Linear Regression

Exploratory data analysis

Jenine Harris
Brown School

Importing and merging data sources

```
# distance to syringe program data
dist.ssp <- read.csv(file = "/Users/harrisj/Box/teaching/Teaching/Fall20

# summary
summary(object = dist.ssp)
```

```
##     county           STATEABBREVIATION     dist_SSP       HIVprevalence
##  Length:500         Length:500         Min.   :  0.00   Min.   :  -1.00
##  Class :character   Class :character   1st Qu.: 35.12   1st Qu.:  52.98
##  Mode  :character   Mode  :character   Median : 75.94   Median : 101.15
##                                        Mean   :107.74   Mean   : 165.75
##                                        3rd Qu.:163.83   3rd Qu.: 210.35
##                                        Max.   :510.00   Max.   :2150.70
##  opioid_RxRate       pctunins        metro
##  Min.   :  0.20   Min.   : 3.00   Length:500
##  1st Qu.: 45.12   1st Qu.: 8.60   Class :character
##  Median : 62.40   Median :11.70   Mode  :character
##  Mean   : 68.33   Mean   :12.18
##  3rd Qu.: 89.95   3rd Qu.:15.00
##  Max.   :345.10   Max.   :35.90
```

Codebook

Leslie looked through the variables and the codebook and determined that the variables had the following meanings:

- county: the county name
- STATEABBREVIATION: the two-letter abbreviation for the state the county is in
- dist_SSP: distance in miles to the nearest syringe services program
- HIVprevalence: people age 13 and older living with diagnosed HIV per 100,000
- opioid_RxRate: number of opioid prescriptions per 100 people
- pctunins: percentage of the civilian noninstitutionalized population with no health insurance coverage
- metro: county is non-metro, which includes open countryside, rural towns, or smaller cities with up to 49,999 people, or metro

Checking the descriptive statistics

```
# descriptive statistics for syringe data
tableone::CreateTableOne(data = dist.ssp,
                         vars = c('dist_SSP', 'HIVprevalence',
                                  'opioid_RxRate', 'pctunins',
                                  'metro'))
```

```
##
##                                Overall
##   n                               500
##   dist_SSP (mean (SD))         107.74 (94.23)
##   HIVprevalence (mean (SD)) 165.75 (208.97)
##   opioid_RxRate (mean (SD))  68.33 (36.81)
##   pctunins (mean (SD))        12.18 (4.97)
##   metro = non-metro (%)         274 (54.8)
```

Checking the distribution of HIV prevalence

```
# open the tidyverse
library(package = "tidyverse")

# check distribution of HIV rate
dist.ssp %>%
  ggplot(aes(x = HIVprevalence)) +
  geom_histogram(fill = "#7463AC", color = "white") +
  labs(x = "HIV cases per 100,000 people", y = "Number of counties") +
  theme_minimal()
```

Fixing the table

```
# descriptive statistics for syringe data
syringe.desc <- tableone::CreateTableOne(data = dist.ssp,
                                         vars = c('dist_SSP', 'HIVpreval
                                                  'opioid_RxRate', 'pctu
                                                  'metro'))
print(syringe.desc, nonnormal = c("HIVprevalence"))
```

```
##
##                                 Overall
##   n                                500
##   dist_SSP (mean (SD))          107.74 (94.23)
##   HIVprevalence (median [IQR]) 101.15 [52.98, 210.35]
##   opioid_RxRate (mean (SD))     68.33 (36.81)
##   pctunins (mean (SD))          12.18 (4.97)
##   metro = non-metro (%)           274 (54.8)
```

Linear regression is about relationships

- Linear regression is about examining relationships among variables.
- Specifically, linear regression is used to examine how one or more variables can predict or explain some continuous outcome variable.
- The research question to address with linear regression could be: How can uninsurance, metro or non-metro status, HIV prevalence, and number of opioid prescriptions predict or explain distance to the nearest syringe program at the county level?

Using scatterplots to explore relationships

- Start by examining whether the distance to a syringe program could be explained or predicted by percentage of county residents without insurance.

```
# percent without health insurance and distance to needle exchange
dist.ssp %>%
  ggplot(aes(x = pctunins, y = dist_SSP)) +
  geom_point(aes(size = "County"), color = "#7463AC", alpha = .6) +
  theme_minimal() +
  labs(x = "Percent without health insurance",
       y = "Miles to syringe program",
       title = "Relationship between percentage without health insurance
  scale_size_manual(values = 2, name = "")
```

Interpreting the scatterplot

- The plot showed that, as percentage without health insurance went up, so did distance to the nearest syringe program.
- That is, counties with a higher percentage of uninsured people were further from the nearest needle exchange.
- These two variables have a positive correlation.
- Use a `geom_smooth()` layer to add a line to the plot and get a better understanding of the relationship between the variables.
- The `method = "lm"` argument with `geom_smooth()` added a line to the plot that represents the linear model for the relationship between the variables.
- Added the line to the legend to clarify what the line represents; which can be done using aesthetics within `geom_point()` and `geom_smooth()`.

Adding a line and modifying the scatterplot legend

```
# percent without health insurance and distance to needle exchange
dist.ssp %>%
  ggplot(aes(x = pctunins, y = dist_SSP)) +
  geom_point(aes(size = "County"), color = "#7463AC", alpha = .6) +
  geom_smooth(aes(linetype = "Linear fit line"), method = "lm", se = FAL
  theme_minimal() +
  labs(x = "Percent uninsured", y = "Miles to syringe program") +
  scale_size_manual(values = 2, name = "") +
  scale_linetype_manual(values = 1, name = "")
```

Using a correlation coefficient to explore the relationship

```
# correlation between percent uninsured and distance
dist.ssp %>%
  summarize(cor.dist.uninsur = cor(x = dist_SSP,
                                   y = pctunins),
            samp.n = n())
```

```
##   cor.dist.uninsur samp.n
## 1        0.4126744    500
```

- The correlation coefficient was positive (r = 0.41).
- The strength is between weak and moderate.
- The mean distance from a county to the nearest syringe program is 107.74 miles with a standard deviation of 94.23
- The mean percent of county residents without insurance is 12.18% with a standard deviation of 4.97
- The relationship between uninsured percentage and distance to syringe program is weak to moderate and positive; counties with a higher percentage of uninsured are further from syringe programs (r = 0.41)

Explore the data by comparing means across groups

- Examine the other bivariate relationships between distance to syringe program and opioid prescriptions (`opioid_RxRate`), HIV prevalence (`HIVprevalence`), and metro or non-metro status (`metro`).

```
# bivariate relationships with distance to SSP
dist.ssp %>%
  summarize(cor.rx.rate = cor(dist_SSP, opioid_RxRate),
            cor.s.hiv = cor(dist_SSP, HIVprevalence, method = "spearman"
            cor.unins = cor(dist_SSP, pctunins))
```

```
##   cor.rx.rate  cor.s.hiv cor.unins
## 1 -0.09979404 0.06210425 0.4126744
```

- The correlation between `dist_SSP` and `HIVprevalence` was still weak and positive, $r_s = .06$, indicating that distance to syringe programs increases as HIV prevalence increases in a county.

- Check the mean distance to a syringe program for metro and non-metro counties:

```
# metro and distance to SSP
dist.ssp %>%
  group_by(metro) %>%
  summarize(m.dist = mean(dist_SSP))
```

Exploring the data with boxplots

```
# metro and distance to SSP
dist.ssp %>%
  ggplot(aes(x = metro, y = dist_SSP, fill = metro)) +
  geom_violin(aes(color = metro), fill = "white", alpha = .8) +
  geom_boxplot(aes(fill = metro, color = metro), width = .2, alpha = .3)
  geom_jitter(aes(color = metro), alpha = .4) +
  labs(x = "Type of county",
       y = "Miles to syringe program",
       title = "Distance to syringe programs by metro or non-metro statu
  scale_fill_manual(values = c("#78A678", "#7463AC"), guide = FALSE) +
  scale_color_manual(values = c("#78A678", "#7463AC"), guide = FALSE) +
  theme_minimal() +
  coord_flip()
```

Exploring the data with boxplots