SOC542 Statistical Methods in Sociology II

Dummy, Categorical, and Non-Linear Variables

Thomas Davidson

Rutgers University

February 20, 2023

Course updates

- Homework 2 released after class
  - Multivariate regression: specification, estimation, and interpretation
  - Due next Tuesday (2/27) at 5pm via Github Classroom

Paper propsals

- Plan for final paper
  - Relevant literature and research question(s)
    - Literature review does not need to be exhaustive!
  - Data
    - Outcome variable
    - Key predictors
  - Methodology
    - Regression approach
- Due 7/3 at 5pm (via email)

Plan

- Dummy variables
- Categorical variables
- Logarithms
- Polynomials

Dummy variables

Definitions

- A **dummy variable** is used to measure the difference between two possible states.
- Dummy variables are binary, taking a value of either zero or one.
- These values stand in for social categories of interest.
  - e.g. Male/female, employed/unemployed, liberal/conservative.

Dummy variables

Dummy variables as random variables

- We can generate dummy variables using the Binomial distribution, where $P(x = 1) = p$ and $P(x = 0) = 1 - p$.

$$x \sim Binomial(N, p)$$

Dummy variables

A simple model

```
N <- 10000
x <- rbinom(N, 1, .4) # p = .4
y <- 3*x + rnorm(N, 10, 1)
m <- lm(y ~ x)
round(m$coefficients,2)
```

```
## (Intercept)           x
##        9.99        3.01
```

Dummy variables

Interpretation

```
as.data.frame(cbind(x,y)) %>% group_by(x) %>%
    summarize(mean = mean(y)) %>% as.matrix()
```

```
##      x      mean
## [1,] 0  9.991578
## [2,] 1 13.002374
```

Dummy variables

Interpretation

- The coefficient represents the expected difference in the outcome when $x = 1$ compared to $x = 0$.
- Consider the following population model, predicting income as a function of union membership.

$$Income = \beta_0 + \beta_1 Union + u$$

- β_1 represents the expected difference in income for union members compared to non-union members.
- The dummy variable also impacts the meaning of the intercept β_0, which is the average income for non-unionized workers.

Dummy variables

Example: Union wage returns

```
gss <- haven::read_dta("../../2022/labs/lab-data/GSS2018.dta")

data <- gss %>% select(realrinc, union, sex) %>%
    drop_na(realrinc, union) %>%
    mutate(union_dummy = ifelse(union == 1, "U", "nU"),
           sex = ifelse(sex == 1, "Male", "Female"))

u.reg <- lm(realrinc ~ union_dummy, data = data)
print(u.reg)
```

```
##
## Call:
## lm(formula = realrinc ~ union_dummy, data = data)
##
## Coefficients:
##  (Intercept)  union_dummyU
##        24572          5646
```

Dummy variables

Example: Union wage returns

```
means <- data %>% group_by(union_dummy) %>%
    summarize(m = mean(realrinc))
print(means)
```

```
## # A tibble: 2 x 2
##   union_dummy      m
##   <chr>        <dbl>
## 1 U           30218.
## 2 nU          24572.
```

```
mean.nonunionized <- means %>% filter(union_dummy == "nU")
print(mean.nonunionized$m + u.reg$coefficients[2])
```

```
## union_dummyU
##      30218.2
```

Dummy variables

Reference category

- The interpretation of the model depends on which value we assign to 1 or 0.
  - The value assigned to 0 is known as the **reference category**.
- For a statistical perspective, the choice is arbitrary.
- But often there are theoretical reasons for selecting a certain reference category and the choices encode certain assumptions about the social world.[1]

[1] See Johfre and Freese (2021) for further discussion of reference categories.

Dummy variables

Reversing the reference category

```
x.rev <- ifelse(x, 0, 1)
m2 <- lm(y ~ x.rev)
round(m2$coefficients,2)
```

```
## (Intercept)       x.rev
##       13.00       -3.01
```

Dummy variables

Removing the reference category

If we estimate a model with no intercept then we get a separate parameter for each category.

```
u.reg.ni <- lm(realrinc ~ 0 + union_dummy, data = data)
print(coefficients(u.reg.ni))
```

```
## union_dummynU  union_dummyU
##       24572.3       30218.2
```

```
diff <- coefficients(u.reg.ni)[2] - coefficients(u.reg.ni)[1]
print(diff[[1]])
```

```
## [1] 5645.907
```

```
print(coefficients(u.reg)[2])
```

```
## union_dummyU
##     5645.907
```

Dummy variables

Removing the reference category

- In the model with no intercept we get a separate coefficient for each category.
- The difference between these coefficients is equivalent to the dummy variable in the model with the intercept.

$$\beta_{Union} = \beta^*_{NoUnion} - \beta^*_{Union}$$

Dummy variables

Multiple dummy variables

- A multiple regression model can include more than one dummy variable.
- The interpretation of each coefficient is now the difference *holding other variables at their means*.
- The intercept is the mean value of the outcome when all dummy variables are zero.

Dummy variables

Multiple dummy variables

This model of union wage returns includes a dummy variable for sex. What is the reference category and how can we interpret the intercept?

```
u.reg2 <- lm(realrinc ~ union_dummy + sex, data = data)
print(u.reg2)
```

```
##
## Call:
## lm(formula = realrinc ~ union_dummy + sex, data = data)
##
## Coefficients:
##  (Intercept)  union_dummyU       sexMale
##        20077          3699          9757
```

Categorical variables

More than two categories

- **Categorical variables** are a generalization of dummy variables to more than two categories.
  - e.g. Race/ethnicity, highest level of education, region.
- Categories can be **ordinal**, indicating some type of numerical ranking, or **nominal**.

Categorical variables

Categorical variables as dummy variables

```
cats <- sample(c("a", "b", "c"), 10000, replace=TRUE,
               prob=c(0.2, 0.2, 0.6))
print(cats[1:10])
```

```
##  [1] "c" "c" "b" "c" "a" "a" "c" "c" "c" "c"
```

```
a <- ifelse(cats == "a", 1,0)
b <- ifelse(cats == "b", 1,0)
c <- ifelse(cats == "c", 1,0)
y <- 0.3*b + rnorm(N)
```

Categorical variables

Reference categories and regression results

The only difference between these models is the order. By default, the last value will be used as the reference category.

```r
m4 <- lm(y ~ a + b + c)
m5 <- lm(y ~ c + a + b)
m6 <- lm(y ~ b + c + a)
```

Categorical variables

Reference categories and regression results

	(1)	(2)	(3)
(Intercept)	0.004	0.294***	-0.008
	(0.013)	(0.022)	(0.022)
a	-0.011	-0.302***	
	(0.026)	(0.032)	
b	0.291***		0.302***
	(0.026)		(0.032)
c		-0.291***	0.011
		(0.026)	(0.026)
Num.Obs.	10000	10000	10000
R2	0.014	0.014	0.014
R2 Adj.	0.013	0.013	0.013
RMSE	0.99	0.99	0.99

+ $p < 0.1$, * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Categorical variables

Interpretation

- Let's assume we run a survey in North America and find out respondents' country of residence. We want to estimate the following model, using USA as a reference category:

$$Income = \beta_0 + \beta_1 Canada + \beta_2 Mexico + u$$

- β_0 represents the average income for respondents in the USA.
- β_1 represents the expected difference between Canada and the USA.
- β_2 represents the expected difference between Mexico and the USA.

Categorical variables

Degrees of freedom

- Each additional category uses up a degree of freedom in our model.
  - Be careful when including variables with many categories (e.g. state of residence)
- Sometimes it may be defensible to treat *ordinal* variables as if they are continuous.
  - This only uses one degree of freedom.
  - Unit increases should be constant for all values and have a linear interpretation.
    - e.g. We have categories a, b, and c we could translate these values to the sequence 1, 2, 3. In this case, $b - a = c - b$.

Categorical variables

Encoding and categorical variables

- Categorical data in surveys is often coded as numeric. This can lead to misleading results since we might consider *nominal* categories as *ordinal*. Consider the example below using the region variable in the GSS.

```
coefficients(lm(realrinc ~ region, data = gss))
```

```
## (Intercept)      region
##  25217.3093    -42.7065
```

Categorical variables

Encoding and categorical variables

- We can fix this by casting the variable as a factor.

```
gss$region <- as.factor(gss$region)
coefficients(lm(realrinc ~ region, data = gss))
```

```
## (Intercept)     region2     region3     region4     reg
##   29665.891   -4397.330   -5512.035   -7055.731   -3517
##     region7     region8     region9
##   -3834.901   -6178.623   -3194.037
```

Non-linear variables

Specifying non-linearities

- Recall that OLS regression is linear *in parameters*.
- This does not require that predictors or outcomes vary in a linear way.
- We will consider how logarithms and polynomials allow us to specify non-linear relationships between variables.

Logarithms

Logarithms and the exponential function

- The **exponential function** raises a *base* b is raised to a power[2] x.

$$exp(x) = b^x$$

- The **logarithm** is the *inverse* of exponentiation.

$$log_b(b^x) = x$$

[2] Note how this differs from the power function, where we raise x to a specified power. E.g. $power(x, 2) = x^2$

Logarithms

Logarithms and the exponential function

- Here are some examples of common bases:

$$log_2(2^4) = 4$$

$$log_{10}(10^4) = 4$$

- The **natural logarithm** uses the constant $e \approx 2.71828$ as its base:

$$log_e(e^4) = 4$$

Logarithms

Logarithms and exponents

- We can easily verify this in R using the `log` function with specified bases:

```
log(2^4, base = 2)
```

```
## [1] 4
```

```
log(10^4, base = 10)
```

```
## [1] 4
```

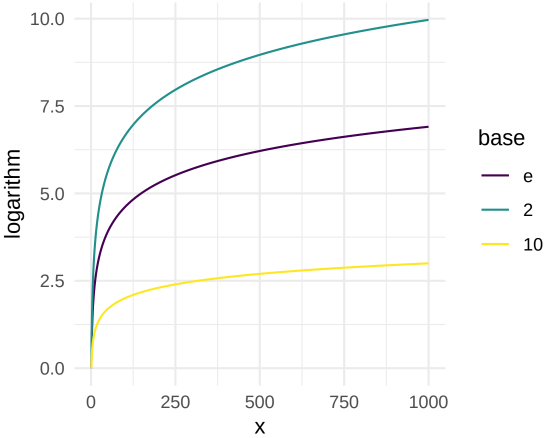
```
log(exp(1)^4)
```

```
## [1] 4
```

- The default base is e. Thus, `exp(1)` $= e^1 = e$.

Logarithms

Graphing logarithms



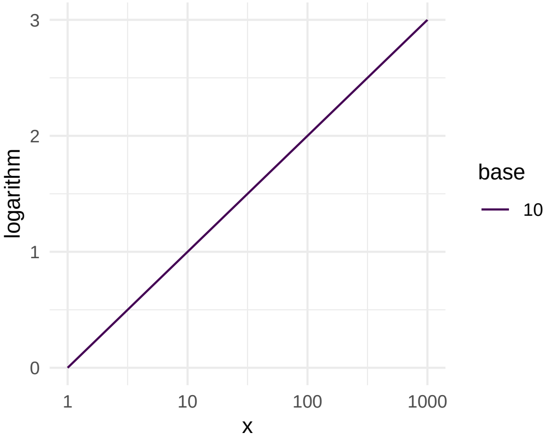
Logarithms

When to use logarithms

- Use logarithms when
  - All values of x are positive
  - x has a wide range (e.g. income)
- Interpretation
  - Logarithms allow us to transform variables to measure differences in *magnitude* and sometimes *percentages*.
- Specification
  - Logarithms can induce normality and reduce variance for if a variable has a **log normal** distribution.

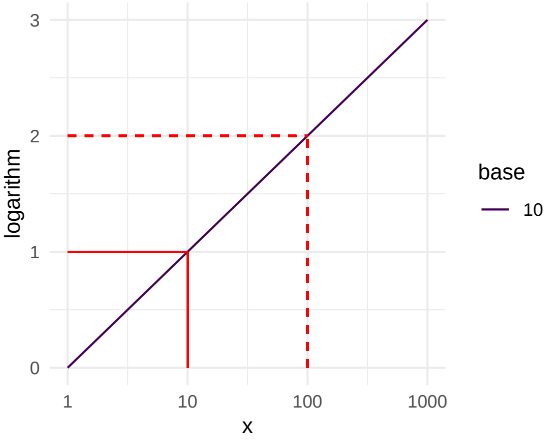
Logarithms

A unit increase of log_{10}



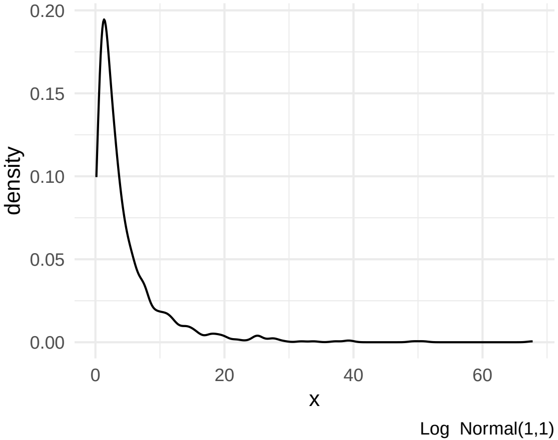
Logarithms

A unit increase of log_{10}



Logarithms

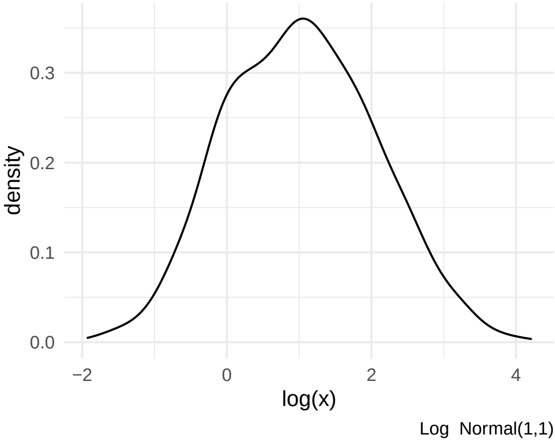
The log-normal distribution



Draws generated using `rlnorm()`.

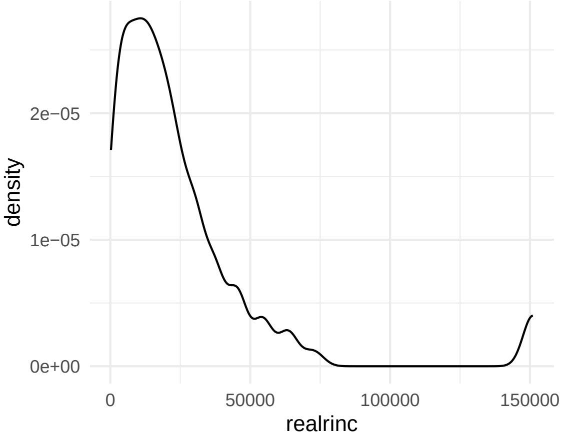
Logarithms

The natural logarithm of the log normal distribution



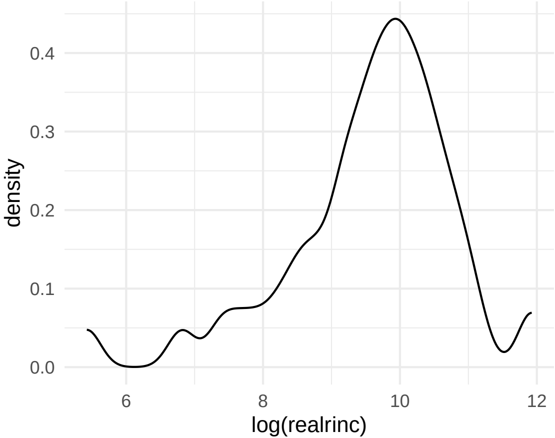
Logarithms

The distribution of 2018 GSS respondent income (`realinc`)



Logarithms

The distribution of 2018 GSS respondent income (`realinc`), natural log



Logarithms

Logarithms and regression

- Logarithms of predictors (**Linear-log models**)
  - If we include $log_e(x)$ as a predictor, β_i now represents the expected change associated with a unit increase in $log_e(x)$

Logarithms

Logarithms as predictors

Here β_1 represents the effect of a one-unit increase in height *on a logarithmic scale*.

```
##
## Call:
## lm(formula = realrinc ~ log(height), data = gss)
##
## Coefficients:
## (Intercept)  log(height)
##     -348333        88987
```

Logarithms

Logarithms and regression

- Logarithms of outcomes (**Log-linear models**)
  - If the outcome is $log_e(y)$. For any variable x, β_i represents the expected change in $log_e(y)$ associated with a unit change in x.

Logarithms

Logarithms as outcomes

Here β_1 represents the effect of a one-unit increase in height on the *logarithm* of income. Note the difference in the coefficient compared to the previous model.

```
##
## Call:
## lm(formula = log(realrinc) ~ height, data = gss)
##
## Coefficients:
## (Intercept)       height
##     5.60849      0.06041
```

Logarithms

Logarithms and regression

- Logarithms of predictors *and* outcomes (**Log-log models**)
  - If both x and y are entered into the model as logarithms, β_i represents the expected change in $log_e(y)$ associated with a unit change in $log_e(x)$
  - Equivalently, this corresponds to the expected percentage change in y as a result of a 1% change in x. Hence, such coefficients can be interpreted as **elasticities**.

Logarithms

Log-log models

Here β_1 represents the effect of a one-unit increase in *logarithm* of height on the *logarithm* of income. A 1% increase in height is associated with a 4% increase in income.

```
##
## Call:
## lm(formula = log(realrinc) ~ log(height), data = gss)
##
## Coefficients:
## (Intercept)  log(height)
##      -7.341        4.044
```

Logarithms

Log-log models

We can still incorporate untransformed variables into these models. Here the coefficient for `sex` can be interpreted as the *difference in the expected logarithm of income* between male and female respondents.

```
##
## Call:
## lm(formula = log(realrinc) ~ log(height) + sex, data = gss)
##
## Coefficients:
## (Intercept)  log(height)          sex
##     -0.4834       2.5136      -0.2774
```

Logarithms

Model comparison

	Log x	Log y	Log-log	Log-log+
(Intercept)	-348333.267***	5.608***	-7.341***	-0.483
	(55342.266)	(0.520)	(2.181)	(3.046)
log(height)	88986.528***		4.044***	2.514***
	(13158.504)		(0.519)	(0.703)
height		0.060***		
		(0.008)		
sex				-0.277**
				(0.086)
Num.Obs.	1194	1194	1194	1194
R2	0.037	0.049	0.049	0.057
R2 Adj.	0.036	0.048	0.048	0.055

+ $p < 0.1$, * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Polynomials

Definitions

- **Polynomial** regression expresses non-linear relationships between continuous predictors in a linear model by adding exponents of x.
- The expected value of y is expressed as an k^{th} degree polynomial:

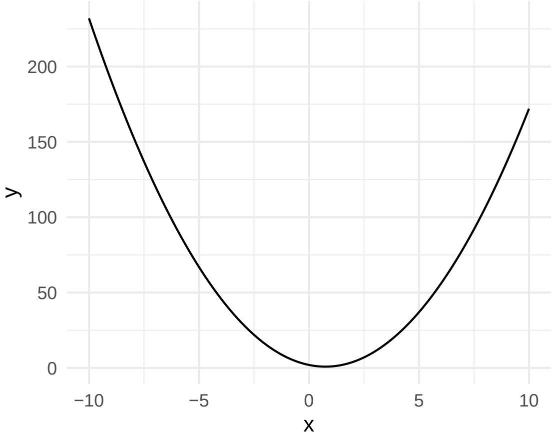
$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + ... + \beta_k x^k + u$$

- Generally, we use a restricted form, such as the quadratic model:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + u$$

Polynomials

Quadratic functions and parabolas



$$y = 2 + -3x + 2x^2.$$

Polynomials

When to use polynomial regression

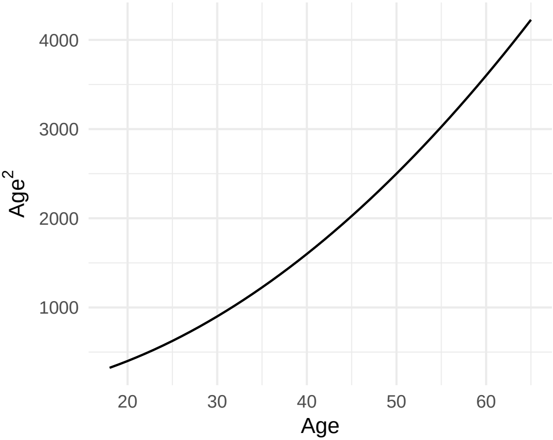
- We add polynomials to capture non-linear relationships. For example, we might expect a non-linear relationship between age and income. We could express this using the following model:

$$Income = \beta_0 + \beta_1 Age + \beta_2 Age^2 + u$$

- The effect of age is now decomposed into two coefficients:
  - β_1 captures the linear relationship between age and income.
  - β_2 captures a non-linear association between age and income.
- The coefficients do not have a simple interpretation.
  - Consider whether we can change *Age* while holding Age^2 constant.

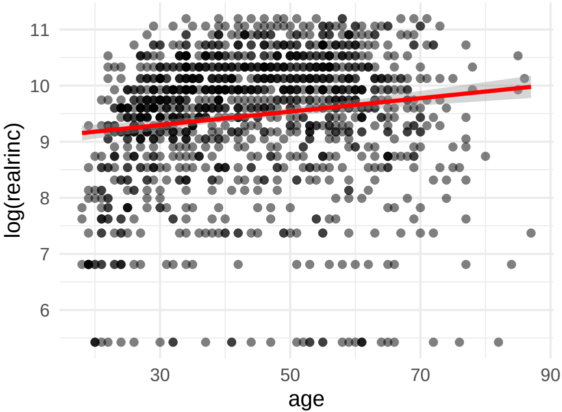
Polynomials

Age and Age-Squared



Polynomials

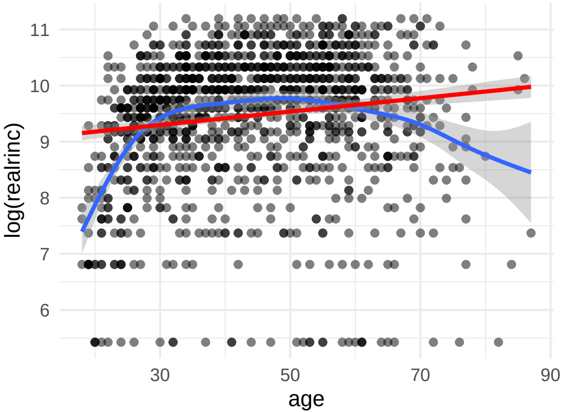
Income and age



Age < 89 and income < 1E5. OLS fitted line.

Polynomials

Income and age



Age < 89 and income < 1E5. OLS & LOESS fitted lines.

Polynomials

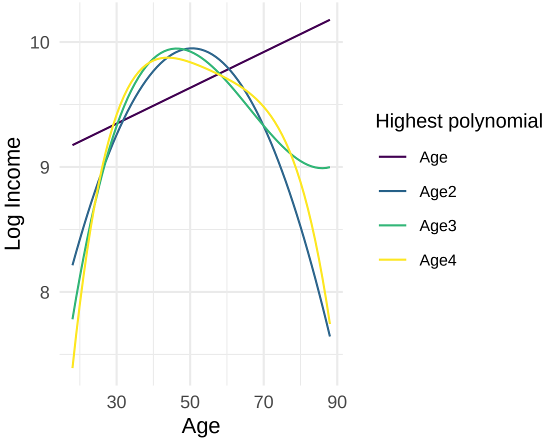
Income and age

	(1)	(2)	(3)	(4)
(Intercept)	8.917***	5.759***	2.958***	-2.453
	(0.108)	(0.294)	(0.764)	(1.964)
age	0.014***	0.166***	0.368***	0.894***
	(0.002)	(0.013)	(0.053)	(0.184)
age2		-0.002***	-0.006***	-0.024***
		(0.000)	(0.001)	(0.006)
age3			0.000***	0.000***
			(0.000)	(0.000)
age4				0.000**
				(0.000)
Num.Obs.	1357	1357	1357	1357
R2	0.027	0.114	0.124	0.130
R2 Adj.	0.027	0.112	0.122	0.127

+ $p < 0.1$, * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Polynomials

Predictions



Predicted values from OLS models. Income measured using realrinc. Respondents under 89 only.

Polynomials

When to use polynomial regression

- A second-order polynomial (e.g. Age^2) is sufficient to capture non-linearity.
  - In this case, there is evidence of a curvilinear relationship between age and income.[3]
- Higher-order polynomial terms can improve model fit and capture more complex non-linearities but *use up degrees of freedom* and become difficult to interpret.

[3] For an example of polynomials used in a different context, see Dokshin, Fedor A. 2016. "Whose Backyard and What's at Issue? Spatial and Ideological Dynamics of Local Opposition to Fracking in New York State, 2010 to 2013." *American Sociological Review* 81 (5): 921–48.

Next week

Interaction terms

- What is an interaction?
- How to specify an interaction
- How to interpret an interaction

Lab

- Regression models using the General Social Survey