Data Analysis with Python

Gregory M. Eirich

Columbia University

(Class #3)

1. The Residual

Implications of the u: Unpacking the u

- Error term
- Disturbance
-Unobservables

Whatever affects our dependent variable but is not included in our equation is captured by u

How are x and u related?

$E(u|x)=0$

This is the zero conditional mean assumption (as long the constant is included in the equation)

This means that for any value of x, the average value of the unobservables is the same

How are x and u related?

Let’s return to our occupational prestige example from last week.

This implies that people with 8 years of education and those with 16 years of education have – on average – the same value on all unobservables that might affect occupational prestige (e.g., assets, connections, ability, etc.)

This is the implication of “all else equal”

How are x and u related?

All other possible variables (that affect occupational prestige) are all randomly distributed among everyone, once educational attainment is taken into account.

No other variables (that affect occupational prestige) are correlated with education.

Is that a reasonable assumption?

More on the OLS assumptions next time …

2. Why Multiple Regression?

- To explicitly account for variables that are likely in *u*.

- To have correctly specified models.

How to build a better model:

1. Find a correlation/association (but correlation ≠ causation)
2. Try to place variables in their proper time order (we will return to this later)
3. Eliminate alternative explanations

How to deal with alternative explanations:

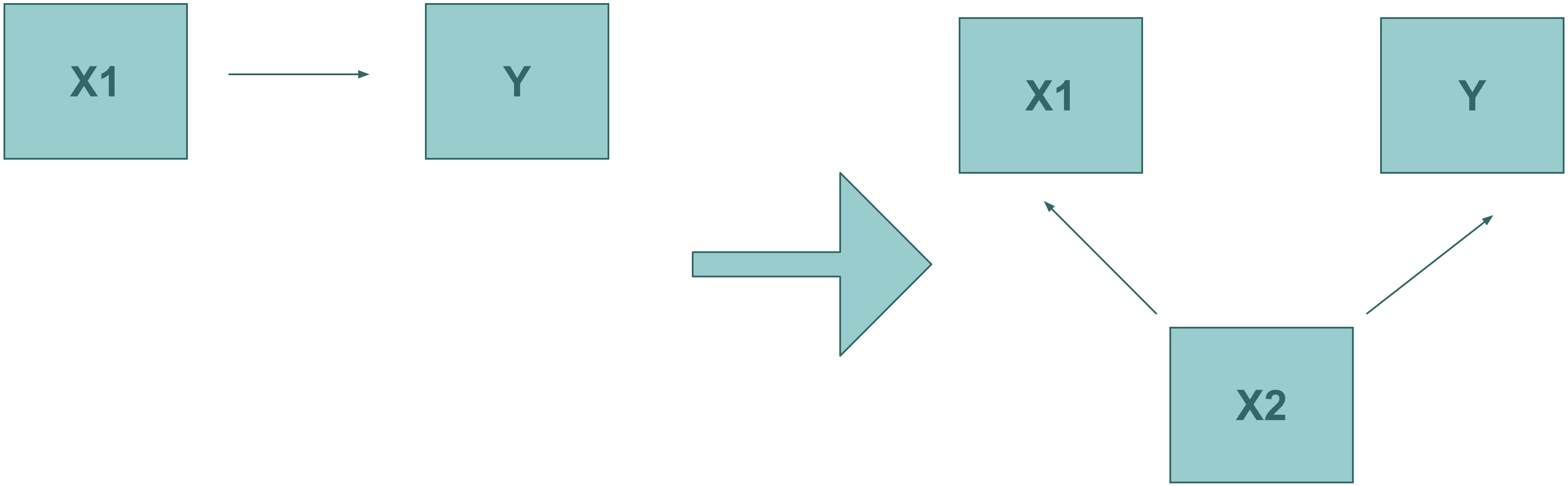
Consider omitted variables

What do omitted variables turn out to be?

- **Spurious**
- **A mediating variables in a process:**
  - The whole link in a chain of causation
  - Part of the link in a chain of causation
- **An interaction with X1**
- **A cause, but unrelated to the other variables**

To account for true relationships

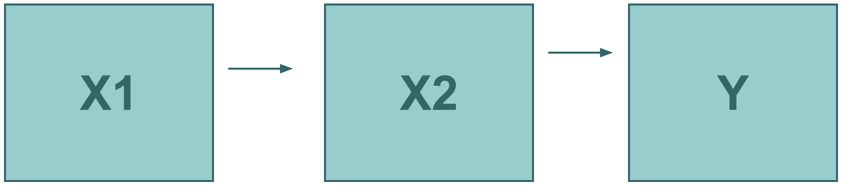
- Spuriousness: Some omitted variable is fully driving the relationship between our X and Y



To account for true relationships

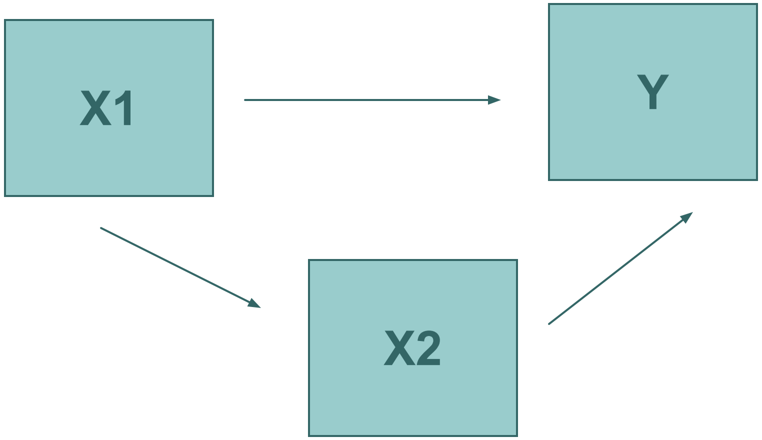
- Mediation: Some variable is the mechanism behind the relationship between our X and Y

Chain Mechanism



X2 fully accounts for the relationship between X1 and Y

Both Direct and Indirect Effects

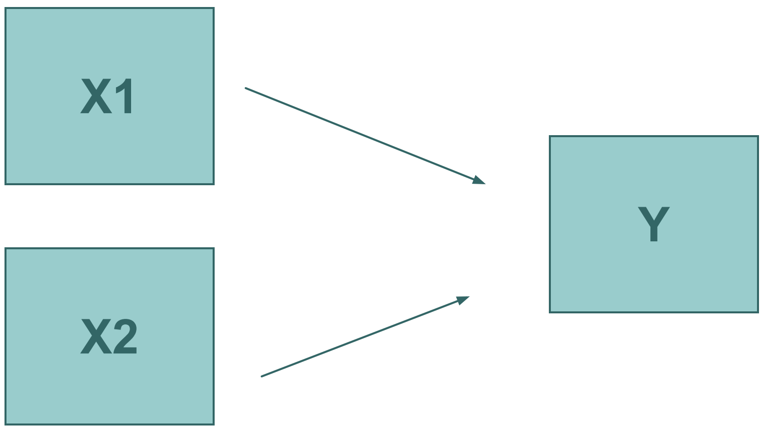


To account for true relationships

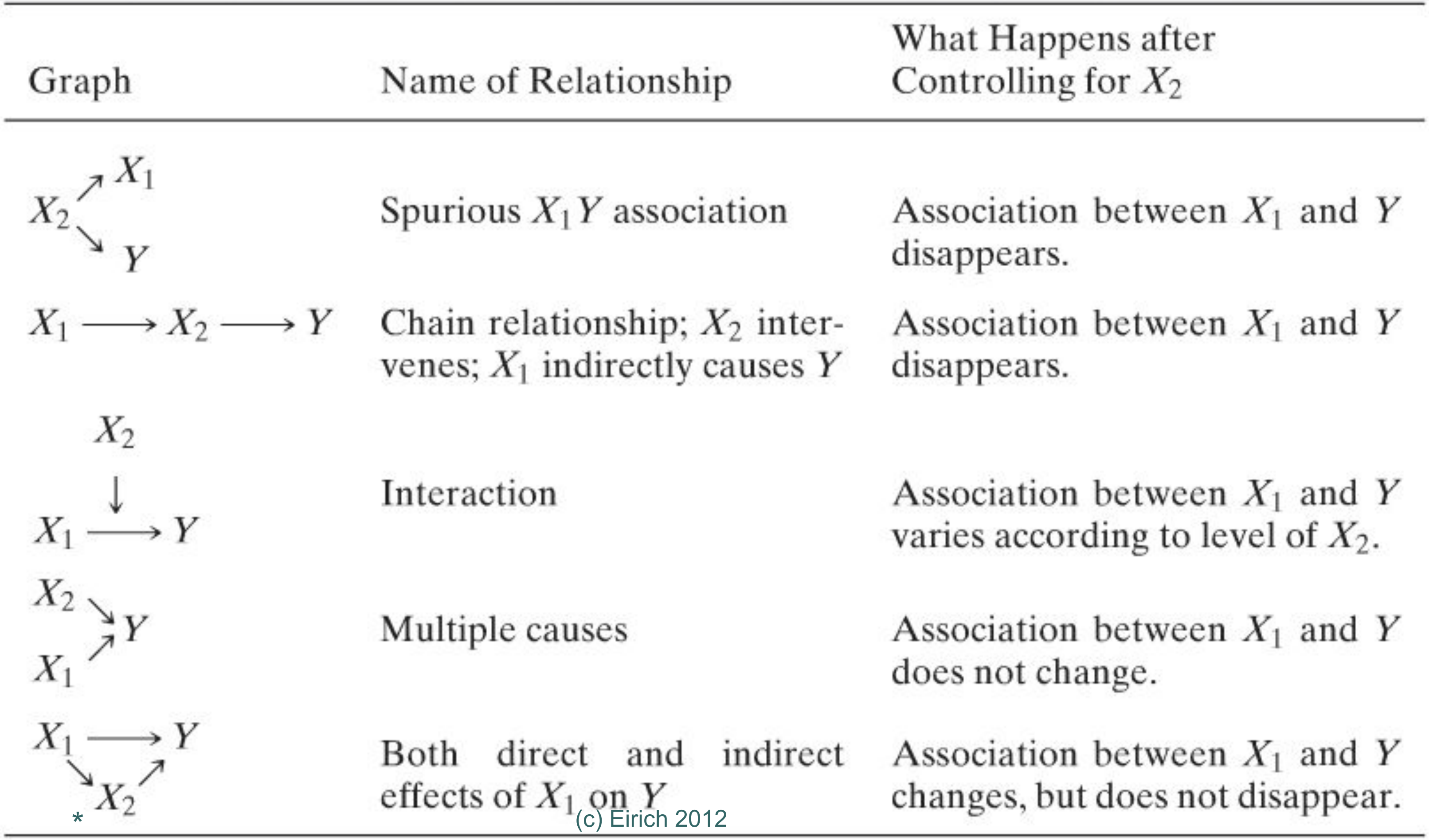
- Interaction: To come in one week!

To account for true relationships

- Multiple Causes: X2 is cause of Y but is unrelated to X1



Another way to look at these relationships …



Graph	Name of Relationship	What Happens after Controlling for X_2
$\begin{tikzcd} & X_1 \\ X_2 \arrow[ur] \arrow[dr] & \\ & Y \end{tikzcd}$	Spurious X_1Y association	Association between X_1 and Y disappears.
$X_1 \longrightarrow X_2 \longrightarrow Y$	Chain relationship; X_2 intervenes; X_1 indirectly causes Y	Association between X_1 and Y disappears.
$\begin{tikzcd} & X_2 \arrow[d] & \\ X_1 \arrow[r, dash] & {} \arrow[r] & Y \end{tikzcd}$	Interaction	Association between X_1 and Y varies according to level of X_2.
$\begin{tikzcd} X_2 \arrow[dr] & \\ & Y \\ X_1 \arrow[ur] & \end{tikzcd}$	Multiple causes	Association between X_1 and Y does not change.
$\begin{tikzcd} X_1 \arrow[rr] \arrow[dr] & & Y \\ & X_2 \arrow[ur] & \end{tikzcd}$	Both direct and indirect effects of X_1 on Y	Association between X_1 and Y changes, but does not disappear.

3. A spurious example

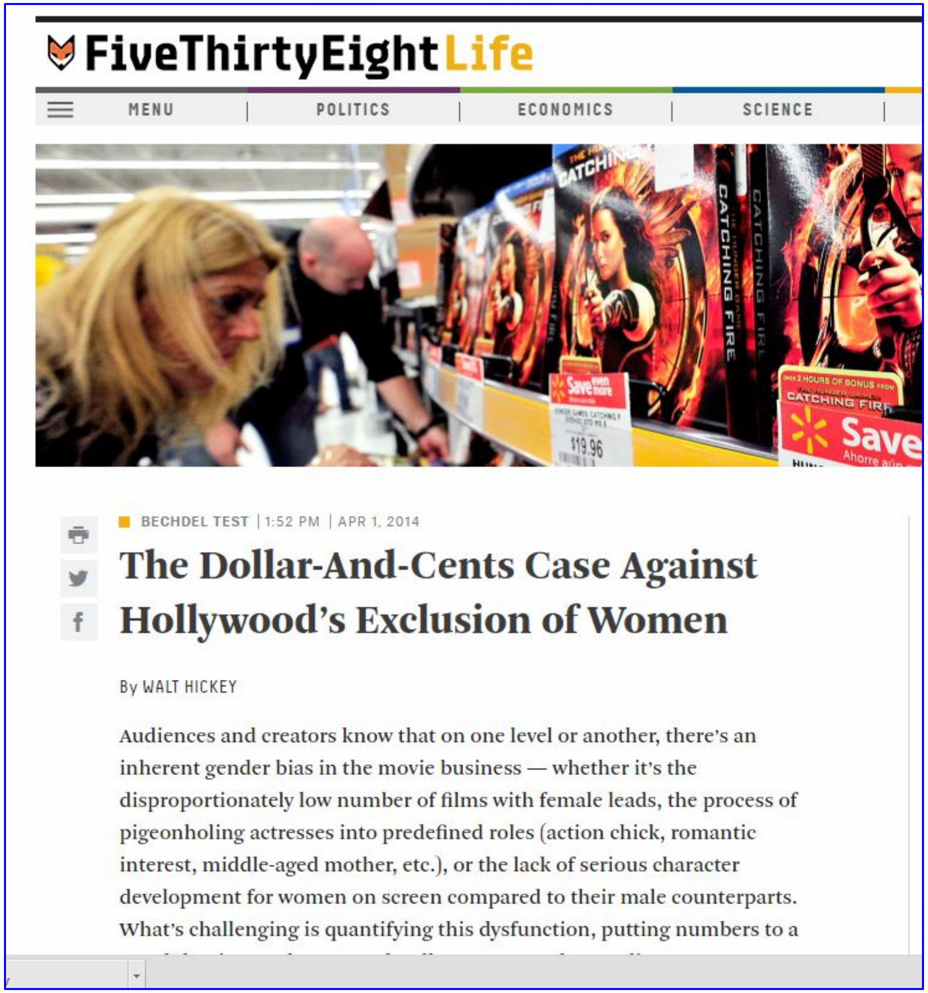
Let’s do a multiple regression example …

$$Y=a + B_1X_1 + B_2X_2 + u$$

Let’s do an example …

Do movies that include women earn less money at the box office?

The inspiration



BECHDEL TEST | 1:52 PM | APR 1, 2014

The Dollar-And-Cents Case Against Hollywood's Exclusion of Women

By WALT HICKEY

Audiences and creators know that on one level or another, there's an inherent gender bias in the movie business — whether it's the disproportionately low number of films with female leads, the process of pigeonholing actresses into predefined roles (action chick, romantic interest, middle-aged mother, etc.), or the lack of serious character development for women on screen compared to their male counterparts. What's challenging is quantifying this dysfunction, putting numbers to a

What does “include” mean?

- The Bechdel test
- Created by cartoonist Alison Bechdel in a 1985 comic strip
- Created 3 criteria to determine if a movie gave female characters a bare minimum of depth:

  — (1) there are at least 2 named women in the picture

The Bechdel test, continued

- Created 3 criteria to determine if a movie gave female characters a bare minimum of depth:

  …

  (2) the 2 women have a conversation with each other at some point, and

  (3) that conversation isn’t about a male character

*

Bechdel example

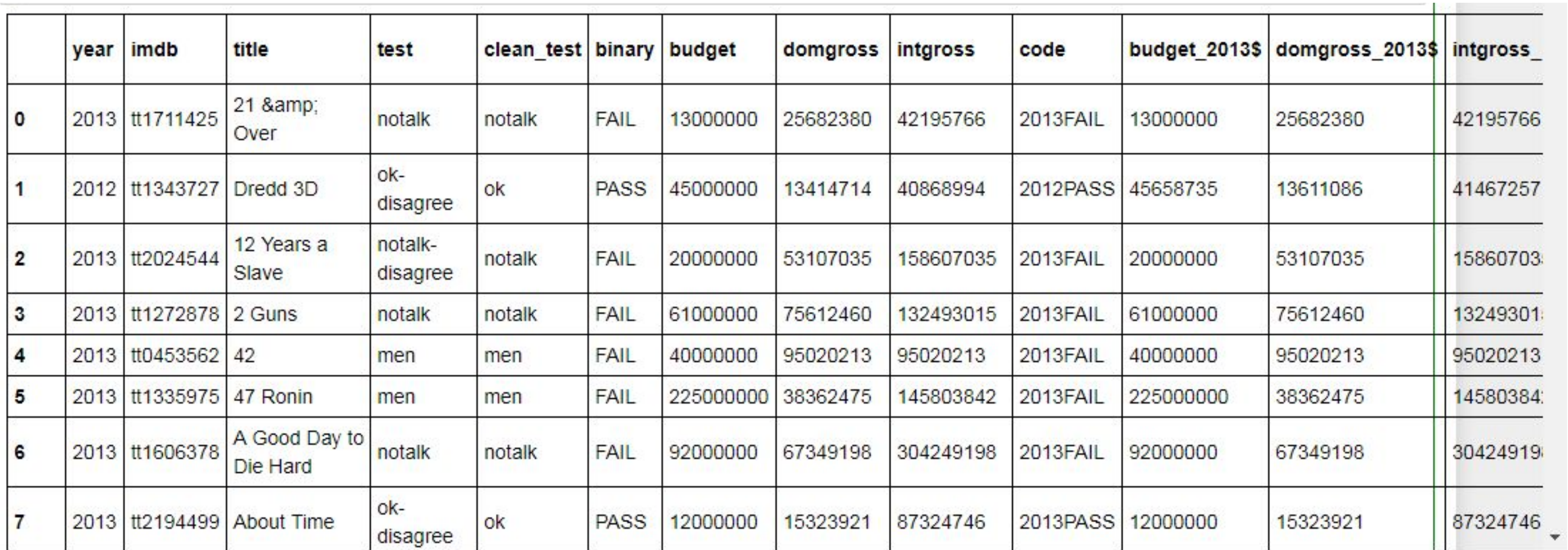
- Preliminary steps:

```
from   future   import division  # In Python 2.x to allow the default floor
division operation of / be replaced by true division
import pandas as pd
import numpy as np
import statsmodels.formula.api as smf
import os
import matplotlib.pyplot as plt
```

Bechdel example

- Data looks like this:

```
os.chdir('C:/Users/gme2101/Desktop/Data Analysis Data') # change working
directory
d = pd.read_csv("movies-bechdel.csv")
d
```



	year	imdb	title	test	clean_test	binary	budget	domgross	intgross	code	budget_2013$	domgross_2013$	intgross_
0	2013	tt1711425	21 &amp; Over	notalk	notalk	FAIL	13000000	25682380	42195766	2013FAIL	13000000	25682380	42195766
1	2012	tt1343727	Dredd 3D	ok-disagree	ok	PASS	45000000	13414714	40868994	2012PASS	45658735	13611086	41467257
2	2013	tt2024544	12 Years a Slave	notalk-disagree	notalk	FAIL	20000000	53107035	158607035	2013FAIL	20000000	53107035	15860703
3	2013	tt1272878	2 Guns	notalk	notalk	FAIL	61000000	75612460	132493015	2013FAIL	61000000	75612460	13249301
4	2013	tt0453562	42	men	men	FAIL	40000000	95020213	95020213	2013FAIL	40000000	95020213	95020213
5	2013	tt1335975	47 Ronin	men	men	FAIL	225000000	38362475	145803842	2013FAIL	225000000	38362475	14580384
6	2013	tt1606378	A Good Day to Die Hard	notalk	notalk	FAIL	92000000	67349198	304249198	2013FAIL	92000000	67349198	30424919
7	2013	tt2194499	About Time	ok-disagree	ok	PASS	12000000	15323921	87324746	2013PASS	12000000	15323921	87324746

The variables

- Create new columns in the DataFrame:

```
d["tg13"] = d["domgross 2013$"] + d["intgross 2013$"]
d["tot gross 13 mil"] = d["tg13"] / (1000000)
d["budget_13_mil"] = d["budget_2013$"] / (1000000)
```

The variables

- Get summary statistics for new variables:

```
d["tot_gross_13_mil"].describe()


count    1776.000000
mean      293.743660
std       403.429718
min         0.001798
25%        55.985323
50%       156.635011
75%       365.059476
max      4838.129232
Name: tot_gross_13_mil, dtype: float64
```

The variables: Gross Revenue

- We can also round the results to a specific number of decimal places (in this case, 3 decimal places) using the following code:

```
a = d["tot gross 13 mil"].describe()
a.map(lambda e: round(e, 3))
```

```
count    1776.000
mean      293.744
std       403.430
min         0.002
25%        55.985
50%       156.635
75%       365.059
max      4838.129
Name: tot_gross_13_mil, dtype: float64
```

The variables: Film budget

- Descriptive statistics:

```
d["budget_13_mil"].describe()

count    1794.000000
mean       55.464608
std        54.918636
min         0.008632
25%        16.068918
50%        36.995786
75%        78.337905
max       461.435929
Name: budget_13_mil, dtype: float64
```

Tabulate the "binary" variable (indicator of Pass/Fail of the Bechdel Test)

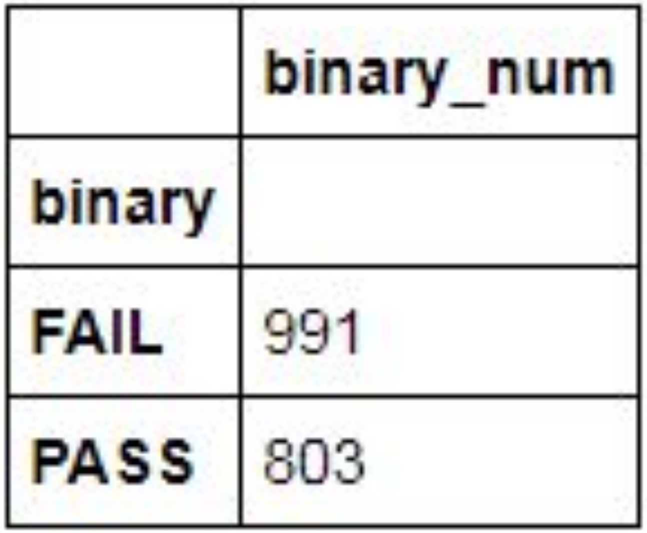
- **Method 1: Create a dictionary:**

```
In [8]:
binary temp = {}
for a, a table in d.groupby("binary"):
    binary temp[a] = len(a_table)
binary temp
Out[8]:
{'FAIL': 991, 'PASS': 803}
```

Tabulate the "binary" variable (indicator of Pass/Fail of the Bechdel Test)

- **Method 2: create a table using Pandas "pivot_table" function:**

```
d["binary num"] = 1
pd.pivot table(d, index = ["binary"], values = ["binary num"], aggfunc =
np.sum, fill_value = 0) # "fill_value = 0" replaces missing values with 0
```



	binary_num
binary	
FAIL	991
PASS	803

(c) Eirich 2012

The simple association

```
lm1 = smf.ols(formula = 'tot_gross_13_mil~binary',data = d).fit()
print (lm1.summary())

                            OLS Regression Results
==============================================================================
Dep. Variable:       tot_gross_13_mil   R-squared:                       0.011
Model:                            OLS   Adj. R-squared:                  0.010
Method:                 Least Squares   F-statistic:                     18.87
Date:                Fri, 09 Jun 2017   Prob (F-statistic):           1.48e-05
Time:                        09:41:47   Log-Likelihood:                -13166.
No. Observations:                1776   AIC:                         2.634e+04
Df Residuals:                    1774   BIC:                         2.635e+04
Df Model:                           1
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept        330.9495     12.810     25.836      0.000       305.826   356.073
binary[T.PASS]   -83.2211     19.158     -4.344      0.000      -120.796   -45.647
==============================================================================
Omnibus:                     1456.112   Durbin-Watson:                   2.032
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            44117.228
Skew:                           3.678   Prob(JB):                         0.00
Kurtosis:                      26.282   Cond. No.                         2.51
==============================================================================
```

Output gives range of residuals

- **Describe quantiles of the residuals (**This output is rounded to one decimal place using the "map(lambda e: round(e,1))" command.)

```
lm1.resid.describe().map(lambda e: round(e,1))
```

```
count    1776.0
mean       -0.0
std       401.3
min      -330.9
25%      -228.0
50%      -136.8
75%        70.1
max      4507.2
dtype: float64
```

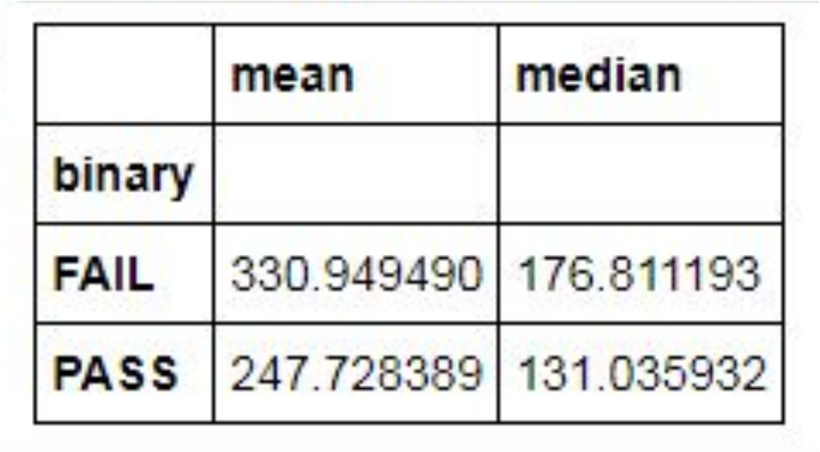
The simple association

```
==================================================================================
                     coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept        330.9495     12.810     25.836      0.000       305.826   356.073
binary[T.PASS]   -83.2211     19.158     -4.344      0.000      -120.796   -45.647
==================================================================================
```

On average, if a film passes the Bechdel test, its total gross revenue (in 2013 $s) is $83M less than a movie that fails the Bechdel test (p<.0001)

There is the same number -- difference in two means

```
pd.pivot_table(d, index = "binary", values = "tot_gross_13_mil", aggfunc = [np.mean,
np.median])
```



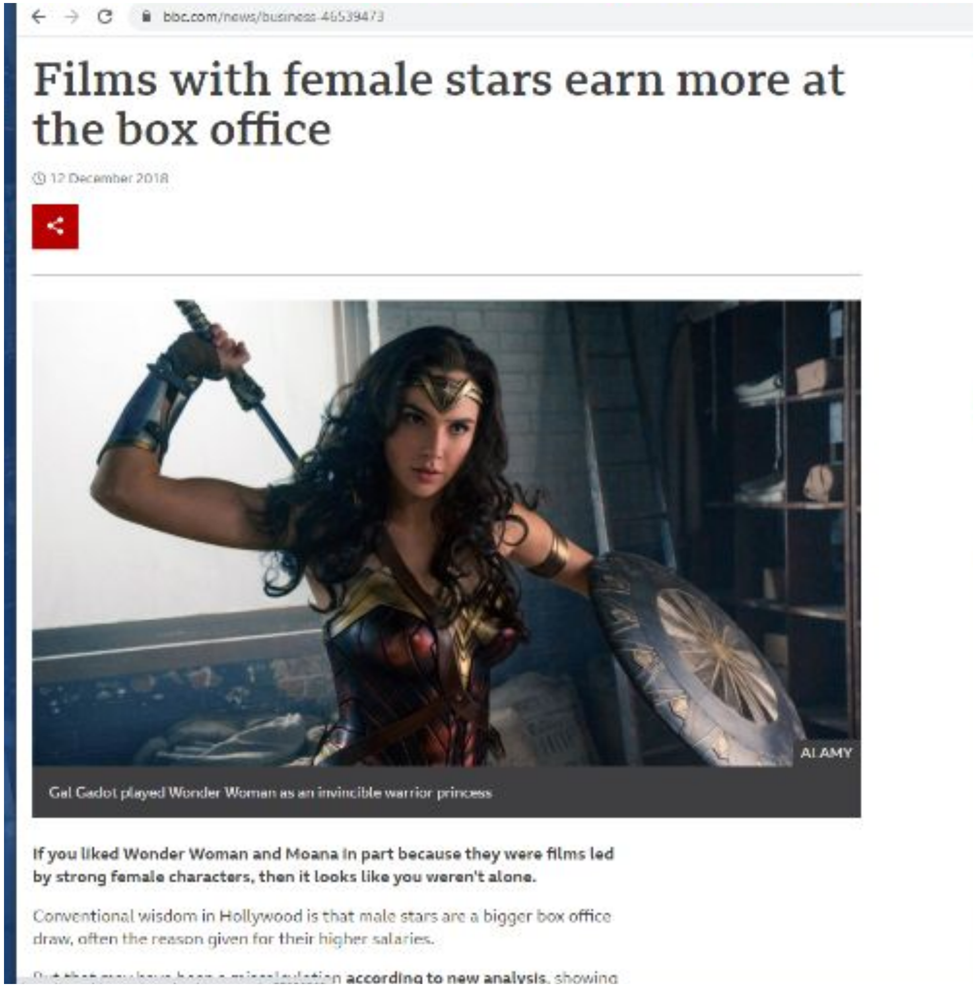
	mean	median
binary		
FAIL	330.949490	176.811193
PASS	247.728389	131.035932

On average, if a film passes the Bechdel test, its total gross revenue (in 2013 $s) is $83M less than a movie that fails the Bechdel test (p<.0001)

Alternative explanations?

Why would a film that includes women earn so much less money than one that excludes women?

This dynamic appears to be changing in recent years...



bbc.com/news/business-46539473

Films with female stars earn more at the box office

12 December 2018

ALAMY

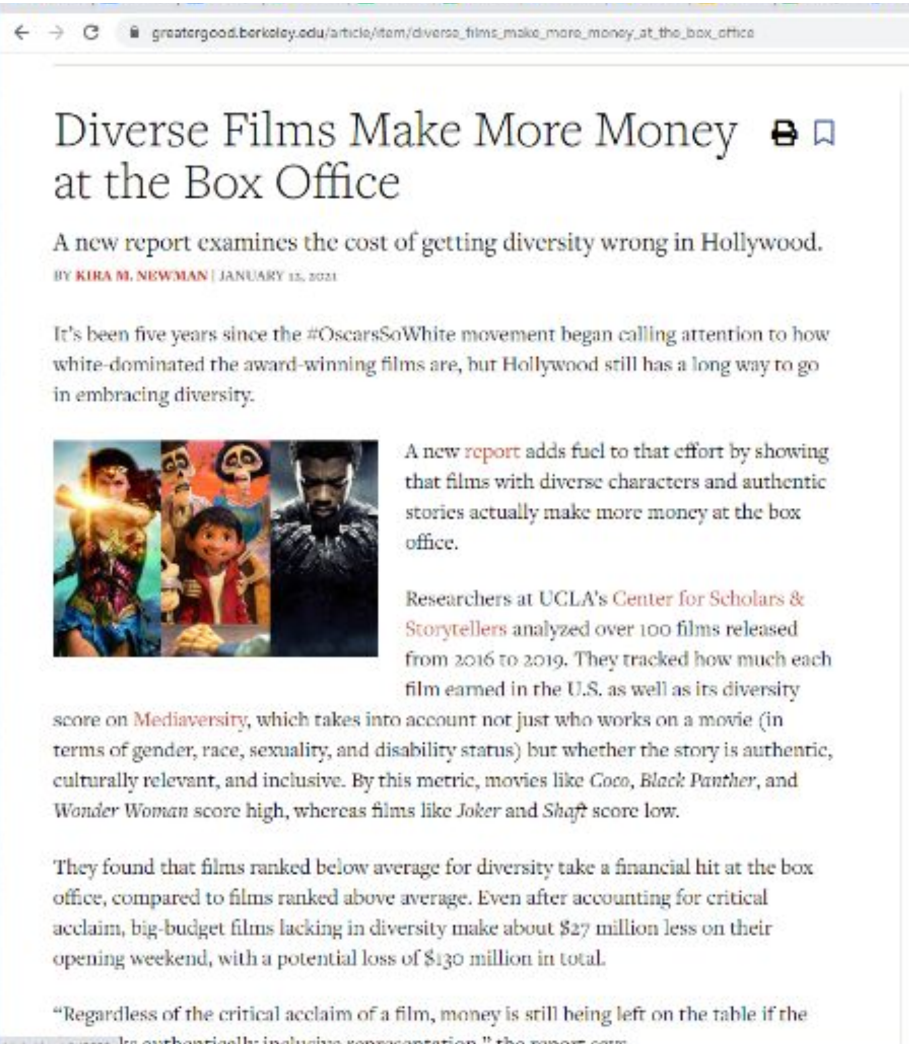
Gal Gadot played Wonder Woman as an invincible warrior princess

If you liked Wonder Woman and Moana in part because they were films led by strong female characters, then it looks like you weren't alone.

Conventional wisdom in Hollywood is that male stars are a bigger box office draw, often the reason given for their higher salaries.

But that may have been a miscalculation according to new analysis, showing

This dynamic appears to be changing in recent years...



greatergood.berkeley.edu/article/item/diverse_films_make_more_money_at_the_box_office

Diverse Films Make More Money at the Box Office

A new report examines the cost of getting diversity wrong in Hollywood.

BY KIRA M. NEWMAN | JANUARY 15, 2021

It's been five years since the #OscarsSoWhite movement began calling attention to how white-dominated the award-winning films are, but Hollywood still has a long way to go in embracing diversity.

A new report adds fuel to that effort by showing that films with diverse characters and authentic stories actually make more money at the box office.

Researchers at UCLA's Center for Scholars & Storytellers analyzed over 100 films released from 2016 to 2019. They tracked how much each film earned in the U.S. as well as its diversity score on Mediaversity, which takes into account not just who works on a movie (in terms of gender, race, sexuality, and disability status) but whether the story is authentic, culturally relevant, and inclusive. By this metric, movies like *Coco*, *Black Panther*, and *Wonder Woman* score high, whereas films like *Joker* and *Shaft* score low.

They found that films ranked below average for diversity take a financial hit at the box office, compared to films ranked above average. Even after accounting for critical acclaim, big-budget films lacking in diversity make about $27 million less on their opening weekend, with a potential loss of $130 million in total.

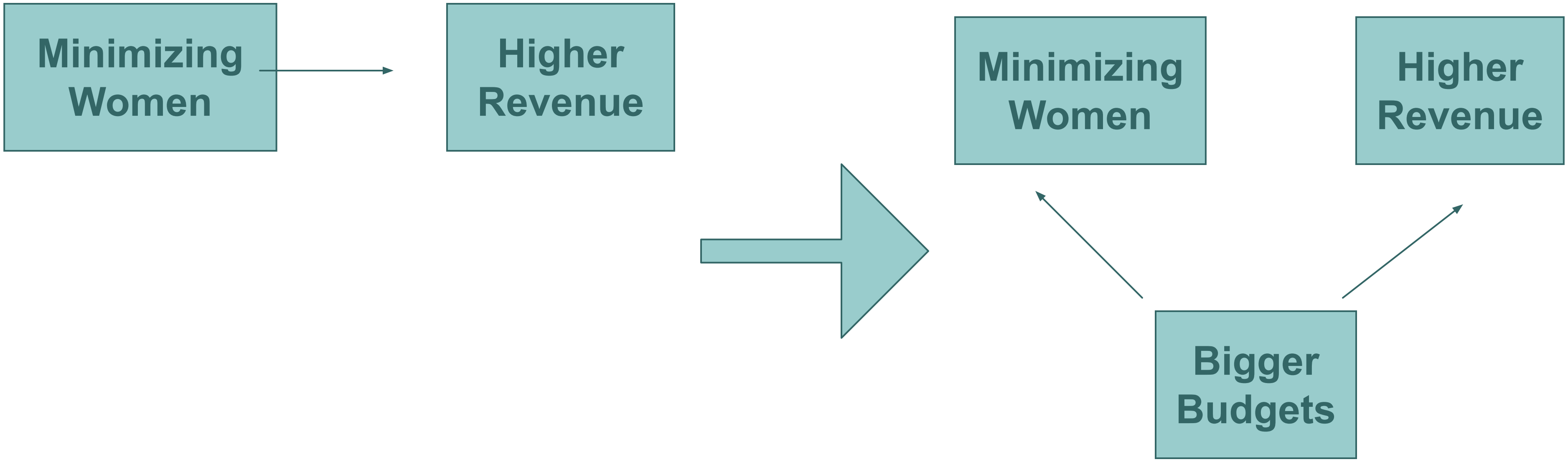
"Regardless of the critical acclaim of a film, money is still being left on the table if the

One avenue to investigate...

Perhaps higher-grossing movies are just “bigger” movies that cost more to make in the first place, and movies that don’t tend to include women also have higher budgets. So we should control for the film’s budget. If we control for the film’s budget, the effect of not including women may disappear.

To account for true relationships

- Spuriousness: Some omitted variable(s) is fully driving the relationship between our X and Y



Minimizing Women

Higher Revenue

Minimizing Women

Higher Revenue

Bigger Budgets

The result is …

```
lm2 = smf.ols(formula = "tot_gross_13_mil ~ binary + budget_13_mil", data = 
d).fit()
print (lm2.summary()) # linear regression model output
lm2.resid.describe().map(lambda f: round(f,1)) # summary of residuals, 
rounded to one decimal place
```

```
             OLS Regression Results
==============================================================================
Dep. Variable:       tot_gross_13_mil   R-squared:                       0.316
Model:                            OLS   Adj. R-squared:                  0.315
Method:                 Least Squares   F-statistic:                     408.7
Date:                Fri, 09 Jun 2017   Prob (F-statistic):          1.10e-146
Time:                        09:41:50   Log-Likelihood:                -12839.
No. Observations:                1776   AIC:                         2.568e+04
Df Residuals:                    1773   BIC:                         2.570e+04
Df Model:                           2
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept         71.4731     14.099      5.069      0.000        43.820    99.126
binary[T.PASS]   -14.9222     16.123     -0.926      0.355       -46.543    16.699
budget_13_mil      4.0963      0.146     28.108      0.000         3.810     4.382
==================================================================================
Omnibus:                     1696.847   Durbin-Watson:                   1.942
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           104439.408
Skew:                           4.389   Prob(JB):                         0.00
Kurtosis:                      39.528   Cond. No.                         190.
==============================================================================
```

The result is …

```
==================================================================================
                      coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept          71.4731     14.099      5.069      0.000        43.820    99.126
binary[T.PASS]    -14.9222     16.123     -0.926      0.355       -46.543    16.699
budget_13_mil       4.0963      0.146     28.108      0.000         3.810     4.382
```

With budget held constant, if a film passes the Bechdel test, it only earns $15M less than a film that fails the test, but the difference is not statistically significant (p=.355)

Or …

```
==================================================================================
                   coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept       71.4731     14.099      5.069      0.000        43.820    99.126
binary[T.PASS] -14.9222     16.123     -0.926      0.355       -46.543    16.699
budget_13_mil    4.0963      0.146     28.108      0.000         3.810     4.382
```

If two films have the same budget, but one film showcases women, that film will earn (on average) a statistically insignificant $15M less

We can think of it the other way too

The result is …

```
==================================================================================
                     coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept         71.4731     14.099      5.069      0.000        43.820    99.126
binary[T.PASS]   -14.9222     16.123     -0.926      0.355       -46.543    16.699
budget_13_mil      4.0963      0.146     28.108      0.000         3.810     4.382
```

Holding passing the Bechdel test constant, for each additional $1M a movie has in its budget, the movie (on average) grosses $4M (p<.0001)

Or …

```
==================================================================================
                      coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept          71.4731     14.099      5.069      0.000        43.820    99.126
binary[T.PASS]    -14.9222     16.123     -0.926      0.355       -46.543    16.699
budget_13_mil       4.0963      0.146     28.108      0.000         3.810     4.382
```

If two films both passed the Bechdel test, but one film spent an additional $1M on its budget, that movie (on average) will gross another $4M (p<.0001)

How to talk about control variables:

"Controlling for all other variables…"
"Holding all other variables constant…"
"Net of all other variables…"
"Ceteris paribus…"
"All else being equal"

What’s it mean to hold X2 constant?

Create a summary table of the budget variable:

```
d.budget_13_mil.describe()

count    1794.000000
mean       55.464608
std        54.918636
min         0.008632
25%        16.068918
50%        36.995786
75%        78.337905
max       461.435929
Name: budget_13_mil, dtype: float64
```

Make budget into a categorical variable

What’s it mean to hold X2 constant?

Here we are categorizing movies based on their budget, using the pd.cut function in Pandas:

number of movies in each category:

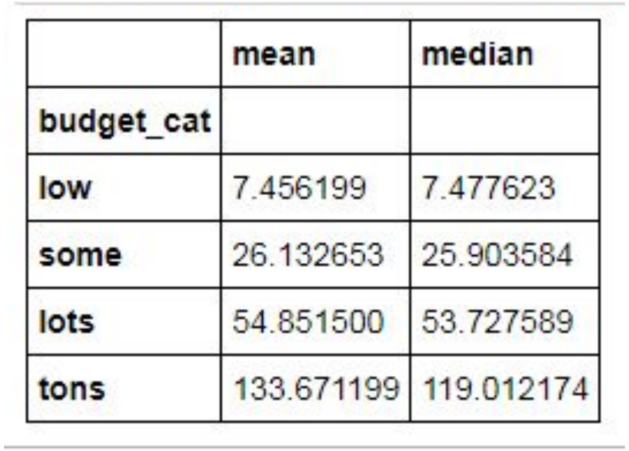
```
d["budget_cat_num"] = 1
d["budget_cat"] = pd.cut(d.budget_13_mil, bins = [-1, 16.0700, 37, 78.34,
462], labels = ["low", "some", "lots", "tons"])
pd.pivot_table(d, index = "budget_cat", values = "budget_cat_num", aggfunc
= np.sum)
```

```
budget_cat
low      449
some     453
lots     443
tons     449
Name: budget_cat_num, dtype: int64
```

What’s it mean to hold X2 constant?

summarize by mean and median:

```
pd.pivot_table(d, index = "budget_cat", values = "budget_13_mil", aggfunc =
[np.mean, np.median])
```



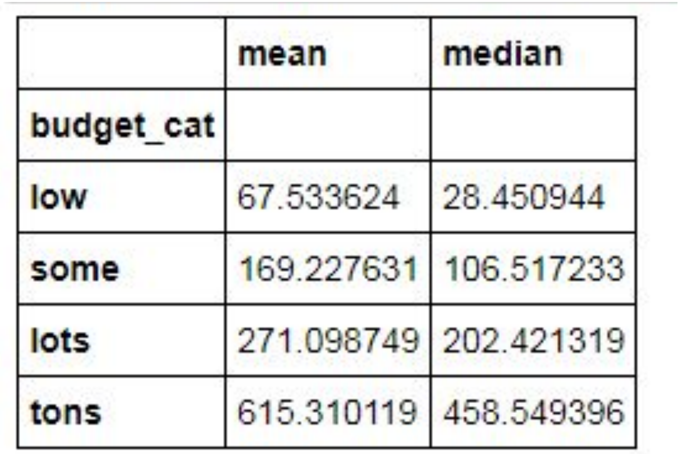
	mean	median
budget_cat		
low	7.456199	7.477623
some	26.132653	25.903584
lots	54.851500	53.727589
tons	133.671199	119.012174

Looking at “passers”

Create two subsets and summary tables
Here, we are making two subsets of the overall data set - one for movies that pass the Bechdel test ("passers"), and one for movies that fail the Bechdel test("failers"). To create the "passers" subset, we select the rows where the variable "binary" = "PASS". We can summarize the subsets using the pivot table function in Pandas.

First, looking at passers:

```
passers = d[d["binary"] == "PASS"]
failers = d[d["binary"] == "FAIL"]
pd.pivot_table(passers, index = "budget_cat", values = "tot_gross_13_mil",
aggfunc = [np.mean, np.median])
```

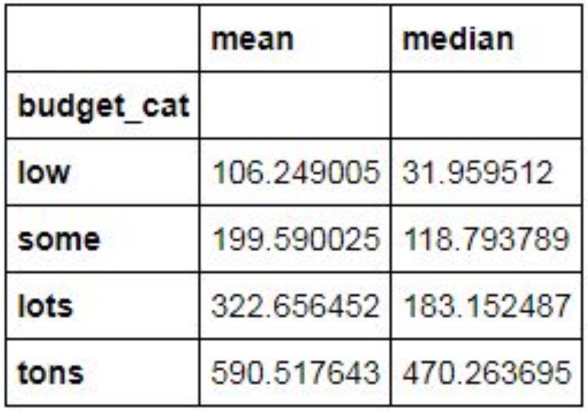


	mean	median
budget_cat		
low	67.533624	28.450944
some	169.227631	106.517233
lots	271.098749	202.421319
tons	615.310119	458.549396

Look at “failers”

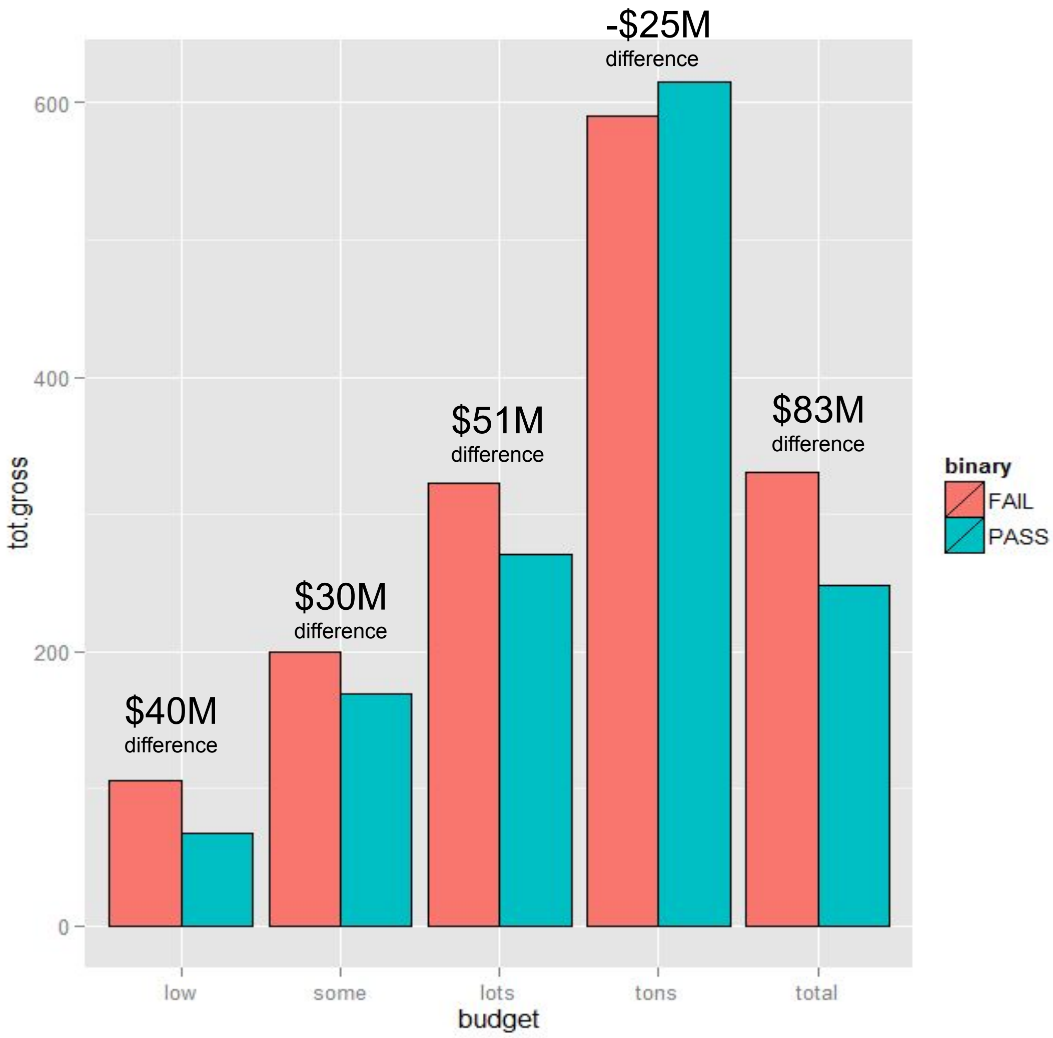
Next, looking at failers:

```
In [19]:
pd.pivot_table(failers, index = "budget_cat", values = "tot_gross_13_mil",
aggfunc = [np.mean, np.median])
```



	mean	**median**
budget_cat		
low	106.249005	31.959512
some	199.590025	118.793789
lots	322.656452	183.152487
tons	590.517643	470.263695

What’s it mean to hold X2 constant?

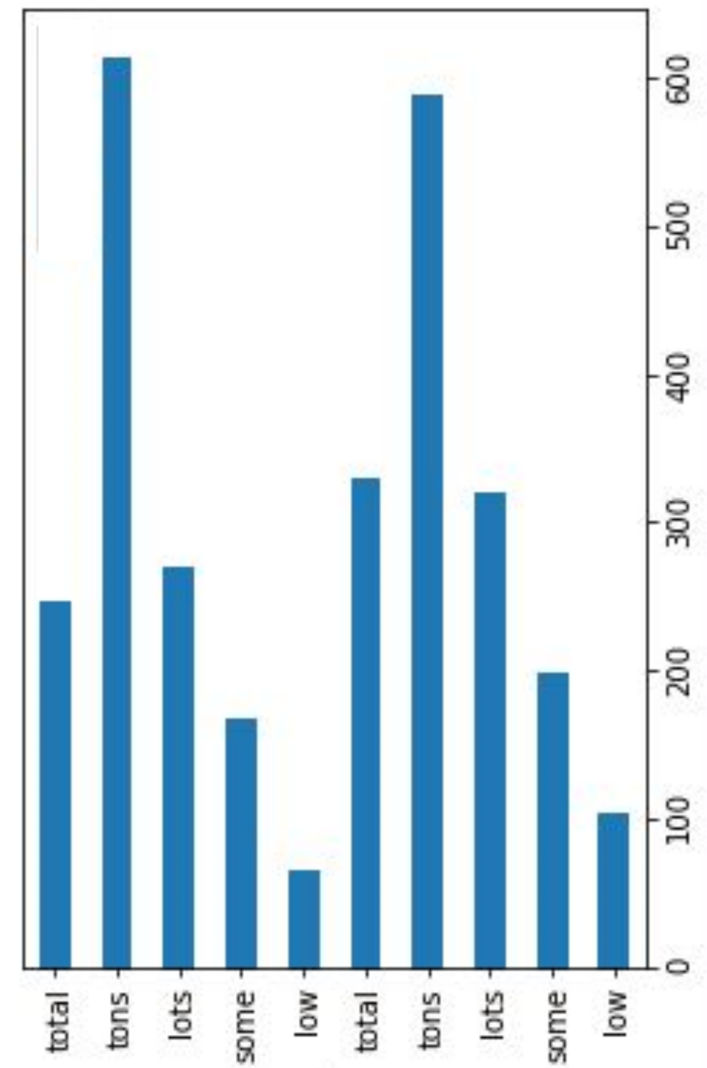


All together, that translates into a B_{pass}=**-14.9** (n.s.) coefficient on passing the Bechdel test, with budget being “controlled for”

How did I make that graph? (in R)

```
> df1 <- data.frame(binary = factor(c("FAIL","FAIL","FAIL","FAIL", "FAIL",
"PASS",  "PASS",  "PASS",  "PASS",  "PASS" )), budget= factor(c("low",
"some", "lots", "tons", "total", "low", "some", "lots", "tons", "total"),
levels=c("low", "some", "lots", "tons", "total")), tot.gross = c( 106.2490,
199.5900, 322.6565, 590.5176, 330.9495, 67.53362, 169.22763, 271.09875,
615.31012, 247.7284))
> df1
   binary budget tot.gross
1    FAIL    low 106.24900
2    FAIL   some 199.59000
3    FAIL   lots 322.65650
4    FAIL   tons 590.51760
5    FAIL  total 330.94950
6    PASS    low  67.53362
7    PASS   some 169.22763
8    PASS   lots 271.09875
9    PASS   tons 615.31012
```

What’s it mean to hold X2 constant?



--------Passers------ -----Failers------

All together, that translates into a B_{pass}=**-14.9** (n.s.) coefficient on passing the Bechdel test, with budget being “controlled for”

How did I make that graph?

```
# graph (slide 40)
data = {'binary': ['FAIL','FAIL','FAIL','FAIL','FAIL','PASS','PASS',
'PASS', 'PASS', 'PASS'], 'budget':
['low','some','lots','tons','total','low','some','lots','tons','total'],
'tot.gross':[106.2490, 199.5900, 322.6565, 590.5176, 330.9495, 67.53362,
169.22763, 271.09875, 615.31012,247.7284]}
df1 = pd.DataFrame(data)
df1.plot(kind = 'barh', x = 'budget', y = 'tot.gross')
plt.show()
```

This also means ...

```
binary_dict = {"PASS":1, "FAIL":0}
d["pass"] = d["binary"].map(binary_dict.get)
pd.pivot_table(d, index = "budget_cat", values = "pass", aggfunc = np.mean)

budget_cat
low       0.518931
some      0.518764
lots      0.419865
tons      0.331849
Name: pass, dtype: float64
```

Additional Information - proportion of films passing the Bechdel test by budget category. That also means that just fewer high budget films passed the Bechdel test

Is this relationship spurious?

Simple Regression B_1 = 83.2***

vs.

Multiple Regression B_1 = 14.9 (n.s)

Is this relationship spurious?

The original (highly significant) B shrinks to non-significance, once we control for film budget size.

The higher revenues that non-Bechdel movies display are due to the fact that higher budget films are less likely to pass the Bechdel test.

Higher budget films are less likely to pass the Bechdel test, or vice versa

```
lm3 = smf.ols(formula = "budget_13_mil ~ binary", data = d).fit()
print (lm3.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:          budget_13_mil   R-squared:                       0.023
Model:                            OLS   Adj. R-squared:                  0.022
Method:                 Least Squares   F-statistic:                     41.63
Date:                Thu, 18 May 2017   Prob (F-statistic):           1.41e-10
Time:                        14:57:10   Log-Likelihood:                -9711.0
No. Observations:                1794   AIC:                         1.943e+04
Df Residuals:                    1792   BIC:                         1.944e+04
Df Model:                           1
Covariance Type:            nonrobust
==================================================================================
                     coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept         62.9116      1.725     36.468      0.000        59.528    66.295
binary[T.PASS]   -16.6374      2.579     -6.452      0.000       -21.695   -11.580
==============================================================================
Omnibus:                      617.742   Durbin-Watson:                   1.916
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2084.633
Skew:                           1.714   Prob(JB):                         0.00
Kurtosis:                       7.018   Cond. No.                         2.51
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Higher budget films are less likely to pass the Bechdel test, or vice versa

```
==================================================================================
                       coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------
Intercept           62.9116      1.725     36.468      0.000        59.528    66.295
binary[T.PASS]     -16.6374      2.579     -6.452      0.000       -21.695   -11.580
==============================================================================
```

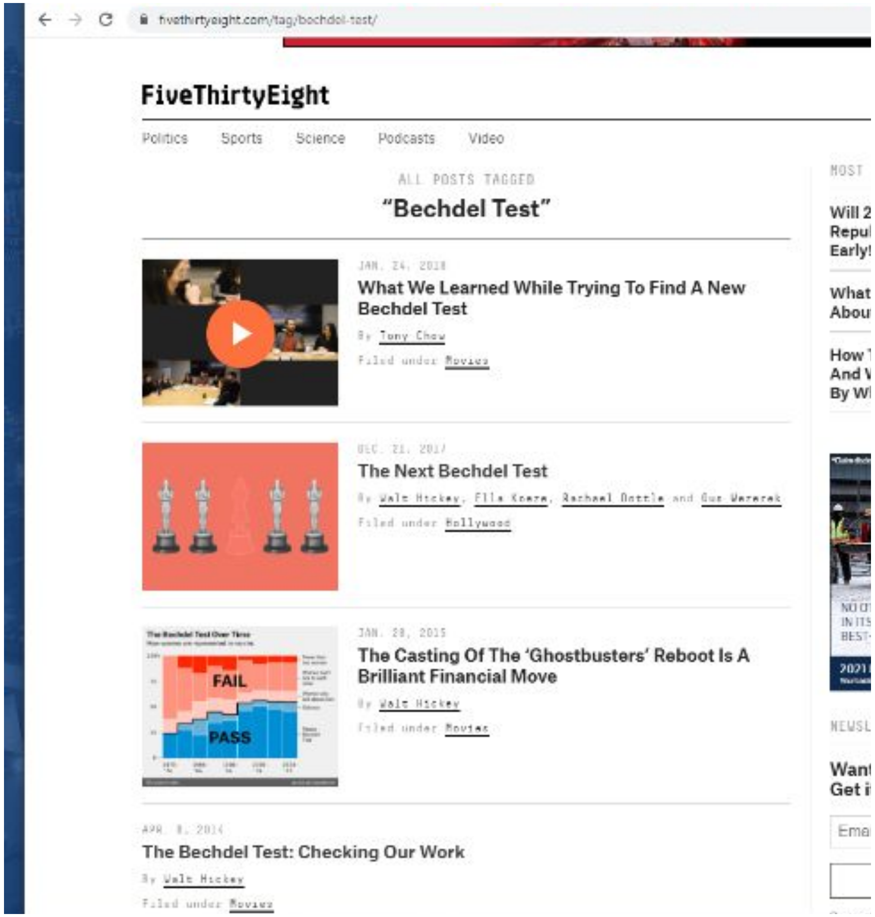
If a film passes the Bechdel test, its budget is (in 2013 $s) $16M less than a movie that fails the Bechdel test *(p<.0001)

Is this relationship spurious?

Other interpretations are also possible.

BTW - We should return to this example when we do log transformations and median regression and generalized linear models (with Gamma distributions)

More since then … Check it out!



fivethirtyeight.com/tag/bechdel-test/

FiveThirtyEight

Politics
Sports
Science
Podcasts
Video

ALL POSTS TAGGED

"Bechdel Test"

JAN. 24, 2018

What We Learned While Trying To Find A New Bechdel Test

By Tony Chow

Filed under Movies

DEC. 21, 2017

The Next Bechdel Test

By Walt Hickey, Ella Koeze, Rachael Dottle and Gus Wezerek

Filed under Hollywood

The Bechdel Test Over Time
FAIL
PASS

JAN. 28, 2015

The Casting Of The 'Ghostbusters' Reboot Is A Brilliant Financial Move

By Walt Hickey

Filed under Movies

APR. 8, 2014

The Bechdel Test: Checking Our Work

By Walt Hickey

Filed under Movies

Let’s do another regression example …

Does marriage lead you to know more words?

Married vs. Everyone Else

WORDSUM = No. of words correct out of 10

Married = 6.12
Others = 5.87
Diff. = 0.25

Our simple model

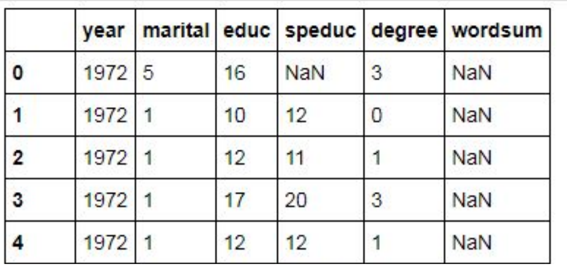
$$Y=a + B_1X_1 + u$$

$$Wordsum=a + B_1(Married) + u$$

Results

This file is too big - just use some columns

```
d = pd.read_csv("GSS_Cum.csv", usecols=["marital", "educ", "year", "speduc", 
"educ", "wordsum", "degree"])
```



	year	marital	educ	speduc	degree	wordsum
0	1972	5	16	NaN	3	NaN
1	1972	1	10	12	0	NaN
2	1972	1	12	11	1	NaN
3	1972	1	17	20	3	NaN
4	1972	1	12	12	1	NaN

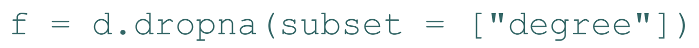
Results

Make “married”

```
d["married"] = pd.get_dummies(d['marital'])[1.0] # set variable 'married' to be 1
where-ever variable marital = 1.0
```

Results

Drop missing values in the "degree" variable: Here we are creating a subset of "d" called "f" which drops the na values in the "degree" variable. The "dropna" function used here only creates a copy and does not affect the original dataset.



```
f = d.dropna(subset = ["degree"])
```

We need to have exactly the same observations across models to compare them; the *dropna* function assures us of this

```
mwlml = smf.ols(formula = "wordsum ~ married", data = f).fit()
print (mwlml.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                wordsum   R-squared:                       0.004
Model:                            OLS   Adj. R-squared:                  0.004
Method:                 Least Squares   F-statistic:                     98.43
Date:                Fri, 09 Jun 2017   Prob (F-statistic):           3.69e-23
Time:                        09:46:24   Log-Likelihood:                -58529.
No. Observations:               26872   AIC:                         1.171e+05
Df Residuals:                   26870   BIC:                         1.171e+05
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      5.8657      0.019    307.392      0.000         5.828     5.903
married        0.2592      0.026      9.921      0.000         0.208     0.310
==============================================================================
Omnibus:                      222.603   Durbin-Watson:                   1.695
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              222.103
Skew:                          -0.209   Prob(JB):                     5.90e-49
Kurtosis:                       2.845   Cond. No.                         2.70
==============================================================================
```

On average, a married person (relatively to a single person) earns 0.26 points higher on the vocabulary test (p*<.000)

Alternative explanations?

Alternative explanations?

But perhaps it is not being married per se that makes someone score higher on the vocab test, but it is instead, higher educated people are more likely to get married (and stay married), so that is why it looks like marriage makes you appear to know more words.

If we were to control for socioeconomic status (proxied by degree), the effect of marriage on Wordsum should go down dramatically.

Let’s see.

The Complex Model

$$Y=a + B_1X_1 + B_2X_2 + u$$

$$\text{Wordsum}=a + B_1(\text{Married}) + B_2(\text{Degree}) + u$$

Results

```
mwlm2 = smf.ols(formula = "wordsum ~ married + degree", data = d).fit()
print (mwlm2.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                wordsum   R-squared:                       0.206
Model:                            OLS   Adj. R-squared:                  0.206
Method:                 Least Squares   F-statistic:                     3481.
Date:                Thu, 06 Apr 2017   Prob (F-statistic):               0.00
Time:                        11:00:01   Log-Likelihood:                -55482.
No. Observations:               26872   AIC:                         1.110e+05
Df Residuals:                   26869   BIC:                         1.110e+05
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      4.8004      0.021    224.746      0.000         4.758     4.842
married        0.1367      0.023      5.849      0.000         0.091     0.183
degree         0.8294      0.010     82.697      0.000         0.810     0.849
==============================================================================
Omnibus:                      366.120   Durbin-Watson:                   1.820
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              397.729
Skew:      *                   -0.259   Prob(JB):                     4.31e-87
Kurtosis:                       3.293   Cond. No.                         4.87
==============================================================================
```

Results

```
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      4.8004      0.021    224.746      0.000         4.758     4.842
married        0.1367      0.023      5.849      0.000         0.091     0.183
degree         0.8294      0.010     82.697      0.000         0.810     0.849
==============================================================================
```

On average, with degree held constant, a married person gets 0.137 more words right than a single person.

Or …

```
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      4.8004      0.021    224.746      0.000         4.758     4.842
married        0.1367      0.023      5.849      0.000         0.091     0.183
degree         0.8294      0.010     82.697      0.000         0.810     0.849
==============================================================================
```

If there are two married people, but one has a degree higher than the other, that person scores 0.829 words higher than the lesser educated person (p<.000)

What about this relationship?

Simple Regression B1 = 0.26

vs.

Multiple Regression B1 = 0.14

Is this relationship spurious?

The B does shrink when Degree is added – and by a lot.

The higher score on Wordsum by married people appears to be partly due to their higher educations that led them to get married in the first place.

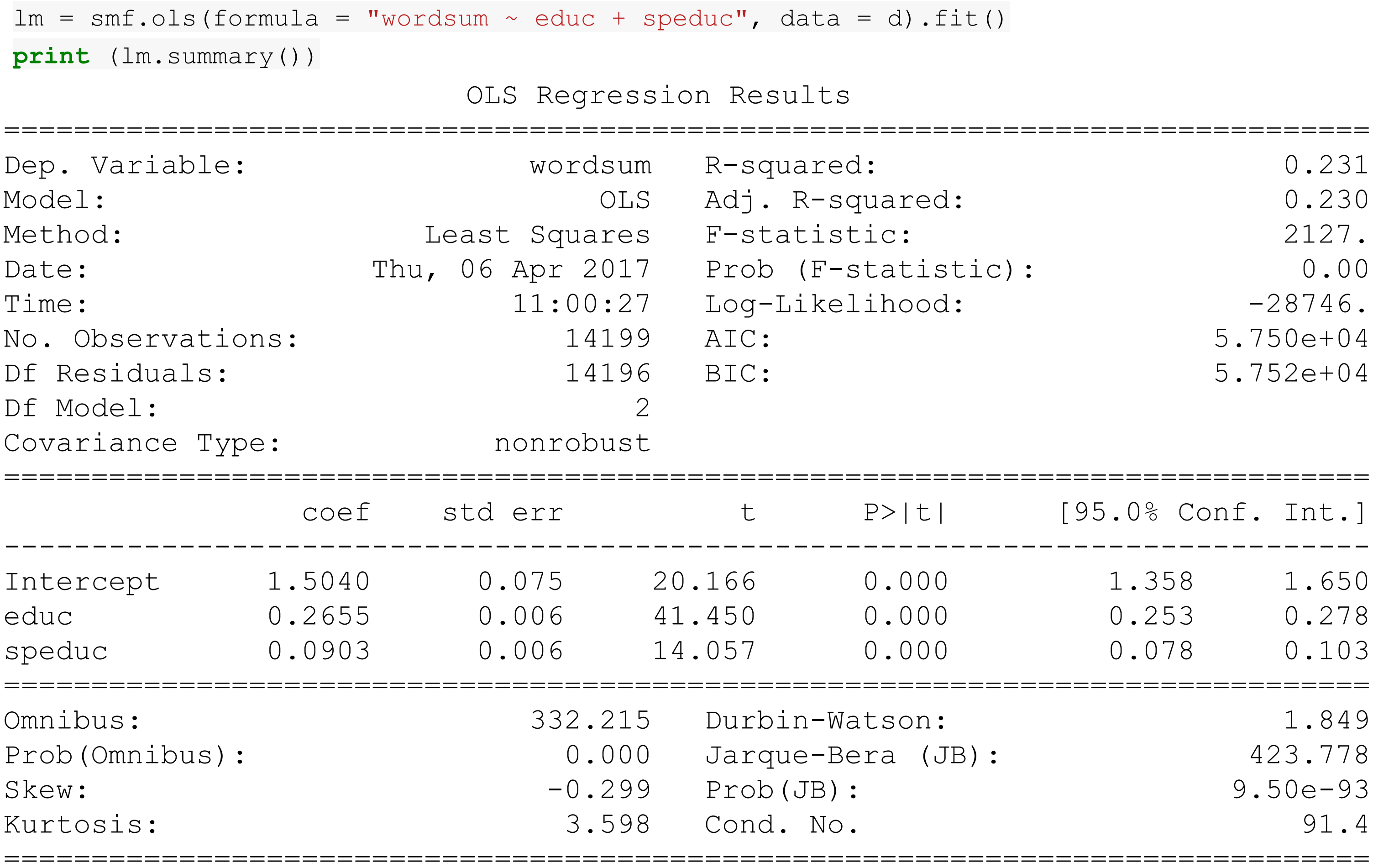
This is often called a "compositional effect," because it is because of the educational composition of married people vs. unmarried that partly drives the results, not marriage per se.

Is this relationship spurious?

But the original “marriage effect” is still statistically significant. So maybe there is something to this …

Other interpretations are possible

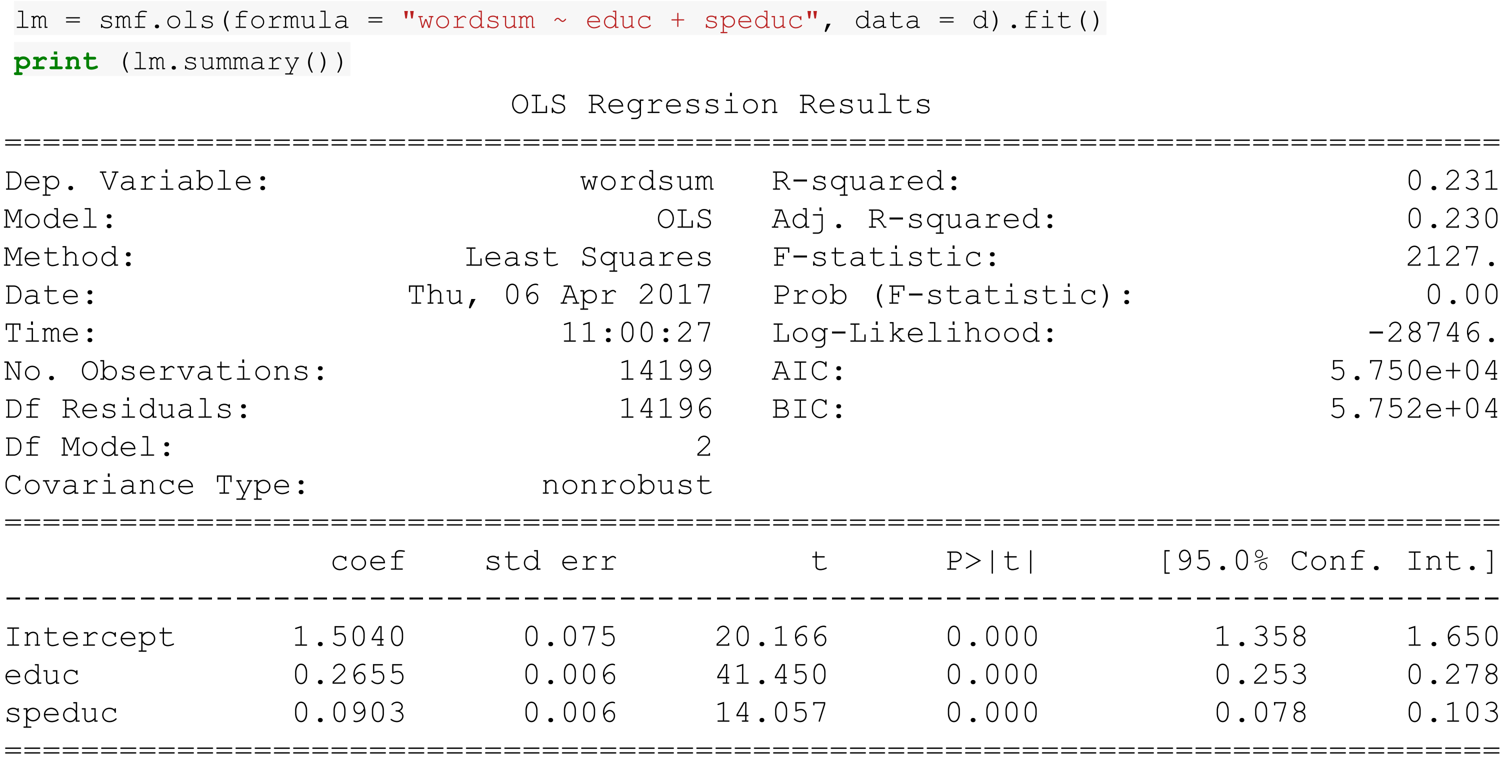
Think about this, for instance



```python
lm = smf.ols(formula = "wordsum ~ educ + speduc", data = d).fit()
print (lm.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                wordsum   R-squared:                       0.231
Model:                            OLS   Adj. R-squared:                  0.230
Method:                 Least Squares   F-statistic:                     2127.
Date:                Thu, 06 Apr 2017   Prob (F-statistic):               0.00
Time:                        11:00:27   Log-Likelihood:                -28746.
No. Observations:               14199   AIC:                         5.750e+04
Df Residuals:                   14196   BIC:                         5.752e+04
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      1.5040      0.075     20.166      0.000         1.358     1.650
educ           0.2655      0.006     41.450      0.000         0.253     0.278
speduc         0.0903      0.006     14.057      0.000         0.078     0.103
==============================================================================
Omnibus:                      332.215   Durbin-Watson:                   1.849
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              423.778
Skew:                          -0.299   Prob(JB):                     9.50e-93
Kurtosis:                       3.598   Cond. No.                         91.4
==============================================================================
```

Think about this, for instance



```
lm = smf.ols(formula = "wordsum ~ educ + speduc", data = d).fit()
print (lm.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                wordsum   R-squared:                       0.231
Model:                            OLS   Adj. R-squared:                  0.230
Method:                 Least Squares   F-statistic:                     2127.
Date:                Thu, 06 Apr 2017   Prob (F-statistic):               0.00
Time:                        11:00:27   Log-Likelihood:                -28746.
No. Observations:               14199   AIC:                         5.750e+04
Df Residuals:                   14196   BIC:                         5.752e+04
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      1.5040      0.075     20.166      0.000         1.358     1.650
educ           0.2655      0.006     41.450      0.000         0.253     0.278
speduc         0.0903      0.006     14.057      0.000         0.078     0.103
==============================================================================
```

Controlling for a person's own level of education, for each year more schooling their spouse has, on average, their word score goes up by 0.09 (p<.000)

Interactions

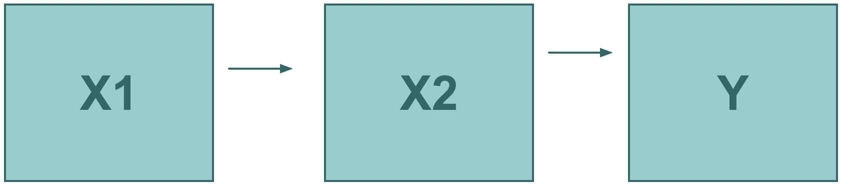
(We will return to this example because there appears to be an interaction between married x degree … but that is the week after next)

4. A mediation example

To account for true relationships

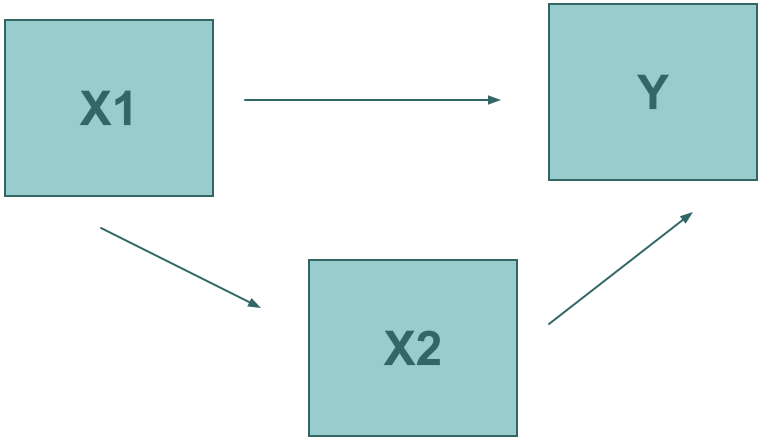
- Mediation: Some variable(s) is the mechanism behind the relationship between our X and Y

Chain Mechanism



X2 fully accounts for the relationship between X1 and Y

Both Direct and Indirect Effects



Let’s do an example …

Do people express lower levels of happiness, after the Great Recession?

Mediation: Our simple model

$$Y=a + B_1X_{1\ +}\ u$$

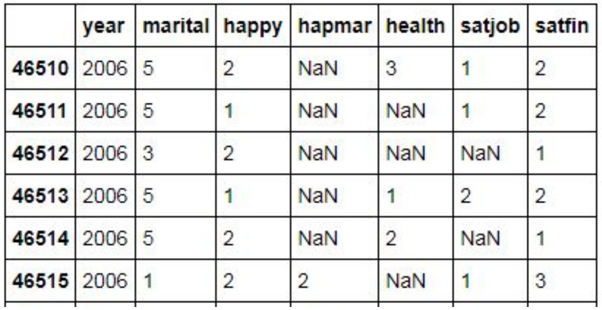
R’s Happiness Score=
a + B_1(Year 2010, compared with 2006) + u

Mediation: Our simple model - Some recodes ...

```
d = pd.read_csv("GSS_Cum.csv", usecols=["happy", "marital", "year", "satfin",
"hapmar", "health", "satjob"])

GSS06and10 = d[(d["year"] == 2006) | (d["year"] == 2010)]

GSS06and10
```



	year	marital	happy	hapmar	health	satjob	satfin
46510	2006	5	2	NaN	3	1	2
46511	2006	5	1	NaN	NaN	1	2
46512	2006	3	2	NaN	NaN	NaN	1
46513	2006	5	1	NaN	1	2	2
46514	2006	5	2	NaN	2	NaN	1
46515	2006	1	2	2	NaN	1	3

Recodes

```
pd.options.mode.chained_assignment = None

# Reverse order variable for happy

GSS06and10["rhappy"] = 4 - GSS06and10.happy



# Pandas' Categorical function is similar to R's factor method

rhappy_temp = pd.Series(pd.Categorical(GSS06and10["rhappy"], categories = [1, 2, 3], ordered = True))



# However, it's not possible with Categorical function to specify labels at creation time. Use s.cat.rename_categories(new_labels) afterwards

GSS06and10["rhappy_fact"] = rhappy_temp.cat.rename_categories(["unhappy", "so-so", "happy"]).values  # pandas.Series has attribute 'values'
```

Another way...

```
# Another way to recode the same thing above without converting 'Categorical' objects
to pandas.Series

rhappy_temp = pd.Categorical(GSS06and10["rhappy"], categories = [1,2,3], ordered =
True)

GSS06and10["rhappy_fact"] = rhappy_temp.rename_categories(["unhappy", "so-so",
"happy"])  # 'Categorical' object has no attribute 'cat' nor 'values'
```

Final recodes

```
b = GSS06and10[["rhappy","year","marital","satfin","hapmar", "health", "satjob"]]
b = b[b.marital == 1]
c = b.dropna(subset = ['satfin','hapmar', 'health','satjob'], how = 'any') # if any
NA values are present in any column pre-specified, drop that label
year_dummy = {2006:0, 2010:1}  # To mimic R's as.factor(year) function that
converts 2006 to 0 and 2010 to 1
c["year_dum"] = c["year"].map(year_dummy.get)
```

Mediation: Our simple model - Results

```
lm1 = smf.ols(formula = "rhappy ~ year_dum", data = c).fit()
print (lm1.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                 rhappy   R-squared:                       0.005
Model:                            OLS   Adj. R-squared:                  0.005
Method:                 Least Squares   F-statistic:                     6.436
Date:                Wed, 15 May 2019   Prob (F-statistic):             0.0113
Time:                        12:43:37   Log-Likelihood:                -1112.5
No. Observations:                1189   AIC:                             2229.
Df Residuals:                    1187   BIC:                             2239.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      2.3759      0.023    103.922      0.000         2.331     2.421
year_dum      -0.0932      0.037     -2.537      0.011        -0.165    -0.021
==============================================================================
Omnibus:                       83.940   Durbin-Watson:                   1.966
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               49.463
Skew:                          -0.358   Prob(JB):                     1.82e-11
Kurtosis: *                     2.303   Cond. No.                         2.44
==============================================================================
```

Mediation: Our simple model - Results

```
lm1 = smf.ols(formula = "rhappy ~ year_dum", data = c).fit()
print (lm1.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                 rhappy   R-squared:                       0.005
Model:                            OLS   Adj. R-squared:                  0.005
Method:                 Least Squares   F-statistic:                     6.436
Date:                Wed, 15 May 2019   Prob (F-statistic):             0.0113
Time:                        12:43:37   Log-Likelihood:                -1112.5
No. Observations:                1189   AIC:                             2229.
Df Residuals:                    1187   BIC:                             2239.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      2.3759      0.023    103.922      0.000         2.331     2.421
year_dum      -0.0932      0.037     -2.537      0.011        -0.165    -0.021
==============================================================================
```

If someone is answering the survey in 2010 , on average, they will express a happiness opinion 0.09* * points lower, compared to 2006

Alternative explanations

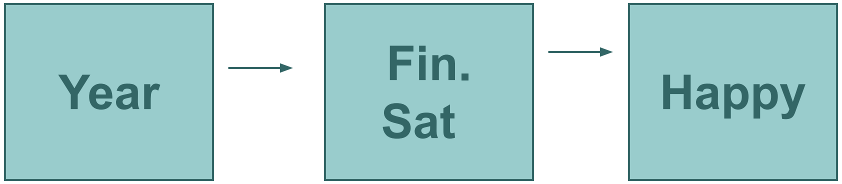
Ideas?

Alternative explanations

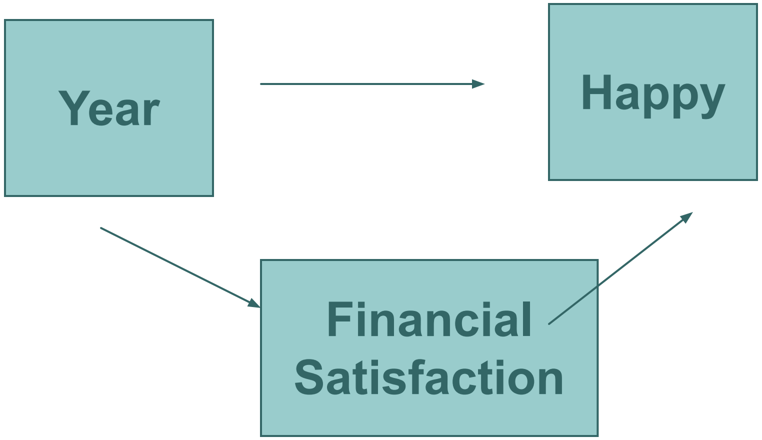
The march of time in itself may not be the reason why people express lower happiness in 2010 vs. 2006. Perhaps it is something that happened to people over that time that lowered their happiness, say, a change in their level of satisfaction with their financial situation

Which form of mediation is it?

Chain Mechanism



Both Direct and Indirect Effects



Mediation: Our complex model - Results

```
lm2 = smf.ols(formula = "rhappy ~ year_dum + satfin", data = c).fit()
print (lm2.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                 rhappy   R-squared:                       0.058
Model:                            OLS   Adj. R-squared:                  0.057
Method:                 Least Squares   F-statistic:                     36.68
Date:                Wed, 15 May 2019   Prob (F-statistic):           3.48e-16
Time:                        12:44:18   Log-Likelihood:                -1080.0
No. Observations:                1189   AIC:                             2166.
Df Residuals:                    1186   BIC:                             2181.
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      2.7596      0.052     53.037      0.000         2.658     2.862
year_dum      -0.0577      0.036     -1.602      0.110        -0.128     0.013
satfin        -0.2030      0.025     -8.159      0.000        -0.252    -0.154
==============================================================================
Omnibus:                       73.276   Durbin-Watson:                   1.998
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               40.984
Skew:                          -0.300   Prob(JB):                     1.26e-09
Kurtosis:                       2.317   Cond. No.                         7.61
==============================================================================
```

Mediation: Our complex model - Results

```
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      2.7596      0.052     53.037      0.000         2.658     2.862
year_dum      -0.0577      0.036     -1.602      0.110        -0.128     0.013
satfin        -0.2030      0.025     -8.159      0.000        -0.252    -0.154
==============================================================================
```

With people's financial satisfaction help constant, their happiness in 2010 will only be 0.057 points lower and not statistically significantly so, compared to 2006

Mediation: Said in the opposite way ...

```
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      2.7596      0.052     53.037      0.000         2.658     2.862
year_dum      -0.0577      0.036     -1.602      0.110        -0.128     0.013
satfin        -0.2030      0.025     -8.159      0.000        -0.252    -0.154
==============================================================================
```

With year held constant, if people increase their financial *dis*satisfaction score by 1 point, they will decrease their happiness by (on average) 0.20 points.

Remember …

What I said about reverse coding all the variables in the GSS?

There's why.

Mediation: Did I just cherry-pick? Look at marital happiness

```
lm3 = smf.ols(formula = "rhappy ~ year_dum + hapmar", data = c).fit()
print (lm3.summary())

                            OLS Regression Results                            
==============================================================================
Dep. Variable:                 rhappy   R-squared:                       0.207
Model:                            OLS   Adj. R-squared:                  0.206
Method:                 Least Squares   F-statistic:                     155.0
Date:                Wed, 15 May 2019   Prob (F-statistic):           1.57e-60
Time:                        12:45:22   Log-Likelihood:                -977.65
No. Observations:                1189   AIC:                             1961.
Df Residuals:                    1186   BIC:                             1977.
Df Model:                           2                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      3.1028      0.047     66.651      0.000         3.011     3.194
year_dum      -0.1049      0.033     -3.195      0.001        -0.169    -0.040
hapmar        -0.5125      0.029    -17.377      0.000        -0.570    -0.455
==============================================================================
```

Maybe people's mood just soured on everything between 2006 and 2010, not just on financial things.

Mediation: Did I just cherry-pick? Look at job satisfaction

```
lm4 = smf.ols(formula = "rhappy ~ year_dum + satjob", data = c).fit()
print (lm4.summary())
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                 rhappy   R-squared:                       0.051
Model:                            OLS   Adj. R-squared:                  0.049
Method:                 Least Squares   F-statistic:                     31.70
Date:                Wed, 15 May 2019   Prob (F-statistic):           3.88e-14
Time:                        12:45:26   Log-Likelihood:                -1084.7
No. Observations:                1189   AIC:                             2175.
Df Residuals:                    1186   BIC:                             2191.
Df Model:                           2                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      2.6519      0.043     61.757      0.000         2.568     2.736
year_dum      -0.0952      0.036     -2.651      0.008        -0.166    -0.025
satjob        -0.1720      0.023     -7.527      0.000        -0.217    -0.127
==============================================================================
```

Maybe people’s mood just soured on everything between 2006 and 2010, not just on financial things.

Mediation: Did I just cherry-pick? Look at health

```
lm5 = smf.ols(formula = "rhappy ~ year_dum + health", data = c).fit()
print (lm5.summary())

                            OLS Regression Results
==============================================================================
Dep. Variable:                 rhappy   R-squared:                       0.057
Model:                            OLS   Adj. R-squared:                  0.055
Method:                 Least Squares   F-statistic:                     35.56
Date:                Wed, 15 May 2019   Prob (F-statistic):           1.00e-15
Time:                        12:45:31   Log-Likelihood:                -1081.1
No. Observations:                1189   AIC:                             2168.
Df Residuals:                    1186   BIC:                             2183.
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      2.7230      0.049     55.940      0.000         2.628     2.819
year_dum      -0.0841      0.036     -2.348      0.019        -0.154    -0.014
health        -0.1882      0.023     -8.021      0.000        -0.234    -0.142
==============================================================================
```

Maybe people’s mood just soured on everything between 2006 and 2010, not just on financial things.

Mediation: Did I just cherry-pick? (in R)

```
install.packages(“stargazer”)
library(stargazer)
stargazer(lm1, lm2, lm3, lm4, lm5, type = "text")

stargazer(lm1, lm2, lm3, lm4, lm5,
          title="Regression Results",
          align=TRUE,
          dep.var.labels=c("Happy"),
          covariate.labels=c("Year","Fin. Sat", "Mar. Sat", "Job Sat", "Health"),
          no.space=TRUE,
          omit.stat=c("LL","ser","f", "rsq"),
          column.labels=c("Model 1", "Model 2", "Model 3", "Model 4", "Model 5"),
          dep.var.caption="",
          model.numbers=FALSE,
          type = "text")
```

Let me put all of this into a table; look here for more: http://dss.princeton.edu/training/NiceOutputR.pdf

Mediation: Did I just cherry-pick? (in R)

```
Regression Results
==============================================================
                                 Happy
             Model 1   Model 2   Model 3   Model 4   Model 5
--------------------------------------------------------------
Year         -0.093**  -0.058   -0.105*** -0.095*** -0.084**
             (0.037)   (0.036)   (0.033)   (0.036)   (0.036)
Fin. Sat              -0.203***
                       (0.025)
Mar. Sat                        -0.513***
                                 (0.029)
Job Sat                                   -0.172***
                                           (0.023)
Health                                              -0.188***
                                                     (0.023)
Constant     2.376*** 2.760***  3.103***  2.652***  2.723***
             (0.023)   (0.052)   (0.047)   (0.043)   (0.049)
--------------------------------------------------------------
Observations  1,189     1,189     1,189     1,189     1,189
Adjusted R2   0.005     0.057     0.206     0.049     0.055
==============================================================
Note:                            *p<0.1; **p<0.05; ***p<0.01
```

No other forms of satisfaction appear to mediate the relationship between time passing and happiness

Is this relationship mediated?

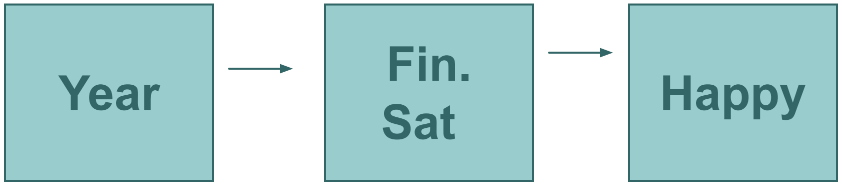
Simple Regression B1 = 0.093*

vs.

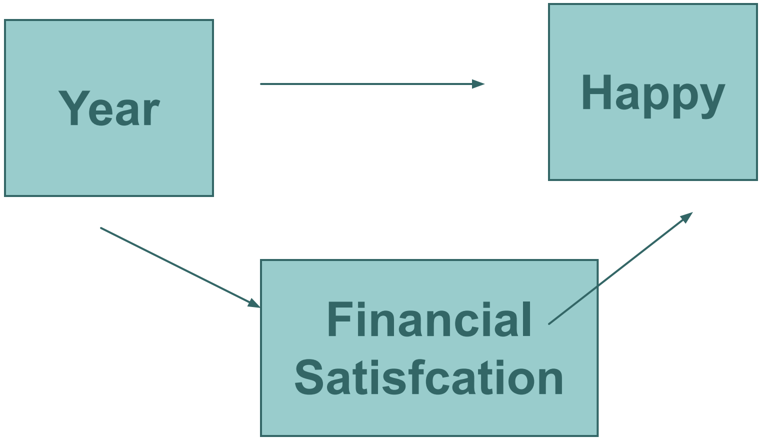
Multiple Regression B1 = 0.057 (n.s.)

Which form of mediation is it?

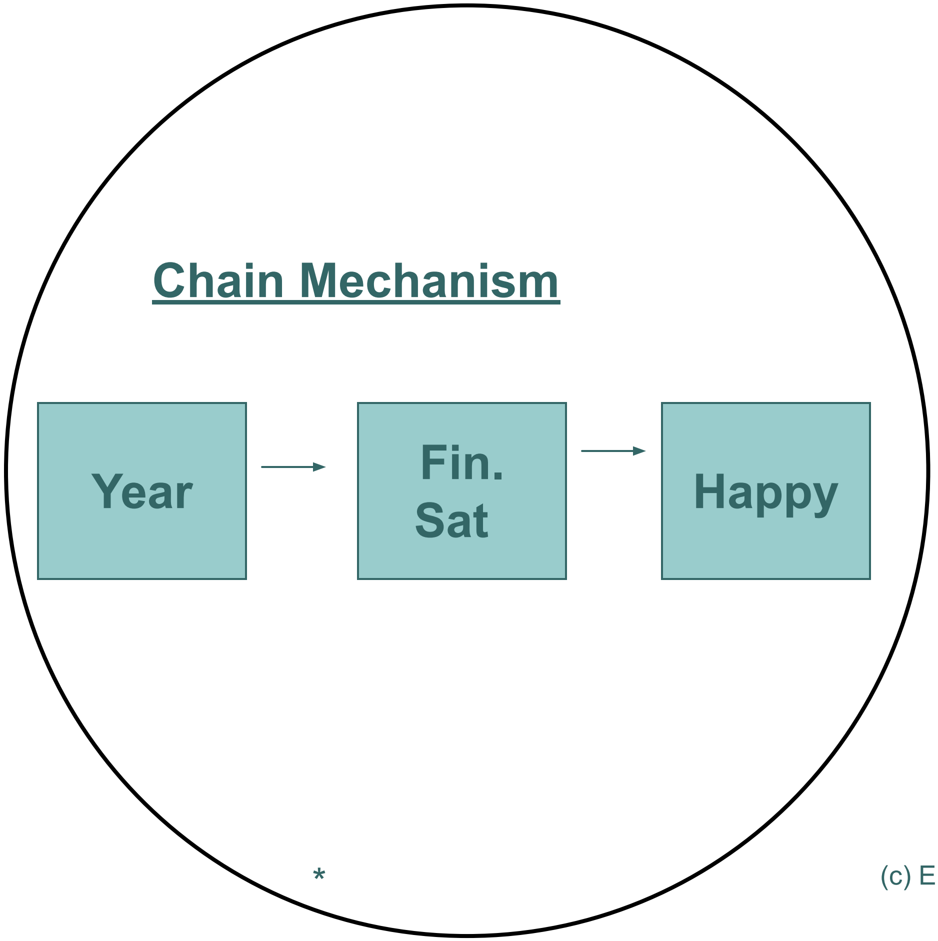
Chain Mechanism



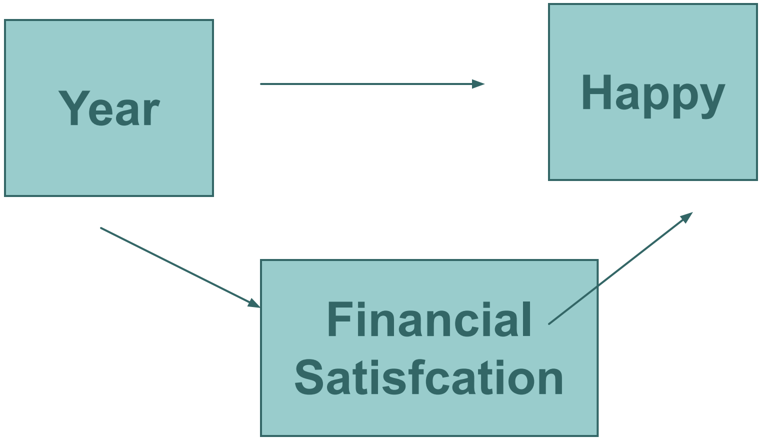
Both Direct and Indirect Effects



Which form of mediation is it?



Chain Mechanism



Both Direct and Indirect Effects

Is this relationship mediated?

Yes and no. From a statistical perspective, we entered a mediating variable that made the original relationship between happiness and 2010 insignificant (from p=0.01 to p=0.11), so that is important.

Is this relationship mediated?-con’t

On the other hand, we didn’t reduce the original B_{2010} very much, only by 38% (=(.093-.058)/.093), so that means practically, there may be other important mediating factors

Mediation: Our simple model - Results

```
lm1 = smf.ols(formula = "rhappy ~ year_dum", data = c).fit()
print (lm1.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                 rhappy   R-squared:                       0.005
Model:                            OLS   Adj. R-squared:                  0.005
Method:                 Least Squares   F-statistic:                     6.436
Date:                Wed, 15 May 2019   Prob (F-statistic):             0.0113
Time:                        12:43:37   Log-Likelihood:                -1112.5
No. Observations:                1189   AIC:                             2229.
Df Residuals:                    1187   BIC:                             2239.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      2.3759      0.023    103.922      0.000         2.331     2.421
year_dum      -0.0932      0.037     -2.537      0.011        -0.165    -0.021
==============================================================================
```

If someone is answering the survey in 2010 , on average, they will express a happiness opinion 0.09* * points lower, compared to 2006

Mediation: Said in the opposite way ...

```
=============================================================================
                    coef    std err          t      P>|t|     [95.0% Conf. Int.]
-----------------------------------------------------------------------------
Intercept         2.7596      0.052     53.037      0.000        2.658     2.862
year_dum         -0.0577      0.036     -1.602      0.110       -0.128     0.013
satfin           -0.2030      0.025     -8.159      0.000       -0.252    -0.154
=============================================================================
```

With year held constant, if people increase their financial *dis*satisfaction score by 1 point, they will decrease their happiness by (on average) 0.20 points.

Additional Mediation Test

Clifford C. Clogg, Eva Petkova, and Adamantios Haritou. “Statistical methods for comparing regression coefficients between models.” *The American Journal of Sociology*, Vol. 100, No. 5 (Mar., 1995), pp. 1261-1293

$$t = \frac{b_{year.model1} - b_{year.model2}}{\sqrt{(SE^2_{year.model2}) - [(SE^2_{year.model1})*(RMSE^2_{model2}/RMSE^2_{model1})]}}$$

(c) Eirich 2013

Additional Mediation Test

Is the slope on *year* in Model 2 (B=0.058, n.s.) statistically significantly smaller than *year* in Model 1 (B=0.093*)?

$$t= \mathbf{-8.15} = \frac{(-\ 0.093) - (-\ 0.058)}{\sqrt{(0.03604^2) - [(0.03676^2)*(0.6009^2/0.6173^2)]}}$$

Additional Mediation Test

Is the slope on *year* in Model 2 statistically significantly smaller than *year* in Model 1? Yes, since t=-8.15, that indicates that there is very little chance (p<.0001) that *year* in Model 2 just by chance is lower than *year* in Model 1. This provides evidence for a mediation effect, as proposed.

Let’s do another example …

Do people whose dads have higher occupational prestige, also have higher occupational prestige themselves?

Mediation: Our simple model

$$Y=a + B_1X_{1\ +}\ u$$

R’s Occ. Prestige=

a + B_1(Dad’s Occ. Pres. when R was 16) + u

Mediation: Our simple model

```
d = pd.read_csv("GSS_Cum.csv", usecols=["papres80", "year", "educ", "prestg80"])

lm_pres = smf.ols(formula = "prestg80 ~ papres80", data = d).fit()
print (lm_pres.summary())
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:               prestg80   R-squared:                       0.051
Model:                            OLS   Adj. R-squared:                  0.051
Method:                 Least Squares   F-statistic:                     1310.
Date:                Wed, 15 May 2019   Prob (F-statistic):          1.73e-279
Time:                        12:53:38   Log-Likelihood:                -97665.
No. Observations:               24286   AIC:                         1.953e+05
Df Residuals:                   24284   BIC:                         1.953e+05
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept     33.4268      0.311    107.506      0.000        32.817    34.036
papres80       0.2495      0.007     36.198      0.000         0.236     0.263
==============================================================================
Omnibus:                      802.518   Durbin-Watson:                   1.853
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              700.365
Skew:                           0.354   Prob(JB):                    8.27e-153
Kurtosis:                       2.563   Cond. No.                         162.
==============================================================================
```

Mediation: Our simple model - Results

```
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept     33.4268      0.311    107.506      0.000        32.817    34.036
papres80       0.2495      0.007     36.198      0.000         0.236     0.263
==============================================================================
```

For each one point increase in dad’s occupational prestige, on average, a child will have 0.249 more prestige points

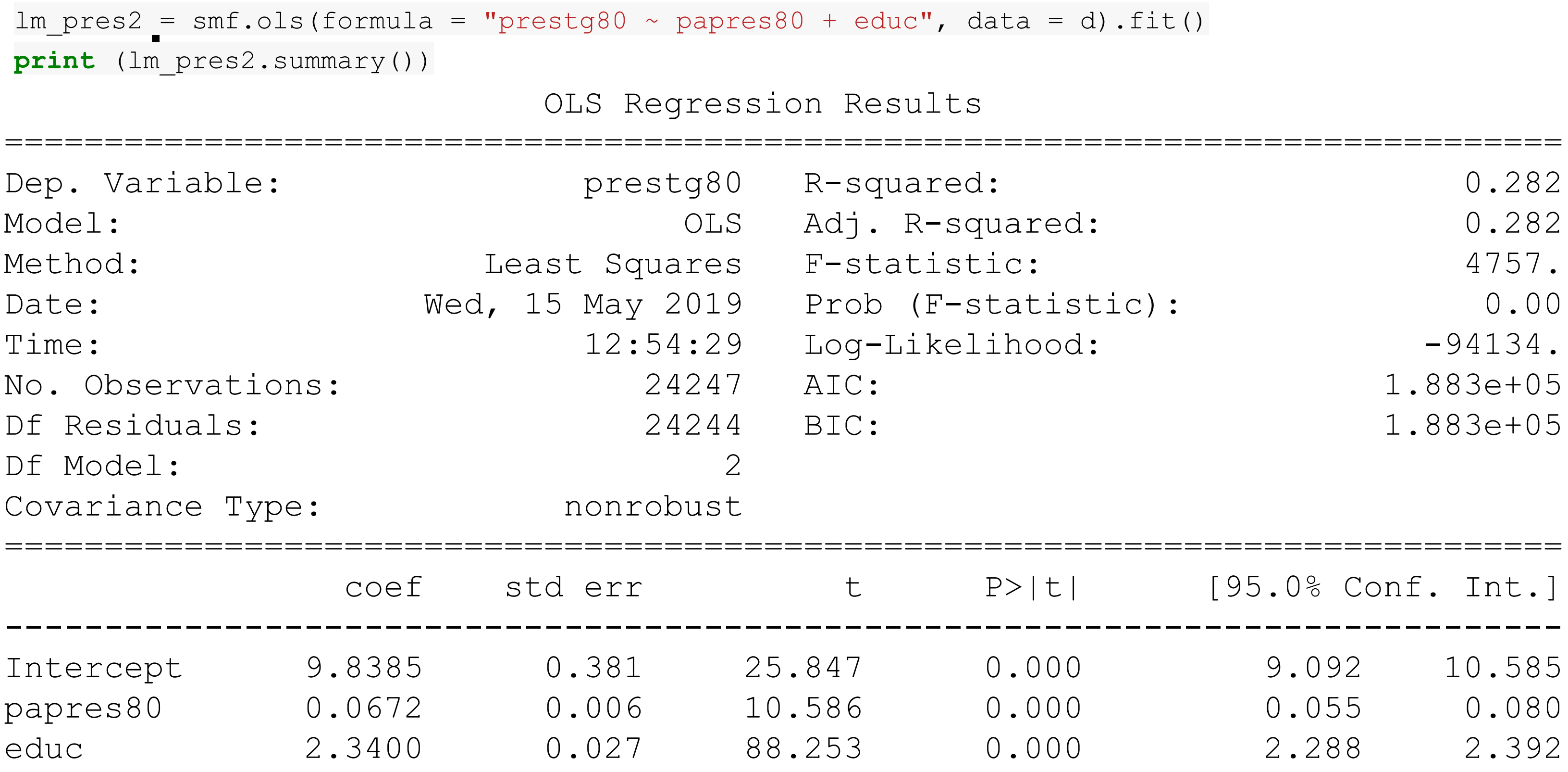
Alternative explanations

Ideas?

Alternative explanations

One thing that dad's with higher occupational prestige do for their kids is help them progress through school. So perhaps that is how occupational prestige levels are passed from one generation to the other.

Mediation: Our complex model - Results



```
lm_pres2 = smf.ols(formula = "prestg80 ~ papres80 + educ", data = d).fit()
print (lm_pres2.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:               prestg80   R-squared:                       0.282
Model:                            OLS   Adj. R-squared:                  0.282
Method:                 Least Squares   F-statistic:                     4757.
Date:                Wed, 15 May 2019   Prob (F-statistic):               0.00
Time:                        12:54:29   Log-Likelihood:                -94134.
No. Observations:               24247   AIC:                         1.883e+05
Df Residuals:                   24244   BIC:                         1.883e+05
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      9.8385      0.381     25.847      0.000         9.092    10.585
papres80       0.0672      0.006     10.586      0.000         0.055     0.080
educ           2.3400      0.027     88.253      0.000         2.288     2.392
```

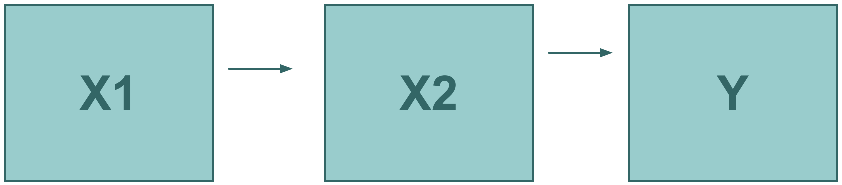
With dad’s occ. prest. held constant, for each year more of schooling, a person will have on average 2.33 more prestige points

Is this relationship mediated?

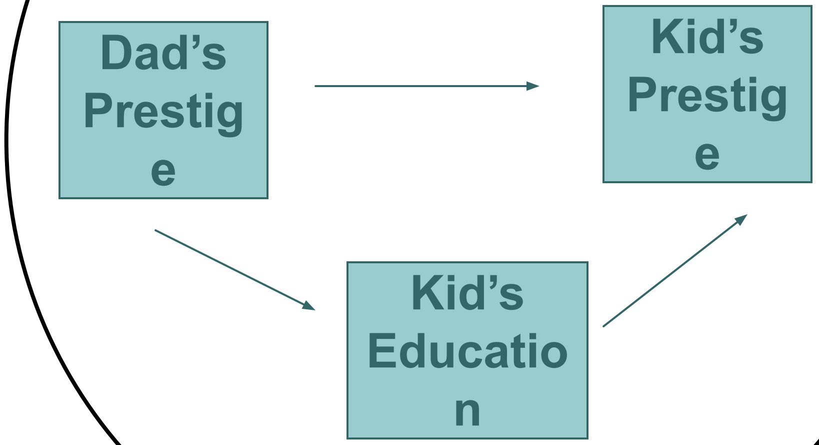
Simple Regression B1 = 0.25
vs.
Multiple Regression B1 = 0.07

Which form of mediation is it?

Chain Mechanism



Both Direct and Indirect Effects



Is this relationship mediated?-con’t

Yes. The vast majority (0.17/0.25=71%) of the way that dad’s occ prestige improves kid’s occ. prestige is through helping the kid get more education.

Is this relationship mediated?-con’t

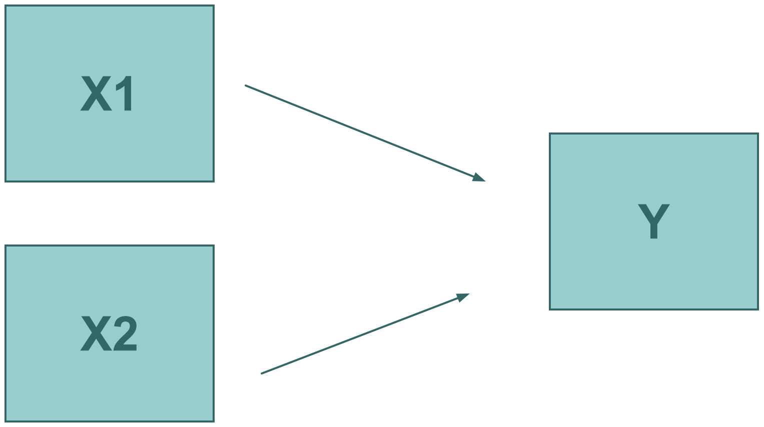
That said, dad’s occ prestige does still have a – smallish – independent effect on kid’s occ prestg, net of the mechanism of increasing kid’s educational attainment

Is this relationship mediated?-con’t

Note: We have *time order* on our side here. A child’s eventual occupational prestige cannot affect their previous education levels, much less their dad’s occupational prestige score when the person was 16.

To account for true relationships

- Multiple Causes: X2 is cause of Y but is unrelated to X1



You will see many of your own of this model!

5. Standardized Coefficients

Standardized Coefficients

Regress the z-score of the independent variable on the z-score of dependent variable

Called “Beta” coefficients

Interpretation

A one-standard deviation increase in the independent variable translates into a ____ standard deviation increase in the dependent variable

Why Standardized Coefficients?

They tell us about the magnitude of the effect of one variable on another. Is the effect large or not?

An example

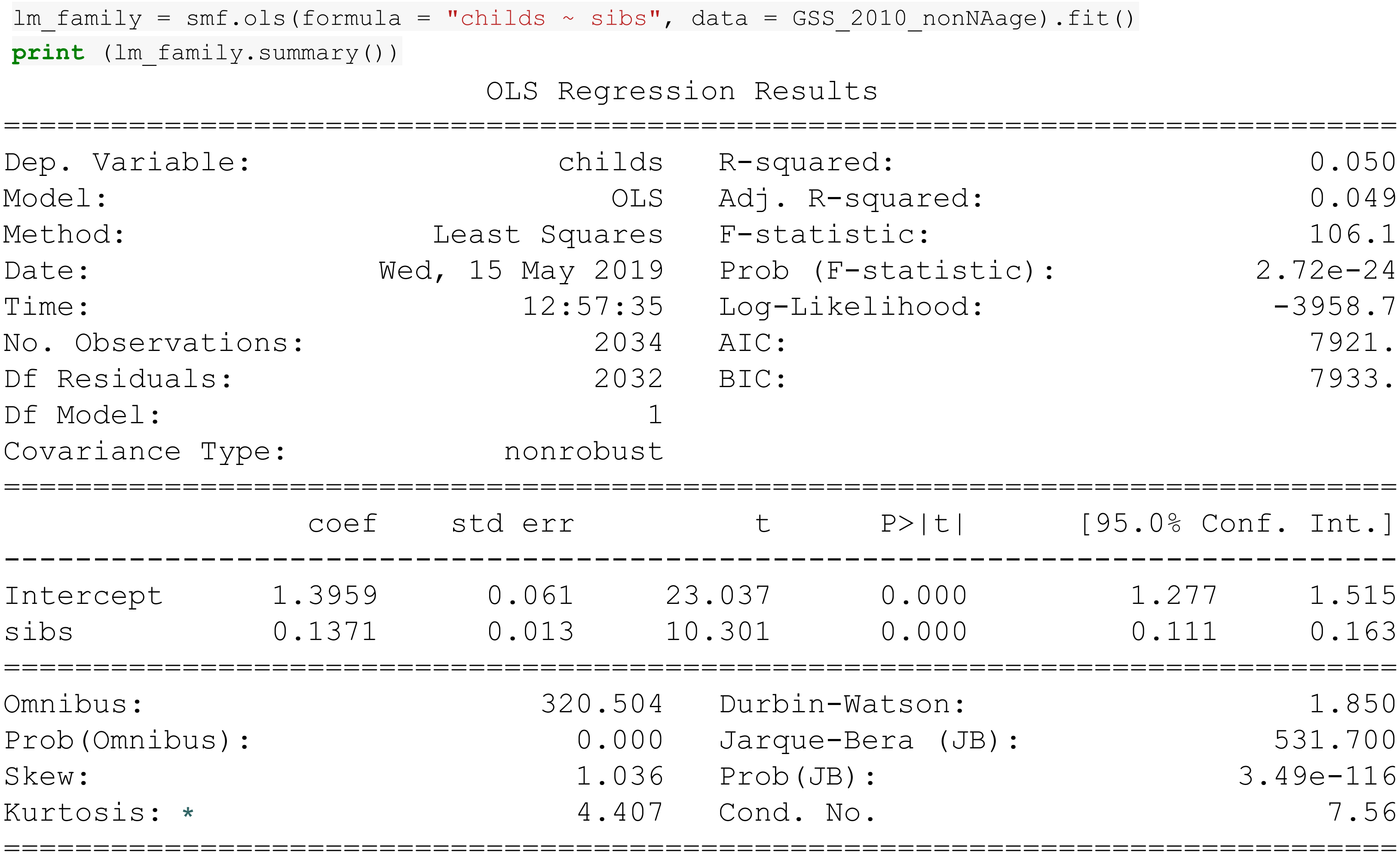
Do people who come from big families reproduce big families? Or the opposite?

Recodes...

```
d = pd.read_csv("GSS_Cum.csv", usecols=["sibs", "year", "childs", "age", "sex", "agekdbrn", "reg16"])

GSS_2010 = d[d.year == 2010]
GSS_2010_nonNAage = GSS_2010.dropna(subset = ["age"])
```

Results



```
lm_family = smf.ols(formula = "childs ~ sibs", data = GSS_2010_nonNAage).fit()
print (lm_family.summary())
```

```
                            OLS Regression Results                            
==============================================================================
Dep. Variable:                 childs   R-squared:                       0.050
Model:                            OLS   Adj. R-squared:                  0.049
Method:                 Least Squares   F-statistic:                     106.1
Date:                Wed, 15 May 2019   Prob (F-statistic):           2.72e-24
Time:                        12:57:35   Log-Likelihood:                -3958.7
No. Observations:                2034   AIC:                             7921.
Df Residuals:                    2032   BIC:                             7933.
Df Model:                           1                                         
Covariance Type:            nonrobust                                         
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      1.3959      0.061     23.037      0.000         1.277     1.515
sibs           0.1371      0.013     10.301      0.000         0.111     0.163
==============================================================================
Omnibus:                      320.504   Durbin-Watson:                   1.850
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              531.700
Skew:                           1.036   Prob(JB):                    3.49e-116
Kurtosis: *                     4.407   Cond. No.                         7.56
==============================================================================
```

Results

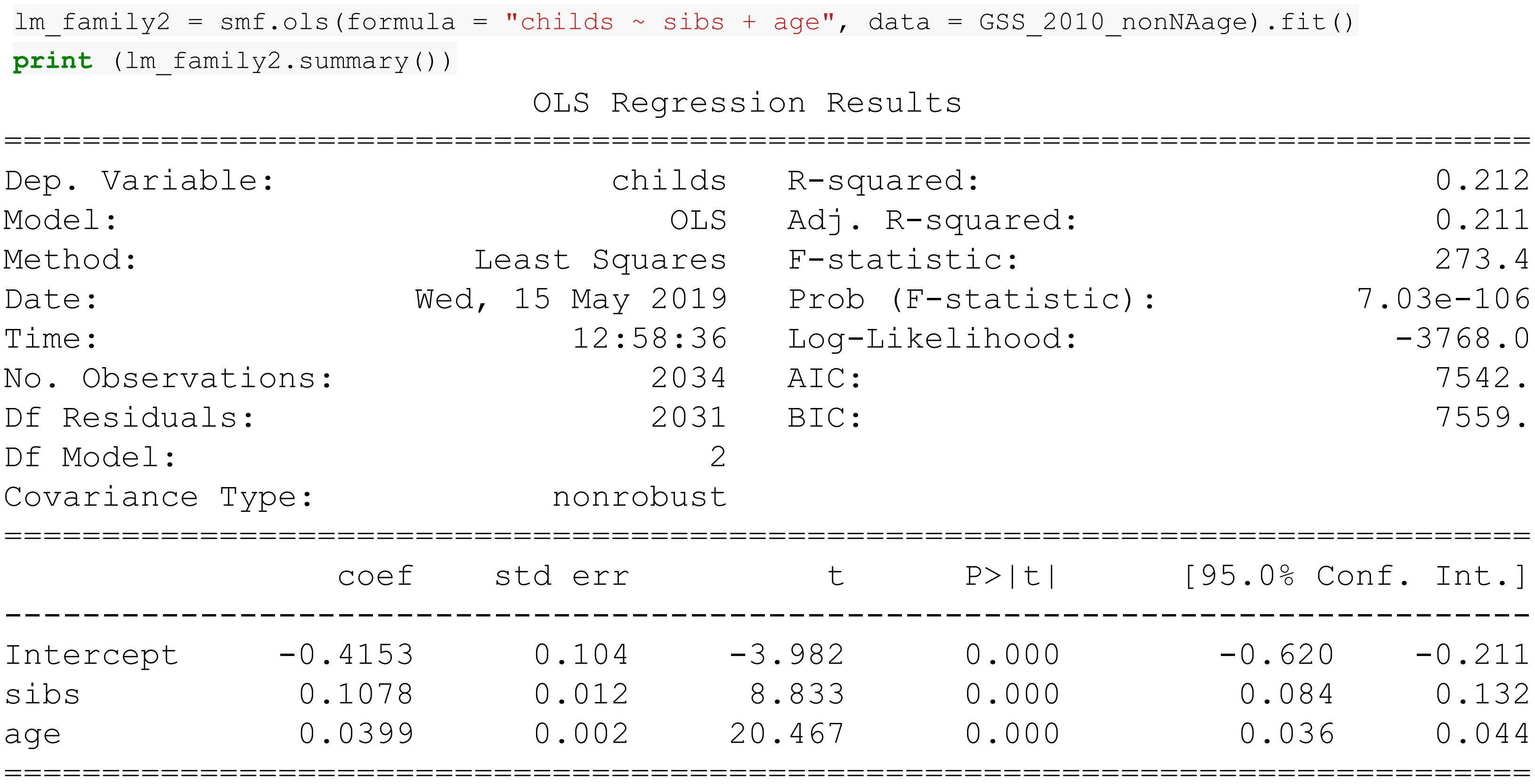
```
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept      1.3959      0.061     23.037      0.000         1.277     1.515
sibs           0.1371      0.013     10.301      0.000         0.111     0.163
==============================================================================
```

For each sibling more someone grew up with, they on average will have 0.137 more children (p<.0001)

An alternate explanation

Maybe we should only compare people of the same age, since it is unfair to compare people who have been around longer to those who have been around less.

Results



```
lm_family2 = smf.ols(formula = "childs ~ sibs + age", data = GSS_2010_nonNAage).fit()
print (lm_family2.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                 childs   R-squared:                       0.212
Model:                            OLS   Adj. R-squared:                  0.211
Method:                 Least Squares   F-statistic:                     273.4
Date:                Wed, 15 May 2019   Prob (F-statistic):          7.03e-106
Time:                        12:58:36   Log-Likelihood:                -3768.0
No. Observations:                2034   AIC:                             7542.
Df Residuals:                    2031   BIC:                             7559.
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept     -0.4153      0.104     -3.982      0.000        -0.620    -0.211
sibs           0.1078      0.012      8.833      0.000         0.084     0.132
age            0.0399      0.002     20.467      0.000         0.036     0.044
==============================================================================
```

Controlling for age, for each sibling more someone grew up with, they will on average have 0.107 more children (p<.000) *

Alternative explanations

Which has a bigger effect on the number of children a person has? Siblings or age?

A beta function ...

```
def stdCoef(fit):
    x = fit.model.data      # Access the original dataset
    sd = x.frame[x.xnames[1:]].std()   # Calculate the standard deviations of
"sibs" and "age"
    sd_dv = x.frame[x.ynames].std()  # Compute the standard deviation of the
dependent variable "childs"
    coefficients = fit.params[1:]
    std_coefs = coefficients * (sd / sd_dv)
    print ("Standardized coefficients are: ")
    return std_coefs

stdCoef(lm_family2)
```

Results

```
stdCoef(lm_family2)

Standardized coefficients are:

sibs    0.175227
age     0.406233
 dtype: float64
```

Thank you, RAs!

Results

```
stdCoef(lm_family2)
```

```
Standardized coefficients are:

sibs    0.175227
age     0.406233
 dtype: float64
```

Controlling for age, a 1 standard deviation increase in the number of siblings someone grew up with, will produce on average a 0.18 st. dev. increase in their number of children

Results

```
Standardized coefficients are:

sibs    0.175227
age     0.406233
 dtype: float64
```

Controlling for number of siblings, a 1 standard deviation increase in a person’s age, will produce on average a 0.41 st. dev. increase in their number of children

Alternative explanations

Which has a bigger effect on the number of children a person has? Siblings or age?

Age.

6. Dummy Variables

Dummies (or Indicator Variables) as Independent Variables

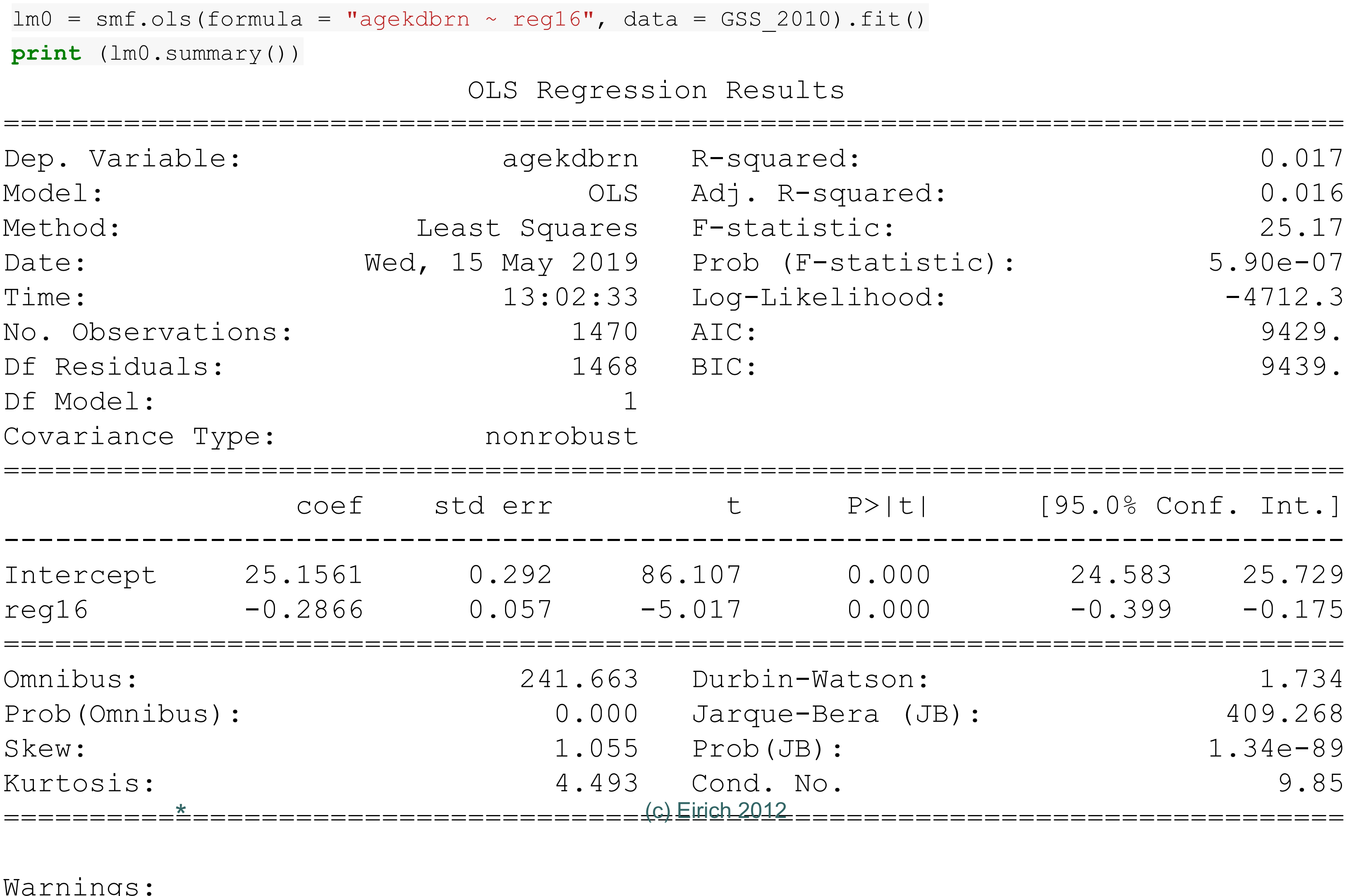
Always leave (at least) one of the dummies out of the equation to avoid perfect collinearity among them

This is called the reference or omitted variable

What about dummy variables?

There are many regions of the US where people grow up. Which one has the lowest average age where people had their first baby?

Don’t do this ...



```
lm0 = smf.ols(formula = "agekdbrn ~ reg16", data = GSS_2010).fit()
print (lm0.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:               agekdbrn   R-squared:                       0.017
Model:                            OLS   Adj. R-squared:                  0.016
Method:                 Least Squares   F-statistic:                     25.17
Date:                Wed, 15 May 2019   Prob (F-statistic):           5.90e-07
Time:                        13:02:33   Log-Likelihood:                -4712.3
No. Observations:                1470   AIC:                             9429.
Df Residuals:                    1468   BIC:                             9439.
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
Intercept     25.1561      0.292     86.107      0.000        24.583    25.729
reg16         -0.2866      0.057     -5.017      0.000        -0.399    -0.175
==============================================================================
Omnibus:                      241.663   Durbin-Watson:                   1.734
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              409.268
Skew:                           1.055   Prob(JB):                     1.34e-89
Kurtosis:                       4.493   Cond. No.                         9.85
==============================================================================

Warnings:
```

Dummy variables

```
lm = smf.ols(formula = "agekdbrn ~ C(reg16, Treatment)", data = GSS_2010).fit()
print (lm.summary())
```

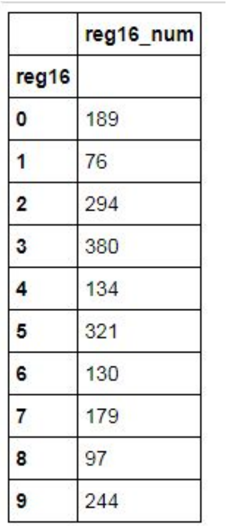
```
                            OLS Regression Results
==============================================================================
Dep. Variable:               agekdbrn   R-squared:                       0.042
Model:                            OLS   Adj. R-squared:                  0.036
Method:                 Least Squares   F-statistic:                     7.106
Date:                Wed, 15 May 2019   Prob (F-statistic):           3.95e-10
Time:                        13:00:28   Log-Likelihood:                -4693.3
============================================================================================
                                coef    std err          t      P>|t|      [95.0% Conf. Int.]
--------------------------------------------------------------------------------------------
Intercept                    25.2092      0.478     52.732      0.000        24.271    26.147
C(reg16, Treatment)[T.1]      0.7031      0.918      0.766      0.444        -1.097     2.503
C(reg16, Treatment)[T.2]      0.3729      0.634      0.588      0.557        -0.872     1.617
C(reg16, Treatment)[T.3]     -1.4853      0.599     -2.479      0.013        -2.661    -0.310
C(reg16, Treatment)[T.4]     -0.7355      0.772     -0.952      0.341        -2.251     0.780
C(reg16, Treatment)[T.5]     -2.1700      0.617     -3.518      0.000        -3.380    -0.960
C(reg16, Treatment)[T.6]     -2.7575      0.778     -3.547      0.000        -4.283    -1.232
C(reg16, Treatment)[T.7]     -3.0915      0.697     -4.436      0.000        -4.459    -1.724
C(reg16, Treatment)[T.8]     -3.2481      0.826     -3.931      0.000        -4.869    -1.627
C(reg16, Treatment)[T.9]     -0.7592      0.669     -1.135      0.256        -2.071     0.552
```

On average, a person who grew up in Region 7 would have had their 1st child 3.09 years earlier than something who grew* up in Region 0 (omitted category)

Adding labels

```
pd.options.mode.chained_assignment = None

GSS_2010["reg16_num"] = 1
pd.pivot_table(GSS_2010, index = ["reg16"], values = ["reg16_num"], aggfunc =
np.sum, fill_value = 0)
```

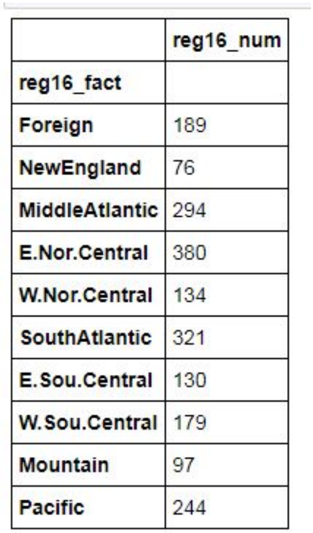


	reg16_num
reg16	
0	189
1	76
2	294
3	380
4	134
5	321
6	130
7	179
8	97
9	244

Adding labels

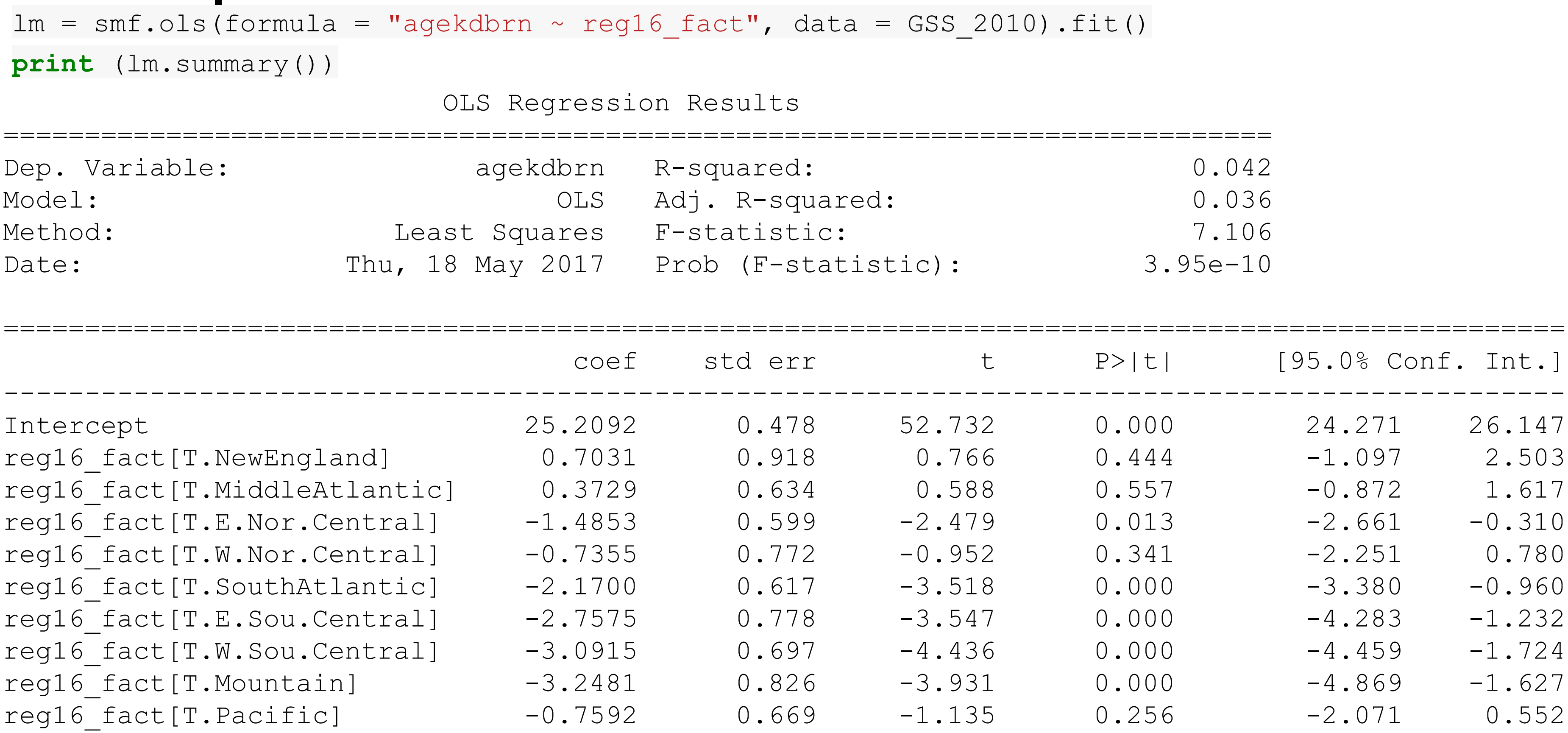
```
GSS_2010["reg16_category"] = pd.Categorical(GSS_2010["reg16"], categories = range(0, 10), ordered = True)

GSS_2010["reg16_fact"] = GSS_2010.reg16_category.cat.rename_categories(["Foreign", "NewEngland", "MiddleAtlantic", "E.Nor.Central", "W.Nor.Central", "SouthAtlantic", "E.Sou.Central", "W.Sou.Central", "Mountain", "Pacific"]).values
pd.pivot_table(GSS_2010, index = ["reg16_fact"], values = ["reg16_num"], aggfunc = np.sum, fill_value = 0)
```



	reg16_num
reg16_fact	
Foreign	189
NewEngland	76
MiddleAtlantic	294
E.Nor.Central	380
W.Nor.Central	134
SouthAtlantic	321
E.Sou.Central	130
W.Sou.Central	179
Mountain	97
Pacific	244

Same results as before, just with labels



```
lm = smf.ols(formula = "agekdbrn ~ reg16_fact", data = GSS_2010).fit()
print (lm.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:               agekdbrn   R-squared:                       0.042
Model:                            OLS   Adj. R-squared:                  0.036
Method:                 Least Squares   F-statistic:                     7.106
Date:                Thu, 18 May 2017   Prob (F-statistic):           3.95e-10

====================================================================================================
                                     coef    std err          t      P>|t|      [95.0% Conf. Int.]
----------------------------------------------------------------------------------------------------
Intercept                         25.2092      0.478     52.732      0.000        24.271    26.147
reg16_fact[T.NewEngland]           0.7031      0.918      0.766      0.444        -1.097     2.503
reg16_fact[T.MiddleAtlantic]       0.3729      0.634      0.588      0.557        -0.872     1.617
reg16_fact[T.E.Nor.Central]       -1.4853      0.599     -2.479      0.013        -2.661    -0.310
reg16_fact[T.W.Nor.Central]       -0.7355      0.772     -0.952      0.341        -2.251     0.780
reg16_fact[T.SouthAtlantic]       -2.1700      0.617     -3.518      0.000        -3.380    -0.960
reg16_fact[T.E.Sou.Central]       -2.7575      0.778     -3.547      0.000        -4.283    -1.232
reg16_fact[T.W.Sou.Central]       -3.0915      0.697     -4.436      0.000        -4.459    -1.724
reg16_fact[T.Mountain]            -3.2481      0.826     -3.931      0.000        -4.869    -1.627
reg16_fact[T.Pacific]             -0.7592      0.669     -1.135      0.256        -2.071     0.552
```

On average, a person who grew up in W. South Central US would have had their 1st child 3.09 years earlier than something who grew up outside of the US

You can change the reference

```
lm = smf.ols(formula ="agekdbrn ~ C(reg16_fact, Treatment(9))", data = GSS_2010).fit()  # we select #9 as reference, which is "Pacific" region
print (lm.summary())
```

```
                           OLS Regression Results
==============================================================================
Dep. Variable:               agekdbrn   R-squared:                       0.042
Model:                            OLS   Adj. R-squared:                  0.036
Method:                 Least Squares   F-statistic:                     7.106
Date:                Thu, 18 May 2017   Prob (F-statistic):           3.95e-10
=================================================================================================================
                                                    coef    std err          t      P>|t|      [95.0% Conf. Int.]
-----------------------------------------------------------------------------------------------------------------
Intercept                                        24.4500      0.467     52.301      0.000        23.533    25.367
C(reg16_fact, Treatment(9))[T.Foreign]            0.7592      0.669      1.135      0.256        -0.552     2.071
C(reg16_fact, Treatment(9))[T.NewEngland]         1.4623      0.912      1.603      0.109        -0.327     3.252
C(reg16_fact, Treatment(9))[T.MiddleAtlantic]     1.1321      0.627      1.807      0.071        -0.097     2.361
C(reg16_fact, Treatment(9))[T.E.Nor.Central]     -0.7261      0.591     -1.229      0.219        -1.885     0.433
C(reg16_fact, Treatment(9))[T.W.Nor.Central]      0.0237      0.766      0.031      0.975        -1.479     1.526
C(reg16_fact, Treatment(9))[T.SouthAtlantic]     -1.4109      0.609     -2.318      0.021        -2.605    -0.217
C(reg16_fact, Treatment(9))[T.E.Sou.Central]     -1.9984      0.771     -2.592      0.010        -3.511    -0.486
C(reg16_fact, Treatment(9))[T.W.Sou.Central]     -2.3324      0.690     -3.382      0.001        -3.685    -0.979
C(reg16_fact, Treatment(9))[T.Mountain]          -2.4890      0.820     -3.035      0.002        -4.098    -0.880
==============================================================================
```

On average, a person who grew up in W. South Central US would have had their 1st child 2.33 years earlier than someone who grew up in the Pacific part of the US