Lab8-Introduction to linear regression

Sean Connin

04/12/21

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)
```

Creating a reproducible lab report

To create your new lab report, in RStudio, go to New File -> R Markdown… Then, choose From Template and then choose `Lab Report for OpenIntro Statistics Labs` from the list of templates.

The data

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

Exercise 1 What are the dimensions of the dataset?

```
#help(package = openintro)

(dm<-dim(hfi)) # --> 1458, 123
```

The dataset has 1458 rows and 123 columns.

Exercise 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?

Since we are evaluating the relationship between two numerical values, I would use a scatterplot.

```
Plot this relationship using the variable `pf_expression_control` as the predictor.
```

```
#glimpse(hfi)

plt <- hfi%>%select(pf_expression_control, pf_score)

ggplot(data = plt)+
  geom_point(aes(x = pf_expression_control, y = pf_score))
```

```
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 polit
ical pressures and controls on media content, would you be comfortable using a linear model to predict the personal freedom score?
```

The relationship between pf_expression_control and pf_expression looks approximately linear but it may also be curvilinear.

I would want to assess the correlation coefficient and R-square value of the linear model before using it to predict the personal freedom score.

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

```
plt %>%
  summarise(cor(pf_expression_control, pf_score, use = "complete.obs"))  #---> 0.796, complete.obs performs listwise deletion of missing da
        ta
```

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.

Exercise 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.

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.

```
# altered code to subset and remove na values

plt<-hfi%>%select(pf_expression_control, pf_score)%>%na.omit()

DATA606::plot_ss(x =plt$pf_expression_control, y = plt$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 =plt$pf_expression_control, y = plt$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.

Exercise 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?

The smallest SS that I computed was 963.095. This value is consistent with other SS estimates shared by classmates in Slack (e.g., 981, 976).

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).

```
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.

```
summary(m1)
```

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 `at_bats`. 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.

Exercise 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?

$\hat{y} = 4.6171 + 0.4914 \times pf\_expression\_control$

The slope (0.4914) tells us that there is a positive relationship between hf_score and pf_expression_control. And that for a unit increase in the latter, hf_score increases by 0.49.

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.

Exercise 6 If someone saw the least squares regression line and not the actual data, how would they predict a country's personal freedom score 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?

To make a prediction and assess the residual for this prediction absent the actual data, we would need the actual equation for the least squares regression line - including the error term.

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.

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

There does not appear to be a pattern in the residuals plot, which is consistent with our condition for linearity.

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

```
ggplot(data = m1, aes(x = .resid)) +
  geom_histogram(binwidth = 0.25) +
  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.

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

Yes, from the histogram and normal probability plot it appears that the residuals approximate a normal model.

Constant variability:

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

Yes, constant variability has been met.

More Practice

1. 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?

```
glimpse(hfi)
help(package = openintro)

#compare military interference with freedom of expression

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

```
# create linear model for this relationship

model <- lm(pf_expression ~ ef_legal_military, data = hfi)

# return model summary

summary(model)
```

1. 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?

The relationship between pf_expression and ef_legal_military n is weaker that that for pf_expression_control and pf_score. The multiple R-square for the the first pair is only 0.26 (due to high variability in dependent variable about the least squares line, indicating that the model only accounts for 26% of the variability in pf_expression. Compare that to 0.63 for pf_expression_control and pf_score.

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

I expected that there would be a strong relationship between the reliability of a country's police force (ef_legal_police) and women's sense of security (pf_ss_women). However, this was not the case (see scatterplot - below). It's also worth noting that the conditions for a linear regression model were not been met given that the residuals for the dependent variable were highly skewed (see histogram and qqplot - below).

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

```
model <- lm(pf_ss_women ~ ef_legal_police, data = hfi)

# return model summary

summary(model)

#check conditions for linear model

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

```
ggplot(data = model, aes(x = .resid)) +
  geom_histogram(binwidth = 0.25) +
  xlab("Residuals")
```

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