Chapter 10: Bivariate Analysis

Answers to Exercises

Brian Fogarty

Contents

Exercise 1

Exercise 1.a

We need to first re-order the values in `vfalter`.

```
library(tidyverse)

vf_england <- read_csv("VF England.csv")

vf_england <- vf_england %>%
                mutate(vfalter1 = factor(vfalter,
                        levels = c("Strongly disagree","Disagree",
                          "Slightly disagree","Neither agree nor disagree",
                          "Slightly agree","Agree","Strongly agree")))

vf_england %>%
  count(vfalter1)
```

```
# A tibble: 7 x 2
  vfalter1                      n
  <fct>                     <int>
1 Strongly disagree            95
2 Disagree                    253
3 Slightly disagree           266
4 Neither agree nor disagree  769
5 Slightly agree              364
```

```
6 Agree                          166
7 Strongly agree                 121
```

```
library(janitor)

vf_england %>%
  filter(!is.na(vote2017_dum)) %>%
  tabyl(vfalter1, vote2017_dum) %>%
  adorn_percentages("col") %>%
  adorn_pct_formatting(digits = 1) %>%
  adorn_ns()
```

```
                  vfalter1       Loser      Winner
          Strongly disagree  5.5%  (51)  4.5%  (34)
                   Disagree 14.0% (129) 12.9%  (98)
          Slightly disagree 13.9% (128) 12.3%  (93)
 Neither agree nor disagree 37.6% (346) 32.1% (243)
             Slightly agree 17.5% (161) 20.2% (153)
                      Agree  7.2%  (66) 10.0%  (76)
             Strongly agree  4.1%  (38)  7.9%  (60)
```

The relationship between `vfalter` and `vote2017_dum` appears similar to what we saw between `vfproblem` and `voter2017_dum` in the chapter. Overall, it appears that a greater percentage of respondents who voted for a candidate from a losing party disagreed, at some level, or were neutral on whether there's enough voter fraud to alter election outcomes in the UK. Meanwhile, a greater percentage of respondents who voted for a candidate from the winning party agreed, at some level, that there's enough voter fraud to alter election outcomes in the UK.

Exercise 1.b

```
vf_england %>%
  filter(!is.na(vote2017_dum)) %>%
  tabyl(vfalter1, vote2017_dum) %>%
  chisq.test()
```

```
    Pearson's Chi-squared test

data:  .
X-squared = 21.578, df = 6, p-value = 0.001444
```

Since $p \leq 0.05$, there is a statistically significant relationship between respondents' 2017 vote and the agreement level that there's enough voter fraud to alter election outcomes in the UK.

Exercise 1.c

We can look at a measure of association since the relationship is statistically significant. As `vote2017_dum` is a nominal-level variable, we will use Cramer's V.

```
library(DescTools)

CramerV(vf_england$vote2017_dum, vf_england$vfalter1)
```

```
[1] 0.1134669
```

Cramer's V is 0.113, which indicates there is a weak relationship or association between respondents' vote choice in 2017 and the agreement level that there's enough voter fraud to alter election outcomes in the UK.

Exercise 2

Exercise 2.a

We will first multiply `Income_rate`, `Employment_rate`, and `not_participating` by 100 to make them proper percentages. This is not required for the correlation analysis, but it's good practice.

```
simd <- read_csv("simd2020.csv", na = "*")

simd <- simd %>%
  mutate(Income_rate = Income_rate*100,
         Employment_rate = Employment_rate*100,
         not_participating = not_participating*100)

cor.test(simd$crime_rate,simd$Income_rate, na.action = na.rm)
```

```
    Pearson's product-moment correlation

data:  simd$crime_rate and simd$Income_rate
t = 29.633, df = 6473, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.3239853 0.3668853
sample estimates:
      cor
0.3456159
```

Since $p \leq 0.05$, we conclude that there is a statistically significant correlation between datazones' percentage of income deprivation and crime rate. We find a correlation of 0.346, which indicates a moderately weak, positive relationship between income deprivation and crime rates in Scottish datazones.

Exercise 2.b

```
cor.test(simd$crime_rate,simd$Employment_rate, na.action = na.rm)
```

```
    Pearson's product-moment correlation

data:  simd$crime_rate and simd$Employment_rate
t = 31.623, df = 6473, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.3445180 0.3867185
sample estimates:
      cor
0.3658063
```

Since $p \leq 0.05$, we conclude that there is a statistically significant correlation between datazones' percentage of employment deprivation and crime rate. We find a correlation of 0.366, which indicates a moderately weak, positive relationship between employment deprivation and crime rates in Scottish datazones.

Exercise 2.c

```
cor.test(simd$crime_rate,simd$not_participating, na.action = na.rm)
```

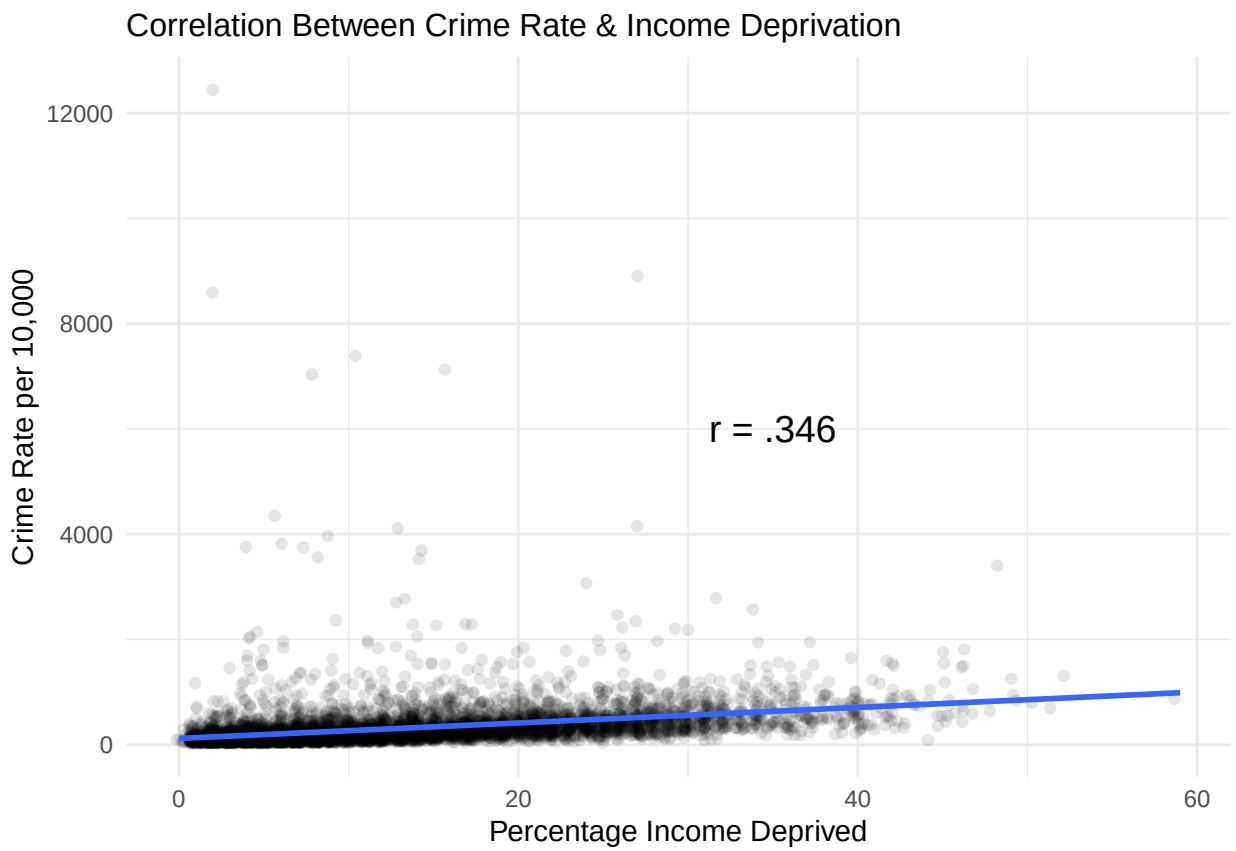
```
    Pearson's product-moment correlation

data:  simd$crime_rate and simd$not_participating
t = 28.348, df = 6472, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.3105005 0.3538419
sample estimates:
      cor
0.3323466
```

Since $p \leq 0.05$, we conclude that there is a statistically significant correlation between datazones' percentage of teens not in education, work, or training and crime rate. We find a correlation of 0.332, which indicates a moderately weak, positive relationship between teens not in education, work, or training and crime rates in Scottish datazones.

Exercise 3

The code from the chapter just needs to be updated with different variables and labels. We also need to figure out a good place for the correlation text.

```
simd %>%
  filter(!is.na(crime_rate) & !is.na(Income_rate)) %>%
ggplot(mapping = aes(x = Income_rate, y = crime_rate)) +
  geom_point(position = "jitter", alpha = .1) +
  geom_smooth(method = "lm", se = FALSE) +
  labs(x = "Percentage Income Deprived",
    y = "Crime Rate per 10,000",
    title = "Correlation Between Crime Rate & Income Deprivation") +
  theme_minimal() +
  theme(
    plot.title = element_text(size=12)
  ) +
  annotate("text", label = "r = .346", x = 35, y = 6000, size = 5)
```



For aesthetic purposes, we could filter out the outliers of `crime_rate`. (If we did filter out the outliers, we probably should re-do the correlation analysis.)