CMDA-3654

Homework 6

Michael La Vance

Due as a .pdf upload

Problem 1: [30 pts] Exploring Relationships between variables.

Load the `DatasaurusDozen.tsv` file into R.

This data consists of x and y observations for **13 sub-datasets** that have the following names:

`dino`, `away`, `h_lines`, `v_lines`, `x_shape`, `star`, `high_lines`, `dots`, `circle`, `bullseye`, `slant_up`, `slant_down`, `wide_lines`

a. Use `dplyr` functions to summarize each dataset in the following way: Compute the mean for x, mean for y, sd for x, sd for y, and the correlation coefficient between x and y. **Please round your answers to 2 decimal places.** The answers should be returned automatically in a tibble. Use `kable()` or `pandoc.table()` (use results='asis' in chunk definition if using `pandoc.table()`) or some other function to make nicely formatted table of your results.

```
dozen <- read.table(file = "DatasaurusDozen.tsv", sep = "\t", header = T)
options(digits = 7)
dozen %>%
  group_by(dataset) %>%
  summarise( mean.x = round(mean(x),2), mean.y = round(mean(y),2), StDev.x = round(sd(x),2), StDev.y = round(s
  kable()
```

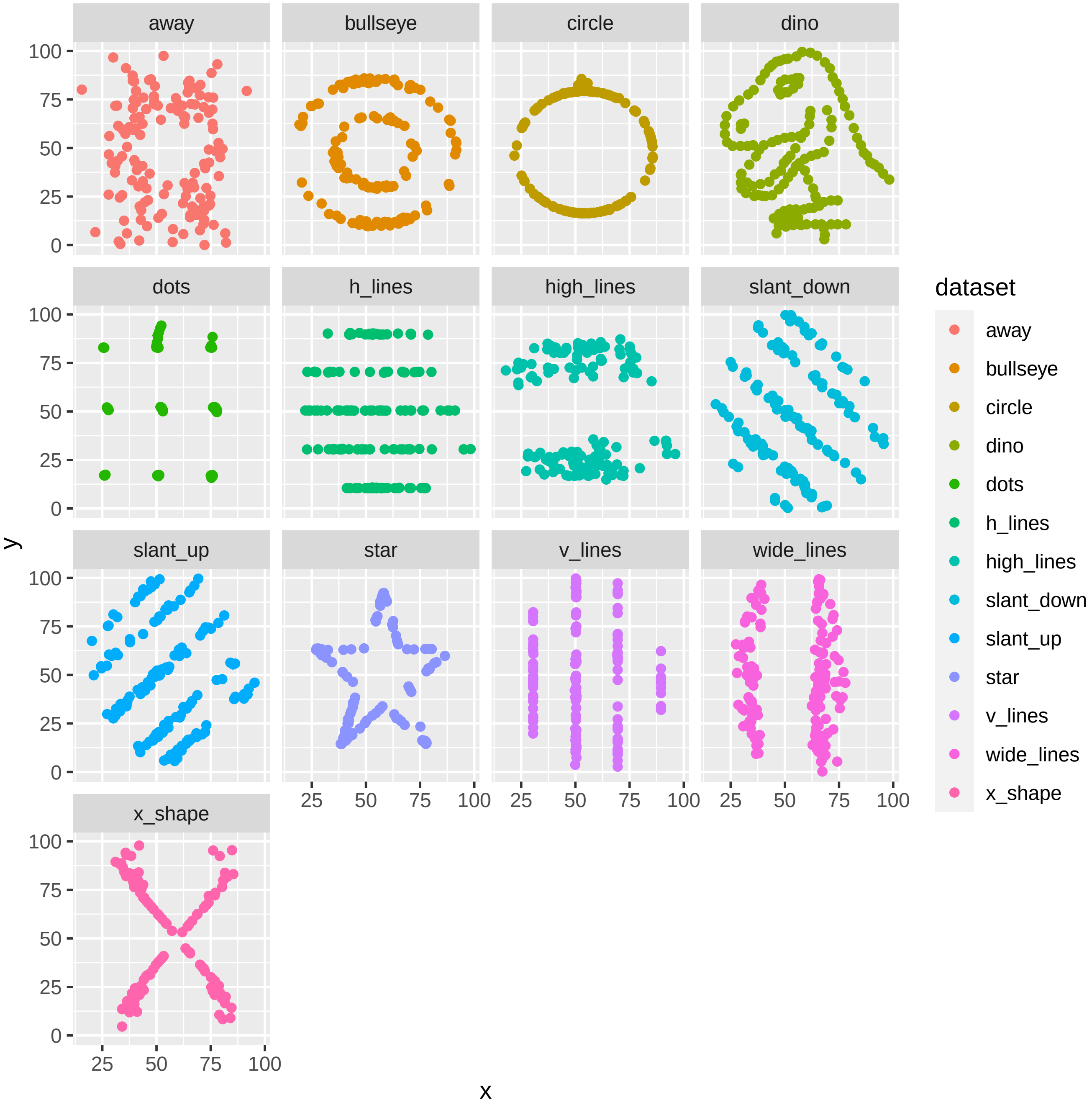
dataset	mean.x	mean.y	StDev.x	StDev.y	Correlation
away	54.27	47.83	16.77	26.94	-0.06
bullseye	54.27	47.83	16.77	26.94	-0.07
circle	54.27	47.84	16.76	26.93	-0.07
dino	54.26	47.83	16.77	26.94	-0.06
dots	54.26	47.84	16.77	26.93	-0.06
h_lines	54.26	47.83	16.77	26.94	-0.06
high_lines	54.27	47.84	16.77	26.94	-0.07
slant_down	54.27	47.84	16.77	26.94	-0.07
slant_up	54.27	47.83	16.77	26.94	-0.07
star	54.27	47.84	16.77	26.93	-0.06
v_lines	54.27	47.84	16.77	26.94	-0.07
wide_lines	54.27	47.83	16.77	26.94	-0.07
x_shape	54.26	47.84	16.77	26.93	-0.07

b. What does the numerical summaries tell you about the data in the 12 different data sets? In particular, does the correlation coefficient provide you with much information about the relationship between x and y?

The correlation coefficient does not provide much information regarding x and y, the coefficient is too close to 0 to be useful.

c. Now make a basic scatterplot of x and y for the 13 different datasets. Use a different color for each dataset. My best advice is to simply use `ggplot()` with `facet_wrap()`, as this can be done in a singe line.

```
ggplot(data = dozen) + geom_point(aes(x = x, y = y, col = dataset)) + facet_wrap(~dataset)
```



d. How does your interpretation about the relationships between x and y change after seeing the plots?

There are some well defined patterns, but none of them would be able to be found with a linear regression.

e. What lesson can be learned here?

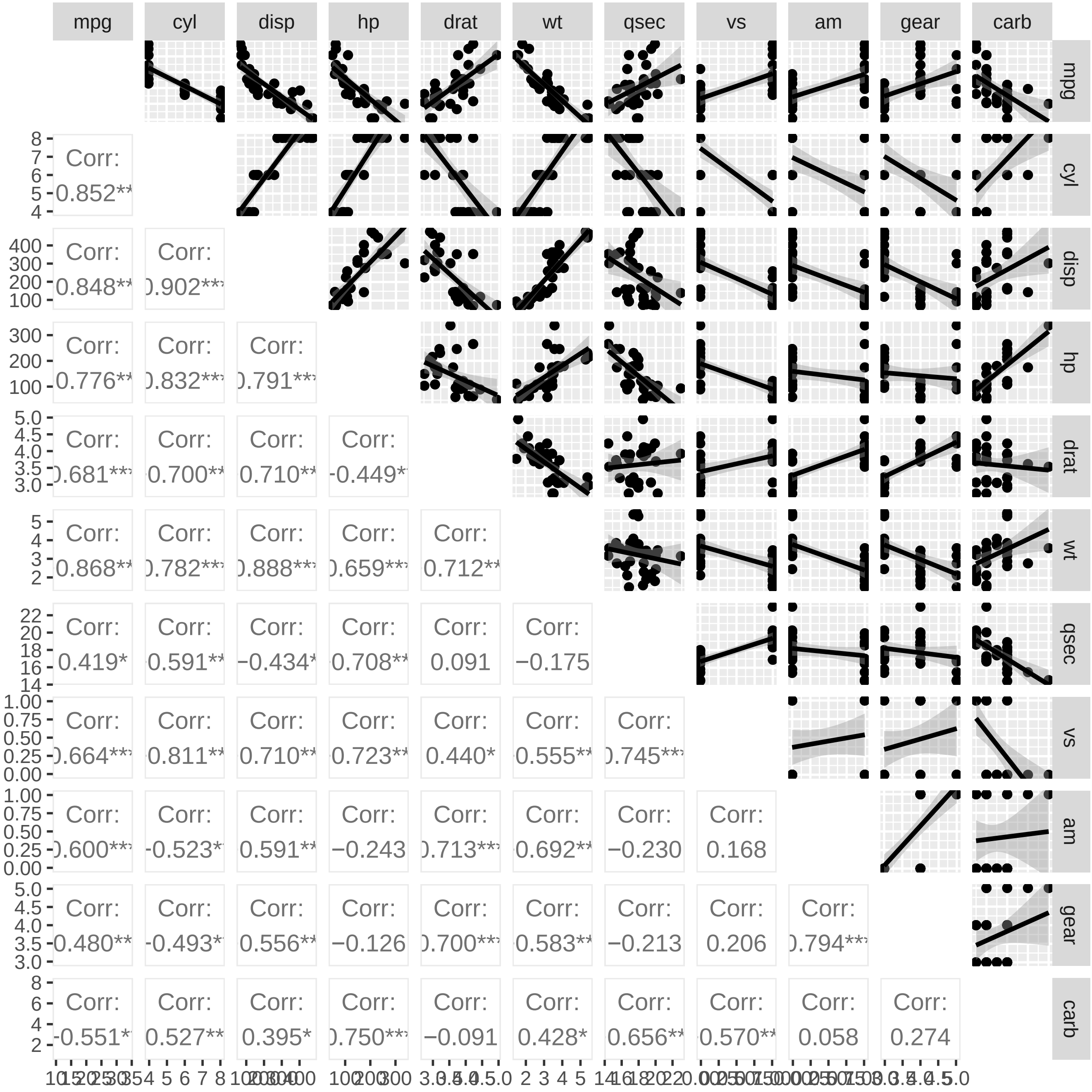
You should always visualize the data before coming to conclusions with it.

Problem 2: [30 pts] Linear Regression

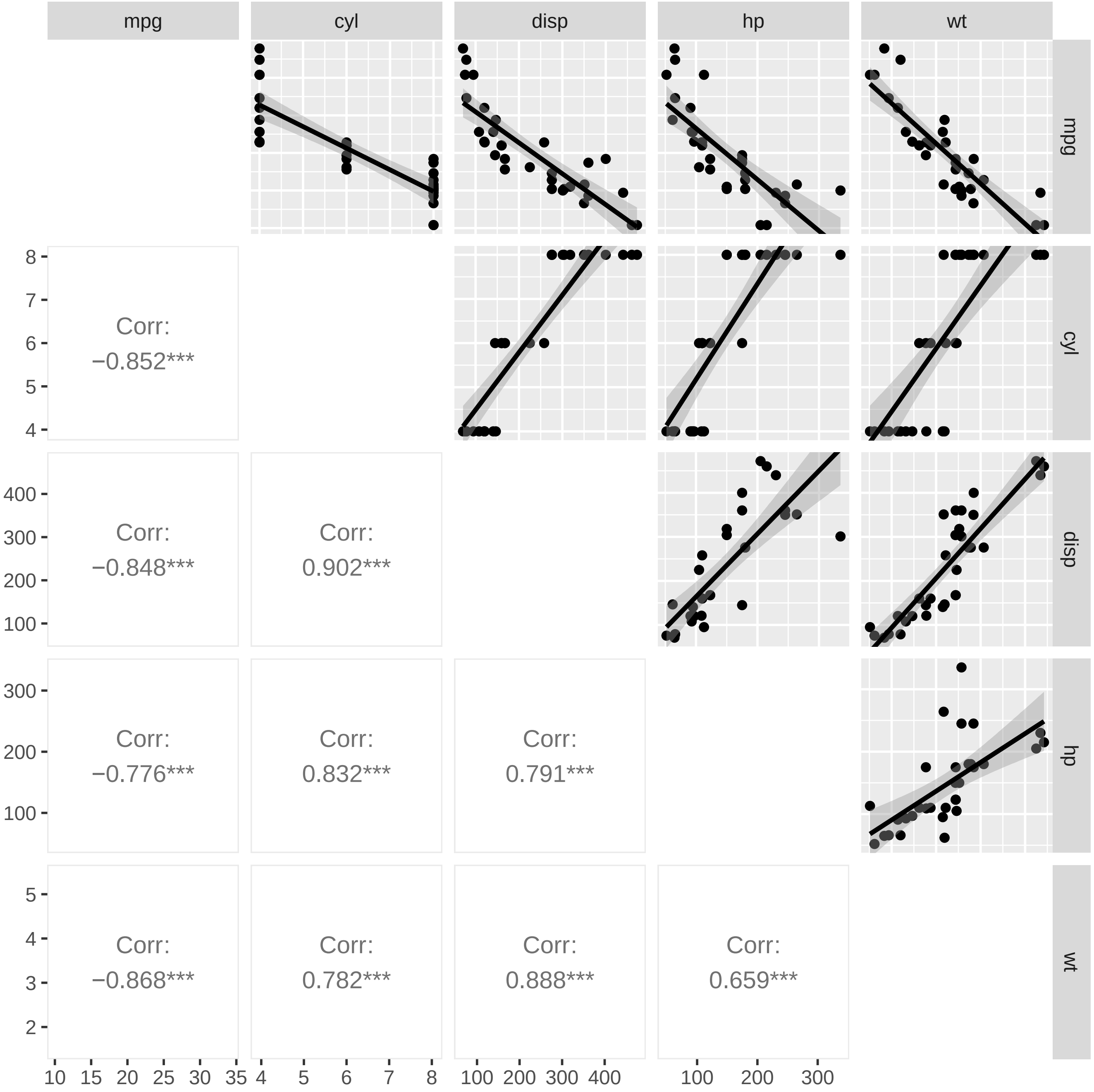
Consider the `mtcars` dataset. Say we want to build a linear regression model that predicts `mpg`, using any subset of the other variables as predictors.

a. Begin by creating a scatterplot matrix between mpg and all other predictors. Report the correlations as well in either the upper or lower half of the scatterplot matrix.

```
ggpairs(mtcars, upper = list(continuous = "smooth"),
        lower = list(continuous = "cor"),
        diag = list(continuous = "blank"))
```



```
ggpairs(mtcars[, c("mpg", "cyl", "disp", "hp", "wt")],
        upper = list(continuous = "smooth"),
        lower = list(continuous = "cor"),
        diag = list(continuous = "blank")
        )
```



b. What are the three variables most highly correlated with mpg?

The number of cylinders, displacement, horse power, and the weight seem to be good predictors of mpg.

c. Fit three simple linear regression models using your previous three variables/predictors. Report summaries for the models. Which model would you choose and why?

```
lm(mpg~cyl, data = mtcars) %>%
  summary()
```

```
Call:
lm(formula = mpg ~ cyl, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.9814 -2.1185  0.2217  1.0717  7.5186 

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

```
(Intercept)  37.8846     2.0738   18.27  < 2e-16 ***
cyl          -2.8758     0.3224   -8.92 6.11e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.206 on 30 degrees of freedom
Multiple R-squared:  0.7262,    Adjusted R-squared:  0.7171
F-statistic: 79.56 on 1 and 30 DF,  p-value: 6.113e-10
```

```
lm(mpg~hp, data = mtcars) %>%
  summary()
```

```
Call:
lm(formula = mpg ~ hp, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-5.7121 -2.1122 -0.8854  1.5819  8.2360

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 30.09886    1.63392  18.421  < 2e-16 ***
hp          -0.06823    0.01012  -6.742 1.79e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.863 on 30 degrees of freedom
Multiple R-squared:  0.6024,    Adjusted R-squared:  0.5892
F-statistic: 45.46 on 1 and 30 DF,  p-value: 1.788e-07
```

```
lm(mpg~wt, data = mtcars) %>%
  summary()
```

```
Call:
lm(formula = mpg ~ wt, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-4.5432 -2.3647 -0.1252  1.4096  6.8727

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  37.2851     1.8776  19.858  < 2e-16 ***
wt           -5.3445     0.5591  -9.559 1.29e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.046 on 30 degrees of freedom
Multiple R-squared:  0.7528,    Adjusted R-squared:  0.7446
F-statistic: 91.38 on 1 and 30 DF,  p-value: 1.294e-10
```

I would choose weight because it has the highest t-value.

d. Create a multiple linear regression (MLR) model using `stepAIC()` to identify the best subset of predictors from all of the variables in `mtcars` (obviously `mpg` is still the response variable). Report these predictors, and a summary of the model these predictors produced.

```
carsfull <- lm(mpg ~ ., mtcars)
cars1 <- lm(mpg ~ 1, data = mtcars)
mlrcar <- stepAIC(cars1, scope = list(lower = cars1, upper = carsfull),
        direction = "both", trace = 0)

coefficients(mlrcar)
```

```
(Intercept)          wt         cyl          hp 
 38.7517874  -3.1669731  -0.9416168  -0.0180381 
```

```
summary(mlrcar)
```

```
Call:
lm(formula = mpg ~ wt + cyl + hp, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.9290 -1.5598 -0.5311  1.1850  5.8986 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 38.75179    1.78686  21.687  < 2e-16 ***
wt          -3.16697    0.74058  -4.276 0.000199 ***
cyl         -0.94162    0.55092  -1.709 0.098480 .  
hp          -0.01804    0.01188  -1.519 0.140015    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.512 on 28 degrees of freedom
Multiple R-squared:  0.8431,    Adjusted R-squared:  0.8263 
F-statistic: 50.17 on 3 and 28 DF,  p-value: 2.184e-11
```

According to the setpAIC() function the best predictive variable from this data set are, weight, #of cylinders, and horsepower.

e. Compare your MLR model to your three simple linar regression models earlier. Are any of those predictors in your MLR model? Are the coefficients the same for those predictors? If not, explain what may have caused the change. All of the variables were in my previous answer, I think the reason that the coefficients are different for each variable is because the amount of variables used for the prediction has increased, thus causing each individual variable to have less impact on the response variable.

Problem 3: [30 pts] More Linear Regression

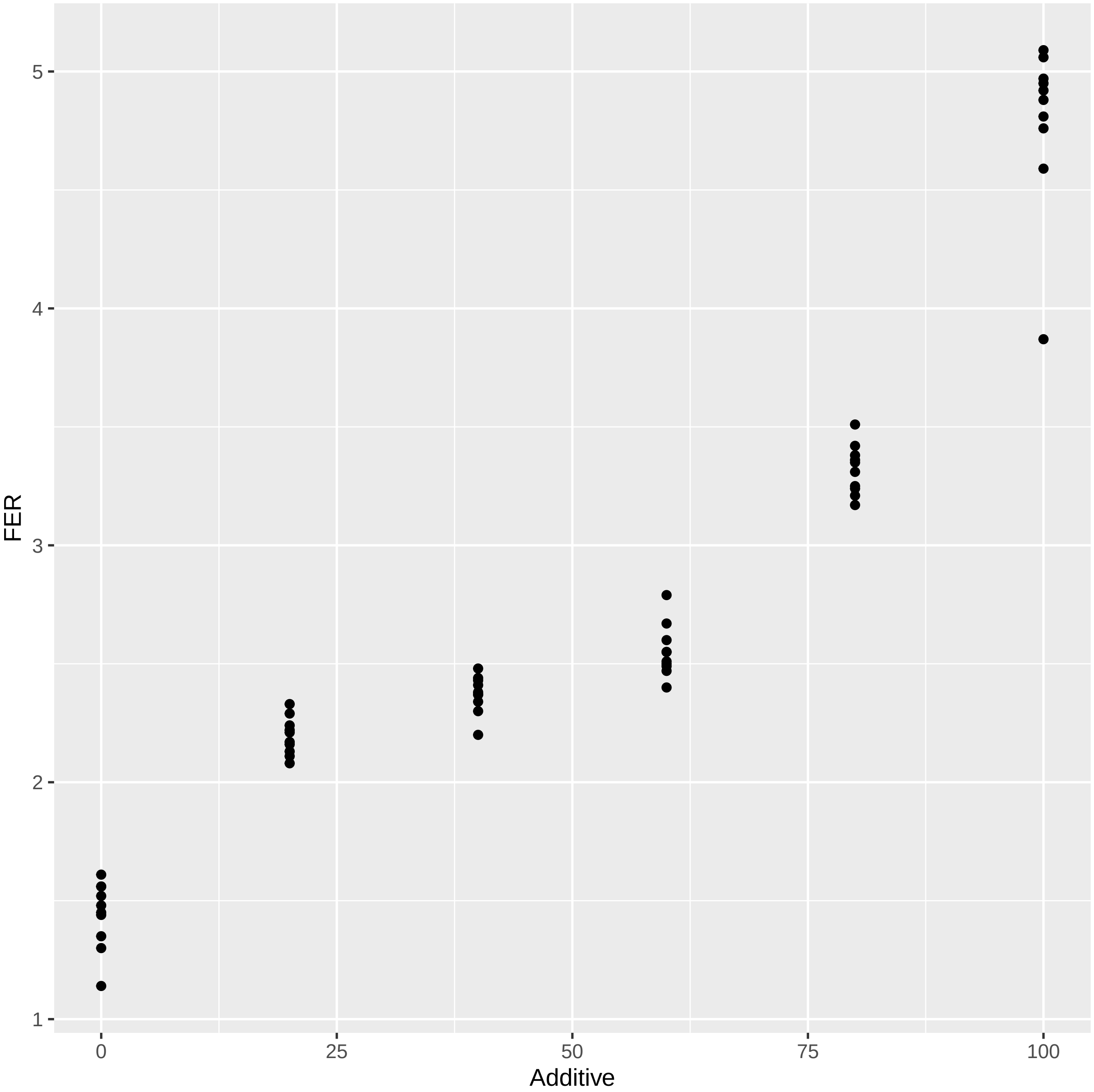
Sometimes your dataset is rather small, but you see that a simple linear regression is not appropriate so you try harder to fit a more complicated model. This is an example of such a situation.

A poultry scientist was studying various dietary additives to increase the rate at which chickens gain weight. One of the potential additives was studied by creating a new diet that consisted of a standard basal diet supplemented with varying amounts of the additive (0, 20, 40, 60, 80, and 100 grams). There were 60 chicks available for the study. Each of the six diets was randomly assigned to 10 chicks. At the end of 4 weeks, the feed efficiency ratio, feed consumed (gm) to weight gain (gm), was obtained for the 60 chicks. The experiment was also concerned with the effects of high levels of copper in the chick feed. Five of the 10 chicks in each level of the feed additive received 400 ppm of copper, while the remaining five chicks received no copper.

The data is contained in the `chicken.csv` data file.

a. In order to explore the relationship between feed efficiency ratio (FER) and feed additive (A), plot the FER versus A.

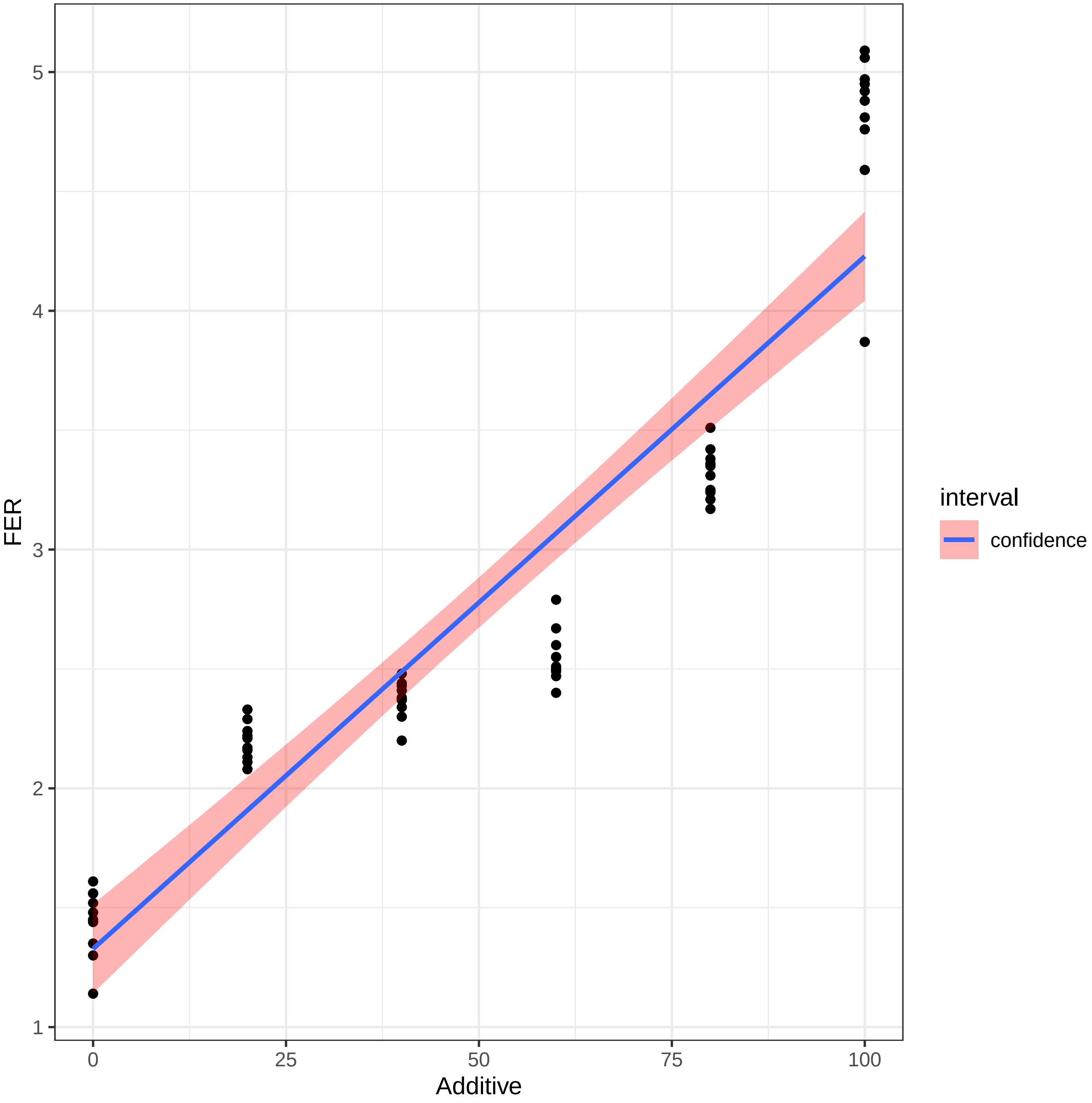
```
chikn <- read.csv(file = "chicken.csv")

ggplot(data = chikn) + geom_point(aes(x = Additive, y = FER))
```



b. What type of regression appears most appropriate? Polynomial regression would appear to be the best representation of the true relationship.

c. Fit first-order, quadratic, and cubic regression models to the data. Which regression equation provides the best fit to the data? Justify your answer using evidence based upon plots and relevant summaries.

```
#first order
first <- lm(FER ~ Additive, data = chikn)
#creating graph
firstreg.df <- data.frame(chikn, predict(first, interval = "predict"))
p1 <- ggplot(firstreg.df, aes(y = FER, x = Additive))
p1 + geom_point() +   geom_smooth(method = "lm", formula = y~x, aes(fill="confidence"), alpha=0.3) +  scale_fi
```

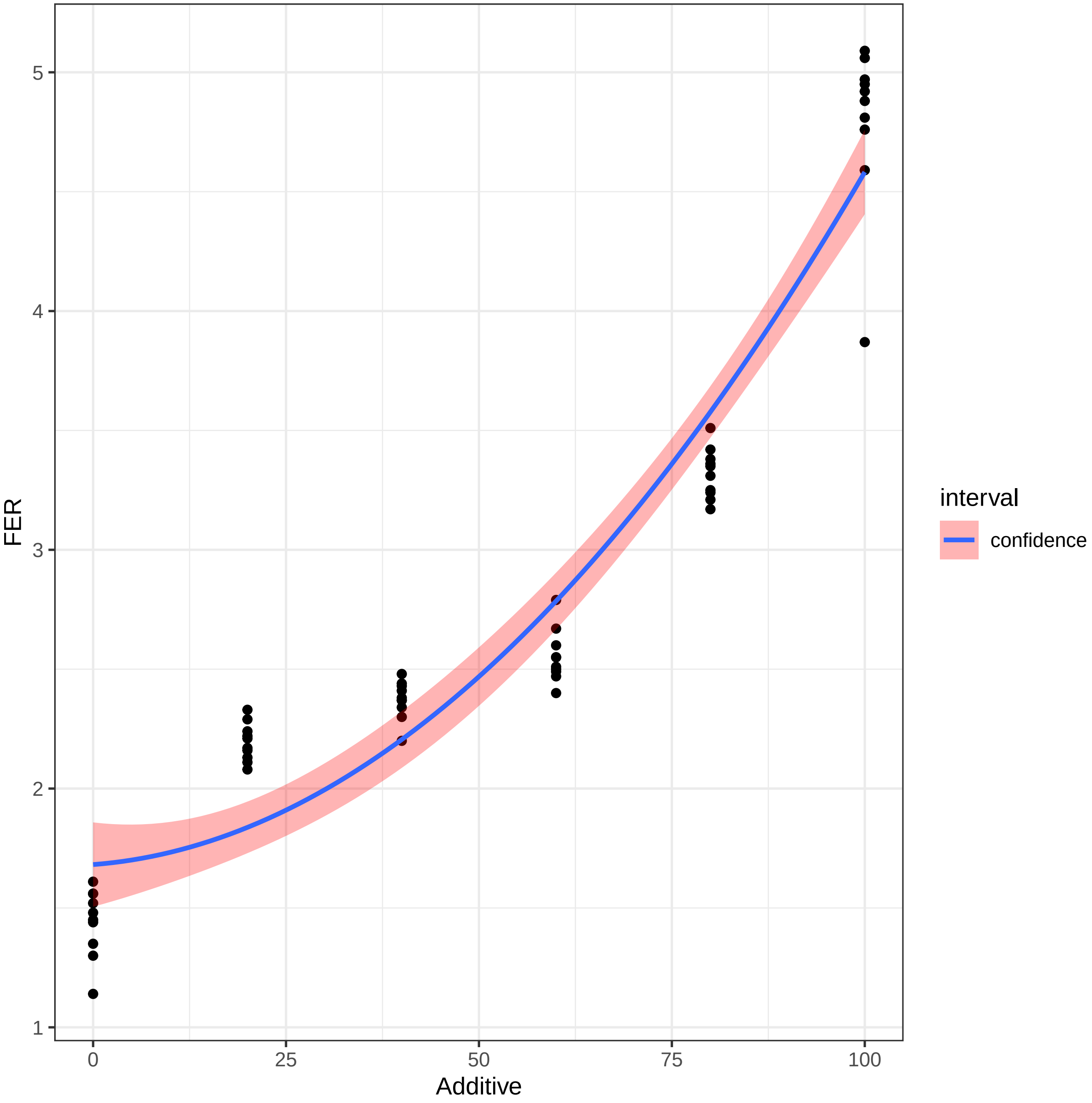


```
summary(first)
```

```
Call:
lm(formula = FER ~ Additive, data = chikn)

Residuals:
     Min       1Q   Median       3Q      Max
-0.66839 -0.30850 -0.05328  0.26687  0.86138

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1.328048   0.093145   14.26   <2e-16 ***
Additive    0.029006   0.001538   18.86   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.407 on 58 degrees of freedom
Multiple R-squared:  0.8598,    Adjusted R-squared:  0.8573
```

```
F-statistic: 355.6 on 1 and 58 DF,  p-value: < 2.2e-16
```

```
#second order
second <- lm(FER ~ Additive + I(Additive^2), data = chikn)
#creating graph
secondreg.df <- data.frame(chikn, predict(second, interval = "predict"))
p2 <- ggplot(secondreg.df, aes(y = FER,  x = Additive))
p2 + geom_point() +   geom_smooth(method = "lm", formula = y~x + I(x^2), aes(fill="confidence"), alpha=0.3) +
```

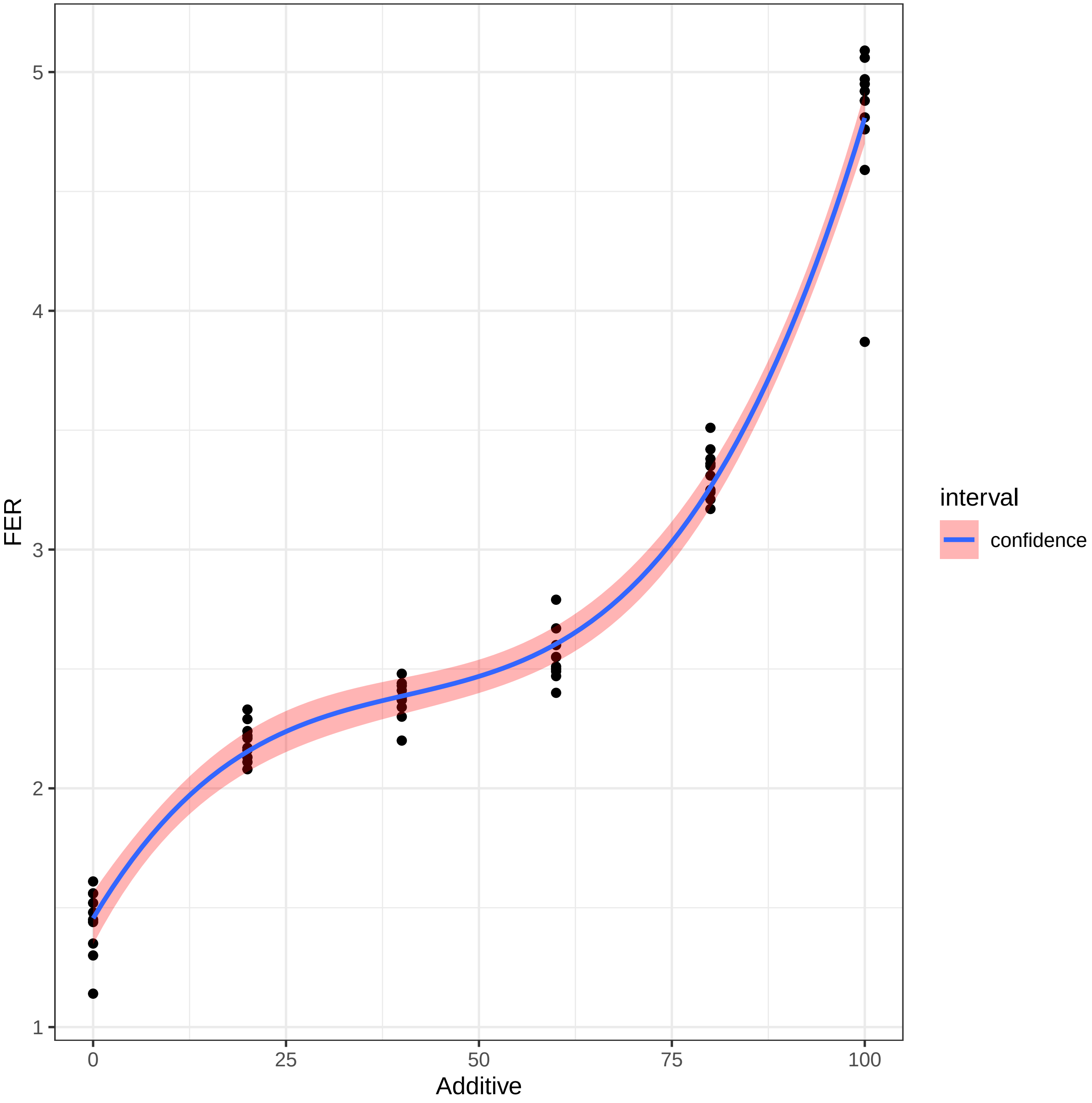


```
summary(second)
```

```
Call:
lm(formula = FER ~ Additive + I(Additive^2), data = chikn)

Residuals:
     Min       1Q   Median       3Q      Max 
-0.71225 -0.23703 -0.03657  0.27308  0.50775 
```

```
Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   1.682e+00  8.825e-02  19.055  < 2e-16 ***
Additive      2.483e-03  4.151e-03   0.598    0.552
I(Additive^2) 2.652e-04  3.984e-05   6.657 1.18e-08 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3079 on 57 degrees of freedom
Multiple R-squared:  0.9211,    Adjusted R-squared:  0.9183
F-statistic: 332.7 on 2 and 57 DF,  p-value: < 2.2e-16
```

```
#third order
third <- lm(FER ~ Additive + I(Additive^2) + I(Additive^3), data = chikn)
#creating graph
thirdreg.df <- data.frame(chikn, predict(third, interval = "predict"))
p3 <- ggplot(thirdreg.df, aes(y = FER,  x = Additive))
p3 + geom_point() +   geom_smooth(method = "lm", formula = y~x + I(x^2) + I(x^3), aes(fill="confidence"), alph
```



```
summary(third)
```

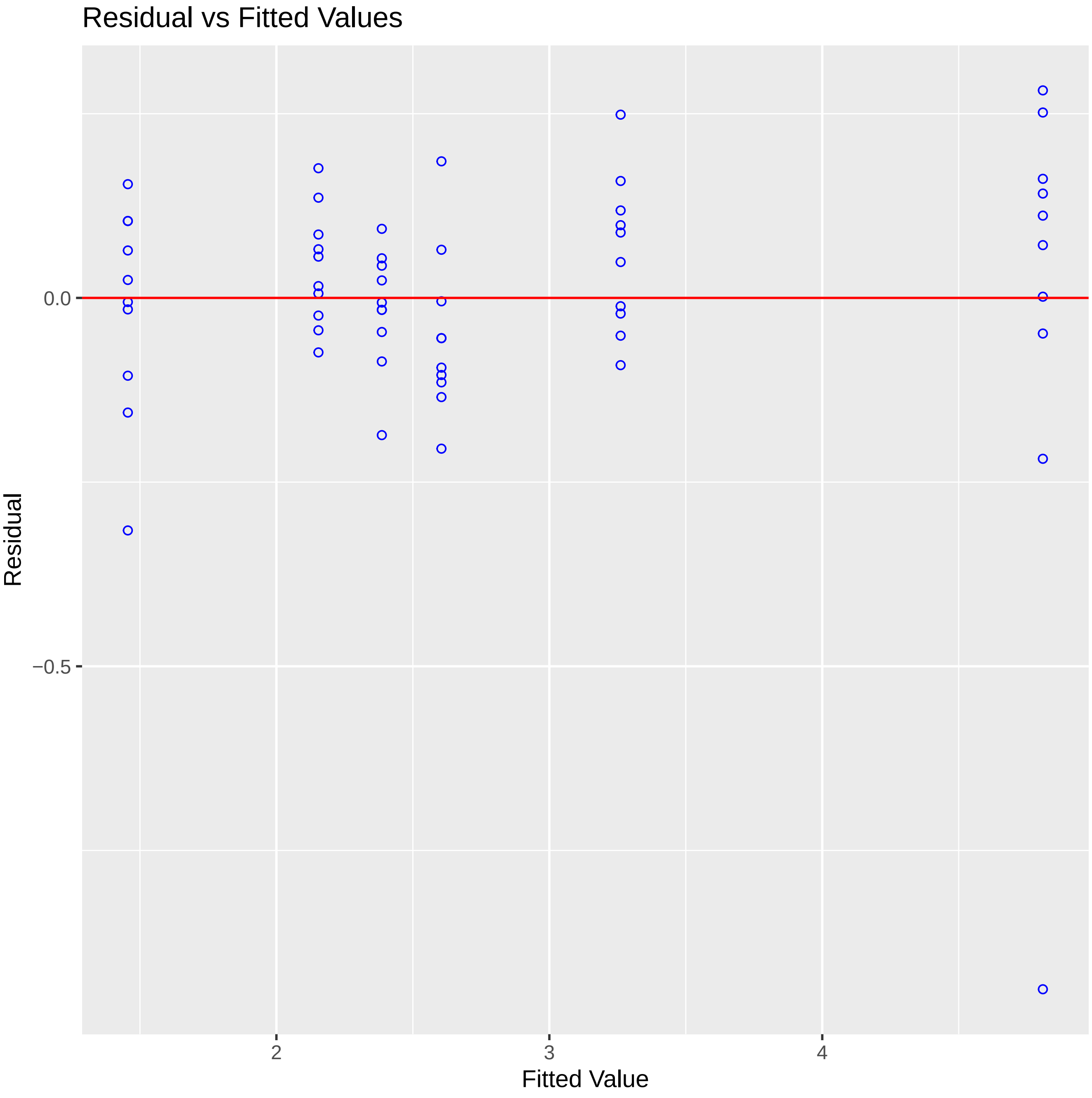
```
Call:
lm(formula = FER ~ Additive + I(Additive^2) + I(Additive^3),
    data = chikn)

Residuals:
     Min       1Q   Median       3Q      Max
-0.93833 -0.05462  0.00386  0.09501  0.28167

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    1.456e+00  5.438e-02  26.766  < 2e-16 ***
Additive       5.411e-02  5.282e-03  10.244 1.89e-14 ***
I(Additive^2) -1.148e-03  1.312e-04  -8.746 4.66e-12 ***
I(Additive^3)  9.420e-06  8.617e-07  10.932 1.64e-15 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 0.1755 on 56 degrees of freedom
Multiple R-squared:  0.9748,    Adjusted R-squared:  0.9735
F-statistic: 722.8 on 3 and 56 DF,  p-value: < 2.2e-16
```

The cubic regression model best fits the data, the regression line goes the center of each additive band. and the t-values of the different exponents is the highest of the three.

d. Is there anything peculiar about any of the data values? Provide an explanation of what may have happened. (Hint: Look at regression diagnostics like plots of the residuals versus the fitted values (or x), plot the leverages, or plot some measure of influence.)

```
ols_plot_resid_fit(third)
```



There seems to be an outlier in the data at the largest additive spot. Based on its location it could either have been recorded wrong and been in the nest lower group, or the chicken might not have eaten all of its food during the testing and causing it to have a lower amount than what was thought.

e. Using your best polynomial model from (b) & (c). Fit a new model that includes the linear addition of copper and display the estimate table. Does Copper provide a significant improvement to the fit? Carry out an F-test that compares the Full model that contains Copper and the reduced model that has your polynomial model fit on the additive only. Discuss the results.

```
fourth <- lm(FER ~ Additive + I(Additive^2) + I(Additive^3) + Copper, data = chikn)
summary(fourth)
```

```
Call:
lm(formula = FER ~ Additive + I(Additive^2) + I(Additive^3) +
    Copper, data = chikn)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.8973 -0.0526  0.0139  0.1018  0.2927 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)    1.415e+00  5.768e-02  24.524  < 2e-16 ***
Additive       5.411e-02  5.171e-03  10.463 1.08e-14 ***
I(Additive^2) -1.148e-03  1.285e-04  -8.933 2.71e-12 ***
I(Additive^3)  9.420e-06  8.437e-07  11.165 9.30e-16 ***
Copper         2.050e-04  1.109e-04   1.848   0.0699 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1718 on 55 degrees of freedom
Multiple R-squared:  0.9763,    Adjusted R-squared:  0.9746 
F-statistic: 566.3 on 4 and 55 DF,  p-value: < 2.2e-16
```

```
fifth <- lm(FER ~ Copper, data = chikn)
summary(fourth)
```

```
Call:
lm(formula = FER ~ Additive + I(Additive^2) + I(Additive^3) +
    Copper, data = chikn)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.8973 -0.0526  0.0139  0.1018  0.2927 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)    1.415e+00  5.768e-02  24.524  < 2e-16 ***
Additive       5.411e-02  5.171e-03  10.463 1.08e-14 ***
I(Additive^2) -1.148e-03  1.285e-04  -8.933 2.71e-12 ***
I(Additive^3)  9.420e-06  8.437e-07  11.165 9.30e-16 ***
Copper         2.050e-04  1.109e-04   1.848   0.0699 .  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1718 on 55 degrees of freedom
Multiple R-squared:  0.9763,    Adjusted R-squared:  0.9746 
F-statistic: 566.3 on 4 and 55 DF,  p-value: < 2.2e-16
```

This data suggests, that the copper content does not seem to influence the feed deficiency ratio. The addition of the Copper content to the prediction model did not improve its predictive capabilities.

Problem 4: [10 pts] Linear Regression with Indicator Variables

Consider the data in `smoking_birthweight.csv`. This data contains 3 variables. The birth weight of a baby (`Weight`), the length of gestation (`Gestation`) in weeks, and the smoking status of the mother (`Smoke`). The smoking status of the mother in this case is coded as `yes` or `no`. This is a categorical variable (aka factor) with 2 categories (a binary variable). We could have coded the levels of this factor as an indicator variable using `TRUE` or `FALSE`, or equivalently `1` or `0`, respectively.

a. Fit a first-order regression model with birth weight as the response variable and the gestation and smoking status as predictors. Write down the fitted regression model equation and interpret the regression coefficients. If you can do this, you should have no problem handling the extra credit.

```
smkbwt <- read.csv(file = "smoking_birthweight.csv", header = T, sep = "\t")
reg.smkbwt <- lm(Weight ~ Gestation + Smoke, data = smkbwt)

#printing summary
summary(reg.smkbwt)
```

```
Call:
lm(formula = Weight ~ Gestation + Smoke, data = smkbwt)

Residuals:
     Min       1Q   Median       3Q      Max 
-223.693  -92.063   -9.365   79.663  197.507 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -2389.573    349.206  -6.843 1.63e-07 ***
Gestation     143.100      9.128  15.677 1.07e-15 ***
Smokeyes     -244.544     41.982  -5.825 2.58e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 115.5 on 29 degrees of freedom
Multiple R-squared:  0.8964,    Adjusted R-squared:  0.8892 
F-statistic: 125.4 on 2 and 29 DF,  p-value: 5.289e-15
```

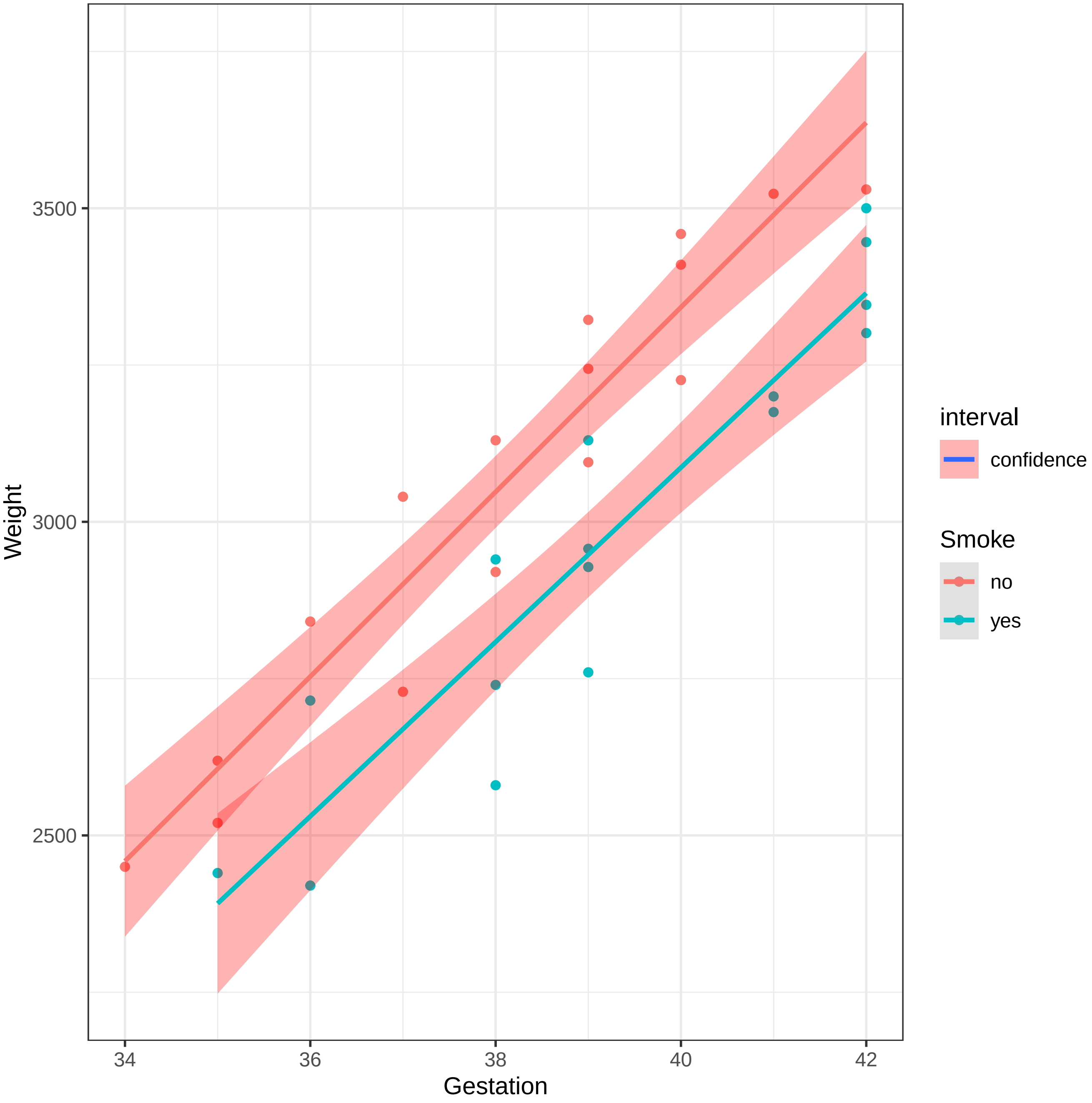
```
coefficients(reg.smkbwt)
```

```
(Intercept)   Gestation    Smokeyes 
 -2389.5729    143.1003   -244.5440 
```

Birthweight = -2389g + 143xGestation(Weeks)g -244.5xSmoke(1/0)g for every week increase in gestation the model predicts an average increase in birthweight by 143 grams, if the mother is a smoker subtract 244.5g from predicted birthweight. Because our data is not near 0 the intercept, by itself, does not make sense.

b. Plot the fitted regression lines (yes plural), why are there two?

```
#creating graph
smkbwt.df <- data.frame(smkbwt, predict(reg.smkbwt, interval = "predict"))
pl1 <- ggplot(smkbwt.df, aes(y = Weight,  x = Gestation, col = Smoke))
pl1 + geom_point() +   geom_smooth(method = "lm", formula = y~x, aes(fill="confidence"), alpha=0.3) +  scale_f
```



Due to our data having a boolean variable as a predictor, a single regression line would not make sense, as the the prediction would lie in the middle of the two different groups predictions.

Problem 5: [15 pts Extra Credit] Parameter Interpretation with Indicator Variables

Recall that indicator variables, sometimes called “dummy” variables, are binary variables that indicate whether an event is recognized or not (i.e., `1` if `TRUE` `0` if `FALSE`). Suppose we have a data set of reported salaries and highest achieved education levels. Suppose the variables are as follows: `salary`, `noHS`, `highSchoolGrad`, `Assoc`, `Bach`, `Masters`, `Doctorate`, where the levels of education are either a `1` or `0` depending on whether that is the given observation’s highest level of achieved education.

- Write down the multiple linear regression model. Specify which β_i are indicator variables.
- Write interpretations for all of your model parameters, that is β_i, for $i \in \{0, 1, 2, 3, 4, 5\}$.

- Now assume we were to add another variable to this data set: an observation's gender. Write down this new model, and now interpret β_0.