# Intermediate statistics: introduction

Frank Edwards

1/27/2021

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, subject to class interest

## Review the syllabus

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

# 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

# The Generalized Linear Model

## The 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:

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

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:

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

# Generalzing the linear model

The linear model:

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

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

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

# Generalzing the linear model

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

# Generalizing the linear model

A linear predictor $\eta$:

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

A linear predictor $\eta$:

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

A link function $g$

$$g\big(E(Y|X)\big) = \eta$$

# Generalizing the linear model

A linear predictor $\eta$:

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

A link function $g$

$$g(E(Y|X)) = \eta$$

A mean expectation $E(Y|X) = \mu$

$$\mu = g^{-1}(\eta)$$

# From OLS to GLM

OLS:

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

GLM, for a likelihood function $f$ with parameters $\theta$:

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

## Diverse likelihood functions

- Binary data: linear probability (Normal/Gaussian) and logistic models

## Diverse likelihood functions

- Binary data: linear probability (Normal/Gaussian) and logistic models
- Categorical data: Multinomial model

# Diverse likelihood functions

- Binary data: linear probability (Normal/Gaussian) and logistic models
- Categorical data: Multinomial model
- Count data: Poisson and negative binomial models

# Diverse likelihood functions

- Binary data: linear probability (Normal/Gaussian) and logistic models
- Categorical data: Multinomial model
- Count data: Poisson and negative binomial models
- Positive continuous data: Gamma model

# Getting started: software

# Required installations

All software we are using is free and open source.

*Install R*:

`https://cran.r-project.org/`

*Install RStudio*:

`https://www.rstudio.com/products/rstudio/download/`

## Recommended software: Git and GitHub

Git and GitHub are powerful tools for backing up and sharing your research.

All course materials, source code, and most of my research are hosted on GitHub (`https://github.com/f-edwards`).

*Install Git*:

`https://git-scm.com/`

*Set up a GitHub account*:

`https://github.com/`

*Using GitHub for social science*:

`https://happygitwithr.com/`

# Recommended software: LaTeX

LaTeX is a powerful typesetting tool that works well with RMarkdown. It makes very attractive academic papers and slides.

Install it from the console

```
install.packages(”tinytex”)

tinytex::install_tinytex()
```

# Break

# Returning to the linear model

# What do we know about the linear regression model?

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

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

1. What forms can y take?

# Review

1. What forms can y take?
2. What assumptions does the linear regression model require?

# Assumptions of linear regression model

1. Validity of data relative to the research question

# Assumptions of linear regression model

1. Validity of data relative to the research question
2. Additive, linear functional form

# Assumptions of linear regression model

1. Validity of data relative to the research question
2. Additive, linear functional form
3. Independent errors

## Assumptions of linear regression model

1. Validity of data relative to the research question
2. Additive, linear functional form
3. Independent errors
4. Equal variance of errors

# Assumptions of linear regression model

1. Validity of data relative to the research question
2. Additive, linear functional form
3. Independent errors
4. Equal variance of errors
5. Normality of errors

# Assumptions of linear regression model

1. Validity of data relative to the research question
2. Additive, linear functional form
3. Independent errors
4. Equal variance of errors
5. Normality of errors

# 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)
- *Recommended approach:* In a terminal (terminal.app on mac, Git Bash on windows):

```
git clone
https://github.com/f-edwards/intermediate_stats.git
```

Before beginning your work each session, pull updates I've pushed to the repo with:

```
git pull
```

Now you have an intermediate_stats folder with all code, slides, and data. Data is in intermediate_stats/data

# Read in

```
#library(tidyverse)
### directly from the web
cj_budgets<-read_csv("./hw/data/revenue_dat.csv")
### from a project directory root
#cj_budgets<-read_csv("./hw/data/revenue_dat.csv")
```

## About the data

Data are for an ongoing research project I'm working on. It's real, and can be a bit messy!

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

Full code for the project is up at:

`https://github.com/f-edwards/police-mort-revenue`

`merge.r` contains the code to make this merged file from a variety of source files (available if you want the raw data).

# Evaluate the structure of the data

```
head(cj_budgets)
```

```
## # A tibble: 6 x 33
##   year_range fips_st fips_cnty deaths exp_tot exp_correction exp
##   <chr>      <chr>   <chr>      <dbl>   <dbl>          <dbl>
## 1 2007-2011  01      001            3  4.97e7       2101800
## 2 2007-2011  01      005            1  2.86e7       1037880.
## 3 2007-2011  01      007            0  1.30e7         80600
## 4 2007-2011  01      009            0  3.66e7       1703760
## 5 2007-2011  01      011            0  1.09e7             0
## 6 2007-2011  01      013            1  3.05e7        487320
## # ... with 26 more variables: exp_welfare <dbl>, rev_tot <dbl>,
## #   rev_fines <dbl>, rev_gen_ownsource <dbl>, rev_int_gov <dbl>,
## #   rev_prop_tax <dbl>, rev_tax <dbl>, pop_tot <dbl>, pop_pct_me
## #   pop_wht <dbl>, pop_blk <dbl>, pop_ami <dbl>, pop_api <dbl>,
## #   pop_pct_pov <dbl>, pop_pct_deep_pov <dbl>, pop_med_income <d
## #   pop_pc_income <dbl>, violent.yr <dbl>, property.yr <dbl>, mu
```

# Evaluate the structure of the data

```r
nrow(cj_budgets)
```

```
## [1] 4286
```

```r
table(cj_budgets$year_range)
```

```
##
## 2007-2011 2012-2016
##      2308      1978
```

# Evaluate the structure of the data

```
names(cj_budgets)
```

```
##  [1] "year_range"         "fips_st"            "fips_cnty"
##  [4] "deaths"             "exp_tot"            "exp_correction"
##  [7] "exp_police"         "exp_welfare"        "rev_tot"
## [10] "rev_fines"          "rev_gen_ownsource"  "rev_int_gov"
## [13] "rev_prop_tax"       "rev_tax"            "pop_tot"
## [16] "pop_pct_men_15_34"  "pop_wht"            "pop_blk"
## [19] "pop_ami"            "pop_api"            "pop_lat"
## [22] "pop_pct_pov"        "pop_pct_deep_pov"   "pop_med_income"
## [25] "pop_pc_income"      "violent.yr"         "property.yr"
## [28] "murder.yr"          "ft_sworn"           "cbsa"
## [31] "metroname"          "dissim_bw"          "dissim_wl"
```

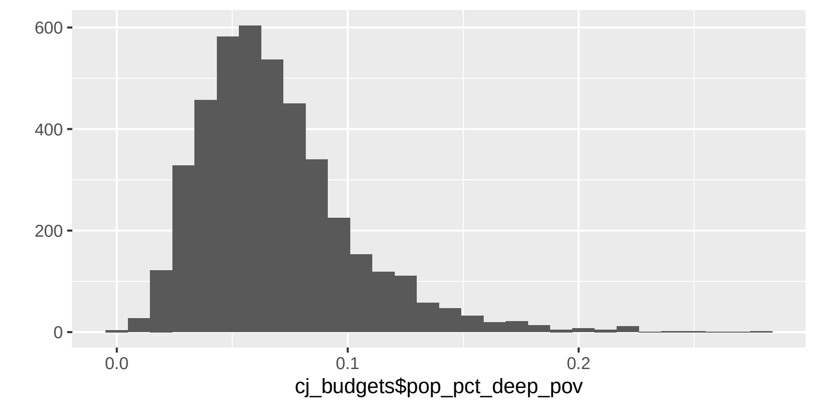
# Descriptives

```
summary(cj_budgets$pop_pct_deep_pov)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.04553 0.06285 0.06884 0.08442 0.27901
```

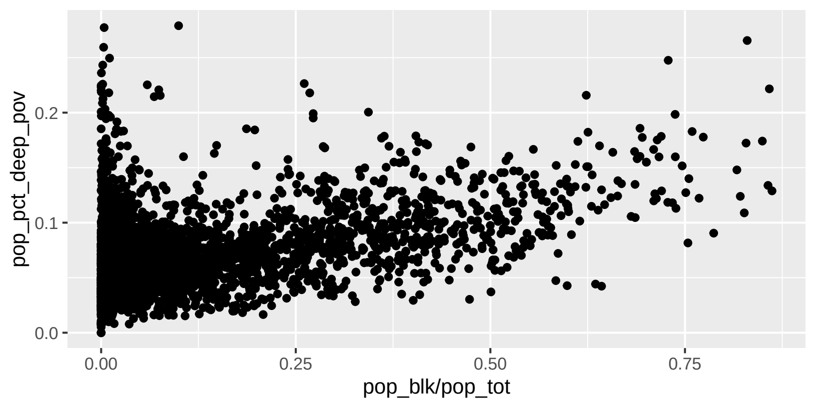
Call individual variables (columns) in a data frame with $, like \texttt{USArrests$Murder}

```
qplot(cj_budgets$pop_pct_deep_pov)
```



# Visualize a bivariate relationship

```
qplot(pop_blk/pop_tot,
      pop_pct_deep_pov,
      data = cj_budgets)
```



# Fitting a linear model

```
model_1<-lm(pop_pct_deep_pov ~
              I(pop_blk/pop_tot),
            data =cj_budgets)
```

# Display the model fit

```
summary(model_1)
```

```
##
## Call:
## lm(formula = pop_pct_deep_pov ~ I(pop_blk/pop_tot), data = cj_bu
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.079709 -0.019343 -0.004579  0.013753  0.217773
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        0.0591712  0.0005603  105.61   <2e-16 ***
## I(pop_blk/pop_tot) 0.0977188  0.0030884   31.64   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

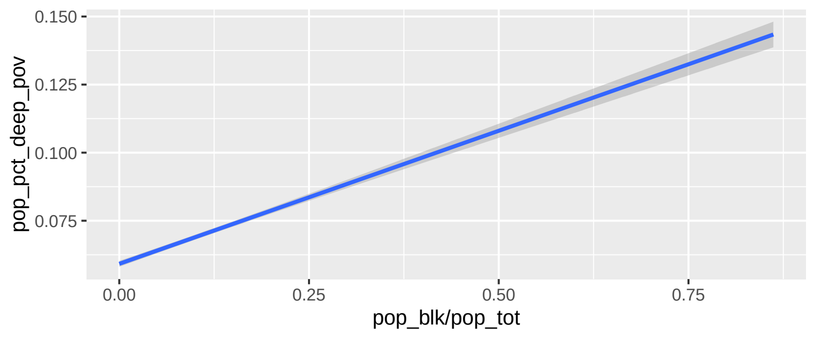
## Display the model fit (nicer)

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

```
## # A tibble: 2 x 5
##   term                estimate std.error statistic   p.value
##   <chr>                  <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)           0.0592  0.000560     106.  0.
## 2 I(pop_blk/pop_tot)    0.0977  0.00309       31.6 1.22e-197
```

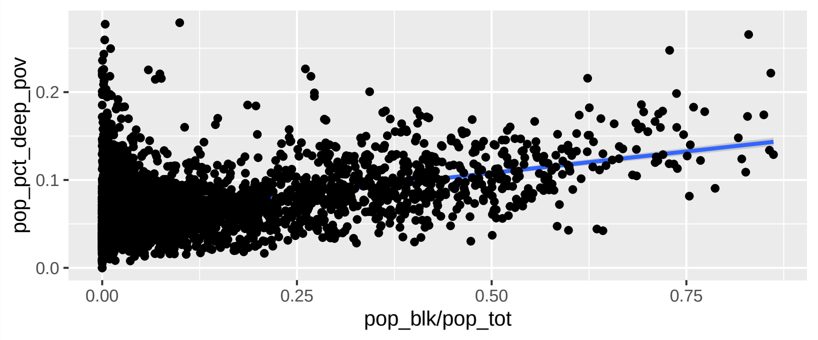
## Visualize the model fit

```
library(ggplot2)
ggplot(cj_budgets,
       aes(x=pop_blk/pop_tot, y=pop_pct_deep_pov))+
  geom_smooth(method = "lm",
              formula = y~x)
```



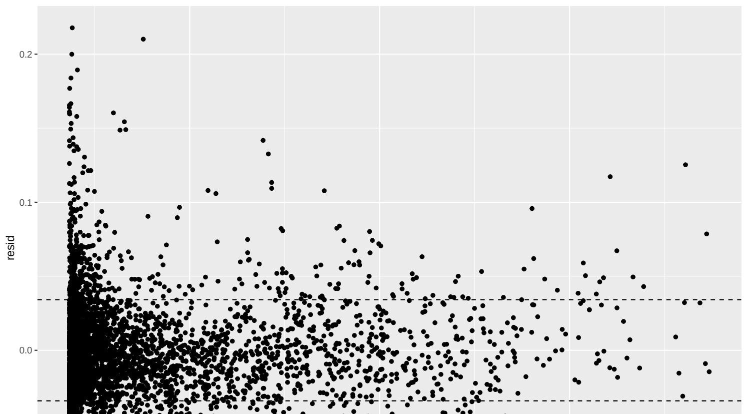
# Visualize the model fit (against the data)

```
library(ggplot2)
ggplot(cj_budgets,
       aes(x=pop_blk/pop_tot, y=pop_pct_deep_pov))+
  geom_smooth(method = "lm",
              formula = y~x) +
  geom_point()
```



# Residuals vs fitted

```
sd_outcome<-sd(cj_budgets$pop_pct_deep_pov)
plot_dat<-data.frame(resid = model_1$residuals, yhat = model_1$fitt
ggplot(plot_dat, aes(y = resid, x = yhat)) +
  geom_point() +
  geom_hline(yintercept = sd_outcome, lty=2) + geom_hline(yintercep
```



# Can we fit a better model?

# Homework programming prep

# Needed concepts / syntax

HW 1 asks you to apply some basic programming, data wrangling, and data visualization to common linear regression challenges.

- Reading data
- Loops
- Lists
- Matrix and data frame indexing
- dplyr::mutate
- ggplot2
- RMarkdown

# HW tips

- Work together!
- Check the Wickham text from the syllabus or other online R courses
- Google it: StackOverflow will become your best friend
- Accept that this is hard and you will probably struggle with it