hw_2_sol_209

2025-10-10

This homework will focus heavily on regression with linear-linear, log-linear, and log-log all making an appearance and indicators as well. My advice is to use the class time to focus on either the log-log models (Q18-25) or indicators (Q26 onwards). The first section is a regular linear model like you explored in the last lab.

HINT: you will make four models (one for linear, one for log-linear, one for log-log and one that deals with indicators) and I suggest you save all four models. My naming scheme is usually along the lines of

mod_line <- LINEAR MODEL CODE

mod_log_line <- LOG-LINEAR MODEL CODE

mod_log_log <- LOG-LOG MODEL CODE

mod_ind <- INDICATOR MODEL CODE

The data set we will be using today is actually a super fun one an old mentor of mine collected on.....LEGOs! I use to love LEGOs growing up so now you get to play with LEGOs (...data set).

First, let's read in the data and look at the first few rows

```
legos <- read.csv('https://vinnys-classes.github.io/data/legos_data.csv')
legos$Year <- as.factor(legos$Year) #RUN THIS!!
head(legos)
```

```
##   Item_Number                   Set_Name Theme Pieces Year Pages Minifigures
## 1       10859          My First Ladybird Duplo      6 2018     9          NA
## 2       10860          My First Race Car Duplo      6 2018     9          NA
## 3       10862       My First Celebration Duplo     41 2018     9          NA
## 4       10864 Large Playground Brick Box Duplo     71 2018    32           2
## 5       10867             Farmers' Market Duplo     26 2018     9           3
## 6       10870                Farm Animals Duplo     16 2018     8          NA
##     Packaging Unique_Pieces  Size amazon_price age
## 1         Box             5 Large        16.00   1
## 2         Box             6 Large         9.45   1
## 3         Box            18 Large        39.89   1
## 4 Plastic box            49 Large        56.69   2
## 5         Box            18 Large        36.99   2
## 6         Box            13 Large         9.99   2
```

The variables are...

1) Item_Number: ID

2) Set_Name: The selling name of the lego set

3) Theme: One of three themes

4) Pieces: Number of pieces in the set
5) Year: Year the set was made as a nominal categorical variable (called factor's in R)
6) Pages: Number of pages in the booklet
7) Minifigures: Number of "people" sold with the set
8) Package: What type of packaging the set comes in
9) Unique Pieces: How many unique lego blocks are in the set
10) Size: The size of the blocks, with two levels
11) amazon_price: Price of the on Amazon as of a few years ago
12) age: the lowest age the company recommends for the data set

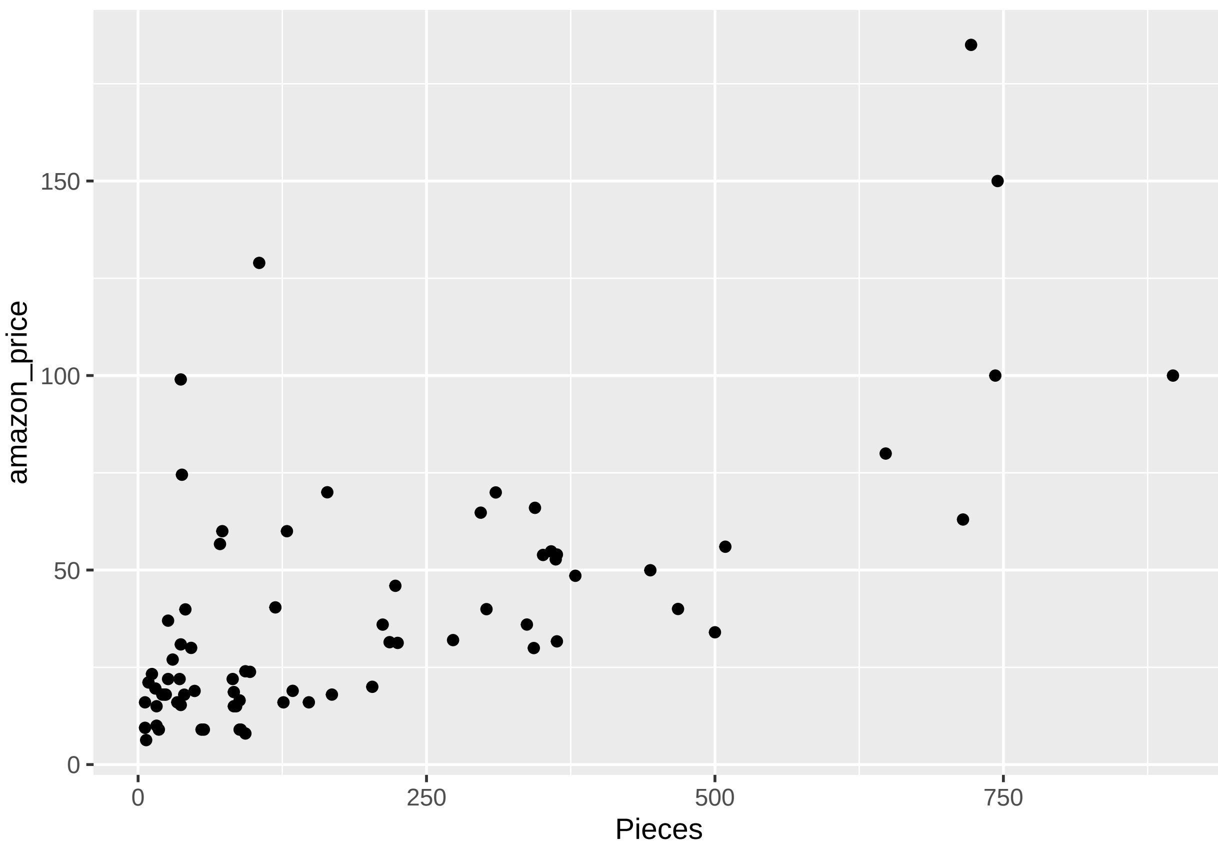
Linear Regression

Q1

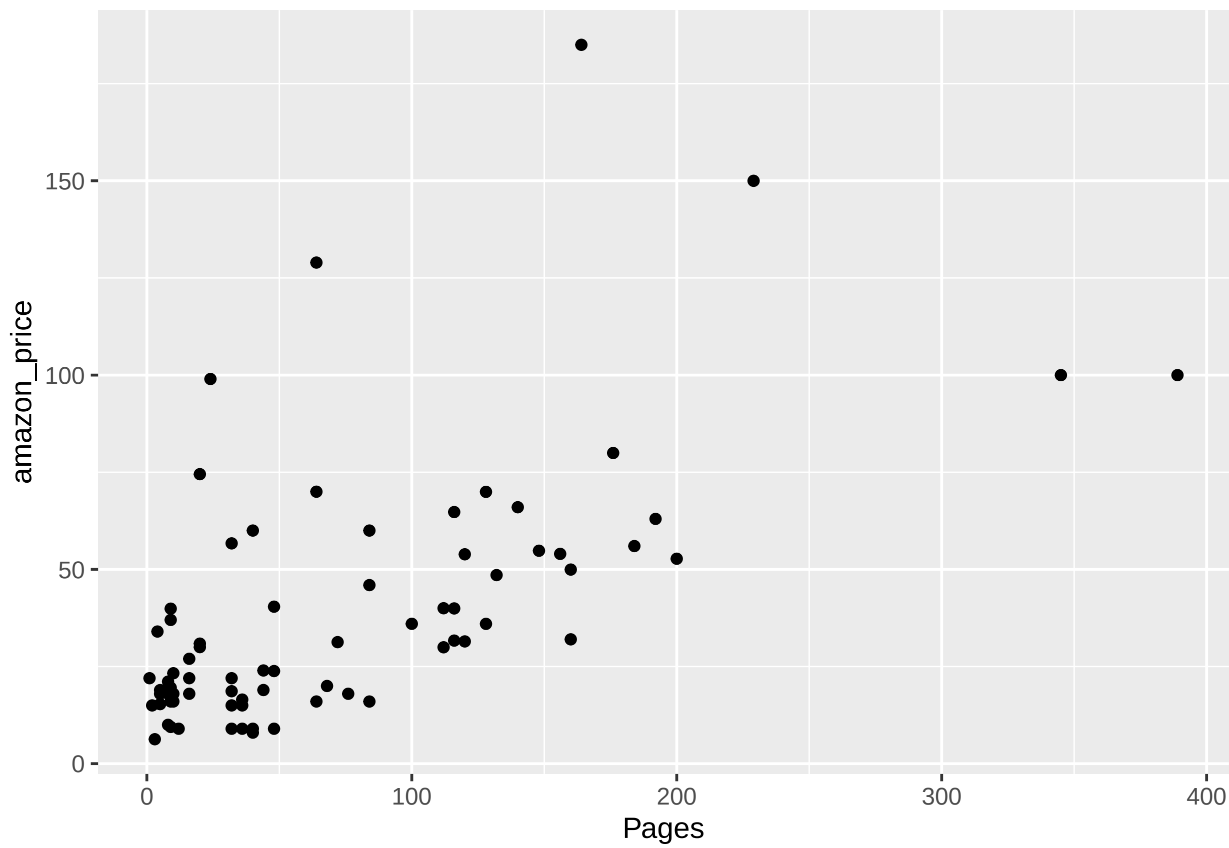
Please make make three scatterplots. All three should have amazon_price as the y-axis and the three x-axis should be the variables Pieces, Pages, and Minifigures.

```r
library(ggplot2)

ggplot(data = legos,
       aes(x = Pieces,
           y = amazon_price)) +
  geom_point()
```

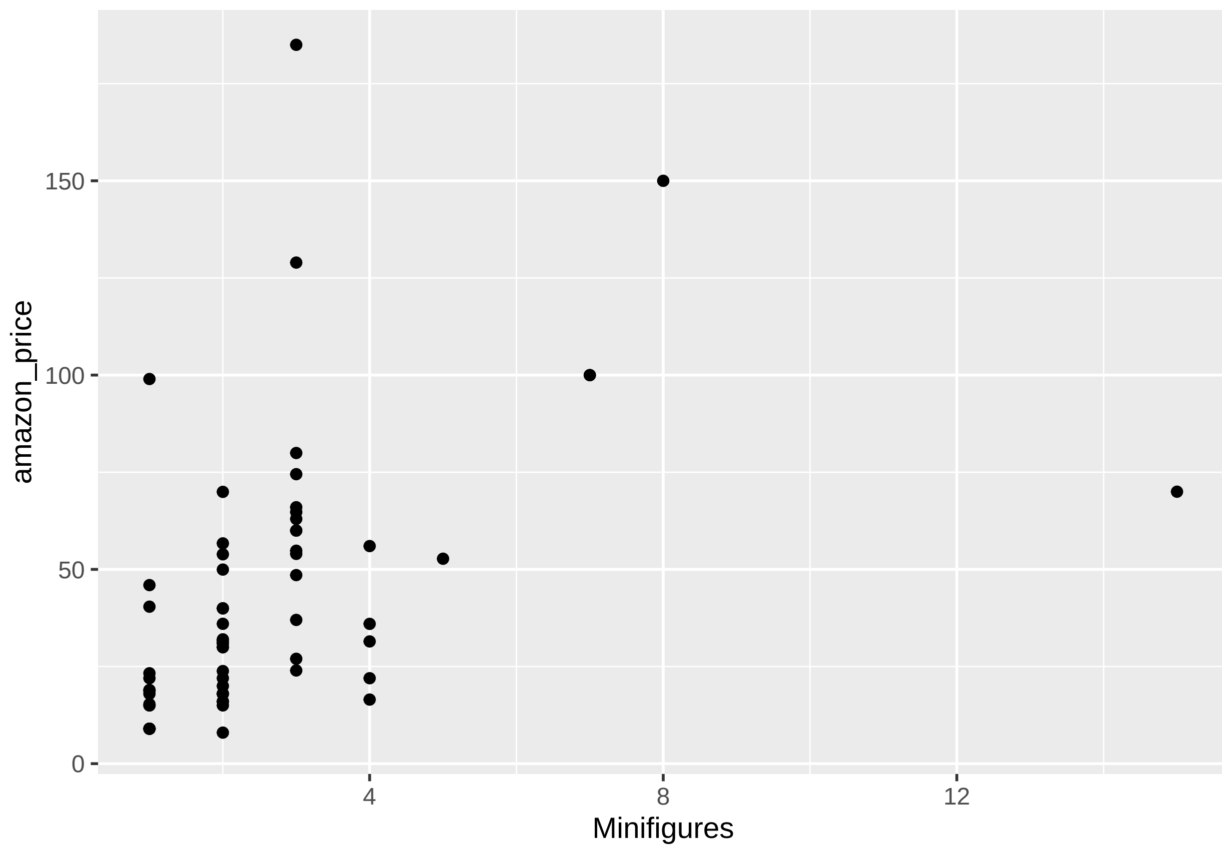



```r
ggplot(data = legos,
       aes(x = Pages,
           y = amazon_price)) +
  geom_point()
```




```
ggplot(data = legos,
       aes(x = Minifigures,
           y = amazon_price)) +
  geom_point()
```

```
## Warning: Removed 10 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



Q2

Spearman's as there are outliers in the y-direction

Q3

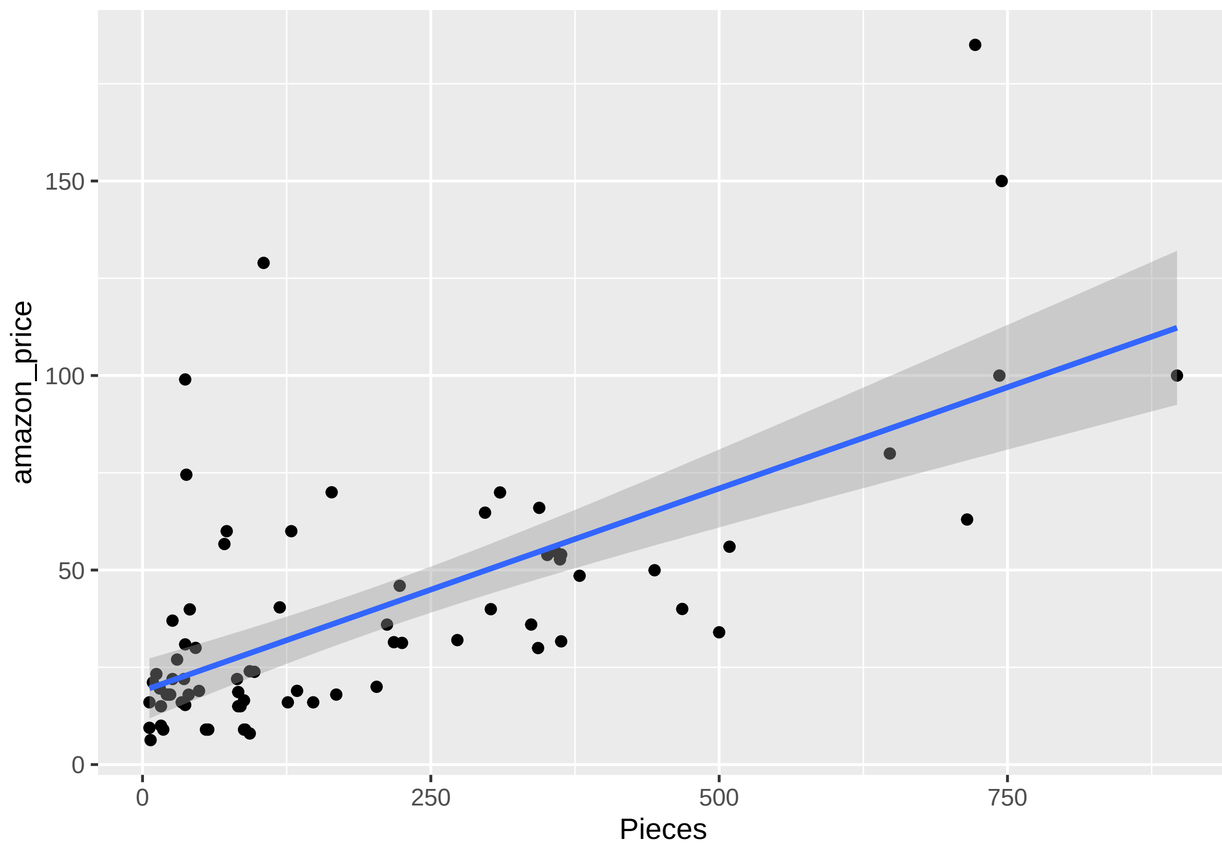
As one variable increases the other, generally, decreases. Try to not sound definitive where the other variable *has* to decrease

Q4

Using geom_smooth(), please plot a best-fit-line (by using the 'lm' method of geom_smooth) to the scatterplot of amazon price by number of pieces. Describe the scatterplot by noting it's direction, form, outliers, and strength please.

```
ggplot(data = legos,
       aes(x = Pieces,
           y = amazon_price)) +
  geom_point() +
  geom_smooth(method = 'lm')
```

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



Q5

Using the lm() function, please fit a linear model with amazon price as the response variable and the number of pieces as the explanatory variable. Print out the summary of the model using the summary() function

```
my_line_mod <- lm(amazon_price ~ Pieces,
                  data = legos)
```

Q6

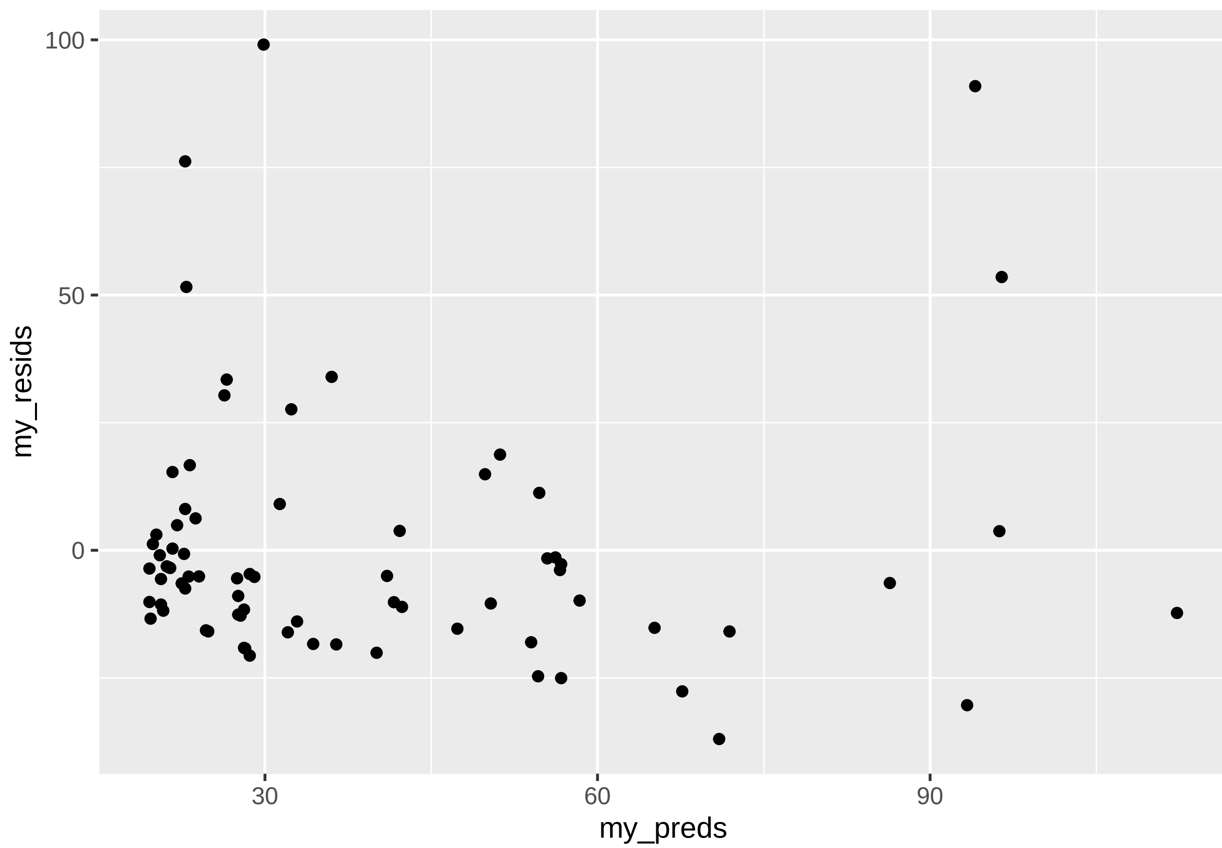
Please save your residuals and your predictions from this model as columns in the data set "legos". The resid() and predict() functions are useful for this.

```
#DATA$NEW_VAR <- function(something)
legos$my_resids <- resid(my_line_mod)
legos$my_preds <- predict(my_line_mod)
```

Q7

Make a residual scatterplot by having the residuals of your model on the y-axis and the predicted price on the x-axis.

```
ggplot(legos,
       aes(x = my_preds,
           y = my_resids)) +
  geom_point()
```



Q8

Please comment on if the homoskedasticity and normality assumptions are met for our linear model by using the graph made in question 10.

Neither are met by a long shot. Too many outliers in the y direction near the left and right hands of the graphs but not in the middle. That either implies normality fails as it's not nearly symmetric around the 0 line (eg look at the far left of the graph, negatives are right below 0 but the positives stretch waaaay out). Homoskedasticity also probably fails as the middle section of the graph seems to have less spread (but that could be just a lack of data in that section… not enough data -> we don't pick up a couple outliers)

Q9

Regardless of your answer to question 6, please write down the estimated linear regression equation. Be sure to use the name of the y and x variables in the equation and to indicate y is predicted (and not observed).

pred. amazon price = 18.96 + .104 * Pieces

Q10

Interpret your intercept from the above equation

If there are 0 pieces in the set, we predict the price to be $18.96

Q11

Interpret your slope from the above equation

If the number of pieces in the sets increase by one, we expect the mean price to increase by 10 cents

Q12

Predict the cost a lego set containing 55 pieces.

$24.67

Q13

The Monster Truck lego set actually has 55 pieces. Using Q15 and the lego set's actual amazon price please calculate the residual. HINT: Monster Truck is the 71st row of our data set.

```
8.99 - 24.67
```

```
## [1] -15.68
```

Q14

Find R^2. There are several ways to do this including using the summary() output for the model earlier or using pearson's correlation coefficient.

.4466

Q15

All said and done, do you think this model explains the relationship well?

No, the assumptions failed too hard and our best-fit-line is off

Transformation

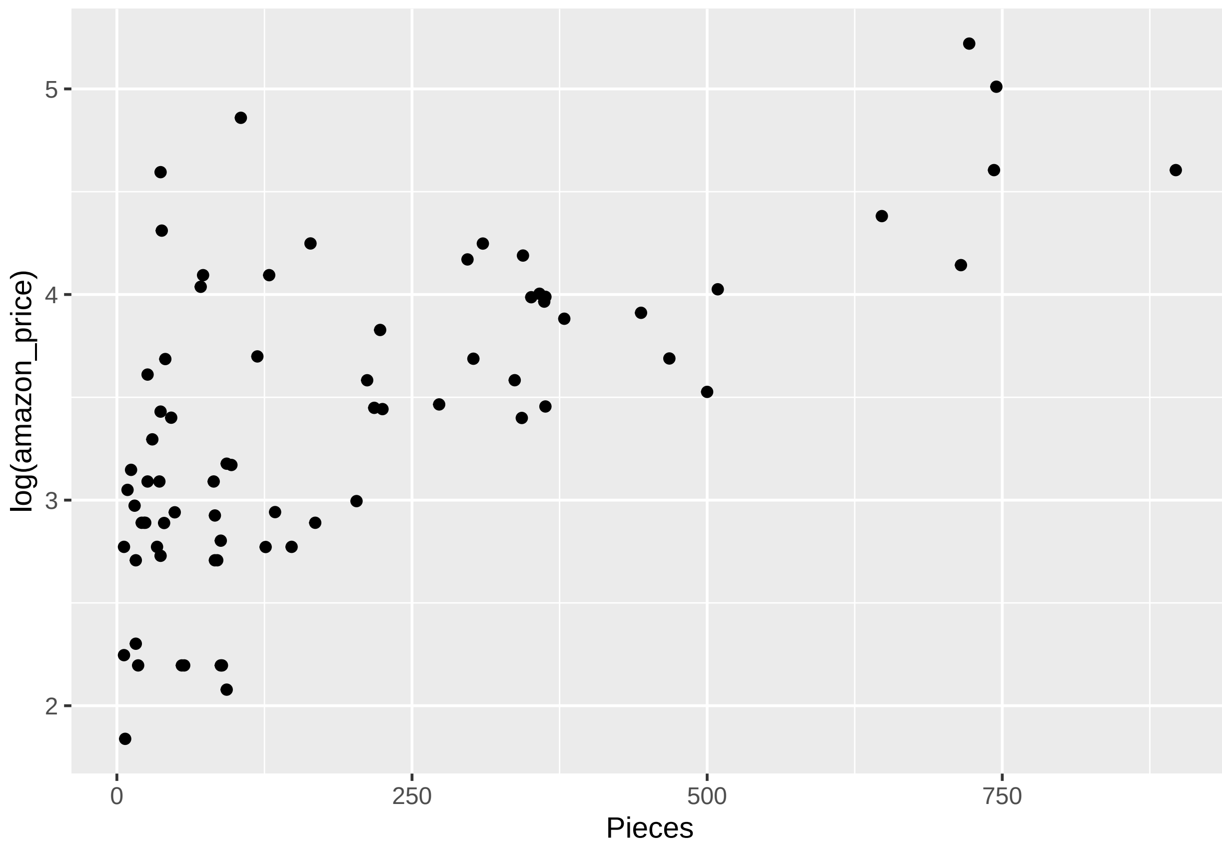
What I dislike about the residual graph I made is that there seemed to be some really outstretched values along the y-axis. That can indicate that the response variable should be transformed via a log() fucntion (but not always!!).

Log-Linear Model

Q16

As such, please make a scatterplot with the log of the amazon price as the y-axis and leave the x-axis as the number of pieces used. Comment on whether you think this graph is sufficently linear.

```
ggplot(legos,
       aes(x = Pieces,
           y = log(amazon_price))) +
  geom_point()
```



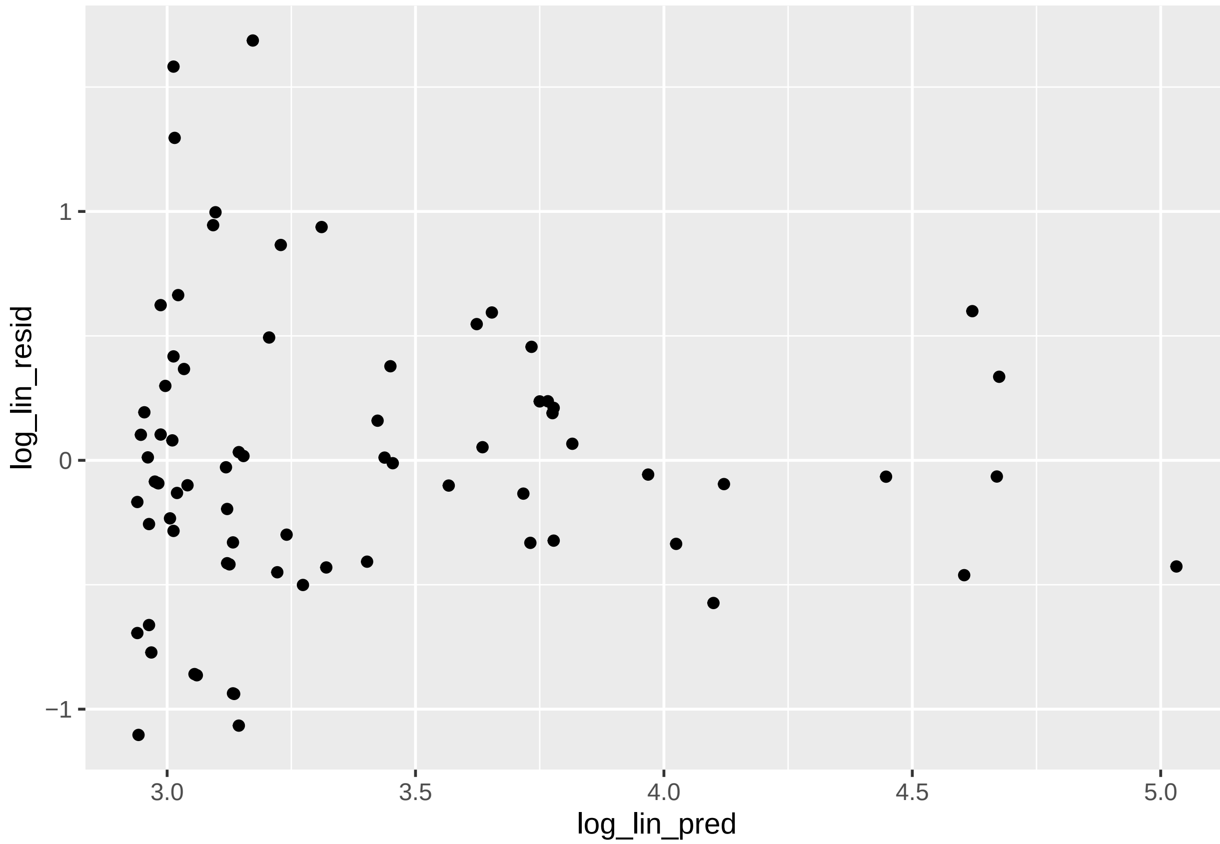
Q17

Homoskedasticity fails as the left side is waaaay more spread out than the right side. Normality is actually probably okay as there is a roughly equal spread of points both above and below the line at all points in the graph

```
mod_log_line <- lm(log(amazon_price) ~ Pieces,
                   data = legos)

legos$log_lin_resid <- resid(mod_log_line)
legos$log_lin_pred <- predict(mod_log_line)
```

```r
ggplot(legos,
       aes(y = log_lin_resid,
           x = log_lin_pred)) +
  geom_point()
```



Let's try one more transformation to see if we can get something closer to what we are after

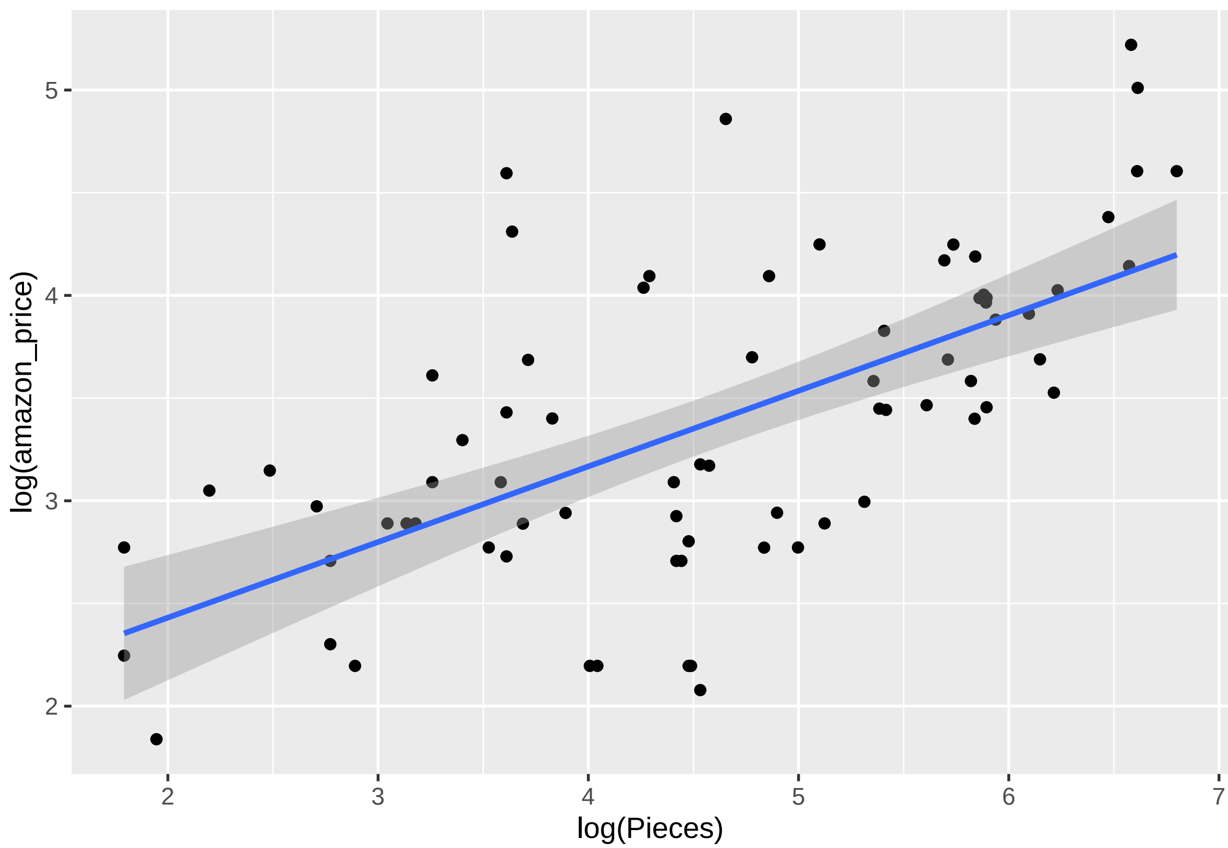
Log-Log Model

Q18

As such, please make a scatterplot with the log of the amazon price as the y-axis and the log of the number of pieces used as the x-axis. Use geom_smooth to fit a best-fit-line similar to question 2.

```r
ggplot(legos,
       aes(x = log(Pieces),
           y = log(amazon_price))) +
  geom_point() +
  geom_smooth(method = 'lm')
```

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



Q19

Fit a linear model using log(amazon_price) as your response and log(Pieces) as your explanatory variable.

```
#Log Log Model format:
# lm(log(RESPONSE) ~ log(EXPLANATORY),
#    data = DATA)
#Note how similar it is to the regular linear model, the
#difference is we are feeding in the log of amazon
#price and the log of the number of pieces
my_log_mod <- lm(log(amazon_price) ~ log(Pieces),
                 data = legos)
```

Q20

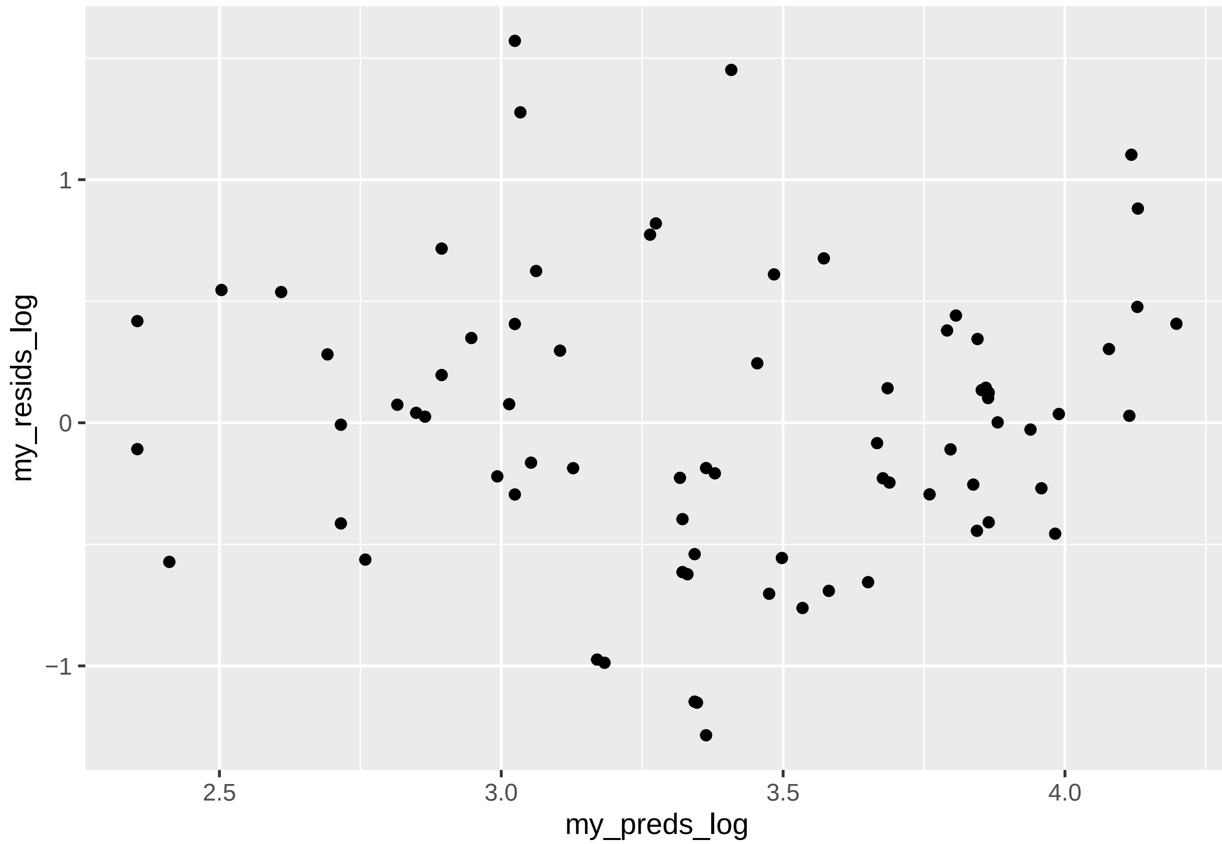
Create a residual graph for the model created in Q22 and comment on whether the normality and homoskedasiticity assumptions are met.

This graph is fine, don't stare toooo long although there does seem to be a minor "bulge" in the middle which raises questions about if there is heteroskedasticity. Buuut I think the spread is so bad as to stop us from using this model

```
#DATA$NEW_VAR <- function(something)
legos$my_resids_log <- resid(my_log_mod)
```

```r
legos$my_preds_log <- predict(my_log_mod)

ggplot(legos,
       aes(x = my_preds_log,
           y = my_resids_log)) +
  geom_point()
```



Q21

Write down your estimated equation. Be sure to indicate what the y and x variables are and that the response is estimated. Also note that in your model both variables are transformed to log()'s. You do not need to back transform for this question.

predicted log amazon price = 1.69 + .368 * log(pieces)

```r
summary(my_log_mod)
```

```
##
## Call:
## lm(formula = log(amazon_price) ~ log(Pieces), data = legos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.28521 -0.40279  0.00156  0.36412  1.57092
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.69477    0.25161   6.736 3.21e-09 ***
## log(Pieces)  0.36814    0.05269   6.987 1.10e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5894 on 73 degrees of freedom
## Multiple R-squared:  0.4007, Adjusted R-squared:  0.3925
## F-statistic: 48.81 on 1 and 73 DF,  p-value: 1.103e-09
```

NOTE: Compare the predicted equation against the R output to get comfortable reading it

Q22

Interpret your value for $\hat{\beta}_0$, the intercept of your model. Be careful to differentiate between predicting the mean vs predicting the median.

If the number of pieces is set to 1 (such that the log(pieces) is 0), then we expect the median price of the lego sets to be exp(1.69) = \$5.45

Q23

Interpret your value for $\hat{\beta}_1$, the slope of your model. Be careful to differentiate between predicting the mean vs predicting the median.

For a 10% increase in the number of pieces we expect the median price of the lego sets to increase by a multiplicative factor 1.1ˆ.3681 = 1.0357 (ie the median increases by 3.5%, we believe)

Q24

Again, please find the predicted price for a lego set with 55 pieces using the model you just created. Be sure that the prediction is reported on the linear scale (ie I want the prediction listed in dollars). You will want to back transform for this problem.

\$23.80

```r
log_y_hat <- 1.6948 + .3681*log(55)

log_y_hat #This is the log scale!!!!!
```

```
## [1] 3.169899
```

```r
exp(log_y_hat) #This is on the linear scale
```

```
## [1] 23.80509
```

Q25

Using Q22's prediction, calculate the residual for the Monster Truck lego set in the data. Be sure that the residual is reported on the linear scale (ie I want the residual listed in dollars).

23.81 - 8.99 =

14.82

Indicators

For this we are going to do something a little odd. We are going to treat Year as a categorical variable and just say that 2018, 2019, and 2020 are just labels (ie nominal) that don't mean anything numerically.

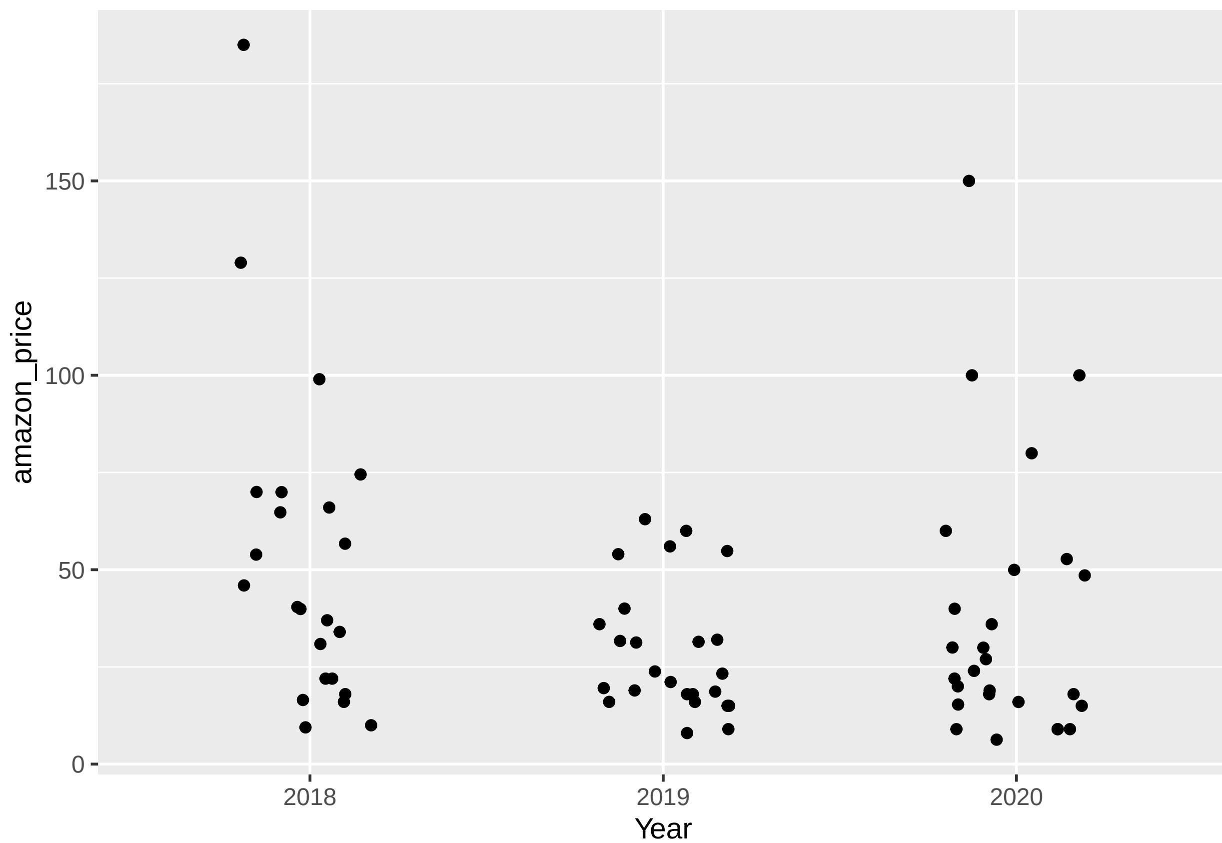
NOTE: This is a common strategy when you have very few numbers (eg only three years)

Q26

Make a plot similar to the one in the class notes. Your x-axis should be Year and your y-axis shoudl be amazon sales price

HINT: Use geom_jitter() and not geom_point(). If the points are spread out too wide, play around with the "width" parameter in geom_jitter()

```r
ggplot(legos,
       aes(x = Year,
           y = amazon_price)) +
  geom_jitter(width = .2)
```



Q27

Make a linear model using Year as an explanatory variable and amazon price as the response variable.

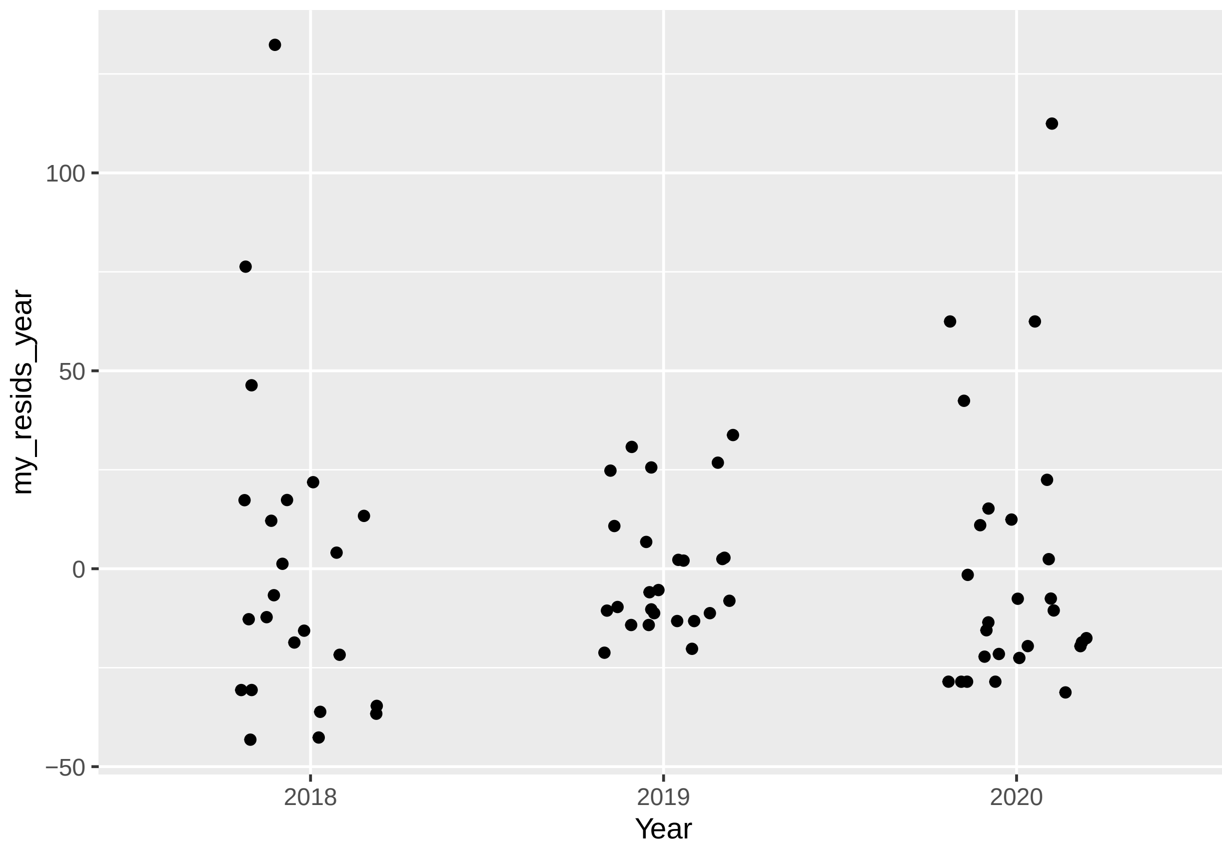
```
my_year_mod <- lm(amazon_price ~ Year,
              data = legos)
```

Q28

Make a scatterplot with your residuals on the y-axis and the x-axis being Year.

```
legos$my_resids_year <- resid(my_year_mod)
legos$my_preds_year <- predict(my_year_mod)

ggplot(legos,
       aes(x = Year,
           y = my_resids_year)) +
  geom_jitter(width = .2)
```



Q29

Comment on if the three categories (years) have heteroskedasticity or if the residuals are not normal.

HINT: Use geom_jitter() and not geom_point(). If the points are spread out too wide, play around with the "widths" parameter in geom_jitter()

Possibly heteroskedastic and only the year 2019 looks really normal… 2020 and 2018 look like they have a right tails

Q30

Write down your best-fit-line equation. Please use the model form which uses β's, and not the one that uses α's.

NOTE: Keep the negative on the coefficients!!!

predicted amazon price = 52.64 - 23.43* (1_{2019}) - 15.11 * (1_{2020})

HINT: run the summary() command on your model and then look at the "Coefficients" table, specifically the "Estimates" column. See the alternative slide deck for indicators for an example

```
summary(my_year_mod)
```

```
##
## Call:
```

```
## lm(formula = amazon_price ~ Year, data = legos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -43.189 -19.553  -9.668  12.264 132.351
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   52.639      6.745   7.804 3.56e-11 ***
## Year2019     -23.422      9.347  -2.506   0.0145 *
## Year2020     -15.106      9.179  -1.646   0.1042
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 32.35 on 72 degrees of freedom
## Multiple R-squared:  0.08186,    Adjusted R-squared:  0.05635
## F-statistic:  3.21 on 2 and 72 DF,  p-value: 0.04622
```

Q31

Predict the cost of a lego set that was made in the 2020.

37.53

```r
52.64 - 23.43*0 - 15.11*1
```

```
## [1] 37.53
```

```r
52.64 - 15.11
```

```
## [1] 37.53
```

Q32

Find the residual (again) for the Monster Truck data set 28.54

```r
37.53 - 8.99
```

```
## [1] 28.54
```

Q33

Interpret your $\hat{\beta}_0$ value

We predict the mean price of lego sets made in 2018 to be $52.64

Q34

Interpret your $\hat{\beta}_1$ value

We predict the change in mean price of lego sets made in 2018 to 2019 to be -$23.42

Q35

Interpret your $\hat{\beta}_2$ value

We predict the change in mean price of lego sets made in 2018 to 2020 to be -$15.11

Q36

Find the different between $\hat{\beta}_2$ and $\hat{\beta}_1$. What *is* this difference? What does it represent?

$8.31, this is the increase in the mean of price of lego sets going from those made in 2019 to those made in 2020, estimated