CM 2.1-2 - Simple linear regression - Solutions

Math 530/630

Contents

Overview

- A complete knitted `html` file is due on Sakai by the beginning of the next class.
- This lab is based on Chapter 5: Basic Regression in ModernDive. Please open it and follow closely!
- You'll need to load these packages to do the lab (make sure they are installed first, not in your .Rmd file!):

```
library(moderndive)
library(tidyverse)
library(skimr)
```

The data

Source: John Clay (1856). "On the Relation Between Crime, Popular Instruction, Attendance on Religious Worship, and Beer-Houses", Journal of the Statistical Society of London, Vol. 20 #1, pp 22-32.

In 1856, the Reverend John Clay felt that it was high time to figure out what societal factors were playing a role in the incidence of criminal behavior in Britain. He stated that:

> "It is a mere truism to say that the progress of popular education, and the formation of religious habits, are fatally opposed by the temptations to animal pleasures, which abound wherever BEER-HOUSES and low ALE-HOUSES abound."

Clearly, the reverend considered public houses in Britain to be a scourge on society, namely that they "promote drunkenness and its consequent evil" (i.e., crime). Let's investigate how well we can predict criminals (per 100k population) from the number of public houses (ale/beer houses per 100k population) using simple linear regression.

You'll first need to download the data from the Reference Plus (scroll down to Data Sets) page on the course website into a "data" directory in your RStudio project. Assuming that, here is how to read in the data into the crime object:

```
crimenames <- c("county", "region_name", "region_code",
                "criminals", "public_houses", "school_attendance",
                "worship_attendance")

# The below assumes that you've downloaded the `beerhall.dat` file to your RStudio project, into a dire
crime <- read_table(here::here("data", "beerhall.dat"),
                    col_names = crimenames)
```

```
crime <- read_table("https://ohsu-math630-fall-2019.netlify.com/data/beerhall.dat",
                    col_names = crimenames)
```

Basics

Note: you don't need to use R to answer these questions, but please create a section header using markdown format (`# Basics`) and type your answers there.

- What is the dependent variable?
- What is the independent variable?

- Copy and paste the provided equation that starts/ends with `$$` into your narrative (not an R code chunk), and replace `y` and `x` in this formula with meaningful variable names (you may wish to reference the `crimenames` object we made above): `$$\hat{y} = b_0 + b_1{x}$$`

$$\hat{criminals} = b_0 + b_1 public\_houses$$

- The "best-fitting" regression line is "best" in that it minimizes what?

  The best-fitting regression is one that minimizes the sum of the squared residuals. Squared residuals are calculated by taking the difference between the observed y values (here, criminals) and the fitted or predicted y values (so y hat!), then squaring each of those differences. The differences are squared so that positive and negative deviations of the same amount are treated equally.

- Why is this method called "simple linear regression" (as opposed to the method in Chapter 6)?

  Linear regression refers to the form of the statistical model we are using- in linear regression, our model is in the form of a line, defined by an intercept and slope. The "simple" part refers to the fact that we have only one predictor variable (or independent variable/IV). In multiple regression, we'll have two or more predictor variables or IVs. All of these cases of regression are univariate, as opposed to multivariate. Univariate vs. multivariate refers to the number of dependent variables or DVs. In this course we will not cover multivariate methods.

EDA

Conduct a new exploratory data analysis, which involves three things:

- Looking at the raw values.
- Computing summary statistics of the variables of interest.
- Creating informative visualizations.

Look at the data

Use `dplyr` to figure out how many counties are in this dataset, and which variable names map onto the independent and dependent variables you identified above.

```
glimpse(crime)
```

```
Observations: 40
Variables: 7
$ county             <chr> "Middlesex", "Surrey", "Kent", "Sussex", "H...
$ region_name        <chr> "South Eastern", "South Eastern", "South Ea...
$ region_code        <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2...
$ criminals          <dbl> 200, 160, 160, 147, 178, 205, 183, 156, 173...
$ public_houses      <dbl> 541, 504, 552, 295, 409, 568, 708, 624, 463...
$ school_attendance  <dbl> 560, 630, 790, 820, 990, 930, 1020, 1130, 9...
$ worship_attendance <dbl> 434, 482, 680, 678, 798, 698, 888, 970, 848...
```

Look at summary statistics

Use `select` to select only the independent and dependent variables you identified above, then pipe those variables to the `skim` function from the `skimr` package (you should have loaded this package at the top) to

see summary statistics for each. Use `dplyr::summarize` to calculate the correlation coefficient.

```
crime %>%
  select(criminals, public_houses) %>%
  skim()
```

```
Skim summary statistics
 n obs: 40
 n variables: 2

-- Variable type:numeric -------------------------------------------------------------------------
      variable missing complete  n   mean     sd p0 p25   p50    p75 p100
     criminals       0       40 40 152.9   41.42 66 127 157.5 174.25  241
 public_houses       0       40 40 374.85 164.97 87 209 407   490.75  708
     hist
 <U+2582><U+2582><U+2583><U+2585><U+2587><U+2582><U+2583><U+2581>
 <U+2583><U+2586><U+2582><U+2582><U+2587><U+2585><U+2583><U+2582>
```

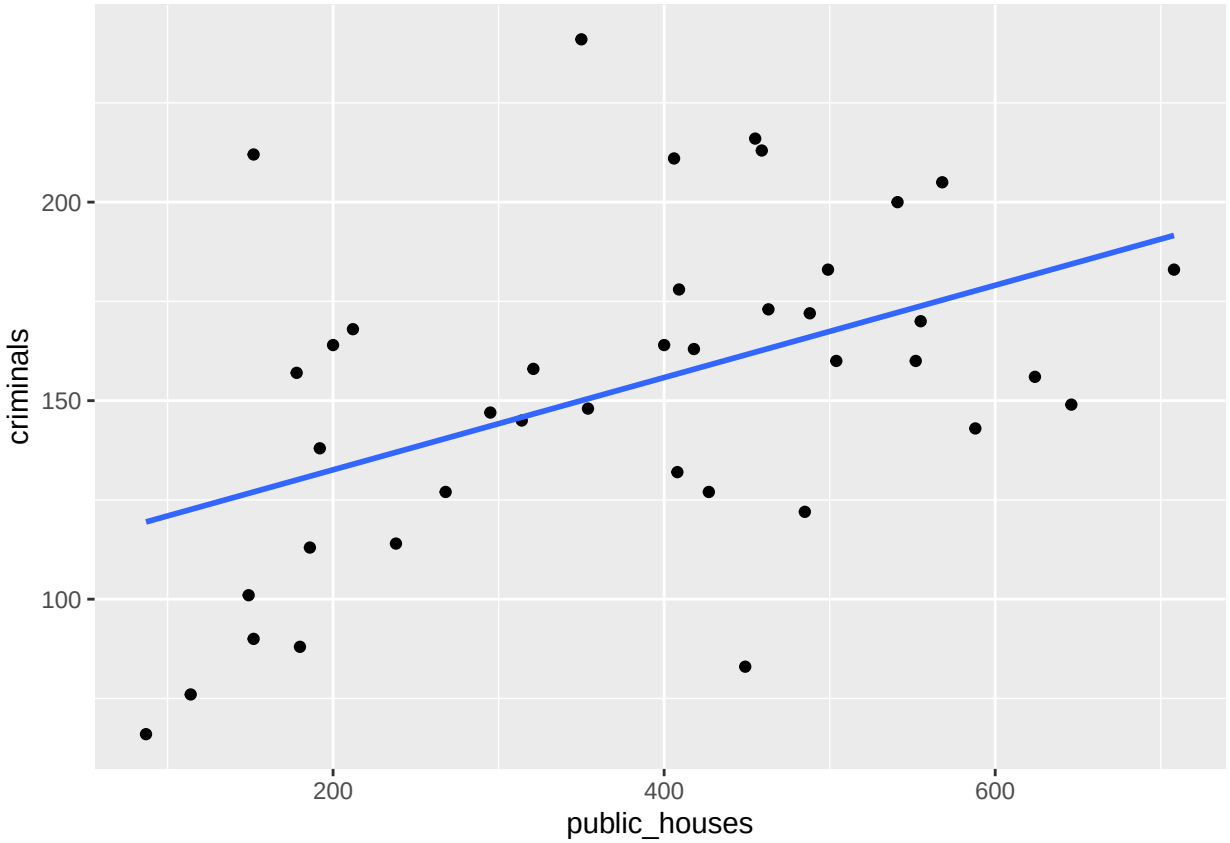
```
crime %>%
  summarize(corr = cor(criminals, public_houses))
```

```
# A tibble: 1 x 1
   corr
  <dbl>
1 0.463
```

Visualize the data

Recreate the scatterplot below of ale/beer houses per 100K on the x-axis and criminals per 100K population on the y-axis. What can you say about the relationship between public houses and criminals based on this exploration?

```
ggplot(crime, aes(x = public_houses, y = criminals)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE)
```



Do this in three sentences: There seems to be a moderately strong relationship. The slope of the regression line is positive, also the correlation coefficient of 0.46 is moderate. Meaning as public houses increase, there is an associated increase in criminal activity.

Simple linear regression: (see here)

Part 1: First "fit" the linear regression model to the data using the `lm()` function, then apply the `get_regression_table()` function from the `moderndive` R package to the model object. Use the output to fill in this formula with `y`, `x`, and the intercept and slope coefficients: (copy and paste into your narative: `$$\hat{y} = b_0 + b_1{x}$$`)

$$\hat{y} = b_0 + b_1 x$$

Part 2: Interpret the intercept coefficient and the slope coefficient. How do the regression results match up with the results from your exploratory data analysis above?

```
crime_lm <- lm(criminals ~ public_houses, data = crime)
get_regression_table(crime_lm)
```

```
# A tibble: 2 x 7
  term          estimate std_error statistic p_value lower_ci upper_ci
  <chr>            <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept      109.       14.8        7.41   0       79.5    139.
2 public_houses    0.116     0.036      3.22   0.003    0.043    0.189
```

The intercept anchors the linear model, but is uninterprettable as there are no counties with 0 public house in the data. Both the slope coefficient and the correlation coefficient suggest a positive relationship beween the independent and dependent variable, with the slope showing the expected increase in criminals per 100K population (0.116) as the number of public houses per 100K population increases. Conceptually, the correlation cofficient shows the strength of the linear association between the variables, whereas the slope coefficient shows the expected, on average, magnitude of change in the dependent variable given a unit change in the independent variable.

Observed/fitted values: (see here)

Part 1: What are the observed and fitted values for the Cornwall (`ID` = 20) and Monmouth regions (`ID` = 23)? Which region do you think the reverend called "the happiest example of the infrequency of crime"?

```
regression_points <- get_regression_points(crime_lm)
regression_points %>%
  filter(ID %in% c(20, 23))
```

```
# A tibble: 2 x 5
     ID criminals public_houses criminals_hat residual
  <int>     <dbl>         <dbl>         <dbl>    <dbl>
1    20        66            87          119.    -53.4
2    23       241           350          150.     91.0
```

The reverend would have used Cornwall as an example, as it has the least criminals and public houses per capita.

Part 2: In fact, we could argue with the reverend about the happiest example of the infrequency of crime. There are two ways you could define this. The first way is that the county had the lowest criminals overall. The second is that the county had the lowest criminals *given their number of public houses.* This would mean that the region has the lowest observed criminals, compared to the fitted value based on predicting criminals from public houses.

Filter the regression ouput using `filter` and the `|` operator (think: `or`) to extract the two alternative definitions above. You'll need to match the `ID` column as a row in the original data (you can just do this visually by comparing the tibbles and typing your answers in your narrative- you don't have to do a join here).

```
# Derby is lowest residual (ID = 20)
# Cornwall is the lowest observed (ID = 33)
regression_points %>%
  filter(criminals == min(criminals) | residual == min(residual))
```

```
# A tibble: 2 x 5
     ID criminals public_houses criminals_hat residual
  <int>     <dbl>         <dbl>         <dbl>    <dbl>
1    20        66            87          119.    -53.4
2    33        83           449          162.    -78.5
```

```
regression_points %>%
  left_join(select(crime, county, criminals, public_houses)) %>%
  filter(criminals == min(criminals) | residual == min(residual))
```

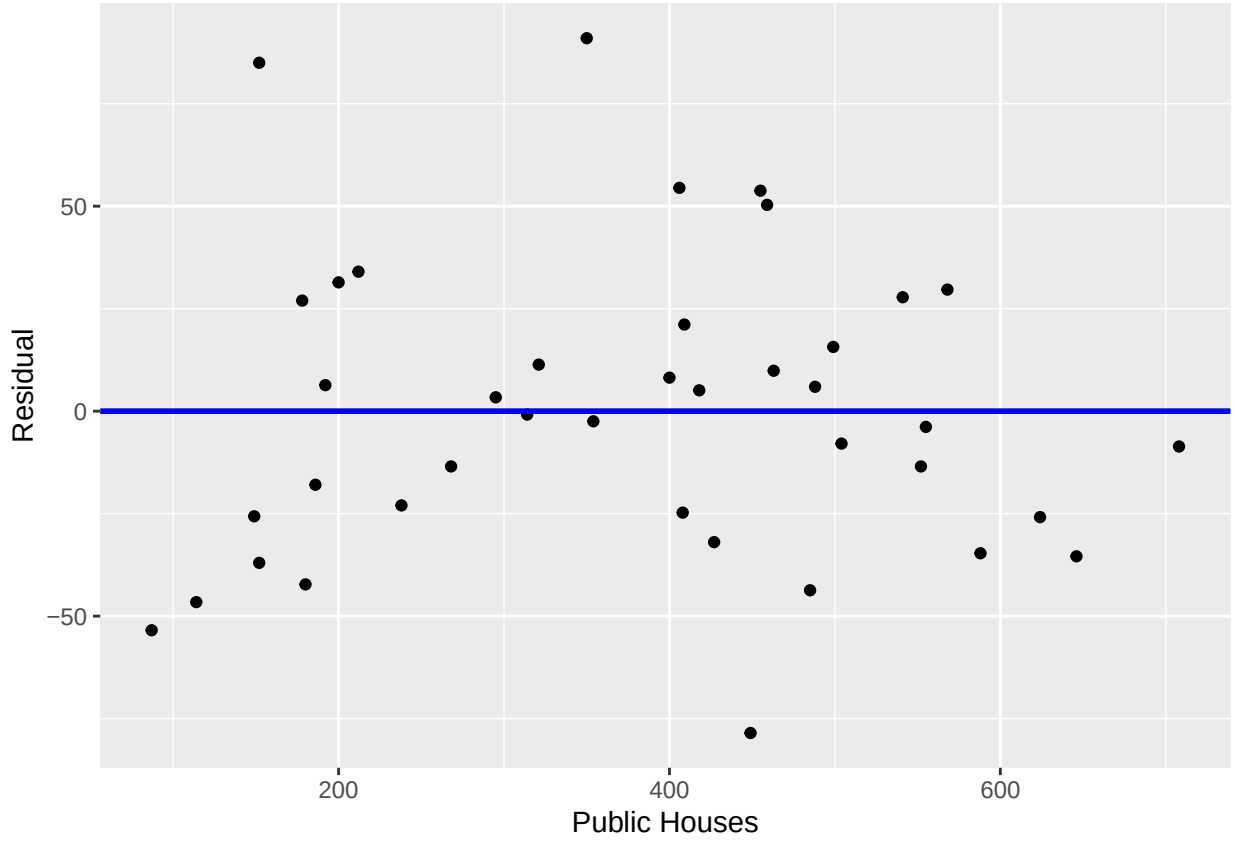
```
# A tibble: 2 x 6
     ID criminals public_houses criminals_hat residual county
  <int>     <dbl>         <dbl>         <dbl>    <dbl> <chr>
1    20        66            87          119.    -53.4 Cornwall
```

```
2    33        83           449          162.    -78.5 Derby
```

Residual analysis: (see here)

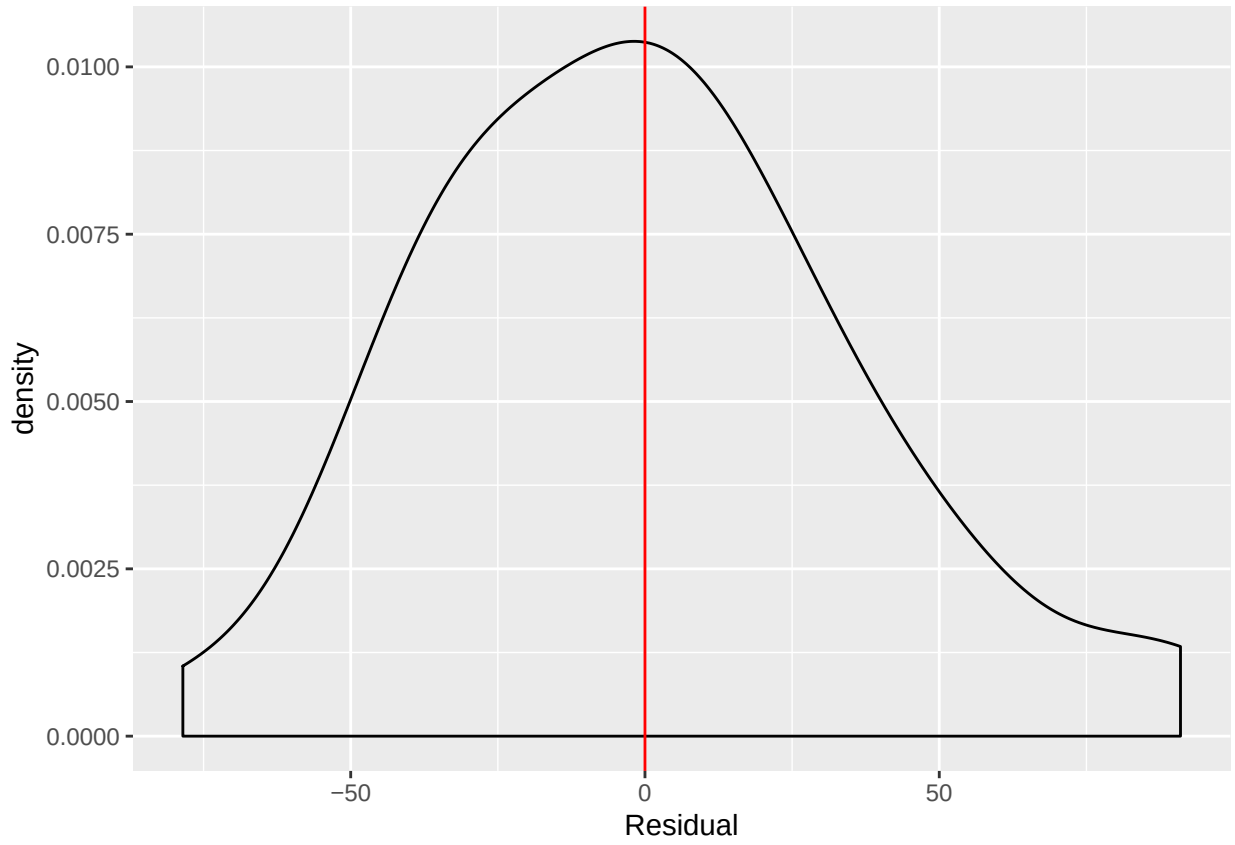
Part 1: Perform a residual analysis and look for any systematic patterns in the residuals. Ideally, there should be little to no pattern- why? Does it seem like there is no systematic pattern to the residuals?

```
ggplot(regression_points, aes(x = public_houses, y = residual)) +
  geom_point() +
  labs(x = "Public Houses", y = "Residual") +
  geom_hline(yintercept = 0, col = "blue", size = 1)
```



A visible pattern in the residuals suggests that the model does not do a good job of explaining the relationship. For this linear model, there is no clear systematic pattern.

Part 2: Recreate this density plot of the residuals. Recall that we would like the residuals to be normally distributed with mean 0 (hint: `?geom_vline()`). Use `dplyr` to calculate the mean of the residuals- is it (pretty close to) 0? Do you think you have more positive residuals than negative, or vice versa?




```
regression_points %>%
  summarize(mean_resid = mean(residual))
```

```
# A tibble: 1 x 1
  mean_resid
       <dbl>
1   0.000025
```

The mean residual is very close to zero, and has more positive residuals than negative (slightly right skewed).