Introduction to linear regression

Ross Boehme

The Human Freedom Index is a report that attempts to summarize the idea of "freedom" through a bunch of different variables for many countries around the globe. It serves as a rough objective measure for the relationships between the different types of freedom - whether it's political, religious, economical or personal freedom - and other social and economic circumstances. The Human Freedom Index is an annually co-published report by the Cato Institute, the Fraser Institute, and the Liberales Institut at the Friedrich Naumann Foundation for Freedom.

In this lab, you'll be analyzing data from Human Freedom Index reports from 2008-2016. Your aim will be to summarize a few of the relationships within the data both graphically and numerically in order to find which variables can help tell a story about freedom.

Getting Started

Load packages

In this lab, you will explore and visualize the data using the **tidyverse** suite of packages. The data can be found in the companion package for OpenIntro resources, **openintro**.

Let's load the packages.

```
library(tidyverse)
library(openintro)
data('hfi', package='openintro')
```

The data

The data we're working with is in the openintro package and it's called `hfi`, short for Human Freedom Index.

1. What are the dimensions of the dataset?

- ANSWER: 1458 rows x 123 columns

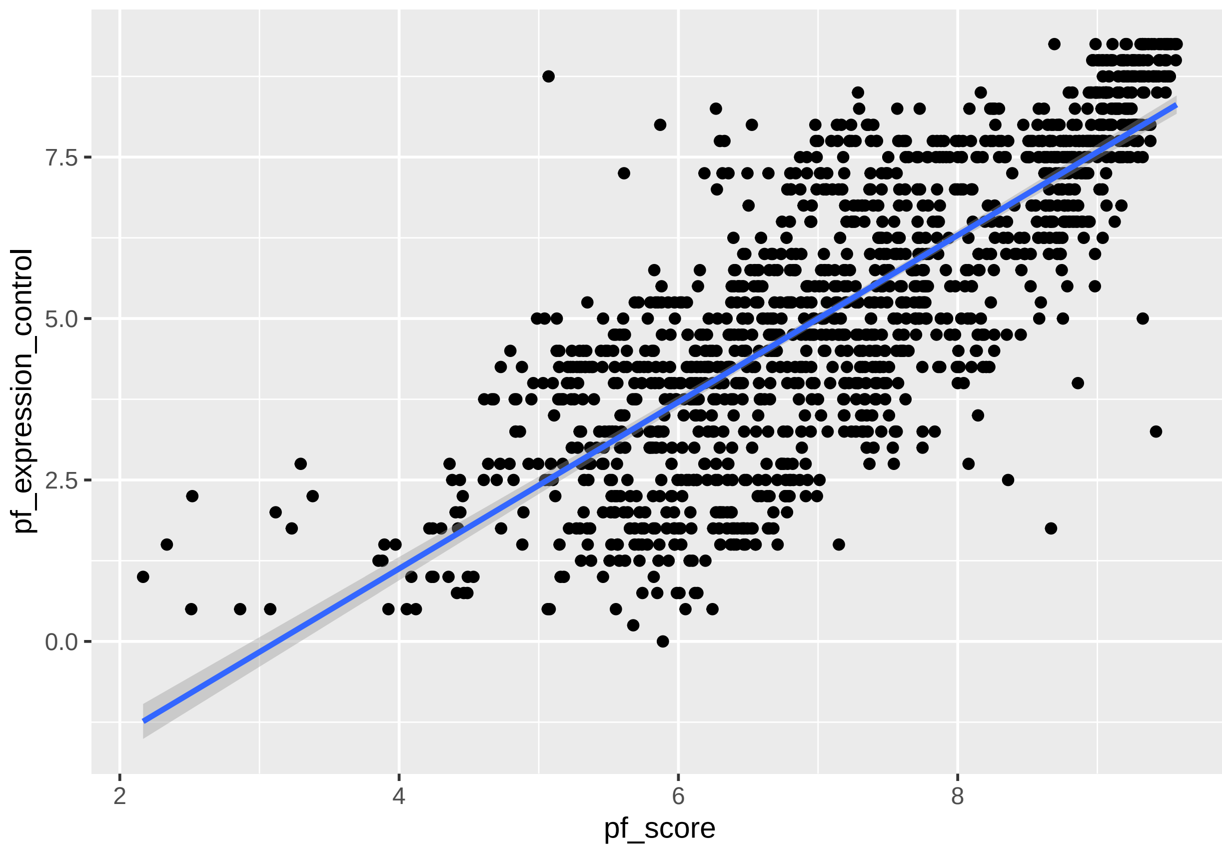
```
dim(hfi)
```

```
## [1] 1458  123
```

2. What type of plot would you use to display the relationship between the personal freedom score, `pf_score`, and one of the other numerical variables? Plot this relationship using the variable `pf_expression_control` as the predictor. Does the relationship look linear? If you knew a country's `pf_expression_control`, or its score out of 10, with 0 being the most, of political pressures and controls on media content, would you be comfortable using a linear model to predict the personal freedom score?

- ANSWER: Scatter plot. The relationship does appear mostly linear, as I create a line of best fit with narrow confidence intervals. However the residual sum of squares is somewhat large, with many data points far from the line of best fit. Therefore if I knew a country's `pf_expression_control` I would be mildly comfortable using a linear model to predict the personal freedom score.

```
ggplot(hfi, aes(x=pf_score, y=pf_expression_control)) +
  geom_point() +
  geom_smooth(method = "lm")
```



If the relationship looks linear, we can quantify the strength of the relationship with the correlation coefficient.

```
hfi %>%
  summarise(cor(pf_expression_control, pf_score, use = "complete.obs"))
```

```
## # A tibble: 1 x 1
##   `cor(pf_expression_control, pf_score, use = "complete.obs")`
##                                                          <dbl>
## 1                                                        0.796
```

Here, we set the `use` argument to "complete.obs" since there are some observations of NA.

Sum of squared residuals

In this section, you will use an interactive function to investigate what we mean by "sum of squared residuals". You will need to run this function in your console, not in your markdown document. Running the function also requires that the `hfi` dataset is loaded in your environment.

Think back to the way that we described the distribution of a single variable. Recall that we discussed characteristics such as center, spread, and shape. It's also useful to be able to describe the relationship of two numerical variables, such as `pf_expression_control` and `pf_score` above.

3. Looking at your plot from the previous exercise, describe the relationship between these two variables. Make sure to discuss the form, direction, and strength of the relationship as well as any unusual observations.

- ANSWER: The relationship between the two variables has a high positive correlation with 0.796. The form of this relationship is primarily linear. However there are a notable number of outliers which means there is a decent chance of error when using pf_expression_control values to predict pf_score values or vice versa.

Just as you've used the mean and standard deviation to summarize a single variable, you can summarize the relationship between these two variables by finding the line that best follows their association. Use the following interactive function to select the line that you think does the best job of going through the cloud of points.

```
# This will only work interactively (i.e. will not show in the knitted document)
hfi <- hfi %>% filter(complete.cases(pf_expression_control, pf_score))
DATA606::plot_ss(x = hfi$pf_expression_control, y = hfi$pf_score)
```

After running this command, you'll be prompted to click two points on the plot to define a line. Once you've done that, the line you specified will be shown in black and the residuals in blue. Note that there are 30 residuals, one for each of the 30 observations. Recall that the residuals are the difference between the observed values and the values predicted by the line:

$$e_i = y_i - \hat{y}_i$$

The most common way to do linear regression is to select the line that minimizes the sum of squared residuals. To visualize the squared residuals, you can rerun the plot command and add the argument `showSquares = TRUE`.

```
DATA606::plot_ss(x = hfi$pf_expression_control, y = hfi$pf_score, showSquares = TRUE)
```

Note that the output from the `plot_ss` function provides you with the slope and intercept of your line as well as the sum of squares.

4. Using `plot_ss`, choose a line that does a good job of minimizing the sum of squares. Run the function several times. What was the smallest sum of squares that you got? How does it compare to your neighbors?

- ANSWER: The lowest sum of squares I managed was 1415. However that was only after multiple attempts. My pretend neighbors (in this case my prior attempts) were around 1500-1600 sum of squares.

The linear model

It is rather cumbersome to try to get the correct least squares line, i.e. the line that minimizes the sum of squared residuals, through trial and error. Instead, you can use the `lm` function in R to fit the linear model (a.k.a. regression line).

```r
m1 <- lm(pf_score ~ pf_expression_control, data = hfi)
```

The first argument in the function `lm` is a formula that takes the form `y ~ x`. Here it can be read that we want to make a linear model of `pf_score` as a function of `pf_expression_control`. The second argument specifies that R should look in the `hfi` data frame to find the two variables.

The output of `lm` is an object that contains all of the information we need about the linear model that was just fit. We can access this information using the summary function.

```r
summary(m1)
```

```
##
## Call:
## lm(formula = pf_score ~ pf_expression_control, data = hfi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8467 -0.5704  0.1452  0.6066  3.2060
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            4.61707    0.05745   80.36   <2e-16 ***
## pf_expression_control  0.49143    0.01006   48.85   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8318 on 1376 degrees of freedom
##   (80 observations deleted due to missingness)
## Multiple R-squared:  0.6342, Adjusted R-squared:  0.634
## F-statistic:  2386 on 1 and 1376 DF,  p-value: < 2.2e-16
```

Let's consider this output piece by piece. First, the formula used to describe the model is shown at the top. After the formula you find the five-number summary of the residuals. The "Coefficients" table shown next is key; its first column displays the linear model's y-intercept and the coefficient of `pf_expression_control`. With this table, we can write down the least squares regression line for the linear model:

$$\hat{y} = 4.61707 + 0.49143 \times pf\_expression\_control$$

One last piece of information we will discuss from the summary output is the Multiple R-squared, or more simply, R^2. The R^2 value represents the proportion of variability in the response variable that is explained by the explanatory variable. For this model, 63.42% of the variability in runs is explained by at-bats.

5. Fit a new model that uses `pf_expression_control` to predict `hf_score`, or the total human freedom score. Using the estimates from the R output, write the equation of the regression line. What does the slope tell us in the context of the relationship between human freedom and the amount of political pressure on media content?

- ANSWER: For every increase of 1 in the political pressure score (pf_expression_control), there is an increase of 0.349862 in the total human freedom score (hf_score). This assumes a base total human freedom score (y intercept) of 5.153687 at a level of 0 political pressure score. Equation and model below.

$$\hat{y} = 5.153687 + 0.349862 \times pf\_expression\_control$$

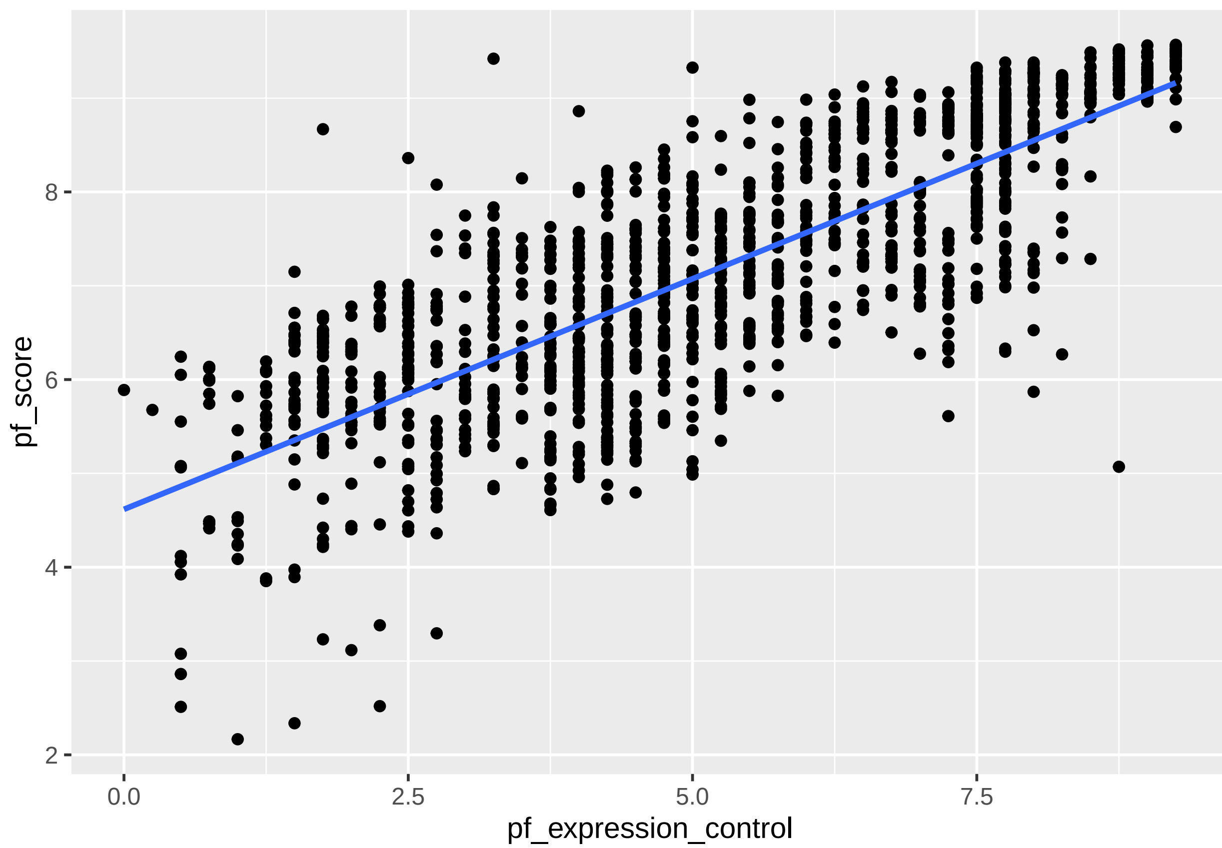
```
m2 <- lm(hf_score ~ pf_expression_control, data = hfi)
summary(m2)
```

```
##
## Call:
## lm(formula = hf_score ~ pf_expression_control, data = hfi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6198 -0.4908  0.1031  0.4703  2.2933
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           5.153687   0.046070  111.87   <2e-16 ***
## pf_expression_control 0.349862   0.008067   43.37   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.667 on 1376 degrees of freedom
##   (80 observations deleted due to missingness)
## Multiple R-squared:  0.5775, Adjusted R-squared:  0.5772
## F-statistic:  1881 on 1 and 1376 DF,  p-value: < 2.2e-16
```

Prediction and prediction errors

Let's create a scatterplot with the least squares line for `m1` laid on top.

```
ggplot(data = hfi, aes(x = pf_expression_control, y = pf_score)) +
  geom_point() +
  stat_smooth(method = "lm", se = FALSE)
```



Here, we are literally adding a layer on top of our plot. `geom_smooth` creates the line by fitting a linear model. It can also show us the standard error `se` associated with our line, but we'll suppress that for now.

This line can be used to predict y at any value of x. When predictions are made for values of x that are beyond the range of the observed data, it is referred to as *extrapolation* and is not usually recommended. However, predictions made within the range of the data are more reliable. They're also used to compute the residuals.

6. If someone saw the least squares regression line and not the actual data, how would they predict a country's personal freedom school for one with a 6.7 rating for `pf_expression_control`? Is this an overestimate or an underestimate, and by how much? In other words, what is the residual for this prediction?

ANSWER: A pf_expression_control of 6.7 would predict a personal freedom score of 8, which would be an overestimate Below I plug in the pf_expression_control of 6.7 into the m1 model, which calculates a pf_score of 7.909651. Therefore the residual is about 0.1.

$$\hat{y} = 4.61707 + 0.49143 \times pf\_expression\_control$$

```
4.61707 + (0.49143 * 6.7)
```

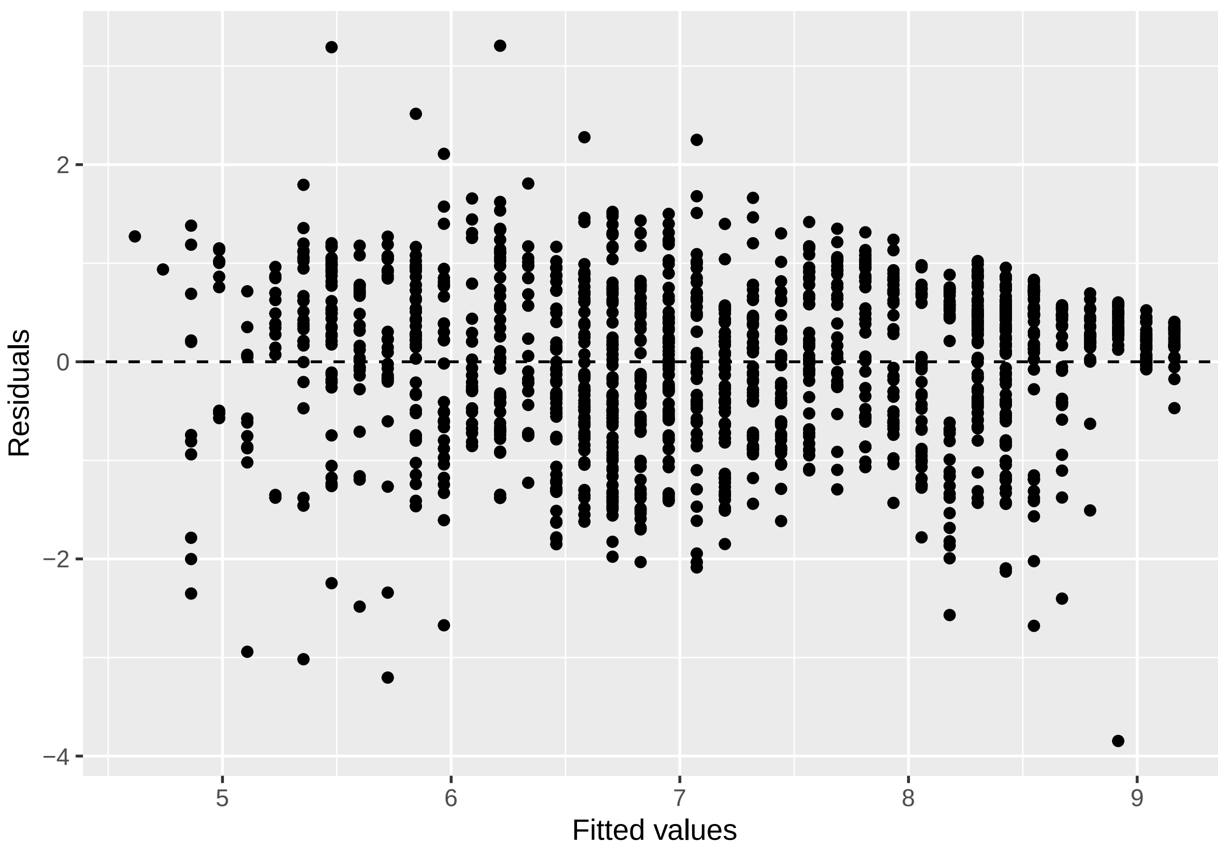
```
## [1] 7.909651
```

Model diagnostics

To assess whether the linear model is reliable, we need to check for (1) linearity, (2) nearly normal residuals, and (3) constant variability.

Linearity: You already checked if the relationship between `pf_score` and 'pf_expression_control' is linear using a scatterplot. We should also verify this condition with a plot of the residuals vs. fitted (predicted) values.

```
ggplot(data = m1, aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, linetype = "dashed") +
  xlab("Fitted values") +
  ylab("Residuals")
```



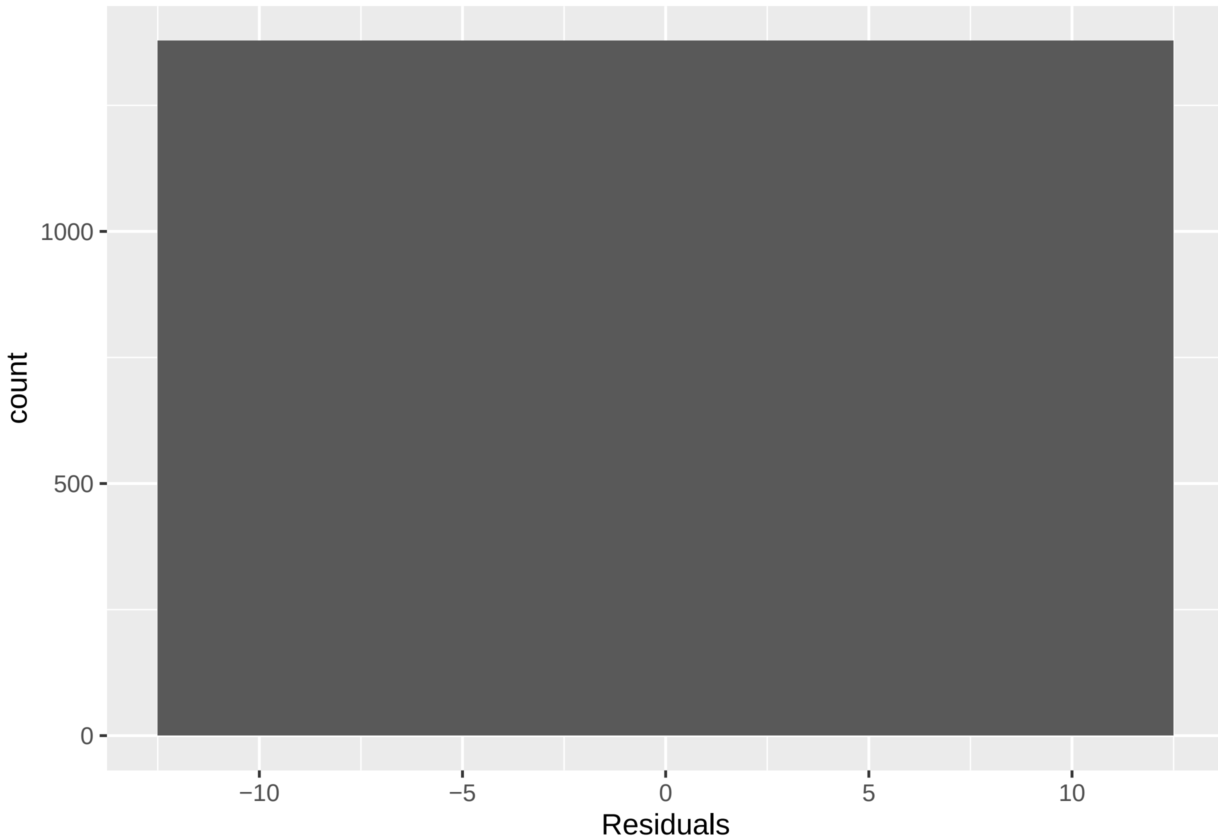
Notice here that `m1` can also serve as a data set because stored within it are the fitted values ($\hat{y}$) and the residuals. Also note that we're getting fancy with the code here. After creating the scatterplot on the first layer (first line of code), we overlay a horizontal dashed line at $y = 0$ (to help us check whether residuals are distributed around 0), and we also reanme the axis labels to be more informative.

7. Is there any apparent pattern in the residuals plot? What does this indicate about the linearity of the relationship between the two variables?

- ANSWER: There is no apparent pattern in the residuals plot, which indicates that the relationship is linear. The linear model is approximating the data points without favoring certain inputs.

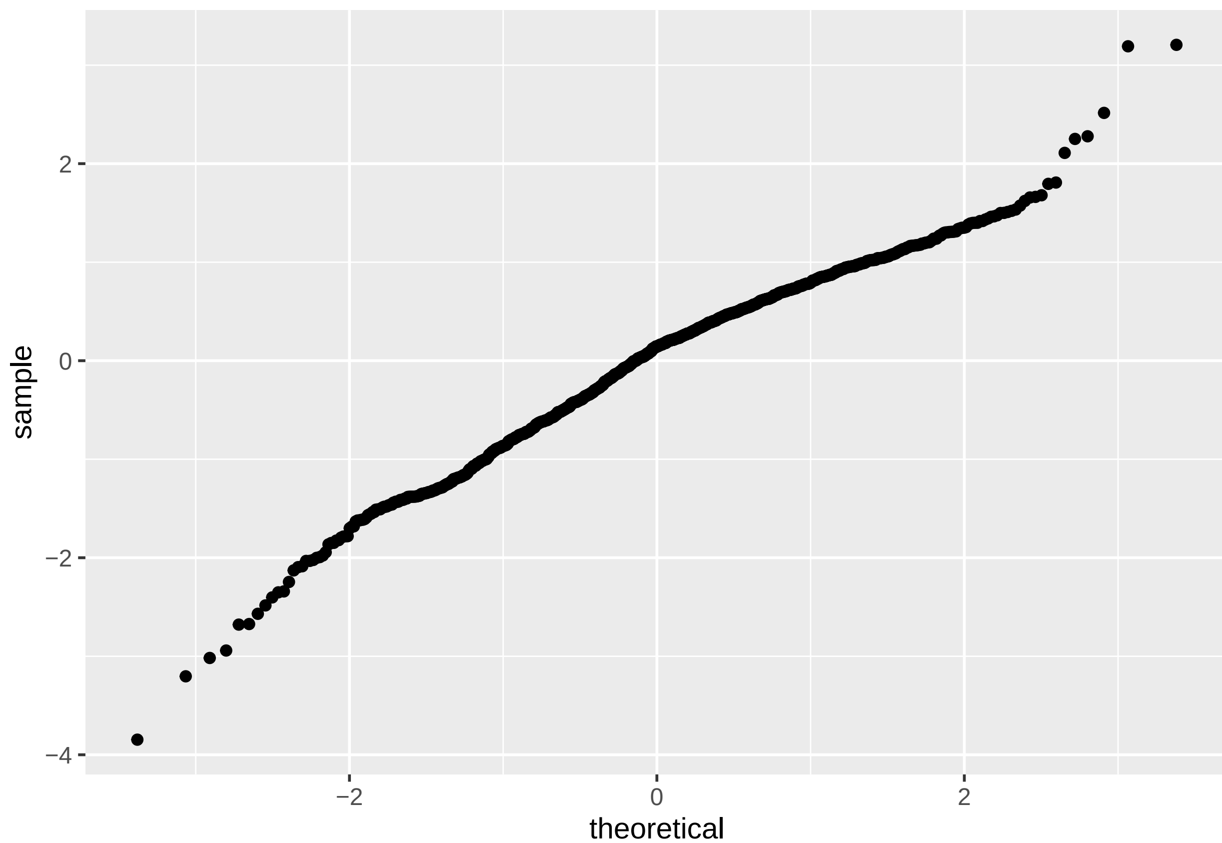
Nearly normal residuals: To check this condition, we can look at a histogram

```
ggplot(data = m1, aes(x = .resid)) +
  geom_histogram(binwidth = 25) +
  xlab("Residuals")
```



or a normal probability plot of the residuals.

```
ggplot(data = m1, aes(sample = .resid)) +
  stat_qq()
```

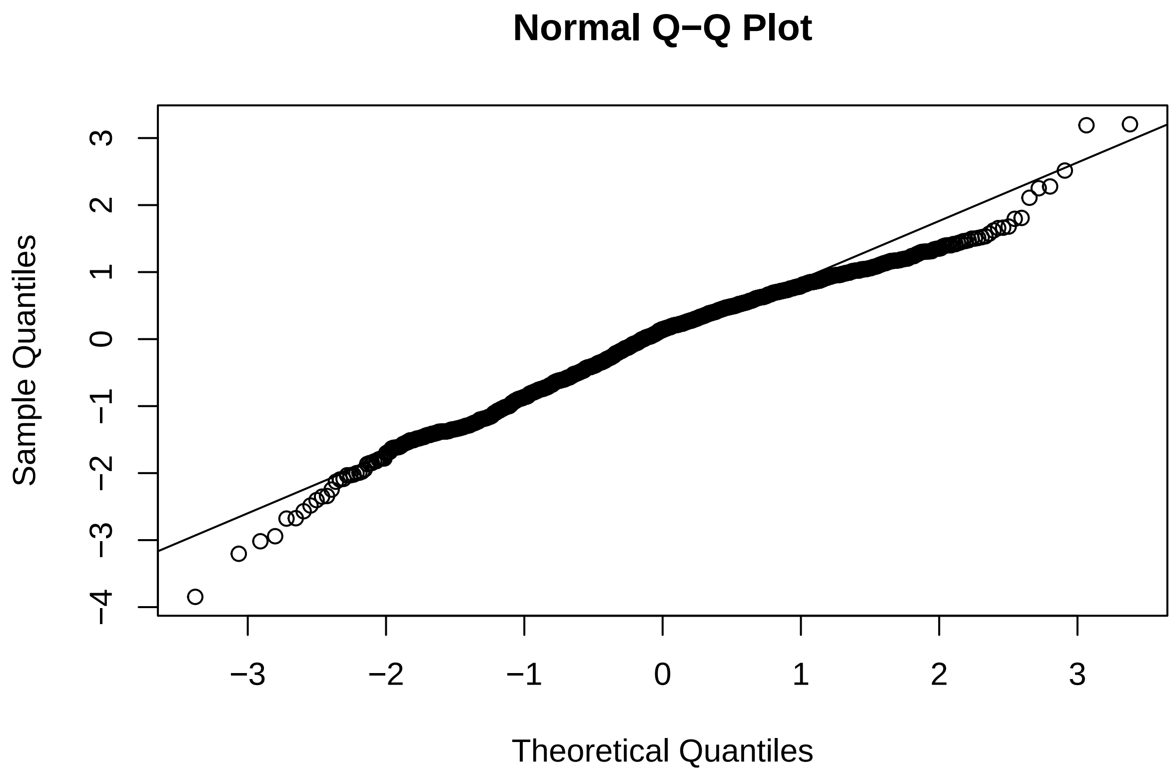


Note that the syntax for making a normal probability plot is a bit different than what you're used to seeing: we set `sample` equal to the residuals instead of `x`, and we set a statistical method `qq`, which stands for "quantile-quantile", another name commonly used for normal probability plots.

8. Based on the histogram and the normal probability plot, does the nearly normal residuals condition appear to be met?

- ANSWER: Yes, the *nearly* normal residuals condition appears to be met. Checked using below commands.

```
qqnorm(m1$residuals)
qqline(m1$residuals)
```



Constant variability:

9. Based on the residuals vs. fitted plot, does the constant variability condition appear to be met?

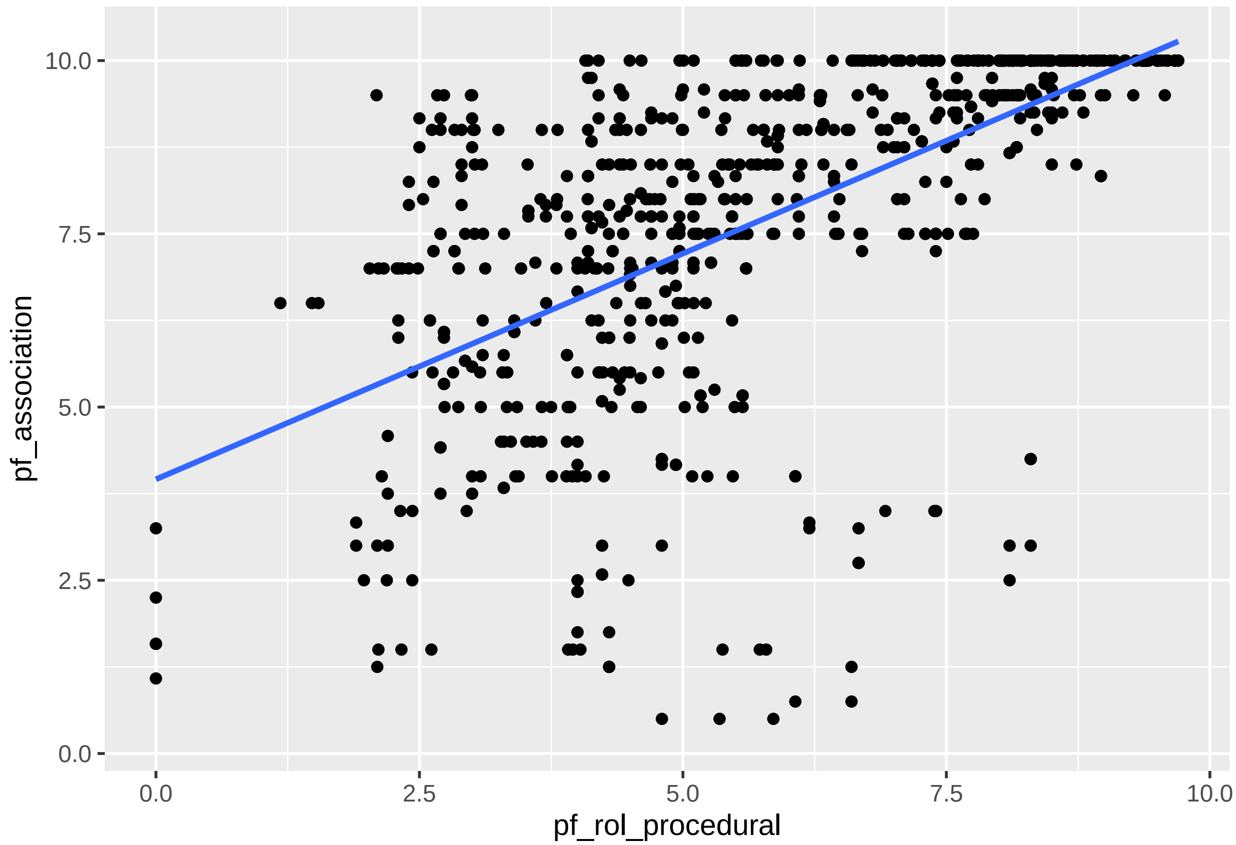
- ANSWER: The constant variability condition states that the variability of points around the least squares line should be roughly constant. When I plotted using qqnorm and qqline above, there was a roughly constant relationship between the two axes. Therefore, yes, the constant variability condition appears to be met.

More Practice

- Choose another freedom variable and a variable you think would strongly correlate with it. Produce a scatterplot of the two variables and fit a linear model. At a glance, does there seem to be a linear relationship?

The two variables I've chosen are `pf_association` (Freedom to associate and assemble with peaceful individuals or organizations) and `pf_rol_procedural` (procedural justice). Procedural justice is about how legal authorities (e.g. police) interact with the public and how those interactions shape the public's views of legal authorities. There is a general positive, weak but linear-ish relationship, however with higher residuals than Personal Freedom Score and Political Pressures and Controls on Media Content.

```
ggplot(hfi, aes(x=pf_rol_procedural, y=pf_association)) +
  geom_point() +
  geom_smooth(method = "lm", se=FALSE)
```



- How does this relationship compare to the relationship between `pf_expression_control` and `pf_score`? Use the R^2 values from the two model summaries to compare. Does your independent variable seem to predict your dependent one better? Why or why not?
- ANSWER: The adjusted R squared for the relationship between pf_expression_control and pf_score ("m1" variable) is 0.634 while the adjusted R squared for my variables pf_rol_procedural ~ pf_association ("m3") is 0.3495. Therefore pf_expression_control appears to predict pf_score better than pf_association predicts pf_rol_procedural. This is because my model has a higher residual standard error (1.867 on 795 degrees of freedom) than the m1 model (residual standard error of 0.8318 on 1376 degrees of freedom).

```
m3 <- lm(pf_association ~ pf_rol_procedural, data = hfi)
summary(m3)
```

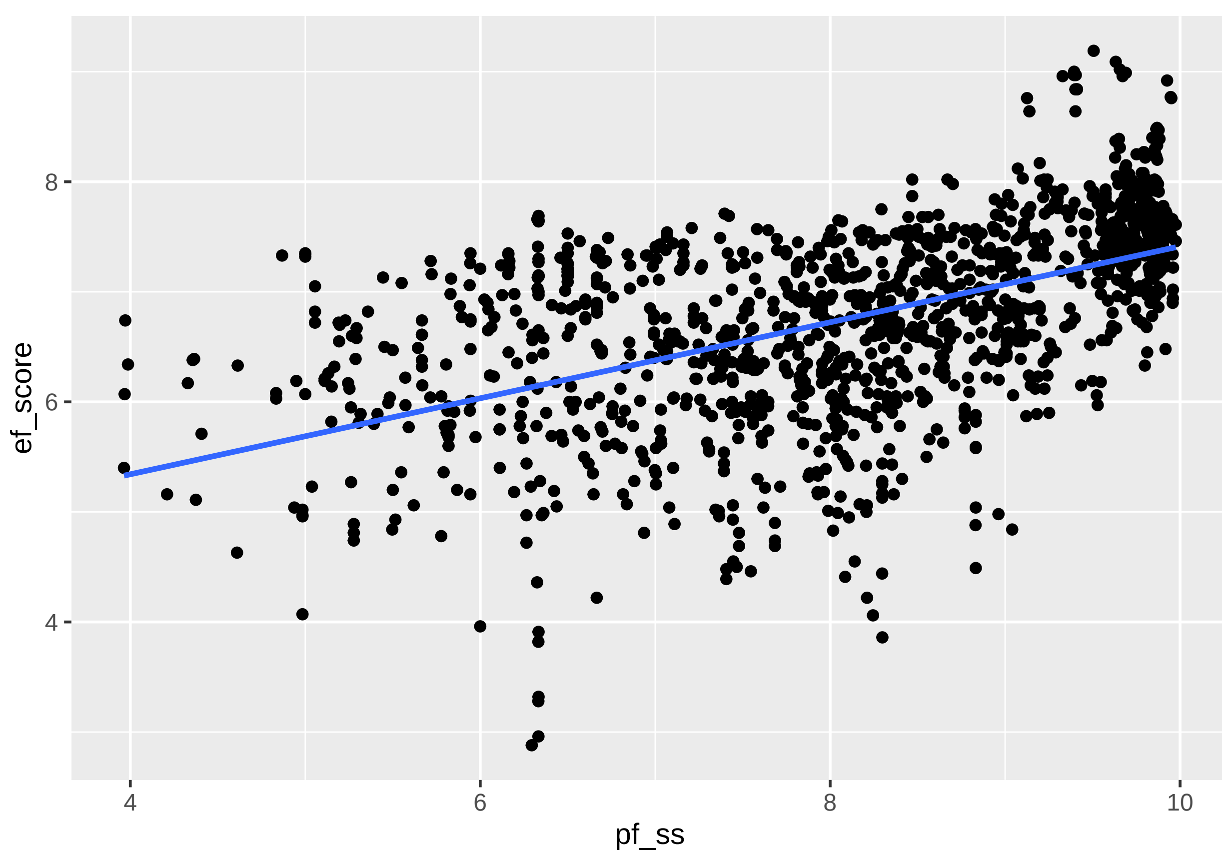
```
##
## Call:
## lm(formula = pf_association ~ pf_rol_procedural, data = hfi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -7.5075 -0.6060  0.3092  1.0782  4.1787
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.95747    0.19103   20.72   <2e-16 ***
## pf_rol_procedural  0.65152    0.03147   20.70   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.867 on 795 degrees of freedom
##   (661 observations deleted due to missingness)
## Multiple R-squared:  0.3503, Adjusted R-squared:  0.3495
## F-statistic: 428.6 on 1 and 795 DF,  p-value: < 2.2e-16
```

- What's one freedom relationship you were most surprised about and why? Display the model diagnostics for the regression model analyzing this relationship.

ANSWER: - I was surprised about the mild, positive relationship (adjusted R squared of 0.3495) between the score for security and safety **pf_ss** with economic freedom's score **ef_score**. I assumed security and safety would be more predictive of economic freedom: Countries that are pro-business need to ensure a safe country to help that business happen. Model diagnostics below.

```
ggplot(hfi, aes(x=pf_ss, y=ef_score)) +
  geom_point() +
  geom_smooth(method = "lm", se=FALSE)
```

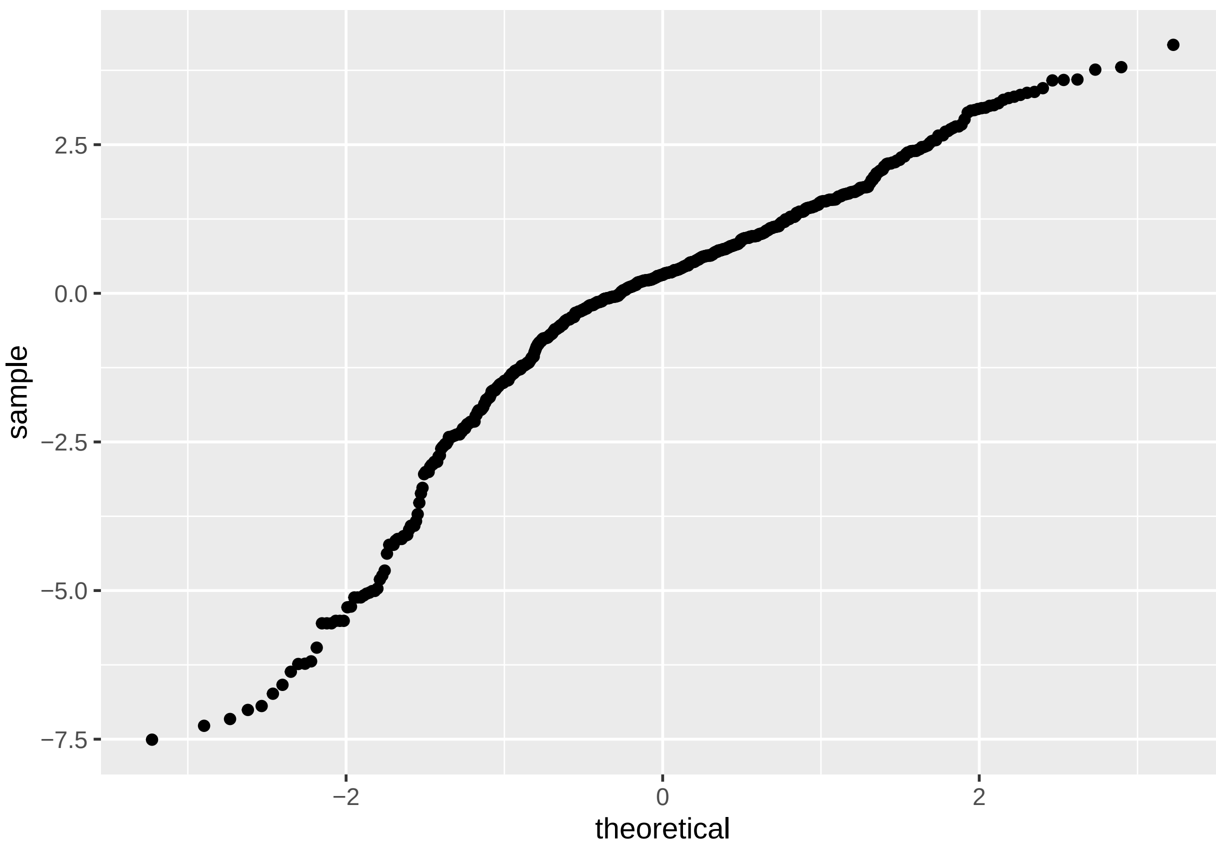


```
m4 <- lm(pf_association ~ pf_rol_procedural, data = hfi)
summary(m4)
```

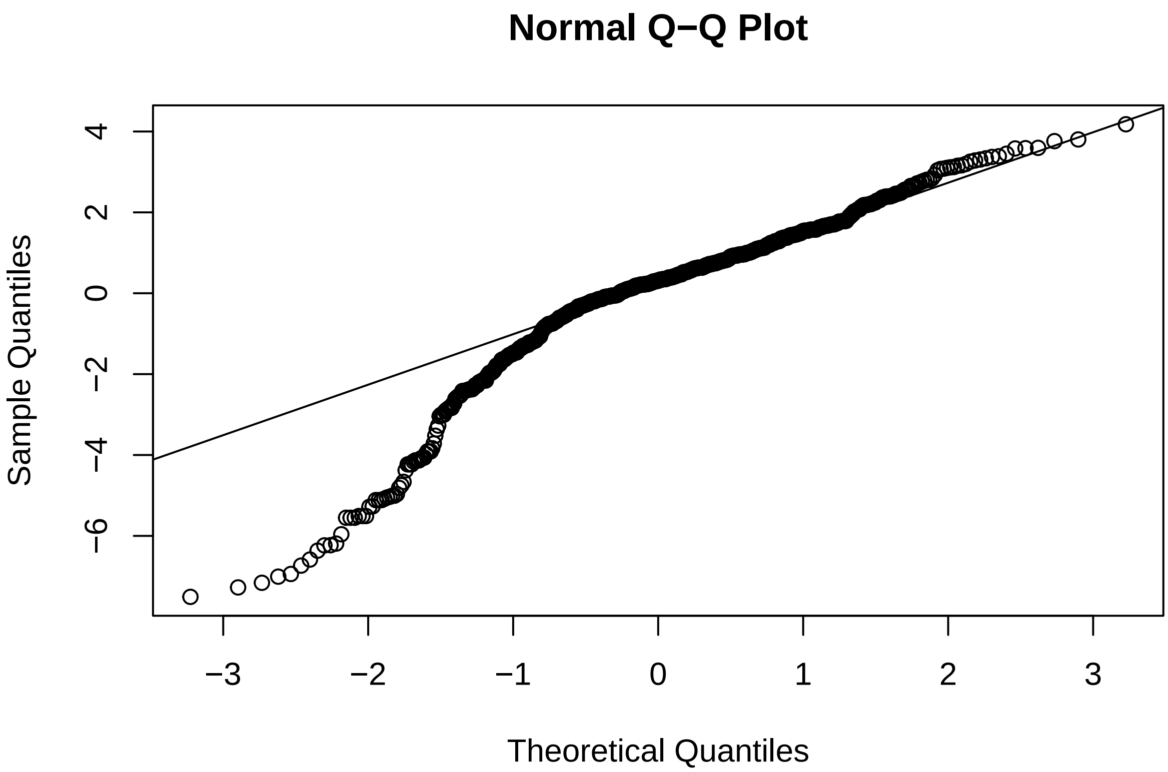
```
##
## Call:
## lm(formula = pf_association ~ pf_rol_procedural, data = hfi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.5075 -0.6060  0.3092  1.0782  4.1787
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.95747    0.19103   20.72   <2e-16 ***
## pf_rol_procedural  0.65152    0.03147   20.70   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.867 on 795 degrees of freedom
##   (661 observations deleted due to missingness)
## Multiple R-squared:  0.3503, Adjusted R-squared:  0.3495
## F-statistic: 428.6 on 1 and 795 DF,  p-value: < 2.2e-16
```

Assessing normal probability plot of the residuals.

```
ggplot(data = m4, aes(sample = .resid)) +
  stat_qq()
```




```
qqnorm(m4$residuals)
qqline(m4$residuals)
```



The *nearly* normal residuals condition *is not* met. The constant variability condition states that the variability of points around the least squares line should be roughly constant. When I plotted using qqnorm and qqline above, there was a somewhat constant but inconsistent relationship between the two axes. Therefore, no, the constant variability condition wasn't met.
