SOLUTION
### --------------

# Compute the prediction accuracy on the test set for each model
# Hint: check the "type" argument of the predict.glm() function
# pa1 = mean( (predict(TODO)>0.5) == lego_te$Amazon_Higher )  # TODO
# pa2 = mean( (predict(TODO)>0.5) == lego_te$Amazon_Higher )  # TODO
# pa3 = mean( (predict(TODO)>0.5) == lego_te$Amazon_Higher )  # TODO
pa1 = mean( (predict(lreg1,lego_te,type="response")>0.5) == lego_te$Amazon_Higher )
pa2 = mean( (predict(lreg2,lego_te,type="response")>0.5) == lego_te$Amazon_Higher )
pa3 = mean( (predict(lreg3,lego_te,type="response")>0.5) == lego_te$Amazon_Higher )

### --------------
### END SOLUTION

paste("Model 1: ", round(pa1,3))
```

```
[1] "Model 1:  0.81"
```

```
paste("Model 2: ", round(pa2,3))
```

```
[1] "Model 2:  0.8"
```

```
paste("Model 3: ", round(pa3,3))
```

```
[1] "Model 3:  0.8"
```

Based on this training/test set procedure, explain which model you would pick and why.

Solution

I will pick model 1, which has only number of pieces,theme and, year as it has the highest prediction accuracy.

F. Consider a model with year as a categorical variable

[2 marks]

The `Year` variable in the data only includes three different values: `2018`, `2019`, and `2020`. Suppose that instead of encoding `Year` as a continuous covariate in our model from part **C**, we encoded this variable as a three-level categorical variable.

Describe one advantage and one disadvantage this model would have compared to the same model that takes `Year` as continuous.

Solution

If the relationship between the year(continuous) and the log odds of the Amazon price is greater than the recommended price is not linear than having year as a categorical variable will be better. In this data set years are discrete, so encoding it as a three-level categorical variable could be more suitable.

The disadvantage could be that we fail to capture the linear trends by years if they exist.