Data Analysis with Python

Gregory M. Eirich
QMSS

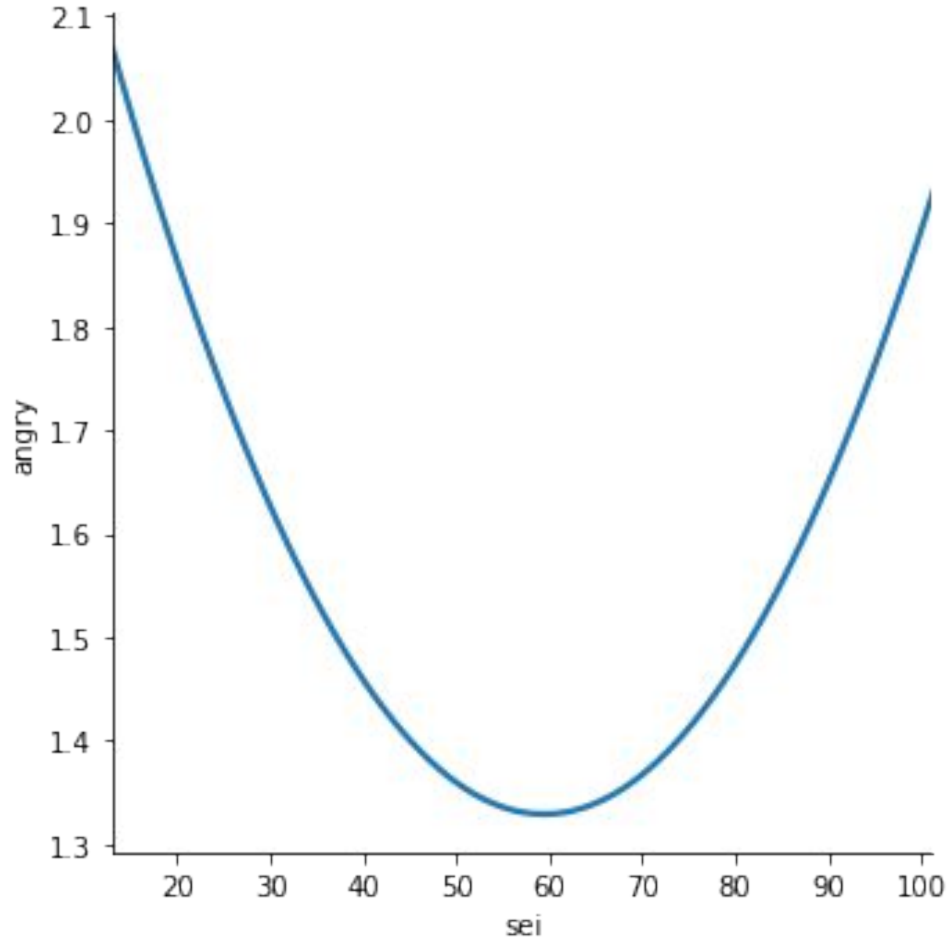
(Class #5)

Agenda

1. Quadratics
2. Adjusted R-sq
3. More on regression assumptions

1. Quadratic terms

Remember: Who gets angry the most?



How I made that plot

```
import seaborn as sns

sns.lmplot(x="sei", y="angry", data=d,
           order=2, ci=None, scatter=False);
```

FIGURE 6.1

Quadratic relationship between $\widehat{wage}$ and *exper*.

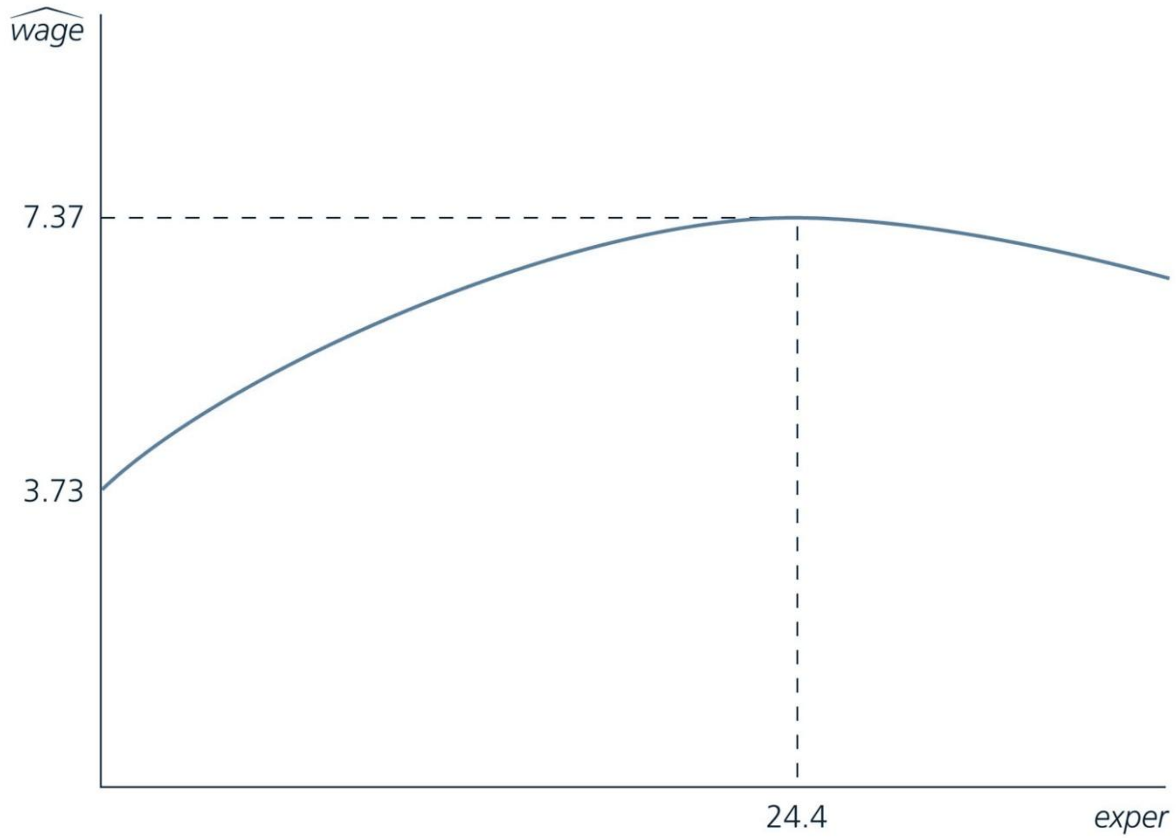
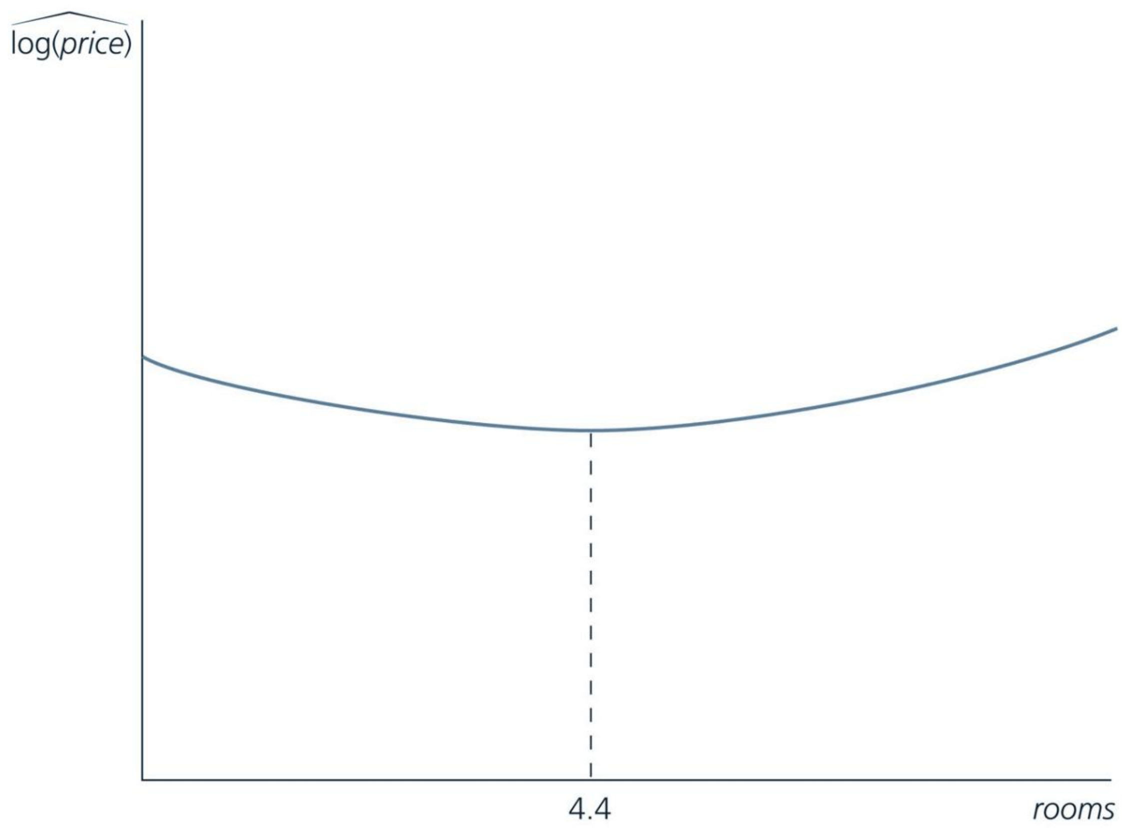


FIGURE 6.2

$\widehat{\log(price)}$ as a quadratic function of *rooms*.



Preliminary codes

```
from __future__ import division
import pandas as pd
import numpy as np
import statsmodels.api as sm
import statsmodels.formula.api as smf
import statsmodels.stats.api as sms
from statsmodels.compat import lzip
import os
import matplotlib.pyplot as plt
from statsmodels.stats.outliers_influence import reset_ramsey
```

Linear regression

```
os.chdir('C:/Users/gme2101/Desktop/Data Analysis Data') # change working directory
d = pd.read_csv("GSS_Cum.csv", usecols=["angry", "sei"])

lm_angry = smf.ols(formula = "angry ~ sei", data = d).fit()
print (lm_angry.summary())

                            OLS Regression Results
==============================================================================
Dep. Variable:                  angry   R-squared:                       0.002
Model:                            OLS   Adj. R-squared:                  0.001
Method:                 Least Squares   F-statistic:                     2.444
Date:                Mon, 03 Jun 2019   Prob (F-statistic):              0.118
Time:                        09:41:01   Log-Likelihood:                -2777.3
No. Observations:                1387   AIC:                             5559.
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      1.6890      0.130     12.960      0.000       1.433       1.945
sei           -0.0040      0.003     -1.563      0.118      -0.009       0.001
==============================================================================
Omnibus:                      370.029   Durbin-Watson:                   1.968
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              770.345
Skew:                           1.544   Prob(JB):                    5.27e-168
Kurtosis:                       4.947   Cond. No.                         139.
==============================================================================
```

For each SEI point, a person's number of angry days goes down by -0.0040 days, but it is not statistically significant

Curvilinear Regression (#1)

```
lm_angry2 = smf.ols(formula = "angry ~ sei + np.power(sei, 2)", data = d).fit()
print (lm_angry2.summary())

                            OLS Regression Results
==============================================================================
Dep. Variable:                  angry   R-squared:                       0.006
Model:                            OLS   Adj. R-squared:                  0.004
Method:                 Least Squares   F-statistic:                     4.073
Date:                Mon, 03 Jun 2019   Prob (F-statistic):             0.0172
Time:                        09:42:26   Log-Likelihood:                -2774.4
No. Observations:                1387   AIC:                             5555.
Df Residuals:                    1384   BIC:                             5571.
Df Model:                           2
Covariance Type:            nonrobust
====================================================================================
                       coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------------
Intercept            2.5430      0.381      6.678      0.000       1.796       3.290
sei                 -0.0409      0.016     -2.608      0.009      -0.072      -0.010
np.power(sei, 2)     0.0003      0.000      2.386      0.017    6.12e-05       0.001
==============================================================================
Omnibus:                      370.176   Durbin-Watson:                   1.969
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              773.345
Skew:                           1.541   Prob(JB):                    1.18e-168
Kurtosis:                       4.969   Cond. No.                     2.66e+04
```

Thanks to Omar Lizardo

http://www.nd.edu/~olizardo/pubs.html

Curvilinear Regression (#2)

```
lm_angry3 = smf.ols(formula = "angry ~ sei + I(sei**2)", data = d).fit()
print (lm_angry3.summary())

                            OLS Regression Results
==============================================================================
Dep. Variable:                  angry   R-squared:                       0.006
Model:                            OLS   Adj. R-squared:                  0.004
Method:                 Least Squares   F-statistic:                     4.073
Date:                Mon, 03 Jun 2019   Prob (F-statistic):             0.0172
Time:                        09:42:31   Log-Likelihood:                -2774.4
No. Observations:                1387   AIC:                             5555.
Df Residuals:                    1384   BIC:                             5571.
Df Model:                           2
Covariance Type:            nonrobust
===============================================================================
                  coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------
Intercept       2.5430      0.381      6.678      0.000       1.796       3.290
sei            -0.0409      0.016     -2.608      0.009      -0.072      -0.010
I(sei ** 2)     0.0003      0.000      2.386      0.017    6.12e-05       0.001
==============================================================================
Omnibus:                      370.176   Durbin-Watson:                   1.969
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              773.345
Skew:                           1.541   Prob(JB):                    1.18e-168
Kurtosis:                       4.969   Cond. No.                     2.66e+04
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.66e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Curvilinear Regression

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	2.5430	0.381	6.678	0.000	1.796	3.290
sei	-0.0409	0.016	-2.608	0.009	-0.072	-0.010
I(sei ** 2)	0.0003	0.000	2.386	0.017	6.12e-05	0.001

At first, a person's number of angry days goes **down**, but then at a certain point, for each SEI point squared, a person's no. of angry days goes **up**

Where does the line reverse direction?

- The point at which the slope is 0, the relationship changes direction from positive to negative (i.e, the maximum) or from negative to positive (i.e., the minimum)
- This happens at $x = -\beta_1/(2\beta_2)$
- So in this case: $x = -(-.04086)/(2*0.00034)$

$$= 59.434$$

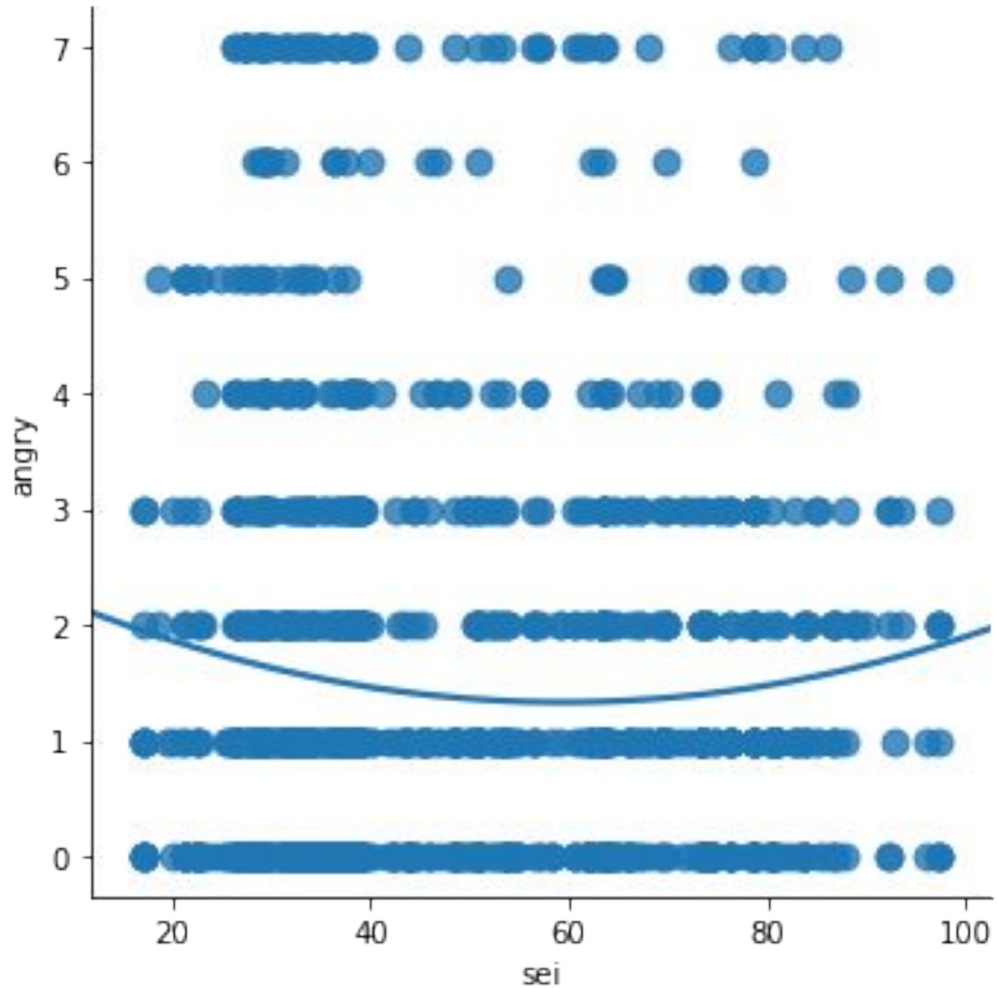
Interpreting Quadratics

- If B1 is positive, but B2 (the quadratic) is negative … the shape is upside-down U
- If B1 is negative, but B2 (the quadratic) is positive … the shape is a U

Interpreting Quadratics

- If B1 is positive, and B2 (the quadratic) is positive … the shape is increasing and even more steeply increasing
- If B1 is negative, and B2 (the quadratic) is negative… the shape is declining and then even more declining

Here is the raw data. Do you see it?



How I made that plot

```
sns.lmplot(x="sei", y="angry", data=d,
           order=2, ci=None, scatter_kws={"s": 80});
```

Statistical significance?

- We want both B1 and B2 to be statistically significant
- Otherwise, it would be easier to just work with a linear assumption

Why would I think there is a quadratic here in the first place?

- Theoretical reasons …

Why would I think there is a quadratic here in the first place?

- Or: A statistical test for an omitted variable, where that omitted variable is a higher power (square, cube, raised to the fourth power) of an X variable already in the model

Omitted variable test

Here, we are using the "reset ramsey" function from the statsmodels outliers_influence package. Source code for this function can be found here:

http://www.statsmodels.org/dev/_modules/statsmodels/stats/outliers_influence.html

```
reset_ramsey(lm_angry, degree=2)
```

```
<class 'statsmodels.stats.contrast.ContrastResults'>

<F test: F=array([[5.69450107]]), p=0.017151877448120086, df_denom=1384, df_num=1>
```

This is a test of the null hypothesis that no higher powers of the Xs would fit the data better

Omitted variable test

This RESET test works thusly:

1. Run the original regression
2. Predict Y as Yhat
3. Standardize Yhat
4. Take Yhat and square, cube and raise to fourth power
5. Rerun original regression but include $Yhat^2$, $Yhat^3$, $Yhat^4$
6. Run F-test that $Yhat^2 = Yhat^3 = Yhat^4 = 0$
7. If $p<.05$, we have evidence of some Xs as higher powers

Omitted variable test

```
<F test: F=array([[5.69450107]]), p=0.017151877448120086, df_denom=1384, df_num=1>
```

This is a test of the hypothesis that no higher powers of the Xs would fit the data better

We could reject the null hypothesis that no higher powers of SEI would fit the data better because $p<.05$

<u>Another example of a quadratic</u>

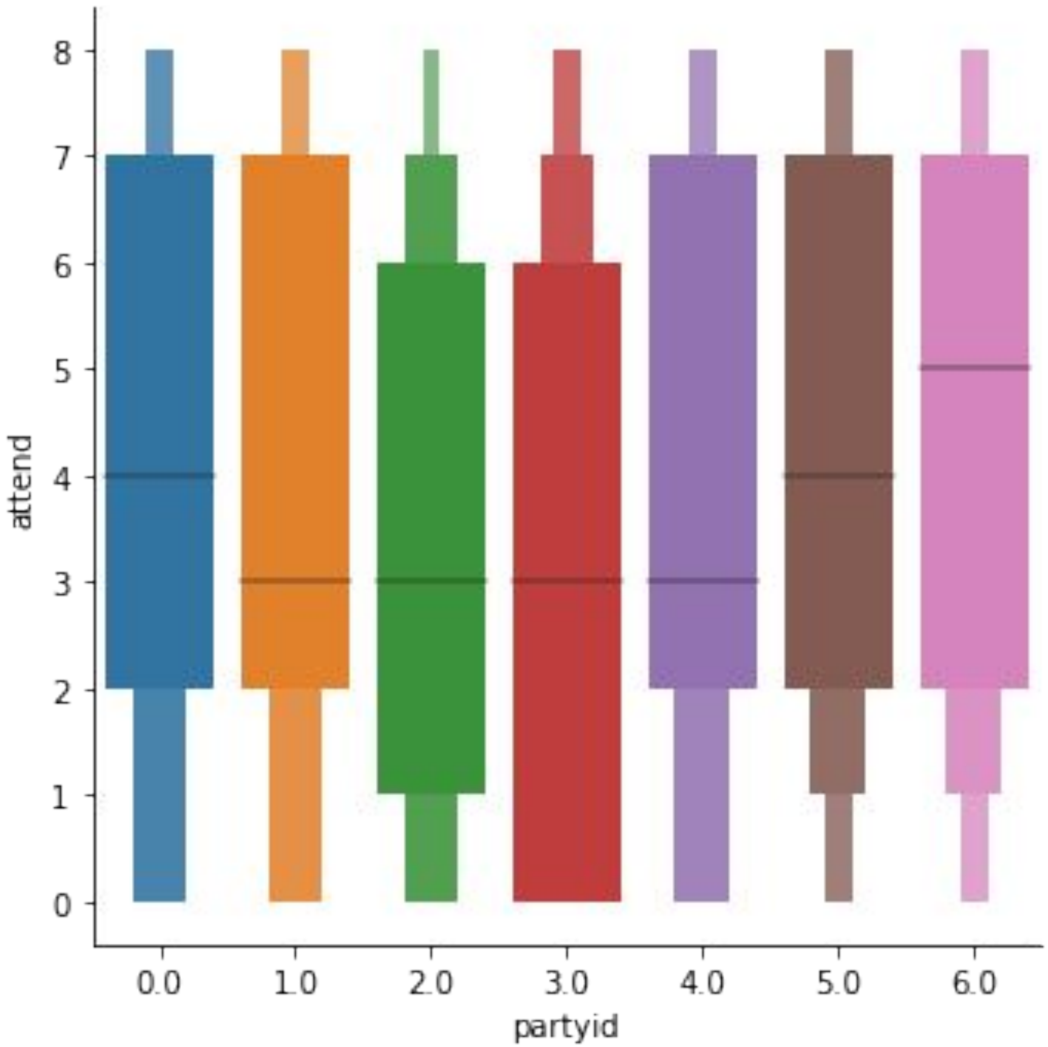
Do Republicans go to religious services more often than Democrats?

Linear regression

```
d = pd.read_csv("GSS_Cum.csv", usecols=["attend", "partyid"])

We also only want to look at cases when partyid < 7:
sub2 = d[d['partyid'] < 7.0]

lm_attend = smf.ols(formula = "attend ~ partyid", data = sub2).fit()

print (lm_attend.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                 attend   R-squared:                       0.003
Model:                            OLS   Adj. R-squared:                  0.003
Method:                 Least Squares   F-statistic:                     145.9
Date:                Mon, 03 Jun 2019   Prob (F-statistic):           1.50e-33
Time:                        09:42:53   Log-Likelihood:            -1.3373e+05
No. Observations:               55401   AIC:                         2.675e+05
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      3.6489      0.019    190.033      0.000       3.611       3.687
partyid        0.0697      0.006     12.079      0.000       0.058       0.081
==============================================================================
Omnibus:                   652561.840   Durbin-Watson:                   1.822
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             4374.347
Skew:                           0.037   Prob(JB):                         0.00
Kurtosis:                       1.625   Cond. No.                         5.89
==============================================================================
```

For each category more strongly someone identifies with the Republican party, they increase their religious attendance by 0.069 categories

Here is the raw data. Do you see it?



How did I do that graph? (in R)

```
sns.catplot(x="partyid", y="attend", kind="boxen",
            data=sub2.sort_values("partyid"));
```

Omitted variable test

```
Omitted variable test

reset_ramsey(lm_attend, degree=2)

<class 'statsmodels.stats.contrast.ContrastResults'>

<F test: F=array([[1259.19968467]]), p=9.68161186333948e-273, df_denom=55398, df_num=1>
```

We cannot reject the null hypothesis that no higher powers of *partyid* would fit the data better because p<.05

So let's consider a quadratic …

Curvilinear Regression

```
lm_attend2 = smf.ols(formula = "attend ~ partyid + I(partyid**2)", data = sub2).fit()
print (lm_attend2.summary())

                            OLS Regression Results
==============================================================================
Dep. Variable:                 attend   R-squared:                       0.025
Model:                            OLS   Adj. R-squared:                  0.025
Method:                 Least Squares   F-statistic:                     704.2
Date:                Mon, 03 Jun 2019   Prob (F-statistic):          9.73e-303
Time:                        09:43:04   Log-Likelihood:            -1.3311e+05
No. Observations:               55401   AIC:                         2.662e+05
===================================================================================
                      coef    std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------------
Intercept           4.1868      0.024    172.324      0.000       4.139       4.234
partyid            -0.6445      0.021    -30.810      0.000      -0.685      -0.603
I(partyid ** 2)     0.1233      0.003     35.485      0.000       0.117       0.130
==============================================================================
Omnibus:                   264433.162   Durbin-Watson:                   1.831
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             3961.041
Skew:                           0.044   Prob(JB):                         0.00
Kurtosis:                       1.693   Cond. No.                         44.5
==============================================================================
```

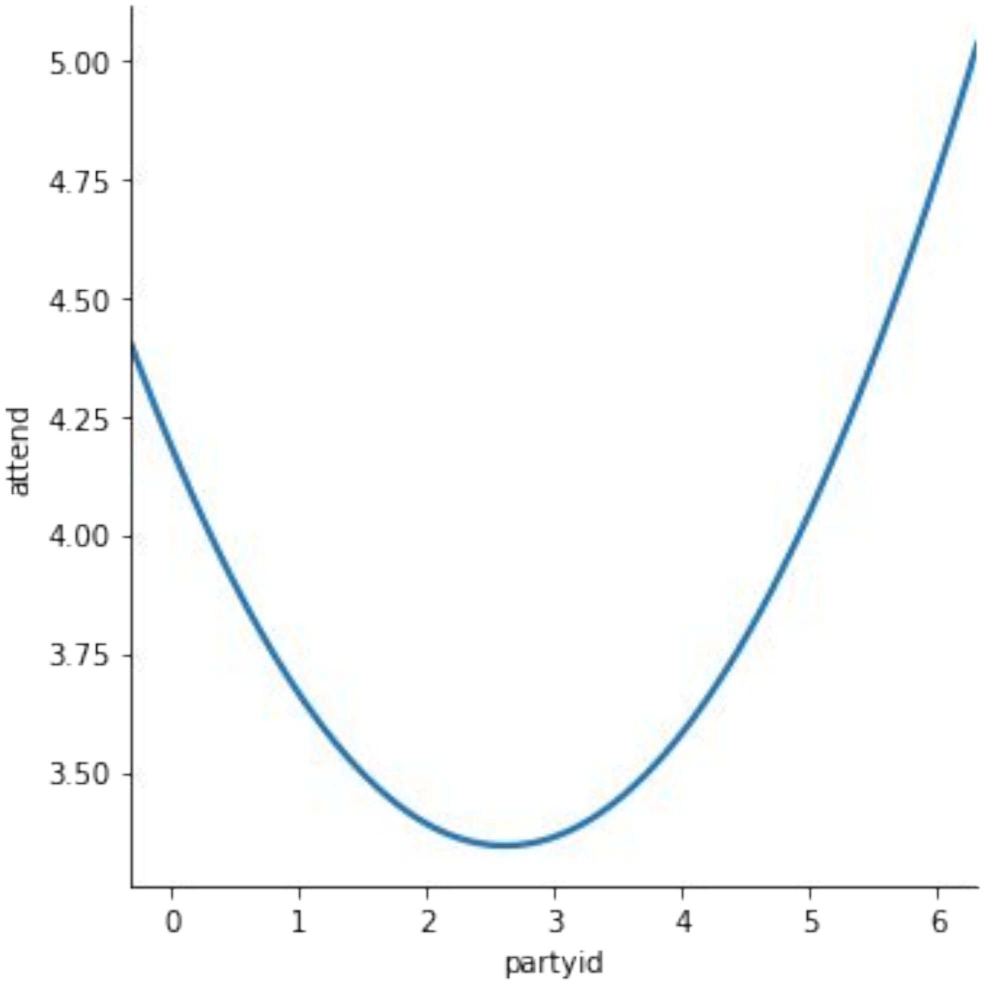
- For each category more strongly someone identifies with the Republican party (ECMSSIWTHRP), they decrease their attendance by -0.644 categories, but at the same time, for ECMSSIWTHRP^2, they increase their attendance by 0.123 categories

Curvilinear Regression

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	4.1868	0.024	172.324	0.000	4.139	4.234
partyid	-0.6445	0.021	-30.810	0.000	-0.685	-0.603
I(partyid ** 2)	0.1233	0.003	35.485	0.000	0.117	0.130

The Adj. R-sq (0.0248) is 10x greater with the quadratic term included vs. the original linear specification only (0.0024).

Here is what that looks like …



Here is what that looks like (in R) …

```
sns.lmplot(x="partyid", y="attend", data=sub2,
           order=2, ci=None, scatter=False);
```

2. Adjusted R-sq

Adjusted R-sq

Adjusted R-sq discounts the original R-sq in light of increased variables being added.

```
lm_maBA_twobio = smf.ols( "educ ~ sibs * maBA + age + twobio" , data = sub_kids).fit()
print (lm_maBA_twobio.summary())  --   OLS Regression Results
==============================================================================
Dep. Variable:                   educ   R-squared:                       0.156
Model:                            OLS   Adj. R-squared:                  0.154
Method:                 Least Squares   F-statistic:                     109.7
Date:                Wed, 22 May 2019   Prob (F-statistic):          1.47e-106
No. Observations:                2977   AIC:                         1.492e+04
Df Residuals:                    2971   BIC:                         1.496e+04
Df Model:                           5
=====================================================================================
                        coef    std err          t      P>|t|      [95.0% Conf. Int.]
-------------------------------------------------------------------------------------
Intercept            13.9321      0.185     75.138      0.000        13.569    14.296
maBA[T.True]          1.4527      0.257      5.657      0.000         0.949     1.956
twobio[T.True]        0.5932      0.121      4.892      0.000         0.355     0.831
sibs                 -0.2902      0.018    -16.206      0.000        -0.325    -0.255
sibs:maBA[T.True]     0.2251      0.075      3.006      0.003         0.078     0.372
age                  -0.0028      0.003     -0.840      0.401        -0.009     0.004
==============================================================================
```

$$R^2_{\text{adj}} = \frac{s_y^2 - s^2}{s_y^2} = 1 - \frac{s^2}{s_y^2},$$

where $s^2 = \sum(y - \hat{y})^2/[n - (k + 1)]$ is the estimated conditional variance (i.e., the mean square error, MSE) and $s_y^2 = \sum(y - \bar{y})^2/(n - 1)$ is the sample variance of y.

Adjusted R-sq

```
lm_maBA_twobio = smf.ols( "educ ~ sibs * maBA + age + twobio" , data = sub_kids).fit()
print (lm_maBA_twobio.summary())  --   OLS Regression Results
==============================================================================
Dep. Variable:                   educ   R-squared:                       0.156
Model:                            OLS   Adj. R-squared:                  0.154
Method:                 Least Squares   F-statistic:                     109.7
Date:                Wed, 22 May 2019   Prob (F-statistic):          1.47e-106
No. Observations:                2977   AIC:                         1.492e+04
Df Residuals:                    2971   BIC:                         1.496e+04
Df Model:                           5
=====================================================================================
                        coef    std err          t      P>|t|      [95.0% Conf. Int.]
-------------------------------------------------------------------------------------
Intercept            13.9321      0.185     75.138      0.000        13.569    14.296
maBA[T.True]          1.4527      0.257      5.657      0.000         0.949     1.956
twobio[T.True]        0.5932      0.121      4.892      0.000         0.355     0.831
sibs                 -0.2902      0.018    -16.206      0.000        -0.325    -0.255
sibs:maBA[T.True]     0.2251      0.075      3.006      0.003         0.078     0.372
age                  -0.0028      0.003     -0.840      0.401        -0.009     0.004
```

The amount of variance that can be explained by this the variables is 15.4%, given the number of variables included

3. OLS assumptions and diagnostics

Assumptions of OLS to get Unbiasedness

1. Linearity in parameters

2. Random sampling

3. Sample variation in explanatory variable (no perfect collinearity)

4. Zero conditional mean

Assumption #1

Linearity of parameters

- We cannot estimate functions of parameters that are not linear
- That said, we can estimate all sorts of non-linear relationships *in the variables* by transformation, like logs or quadratics

Assumption #3

No perfect collinearity

- Sometimes things aren't perfectly collinear but they display multicollinearity … we will deal with this one when we get to scales

Assumption #3 - Continued

No perfect collinearity

- R just bumps out the collinear terms

```
Like in R, Python automatically drops perfectly linear terms:

d = pd.read_csv("GSS_Cum.csv", usecols=["tvhours", "age", "degree"])

lm_tv = smf.ols(formula = "tvhours ~ age + age", data = d).fit()
print (lm_tv.summary())

                            OLS Regression Results
==============================================================================
Dep. Variable:                tvhours   R-squared:                       0.009
Model:                            OLS   Adj. R-squared:                  0.009
Method:                 Least Squares   F-statistic:                     301.1
Date:                Mon, 03 Jun 2019   Prob (F-statistic):           3.71e-67
Time:                        09:44:13   Log-Likelihood:                -76602.
No. Observations:               33735   AIC:                         1.532e+05
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      2.3948      0.036     67.238      0.000       2.325       2.465
age            0.0126      0.001     17.352      0.000       0.011       0.014
==============================================================================
Omnibus:                    19665.763   Durbin-Watson:                   1.875
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           251492.570
Skew:                           2.583   Prob(JB):                         0.00
Kurtosis:                      15.339   Cond. No.                         137.
==============================================================================
```

Assumption #5

Homoskedasticity

- There is a constant variance of u over all the values of the Xs
- At each value of X, u has the same variance
- We care about this because heteroskedasticity leads to inappropriate standard errors and p-values (i.e., inefficiency)

Assumptions of OLS to get Unbiasedness

5. Homoskedasticity

$Var(u|x)=\sigma^2$

Distribution of u is same for any value of x

FIGURE 2.8

The simple regression model under homoskedasticity.

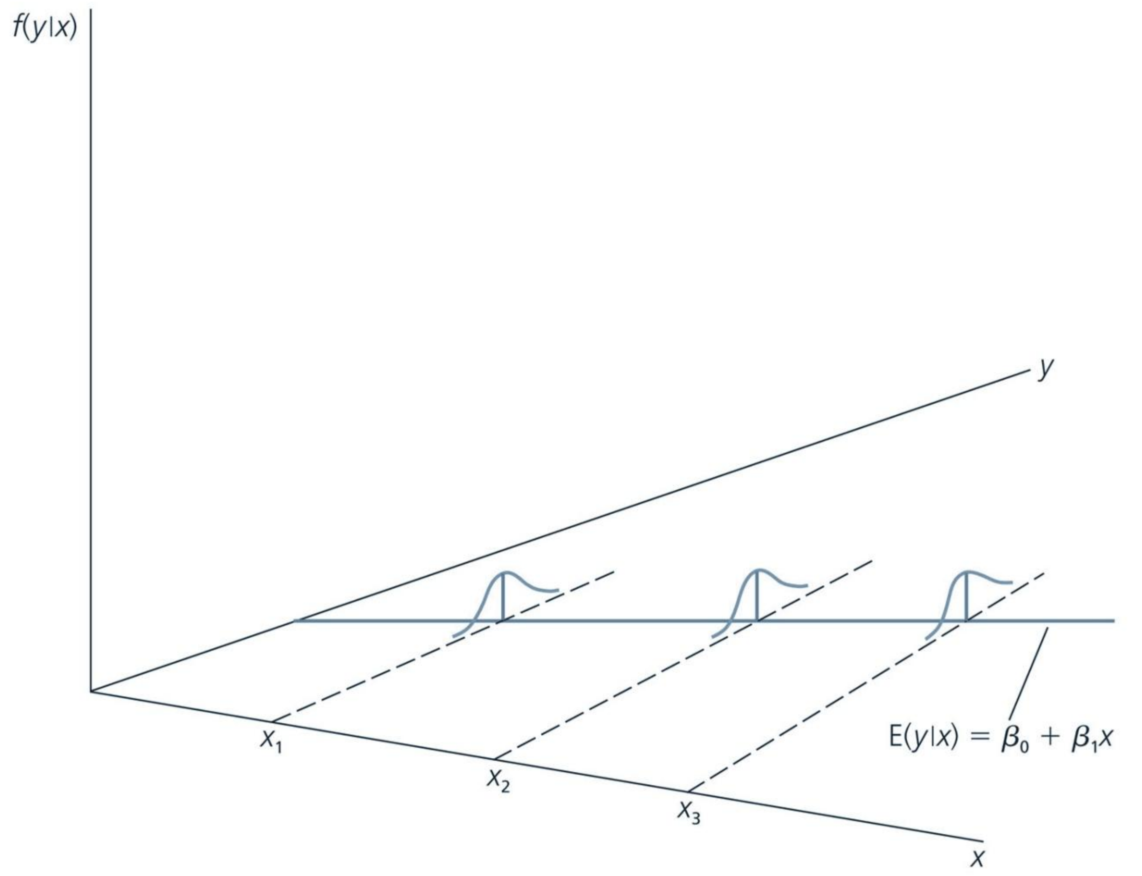
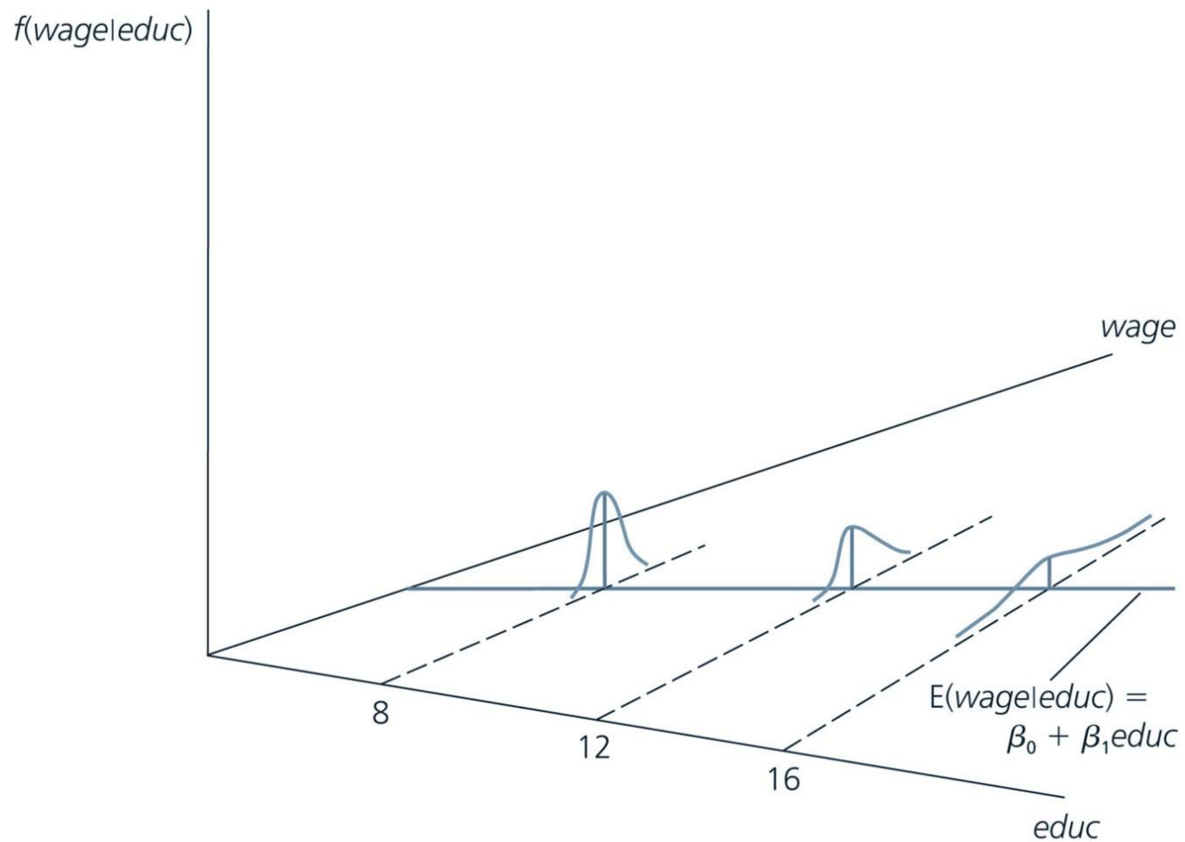


FIGURE 2.9

Var(*wage*|*educ*) increasing with *educ*.



Let's look at this regression ...

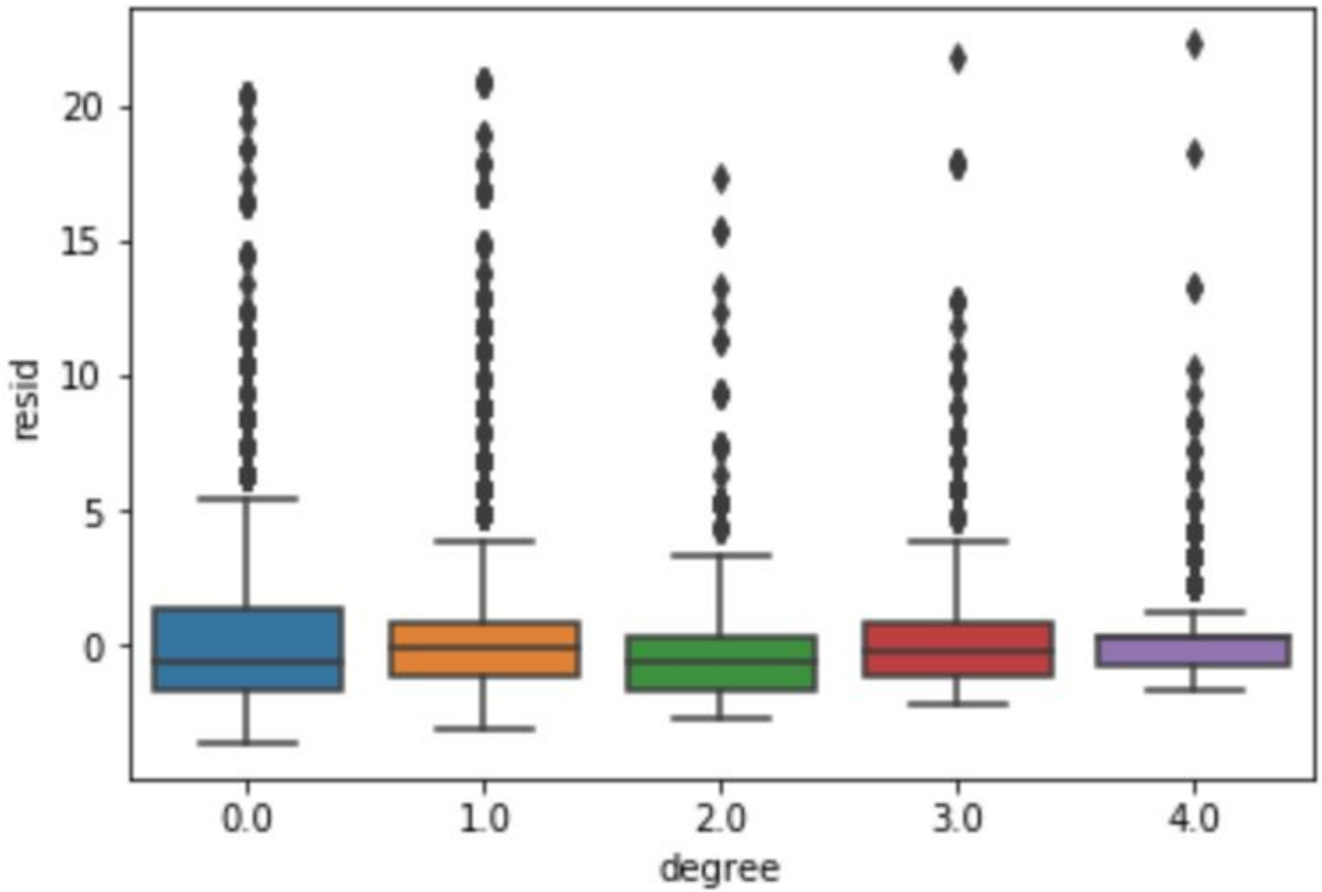
```
lm_tv = smf.ols(formula = "tvhours ~ degree", data = d).fit()
print (lm_tv.summary())

                            OLS Regression Results
==============================================================================
Dep. Variable:                tvhours   R-squared:                       0.054
Model:                            OLS   Adj. R-squared:                  0.054
Method:                 Least Squares   F-statistic:                     1934.
Date:                Mon, 03 Jun 2019   Prob (F-statistic):               0.00
Time:                        09:44:19   Log-Likelihood:                -75921.
No. Observations:               33788   AIC:                         1.518e+05
Df Residuals:                   33786   BIC:                         1.519e+05
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      3.5967      0.019    190.058      0.000       3.560       3.634
degree        -0.4726      0.011    -43.977      0.000      -0.494      -0.452
==============================================================================
Omnibus:                    19786.472   Durbin-Watson:                   1.920
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           269533.997
Skew:                           2.574   Prob(JB):                         0.00
Kurtosis:                      15.844   Cond. No.                         3.23
==============================================================================
```

About heteroskedasticity

```
name = ['Lagrange multiplier statistic', 'p-value',

        'f-value', 'f p-value']

test = sms.het_breuschpagan(lm_tv.resid, lm_tv.model.exog)

lzip(name, test)

[('Lagrange multiplier statistic', 222.95229561145405),
 ('p-value', 2.0532020056763167e-50),
 ('f-value', 224.41994797295303),
 ('f p-value', 1.4285427344176713e-50)]
```

A low p-value indicates that we have heteroskedasticity

At lower levels of degree, the residuals from predicting tvhours have more variance



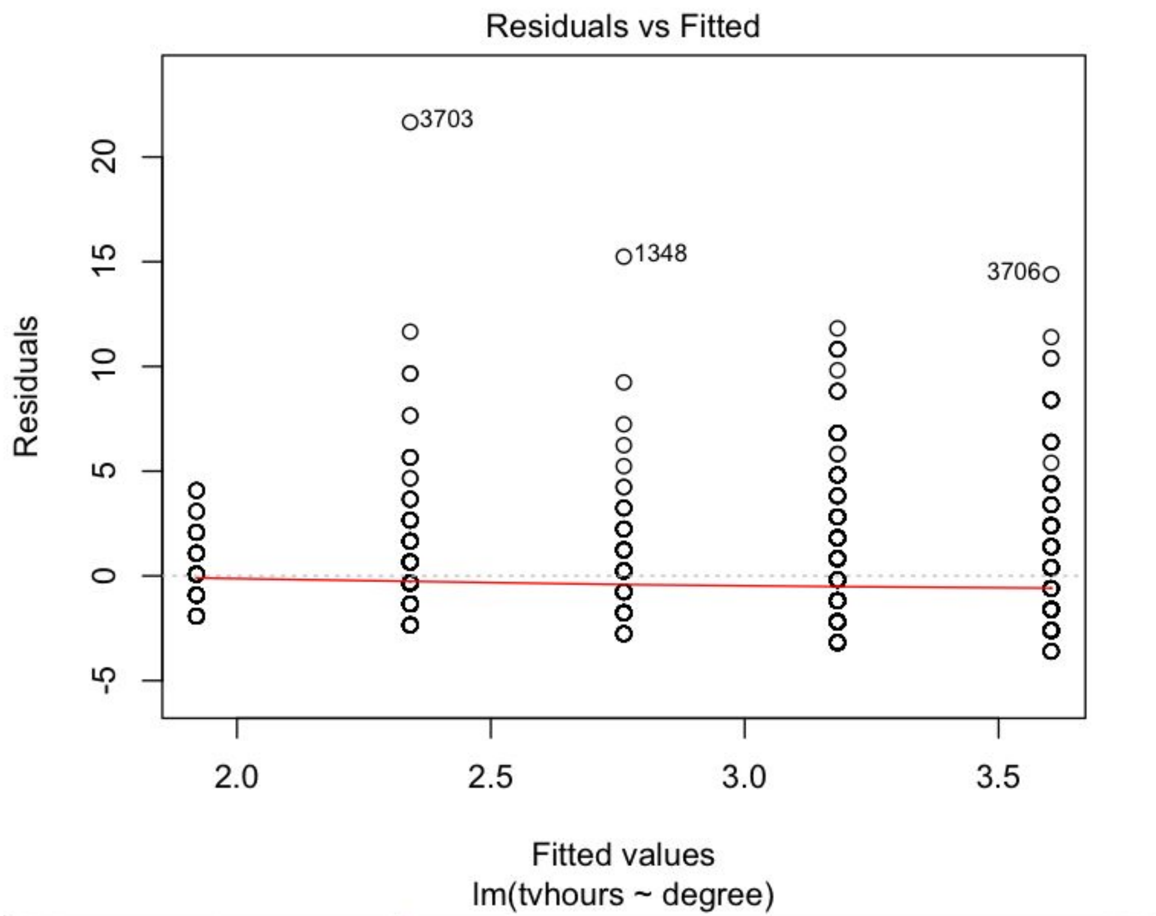
How did I do that graph?

```
d['yhat'] = lm_tv.fittedvalues

d['resid'] = lm_tv.resid


import seaborn as sns

sns.boxplot(x="degree", y="resid",

                        data=d)
```

Or this ...

Residuals versus fitted (predicted) values (in R):

```
plot(lm.tv)
```



*

Huber-White standard errors

From Wooldridge (2009): p 283

It can be shown that a valid estimator of $\mathrm{Var}(\hat{\beta}_j)$, under Assumptions MLR.1 through MLR.4, is

$$\widehat{\mathrm{Var}}(\hat{\beta}_j) = \frac{\sum_{i=1}^{n} \hat{r}_{ij}^2 \hat{u}_i^2}{\mathrm{SSR}_j^2}, \quad \boxed{8.4}$$

where $\hat{r}_{ij}$ denotes the i^{th} residual from regressing x_j on all other independent variables, and SSR_j is the sum of squared residuals from this regression (see Section 3.2 for the partialling out representation of the OLS estimates). The square root of the quantity in (8.4) is called the **heteroskedasticity-robust standard error** for $\hat{\beta}_j$. In econometrics, these robust standard errors are usually attributed to White (1980). Earlier works in statistics, notably those by Eicker (1967) and Huber (1967), pointed to the possibility of obtaining such robust standard errors. In applied work, these are sometimes called *White*, *Huber*, or *Eicker standard errors* (or some hyphenated combination of these names). We will just refer to them as *heteroskedasticity-robust standard errors*, or even just *robust standard*

Huber-White standard errors

Robust or “sandwich” errors

- Huber-White standard errors relax the assumption of i.i.d. errors
- They estimate a new variance of b1 that can be used in the presence of heteroskadasticity

In matrix form, homoskedasticity

$$\mathrm{Var}(\beta) = (\mathbf{X}^{\top}\mathbf{X})^{-1}\mathbf{X}^{\top}\sigma^{2}\mathbf{I}((\mathbf{X}^{\top}\mathbf{X})^{-1}\mathbf{X}^{\top})^{\top}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

In matrix form, robust standard errors

$$\text{Var}(\beta) = (\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\sigma^2\boldsymbol{\Omega}((\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top)^\top$$

$$\begin{bmatrix} \sigma_1^2 & 0 & 0 \\ 0 & \sigma_2^2 & 0 \\ & & \\ 0 & 0 & \sigma_n^2 \end{bmatrix}$$

What do you do if you have heteroskedasticity?

Get robust standard errors

What do you do if you have heteroskedasticity? Get robust standard errors

```
lm_tv_rse = smf.ols(formula = "tvhours ~ degree", data = d).fit(cov_type='HC3')
print (lm_tv_rse.summary())
                            OLS Regression Results
==============================================================================
Dep. Variable:                tvhours   R-squared:                       0.054
Model:                            OLS   Adj. R-squared:                  0.054
Method:                 Least Squares   F-statistic:                     2288.
Date:                Tue, 11 Jun 2019   Prob (F-statistic):               0.00
Time:                        22:05:34   Log-Likelihood:                -75921.
No. Observations:               33788   AIC:                         1.518e+05
Df Residuals:                   33786   BIC:                         1.519e+05
Df Model:                           1
Covariance Type:                  HC3
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      3.5967      0.021    171.894      0.000       3.556       3.638
degree        -0.4726      0.010    -47.828      0.000      -0.492      -0.453
==============================================================================
Omnibus:                    19786.472   Durbin-Watson:                   1.920
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           269533.997
Skew:                           2.574   Prob(JB):                         0.00
Kurtosis:                      15.844   Cond. No.                         3.23
==============================================================================

Warnings:
[1] Standard Errors are heteroscedasticity robust (HC3)
```

Assumption #6

Normality of the errors

- The errors should come from a (standard) normal distribution
- Empirically, this is often not the case, but we can invoke the Central Limit Theorem and Law of Large Numbers to justify using our usual asymptotic inference, especially with reasonable sample sizes