Intermediate statistics: introduction

Frank Edwards

1/16/2024

School of Criminal Justice, Rutgers - Newark

Introductions: What are you planning to do with statistical models?

Before we begin

Remember: All models are wrong, some are useful.

What we will cover

- How to explore, visualize, and model diverse kinds of data with an emphasis on generalized linear models

What we will cover

- How to explore, visualize, and model diverse kinds of data with an emphasis on generalized linear models
- How to program in R

What we will cover

- How to explore, visualize, and model diverse kinds of data with an emphasis on generalized linear models
- How to program in R
- Developing a workflow for producing replicable quantitative social science

What we will cover

- How to explore, visualize, and model diverse kinds of data with an emphasis on generalized linear models
- How to program in R
- Developing a workflow for producing replicable quantitative social science
- Some advanced topics that are relevant for the kinds of data we're dealing with in the course

My general approach to data analysis

1. Explore and visualize data

My general approach to data analysis

1. Explore and visualize data
2. Fit models

My general approach to data analysis

1. Explore and visualize data
2. Fit models
3. Assess model fit

My general approach to data analysis

1. Explore and visualize data
2. Fit models
3. Assess model fit
4. Interpret and describe results through simulation

Review the syllabus

https://f-edwards.github.io/intermediate_stats/

1. Install R

cran.r-project.org

2. Install RStudio

posit.co/download/rstudio-desktop/

3. File structure and project organization basics
4. R console basics

Break

The Generalized Linear Model

We know we can model data as:

$$y = \beta_0 + \beta_1 x_1 \cdots \beta_n x_n + \varepsilon$$

The linear model

We know we can model data as:

$$y = \beta_0 + \beta_1 x_1 \cdots \beta_n x_n + \varepsilon$$

Or, more succinctly:

$$y = \mathsf{X}\beta + \varepsilon$$

The linear model

We know we can model data as:

$$y = \beta_0 + \beta_1 x_1 \cdots \beta_n x_n + \varepsilon$$

Or, more succinctly:

$$y = \mathsf{X}\beta + \varepsilon$$

Where the likelihood for the outcome conditional on the data takes the form:

$$Y|X \sim Normal(\mu, \sigma^2)$$

The linear model:

$$Y|X \sim Normal(\mu, \sigma^2)$$

Can be written as a more general formulation for a likelihood function f

$$Y|X \sim f(\mu, \sigma^2)$$

The linear model:

$$Y|X \sim Normal(\mu, \sigma^2)$$

Can be written as a more general formulation for a likelihood function f

$$Y|X \sim f(\mu, \sigma^2)$$

Now we can extend the (very) useful linear model to data with discrete outcomes

An expected value $E(Y|X) = \mu$

A linear predictor:

$$\mathrm{X}\beta$$

Generalizing the linear model

An expected value $E(Y|X) = \mu$

A linear predictor:

$$\mathsf{X}\beta$$

A link function g

$$g(\mu) = \mathsf{X}\beta$$

Generalizing the linear model

An expected value $E(Y|X) = \mu$

A linear predictor:

$$\mathsf{X}\beta$$

A link function g

$$g(\mu) = \mathsf{X}\beta$$

$$\mu = g^{-1}(\mathsf{X}\beta)$$

From OLS to GLM

OLS:

$$Y|X \sim Normal(\mu, \sigma^2)$$

GLM, for a likelihood function f with parameters θ:

$$Y|X \sim f(\theta)$$

Models we'll consider this semester

- Binary data: logistic models

Models we'll consider this semester

- Binary data: logistic models
- Categorical data: Multinomial models

Models we'll consider this semester

- Binary data: logistic models
- Categorical data: Multinomial models
- Count data: Poisson and negative binomial models

Returning to the linear model

What do we know about the linear regression model?

$$y = \mathsf{X}\beta + \varepsilon$$

$$\varepsilon \sim Normal(0, \sigma^2)$$

OR

$$\mu = \mathsf{X}\beta$$

$$y \sim Normal(\mu, \sigma^2)$$

Let’s analyze some data?

Two ways to access course data

- All data is accessible through the the course website (see the data link, or data folder on the GitHub page)

Read in

```r
library(tidyverse)
### data available in intermediate_stats/data/revenue_dat.csv
cj_budgets <- read_csv("http://tinyurl.com/revenuedata1")
```

About the data

It documents police involved deaths, demographics, and local government budgets at the county-level for two time periods, 2007-11 and 2012-16. Sources include Fatal Encounters, American Community Survey 5-year data, Annual Survey of State and Local Government Finance, and Uniform Crime Reports.

Evaluate the structure of the data

```r
head(cj_budgets)
```

```
## # A tibble: 6 x 26
##   year_range fips_st fips_cnty deaths  exp_tot exp_correction exp_police rev_tot
##   <chr>      <chr>   <chr>      <dbl>    <dbl>          <dbl>      <dbl>   <dbl>
## 1 2007-2011  01      001            3 49742600        2101800    9306200  5.65e7
## 2 2007-2011  01      005            1 28588200        1037880    5537840  3.37e7
## 3 2007-2011  01      007            0 13036120          80600    2421720  1.36e7
## 4 2007-2011  01      009            0 36644480        1703760    6853480  3.33e7
## 5 2007-2011  01      011            0 10940520              0    2285320  1.18e7
## 6 2007-2011  01      013            1 30533760         487320    4067200  2.50e7
## # i 18 more variables: rev_gen_ownsource <dbl>, rev_int_gov <dbl>,
## #   rev_prop_tax <dbl>, rev_tax <dbl>, pop_tot <dbl>, pop_wht <dbl>,
## #   pop_blk <dbl>, pop_lat <dbl>, pop_pct_pov <dbl>, pop_pct_deep_pov <dbl>,
## #   pop_med_income <dbl>, pop_pc_income <dbl>, violent.yr <dbl>,
## #   property.yr <dbl>, murder.yr <dbl>, ft_sworn <dbl>, cbsa <dbl>,
## #   metroname <chr>
```

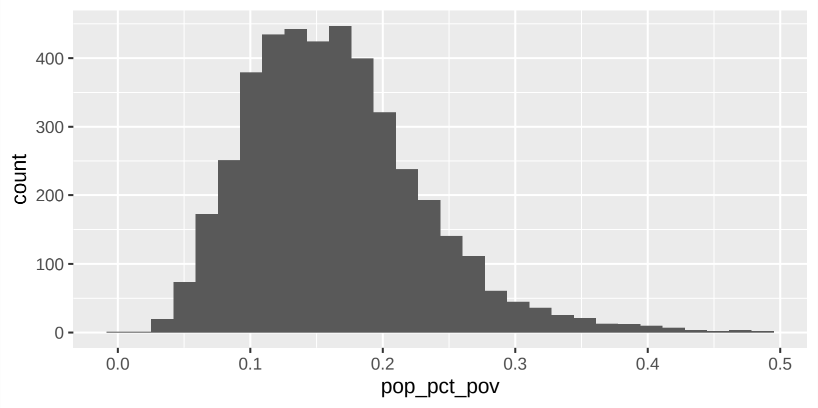
Descriptives

```
summary(cj_budgets)
```

```
##   year_range          fips_st            fips_cnty              de
##  Length:4286        Length:4286        Length:4286        Min.  
##  Class :character   Class :character   Class :character   1st Qu
##  Mode  :character   Mode  :character   Mode  :character   Median
##                                                           Mean  
##                                                           3rd Qu
##                                                           Max.  
##
##     exp_tot          exp_correction       exp_police
##  Min.   :2.531e+05   Min.   :0.000e+00   Min.   :1.221e+04
##  1st Qu.:1.965e+07   1st Qu.:1.996e+05   1st Qu.:1.679e+06
##  Median :4.972e+07   Median :1.280e+06   Median :4.257e+06
##  Mean   :3.614e+08   Mean   :9.856e+06   Mean   :3.005e+07
##  3rd Qu.:1.539e+08   3rd Qu.:4.076e+06   3rd Qu.:1.241e+07
##  Max.   :1.177e+11   Max.   :1.747e+09   Max.   :5.623e+09
```

Visualize the distribution of deep poverty across counties with ggplot

```r
ggplot(cj_budgets, aes(x = pop_pct_pov)) + geom_histogram()
```

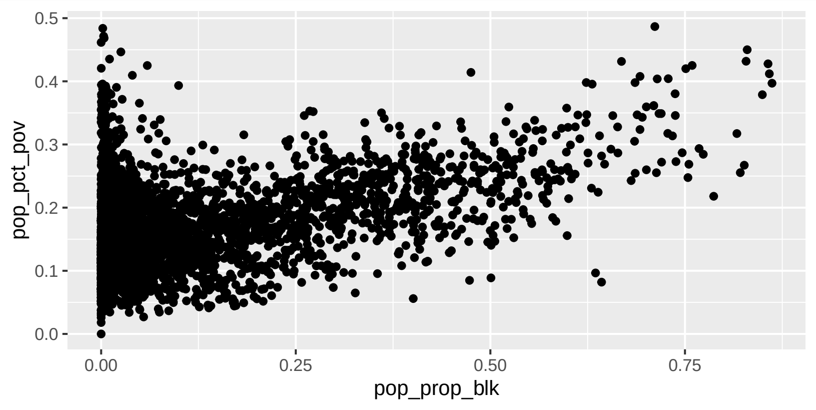


Create a new variable using mutate()

```r
cj_budgets <- cj_budgets %>%
    mutate(pop_prop_blk = pop_blk/pop_tot)
```

Visualize a bivariate relationship with ggplot

```
ggplot(cj_budgets, aes(y = pop_pct_pov, x = pop_prop_blk)) + geom_point()
```



Fitting a linear model with lm()

```r
model_1 <- lm(pop_pct_pov ~ pop_prop_blk, data = cj_budgets)
```

Display the model fit

```r
summary(model_1)
```

```
##
## Call:
## lm(formula = pop_pct_pov ~ pop_prop_blk, data = cj_budgets)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.18870 -0.04002 -0.00586  0.03362  0.33849
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.144985   0.001090  133.05   <2e-16 ***
## pop_prop_blk 0.195611   0.006007   32.56   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05978 on 4284 degrees of freedom
## Multiple R-squared:  0.1984, Adjusted R-squared:  0.1982
## F-statistic:  1060 on 1 and 4284 DF,  p-value: < 2.2e-16
```

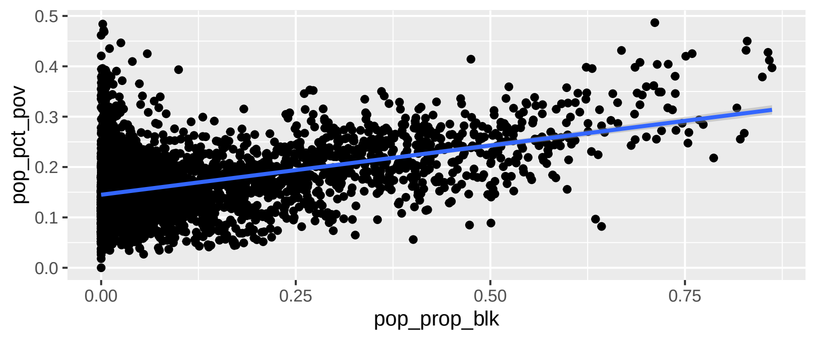
Display the model fit (nicer)

```r
library(broom)
tidy(model_1)
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic   p.value
##   <chr>          <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)    0.145   0.00109     133.  0
## 2 pop_prop_blk   0.196   0.00601      32.6 4.85e-208
```

Visualize the model fit

```
ggplot(cj_budgets, aes(y = pop_pct_pov, x = pop_prop_blk)) + geom_point() + geom_smooth(method = "lm",
    formula = y ~ x)
```



HW 1 guidelines

HW tips

- Work together!
- Google it: StackOverflow will become your best friend
- Accept that this is hard and you will probably struggle with it