POLS/CSSS 503:
Advanced Quantitative Political Methodology

Linear Regression: Specification and Fitting

Christopher Adolph

Department of Political Science
and
Center for Statistics and the Social Sciences
University of Washington, Seattle

Specification

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \varepsilon$$

(Note: for the moment, we use X_1 rather than x_1 to refer to the first covariate)

Model specification refers to the modeler's choice of X_1, X_2, etc.:

which X's we include in our model, which we exclude, & how we transform them

We need to get this right for substantive & statistical reasons

In observational research, a large % of criticism regard specification

The goal of most experimental or quasi-experimental work,
is avoiding specification altogether, but even then not always possible

First, we'll talk about what different specifications imply substantively

Later, we'll talk about how to choose a specification: fitting the model

Outline of topics

Omitted variable bias & Specification

Transforming covariates

Transforming response variables

Diagnosing heteroskedasticity & misspecification

Goodness of Fit tests

Omitted variable bias

Let's imagine we know that the *true* model for some data Y is

$$Y_i = \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i + \varepsilon_i^{\text{true}}$$

Data generating process (DGP): the model generating the population we sample (usually a fiction)

Which quantitative methods will recover the truth about the model's parameters?

Unpacking the question:

Omitted variable bias

Let's imagine we know that the *true* model for some data Y is

$$Y_i = \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i + \varepsilon_i^{\text{true}}$$

Data generating process (DGP): the model generating the population we sample (usually a fiction)

Which quantitative methods will recover the truth about the model's parameters?

Unpacking the question:

- "recover the truth": yield an efficient, unbiased estimate;
  one that is true on average (and close to the truth) over many samples of X, Y, Z

Omitted variable bias

Let's imagine we know that the *true* model for some data Y is

$$Y_i = \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i + \varepsilon_i^{\text{true}}$$

Data generating process (DGP): the model generating the population we sample (usually a fiction)

Which quantitative methods will recover the truth about the model's parameters?

Unpacking the question:

- "recover the truth": yield an efficient, unbiased estimate;
  one that is true on average (and close to the truth) over many samples of X, Y, Z
- "methods": includes the method of estimation;
  e.g. least squares

Omitted variable bias

Let's imagine we know that the *true* model for some data Y is

$$Y_i = \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i + \varepsilon_i^{\text{true}}$$

Data generating process (DGP): the model generating the population we sample (usually a fiction)

Which quantitative methods will recover the truth about the model's parameters?

Unpacking the question:

- "recover the truth": yield an efficient, unbiased estimate;
  one that is true on average (and close to the truth) over many samples of X, Y, Z
- "methods": includes the method of estimation;
  e.g. least squares
- "model": includes the choice of specification;
  e.g., which controls, transformations, and interactions to include on the RHS

Omitted variable bias

Because Y_i was constructed by adding together

$$\beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i \qquad + \qquad \text{a Normal disturbance,}$$

estimation by least squares will be not only be unbiased,
but also the best unbiased method (most efficient)

Omitted variable bias

Because Y_i was constructed by adding together

$$\beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i \qquad + \qquad \text{a Normal disturbance,}$$

estimation by least squares will be not only be unbiased,
but also the best unbiased method (most efficient)

But it will only be unbiased if we choose the right specification

If we estimate

$$Y_i = \beta_0^* + \beta_1^* X_i + \varepsilon_i^*$$

We won't get any estimate for β_2 (because we assumed it was zero by omitting it)

Moreover, it will often be the case that $\hat{\beta}_1^*$ is a *biased* estimate of β_1

Omitted variable bias

The source of this bias can be shown formally

We estimated this

$$Y_i = \beta_0^* + \beta_1^* X_i + \varepsilon_i^*$$

leaving out Z_i. Suppose we ran an auxilliary regression of Z_i on X_i:

$$Z_i = \gamma_0 + \gamma_1 X_i + \nu_i$$

Omitted variable bias

The source of this bias can be shown formally

We estimated this

$$Y_i = \beta_0^* + \beta_1^* X_i + \varepsilon_i^*$$

leaving out Z_i. Suppose we ran an auxilliary regression of Z_i on X_i:

$$Z_i = \gamma_0 + \gamma_1 X_i + \nu_i$$

We could substitute this back into the true model,

$$Y_i = \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i + \varepsilon_i^{\text{true}}$$

Omitted variable bias

The source of this bias can be shown formally

We estimated this

$$Y_i = \beta_0^* + \beta_1^* X_i + \varepsilon_i^*$$

leaving out Z_i. Suppose we ran an auxilliary regression of Z_i on X_i:

$$Z_i = \gamma_0 + \gamma_1 X_i + \nu_i$$

We could substitute this back into the true model,

$$Y_i = \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i + \varepsilon_i^{\text{true}}$$

$$Y_i = \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} (\gamma_0 + \gamma_1 X_i + \nu_i) + \varepsilon_i^{\text{true}}$$

Omitted variable bias

The source of this bias can be shown formally

We estimated this

$$Y_i = \beta_0^* + \beta_1^* X_i + \varepsilon_i^*$$

leaving out Z_i. Suppose we ran an auxilliary regression of Z_i on X_i:

$$Z_i = \gamma_0 + \gamma_1 X_i + \nu_i$$

We could substitute this back into the true model,

$$\begin{aligned} Y_i &= \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} Z_i + \varepsilon_i^{\text{true}} \\ Y_i &= \beta_0^{\text{true}} + \beta_1^{\text{true}} X_i + \beta_2^{\text{true}} (\gamma_0 + \gamma_1 X_i + \nu_i) + \varepsilon_i^{\text{true}} \\ Y_i &= \left(\beta_0^{\text{true}} + \beta_2^{\text{true}} \gamma_0\right) + \left(\beta_1^{\text{true}} + \beta_2^{\text{true}} \gamma_1\right) X_i + \left(\varepsilon_i^{\text{true}} + \beta_2^{\text{true}} \nu_i\right) \end{aligned}$$

Omitted variable bias

The parenthetical terms below are what we recover when we run LS omitting Z_i

$$Y_i = \left(\beta_0^{\text{true}} + \beta_2^{\text{true}}\gamma_0\right) + \left(\beta_1^{\text{true}} + \beta_2^{\text{true}}\gamma_1\right) X_i + \left(\varepsilon_i^{\text{true}} + \beta_2^{\text{true}}\nu_i\right)$$

Omitted variable bias

The parenthetical terms below are what we recover when we run LS omitting Z_i

$$
\begin{aligned}
Y_i &= \left(\beta_0^{\text{true}} + \beta_2^{\text{true}}\gamma_0\right) + \left(\beta_1^{\text{true}} + \beta_2^{\text{true}}\gamma_1\right) X_i + \left(\varepsilon_i^{\text{true}} + \beta_2^{\text{true}}\nu_i\right) \\
Y_i &= \beta_0^* + \beta_1^* X_i + \varepsilon_i^*
\end{aligned}
$$

Omitted variable bias

The parenthetical terms below are what we recover when we run LS omitting Z_i

$$\begin{aligned} Y_i &= \left(\beta_0^{\text{true}} + \beta_2^{\text{true}}\gamma_0\right) + \left(\beta_1^{\text{true}} + \beta_2^{\text{true}}\gamma_1\right) X_i + \left(\varepsilon_i^{\text{true}} + \beta_2^{\text{true}}\nu_i\right) \\ Y_i &= \beta_0^* + \beta_1^* X_i + \varepsilon_i^* \end{aligned}$$

The estimate we get of β_1 is:

$$\mathrm{E}(\hat{\beta}_1) = \beta_1^{\text{true}} + \beta_2^{\text{true}}\gamma_1$$

Omitted variable bias

The parenthetical terms below are what we recover when we run LS omitting Z_i

$$\begin{aligned} Y_i &= \left(\beta_0^{\text{true}} + \beta_2^{\text{true}}\gamma_0\right) + \left(\beta_1^{\text{true}} + \beta_2^{\text{true}}\gamma_1\right) X_i + \left(\varepsilon_i^{\text{true}} + \beta_2^{\text{true}}\nu_i\right) \\ Y_i &= \beta_0^* + \beta_1^* X_i + \varepsilon_i^* \end{aligned}$$

The estimate we get of β_1 is:

$$\begin{aligned} \mathrm{E}(\hat{\beta}_1) &= \beta_1^{\text{true}} + \beta_2^{\text{true}}\gamma_1 \\ &= \beta_1^{\text{true}} + \beta_2^{\text{true}} \left(\frac{\sum_i (X_i - \bar{X})(Z_i - \bar{Z})}{\sum_i (X_i - \bar{X})^2} \right) \end{aligned}$$

Which is unbiased only if $\beta_2^{\text{true}} = 0$ or $\mathrm{corr}(X_i, Z_i) = 0$.

Omitted variable bias

Thus there are only two conditions under which we can safely omit variable Z from our model:

β is really zero; Z has no effect on Y in the "true" model

or

The correlation of Z with the included X's is zero

we should include any Z that is correlated with both Y and some included X

(A major exception – post-treatment variables – discussed below)

This is why the complaint "You should have controlled for . . . " carries so much weight in criticizing empirical research.

Specification arguably *the* major concern in most observational research (along with selection & endogeneity)

Omitted variable bias

Does this mean we should include the kitchen sink in the regression?

Is there a penalty to including irrelevant variables?

Yes, but it is smaller. Lose efficiency in two ways:

- Lost degrees of freedom
- Lost variance in relevant covariates after conditioning on irrelevant ones

So is the kitchen sink safer?

Kevin Clarke: you only *solve* OVB with "right" specification.
If there are countervailing biases, adding a subset of omitted variables could make things *worse*

My view: in observational research, show robustness to specification choice
Try to break your findings, then report how easily they are broken

Omitted variable bias

Is there any other reason why we should omit variables?

Generally, you should omit "post-treatment" variables
or intermediate steps in a (presumed) causal process

If you want to model $X \rightarrow Y$, and one path between X and Y is $X \rightarrow W \rightarrow Y$,
then omit W or risk W masking the underlying effect of X

Omitted variable bias

Is there any other reason why we should omit variables?

Generally, you should omit "post-treatment" variables
or intermediate steps in a (presumed) causal process

If you want to model $X \rightarrow Y$, and one path between X and Y is $X \rightarrow W \rightarrow Y$,
then omit W or risk W masking the underlying effect of X

Consider this hypothesis:

Post-college education makes people more likely to vote Democratic

Suppose we test this by regressing vote choice in 2004 on years of education

A critic objects that we should have controlled for party ID,
which is correlated with both our response and covariate. Is he right?

Omitted variable bias

Is there any other reason why we should omit variables?

Generally, you should omit "post-treatment" variables
or intermediate steps in a (presumed) causal process

If you want to model $X \rightarrow Y$, and one path between X and Y is $X \rightarrow W \rightarrow Y$,
then omit W or risk W masking the underlying effect of X

Consider this hypothesis:

Post-college education makes people more likely to vote Democratic

Suppose we test this by regressing vote choice in 2004 on years of education

A critic objects that we should have controlled for party ID,
which is correlated with both our response and covariate. Is he right?

The same logic says to include vote intention 10 minutes prior to voting!

Omitted variable bias

Is there any other reason why we should omit variables?

Generally, you should omit "post-treatment" variables
or intermediate steps in a (presumed) causal process

If you want to model $X \rightarrow Y$, and one path between X and Y is $X \rightarrow W \rightarrow Y$,
then omit W or risk W masking the underlying effect of X

Consider this hypothesis:

Post-college education makes people more likely to vote Democratic

Suppose we test this by regressing vote choice in 2004 on years of education

A critic objects that we should have controlled for party ID,
which is correlated with both our response and covariate. Is he right?

The same logic says to include vote intention 10 minutes prior to voting!

We can validly omit variables W which lie along the causal chain from affect X to Y's if we want βX to absorb the impact of X through W

Omitted variable bias

Is there any other reason why we should omit variables?

Generally, you should omit "post-treatment" variables
or intermediate steps in a (presumed) causal process

If you want to model $X \rightarrow Y$, and one path between X and Y is $X \rightarrow W \rightarrow Y$,
then omit W or risk W masking the underlying effect of X

Omitted variable bias

Is there any other reason why we should omit variables?

Generally, you should omit "post-treatment" variables
or intermediate steps in a (presumed) causal process

If you want to model $X \rightarrow Y$, and one path between X and Y is $X \rightarrow W \rightarrow Y$,
then omit W or risk W masking the underlying effect of X

Note an implication:

To get a good estimate of the relationship between X and Y from a regression,
we may need to craft the rest of the regression around that causal story

But now suppose we *do* want to understand the relationship between short-run intentions and voting

We'd need to specify the model differently

So testing multiple *hypotheses* might require multiple specifications,
even in the same paper

What makes linear regression "linear"?

Any thoughts?

What makes linear regression "linear"?

Any thoughts?

As long as this remains a valid statement:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \varepsilon$$

the regression is linear

But we can make any algebraic subtitutions we want

To see this, replace the big X's of the above equation with functions of small x's

This is a trick from high school algebra:
I'm just renaming complex algebraic expressions with simple names

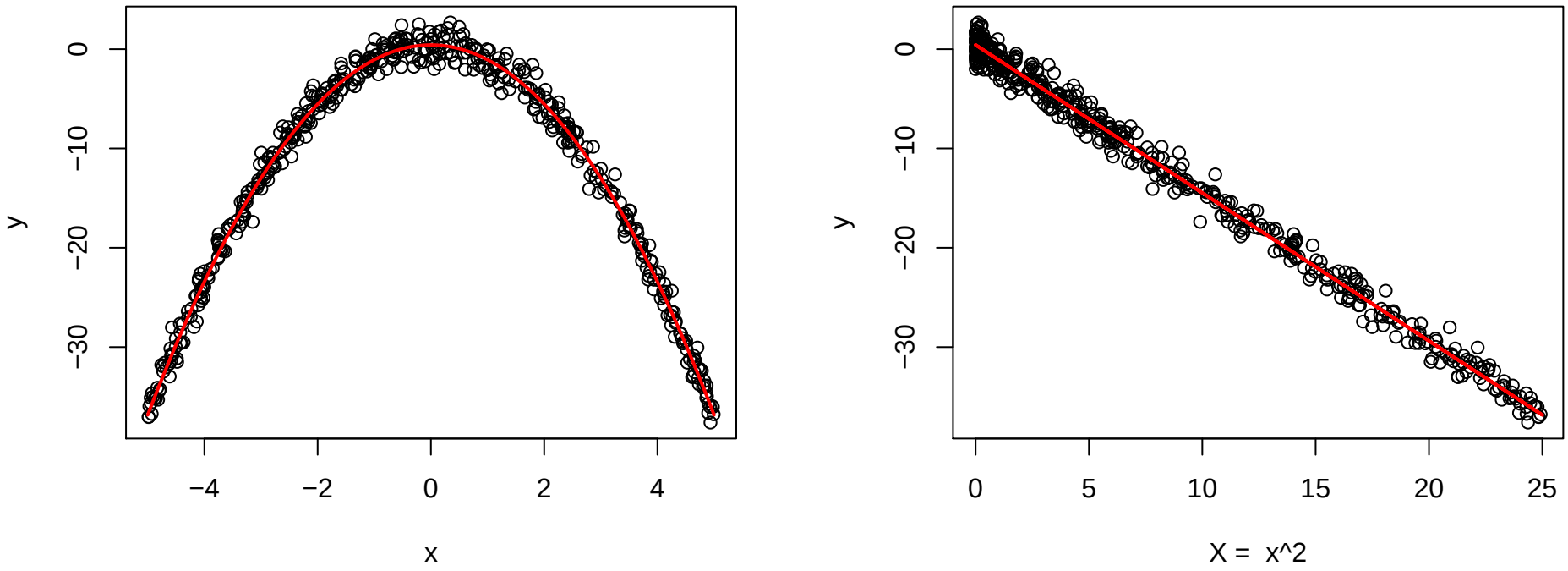
Transformations of covariates

Suppose $X_1 = x_1^2$ and $X_2 = x_1$. Then, by algebraic substitution, the DGP is:

$$Y = \beta_0 + \beta_1 x_1^2 + \beta_2 x_1 + \ldots + \varepsilon$$

Sample output (note we're leaving out x_1 to make a point; normally we need it, too):

```
Call:
lm(formula = y ~ I(x^2))

Residuals:
    Min      1Q  Median      3Q     Max
-4.3876 -0.1713  0.2031  0.3971  0.7495

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.9008332  0.0310244  29.036   <2e-16 ***
I(x^2)      -0.0006920  0.0007175  -0.965    0.335
```



The same regression; two views

Left is the regression in the original, untransformed scale of x

Right is the regression as R sees it, in the transformed scale $x^2 = X$

Linear regression is always linear in the transformed covariate

It may be very curvilinear in the scale we care about

Transformations of covariates

Suppose $X_1 = x_1^3$, $X_2 = x_1^2$, $X_3 = x_1$.

This is a cubic polynomial specification:

$$Y = \beta_0 + \beta_1 x_1^3 + \beta_2 x_1^2 + \beta_3 x_1 + \ldots + \varepsilon$$

Or even add a quartic term, $X_4 = x_1^4$. Then,

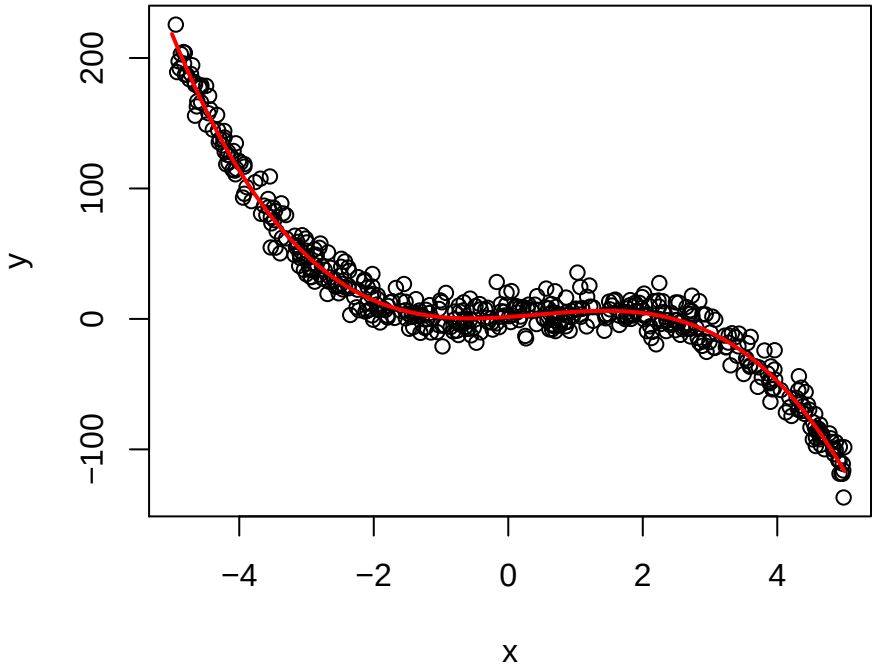
$$Y = \beta_0 + \beta_1 x_1^3 + \beta_2 x_1^2 + \beta_3 x_1 + \beta_4 x_1^4 + \ldots + \varepsilon$$

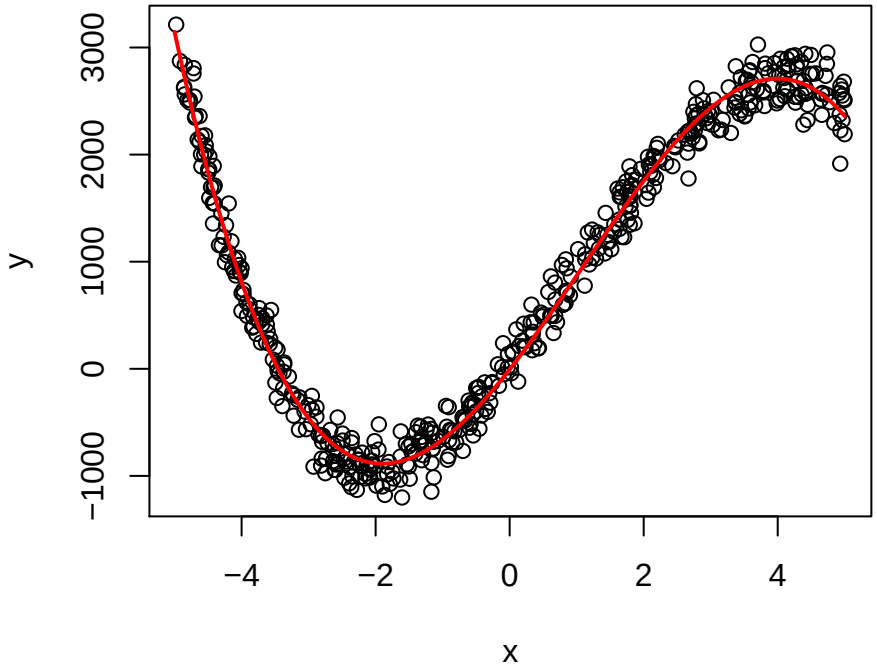
Sample output for a cubic (3rd order) polynomial:

```
Call:
lm(formula = y ~ I(x^3) + I(x^2) + I(x))

Residuals:
    Min      1Q  Median      3Q     Max
-4.3835 -0.1706  0.2006  0.3983  0.7536

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.8840553  0.0830528  10.644   <2e-16 ***
I(x^3)       0.0002337  0.0011134   0.210    0.834
I(x^2)      -0.0044455  0.0168552  -0.264    0.792
I(x)         0.0166821  0.0721171   0.231    0.817
```

None of the coefficients are significant, but they collectively explain a lot of variance. (How is this possible?)

With polynomials (and with other interactions)
t-tests of individual parameters are less interesting than CIs around $\hat{Y}$



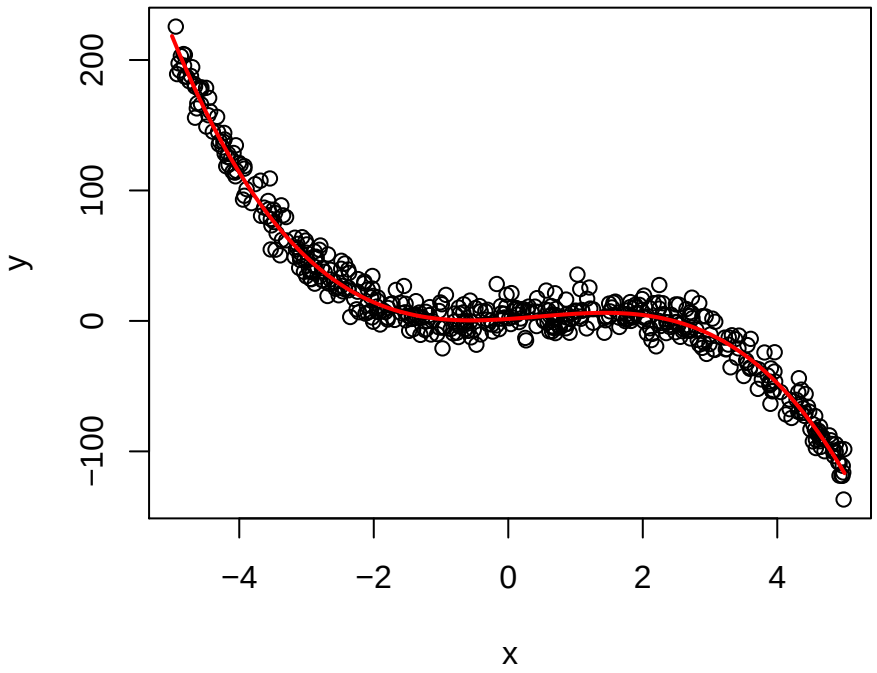


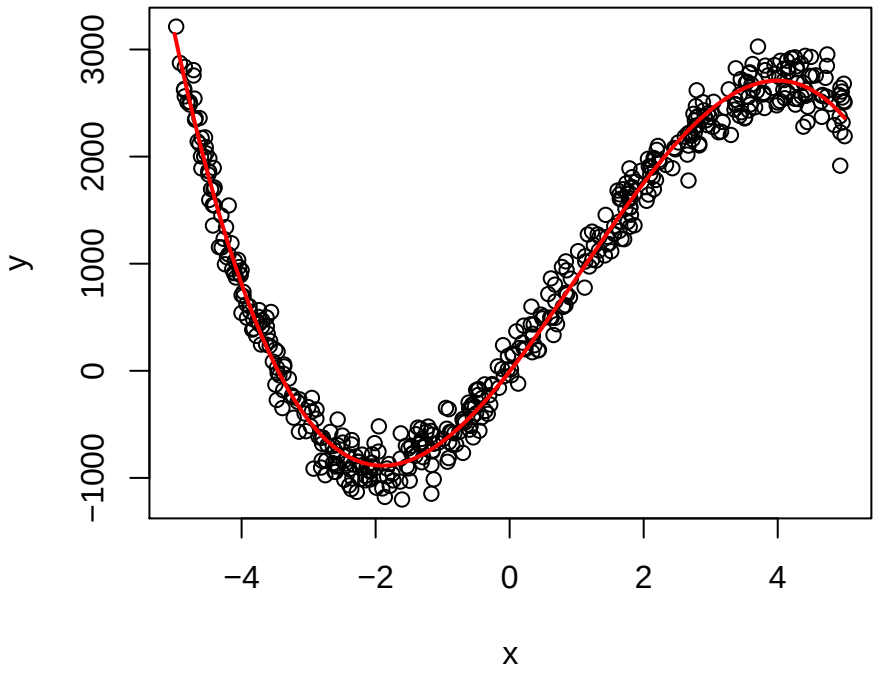
These are two different regressions.

Left is a cubic (3rd order) polynomial specificiation. It has 2 bends

Right is a quartic (4th order) polynomial. It has 3 bends

Each polynomial order we add puts another bend in the line





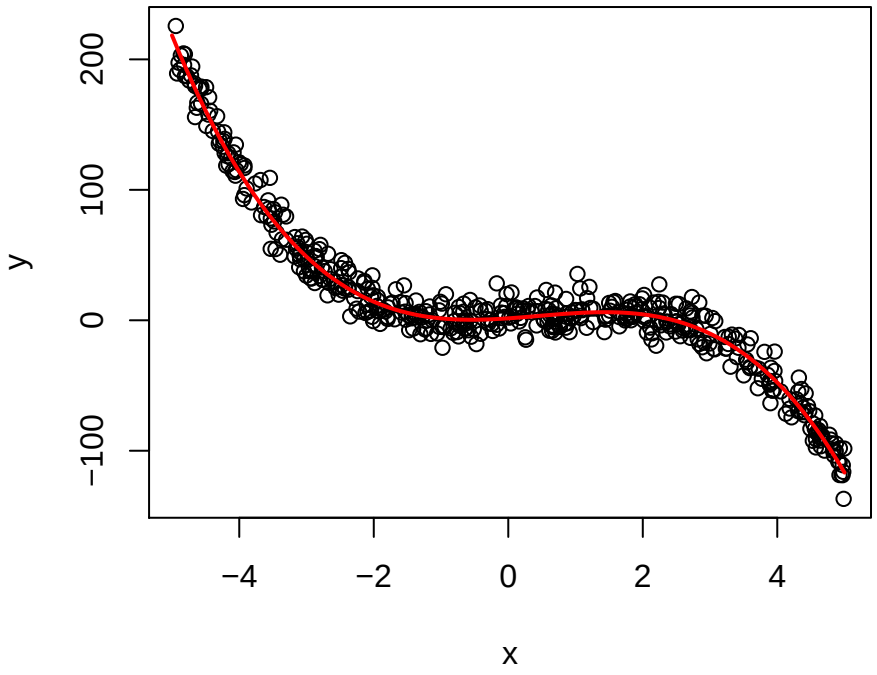
If we include n terms, there will be a bend for every observation

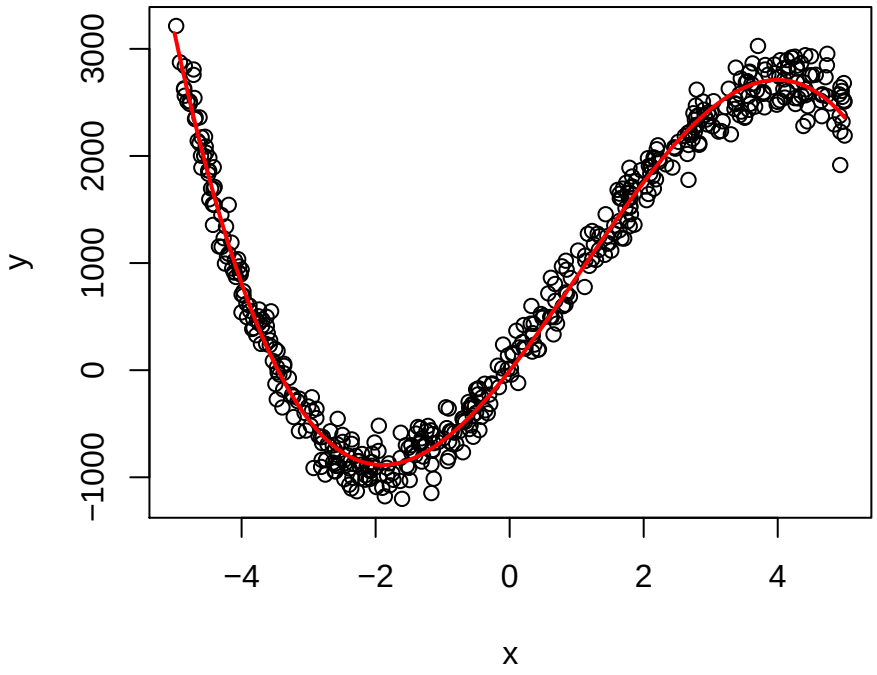
Called "curve-fitting": a perfect (& perfectly useless) model

Always have a theoretical reason to include polynomial terms

Seldom is more than a quadratic justified by theory

If you include polynomial terms, you need to interpret the result graphically

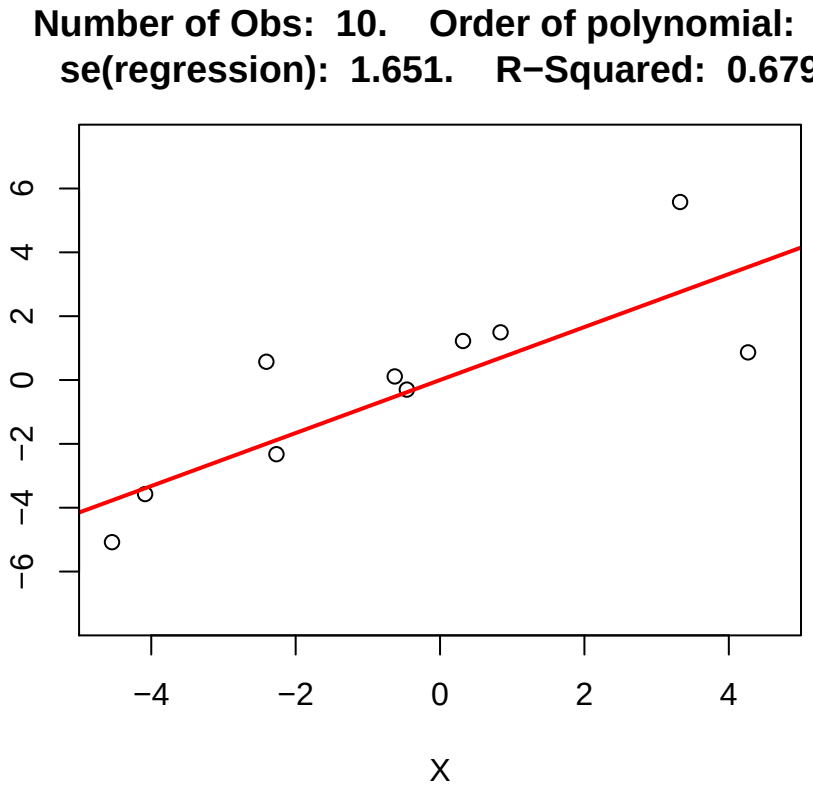


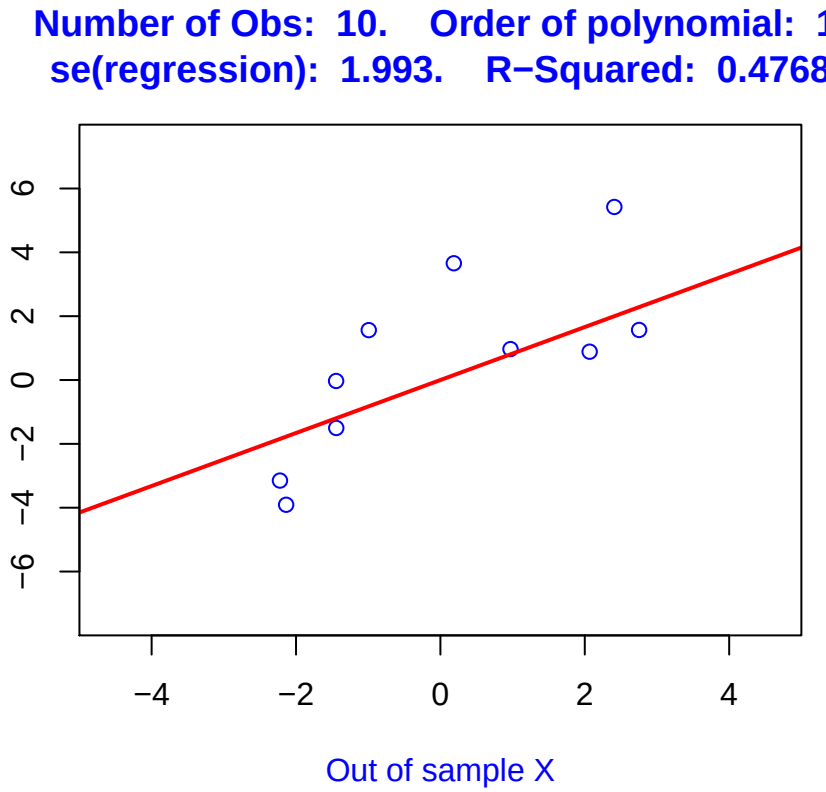


Warnings about polynomial fits:

- High order polynomials will *always* fit the sample well,
  but seldom fit the population well (curve-fitting)

- Extrapolation from polynomial or interactive specifications is dangerous
  These functional forms behave wildly outside the known data



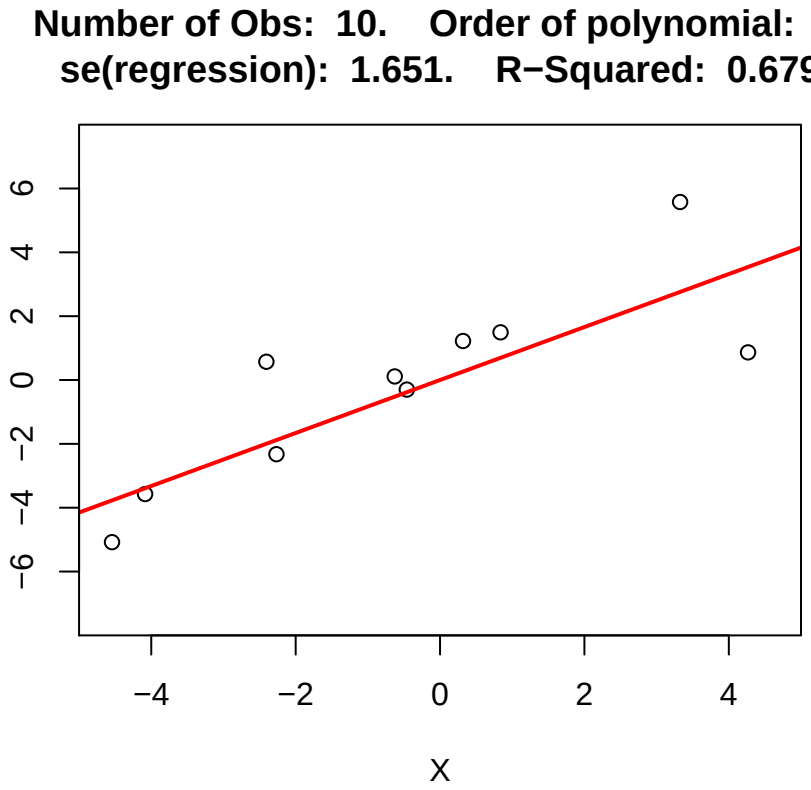


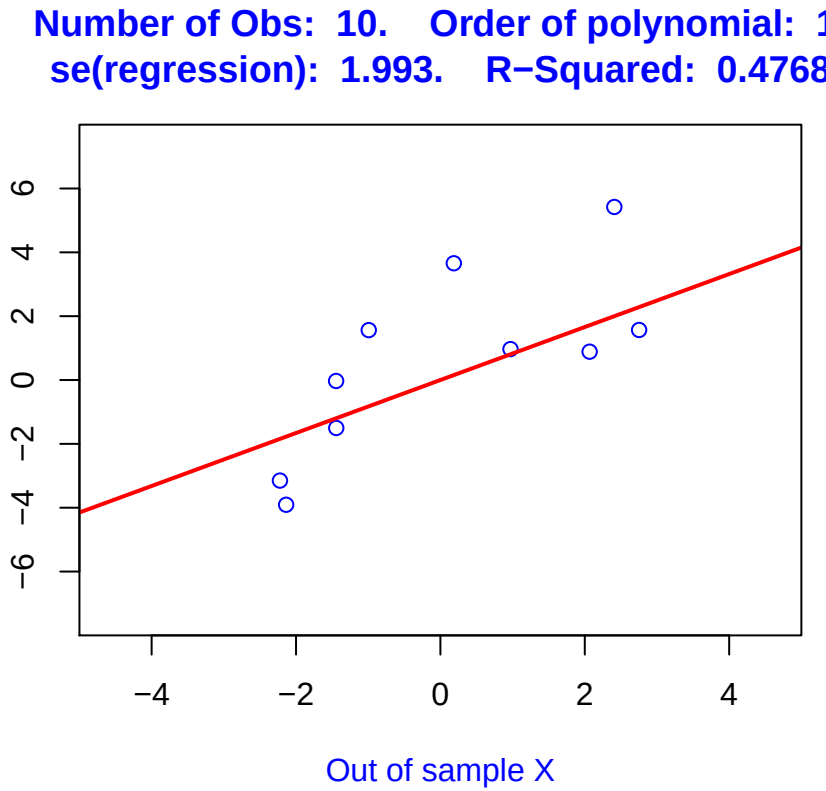
Polynomial overfitting experiment: generate 10 obs from the "true" model:

$$Y = x + \varepsilon, \qquad \varepsilon \sim \mathrm{N}(0, 3)$$

and fit these data using different polynomials of x.

We will show the fit of the model to the original data on the left



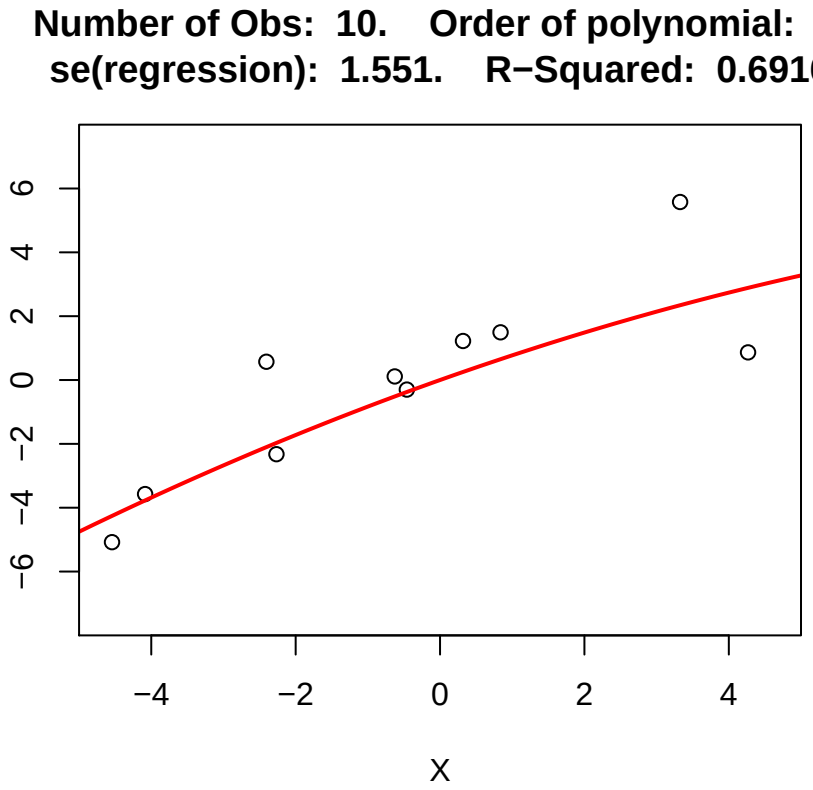


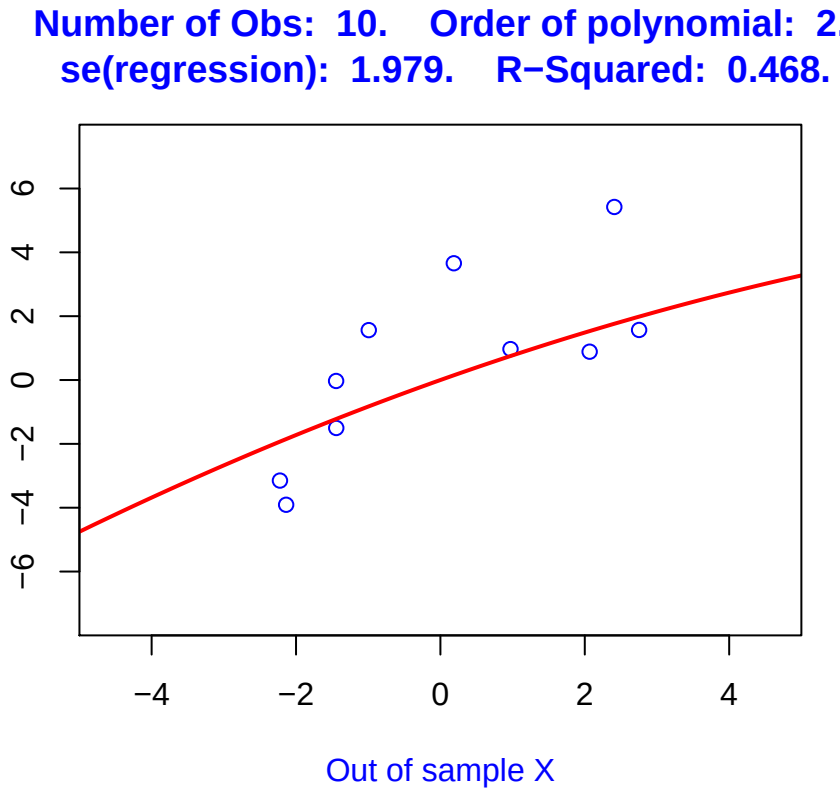
We will also draw new "out of sample data" from the same true model:

$$Y_{\text{OOS}} = x_{\text{OOS}} + \varepsilon_{\text{OOS}}, \qquad \varepsilon \sim \text{N}(0, 3)$$

and use the model as fitted on the original dataset to predict out of sample cases

We will show the fit of the old model to the out-of-sample data on the right

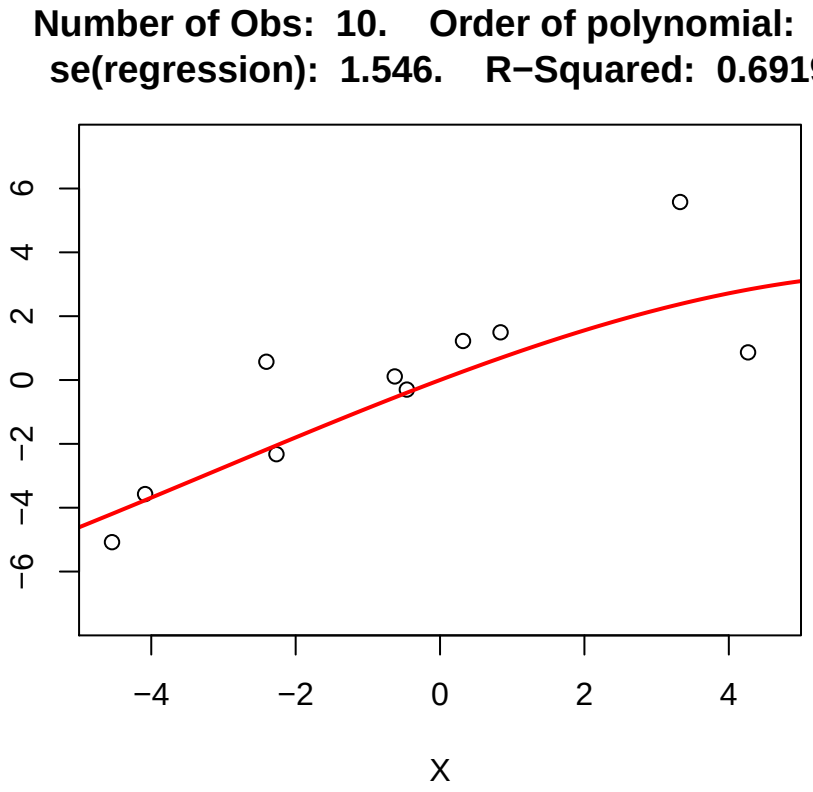


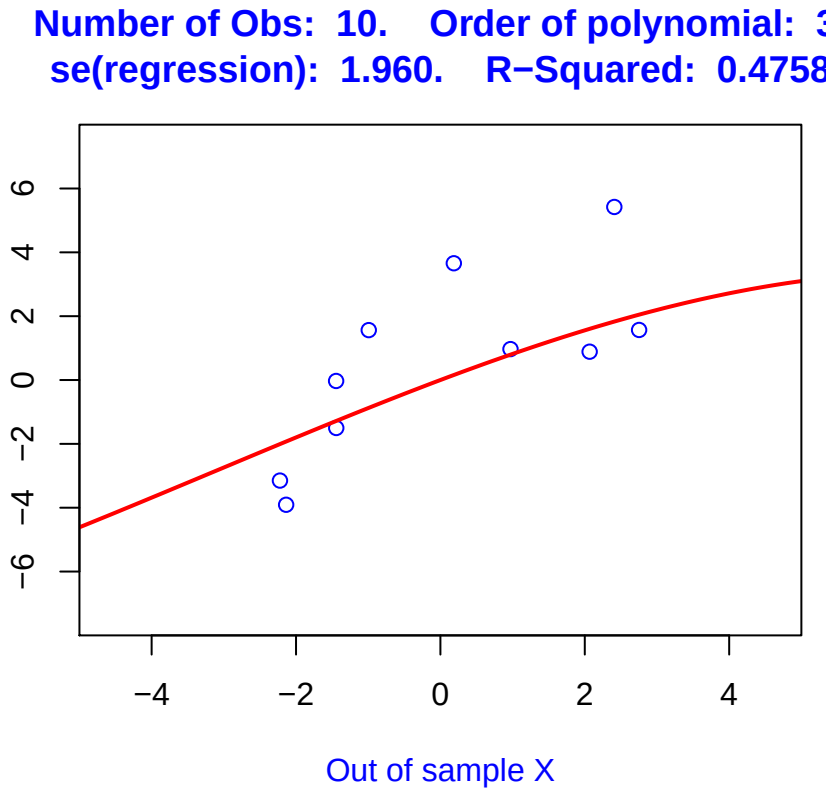


Above is the fit from a quadratic specification of x, ie, we estimated:

$$Y = \hat{\beta}_1 x + \hat{\beta}_2 x^2 + \hat{\varepsilon}$$

Note that we omitted the constant for didactic reasons



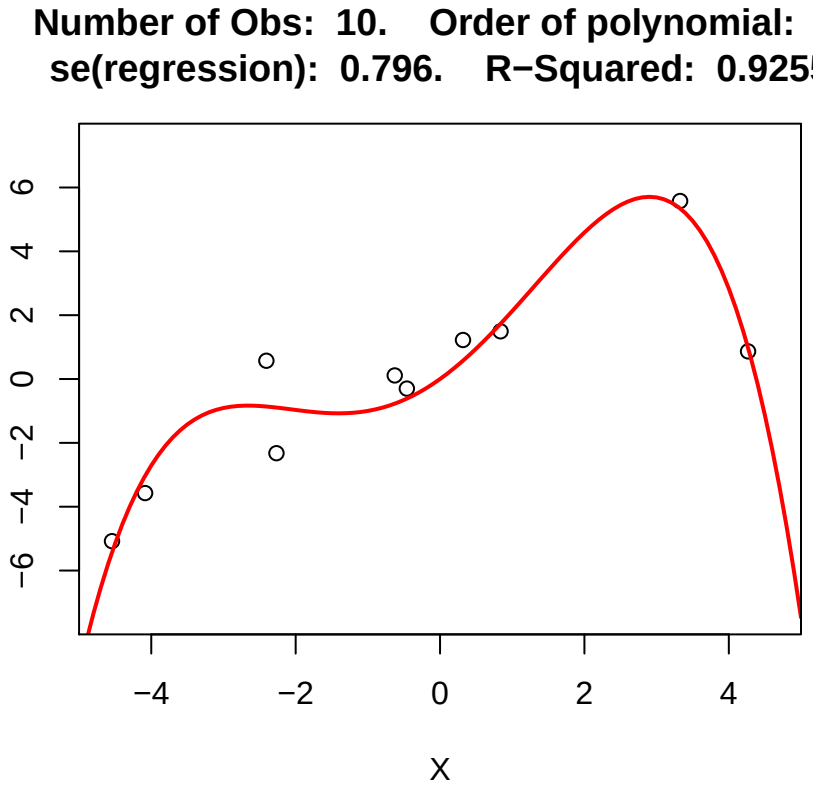


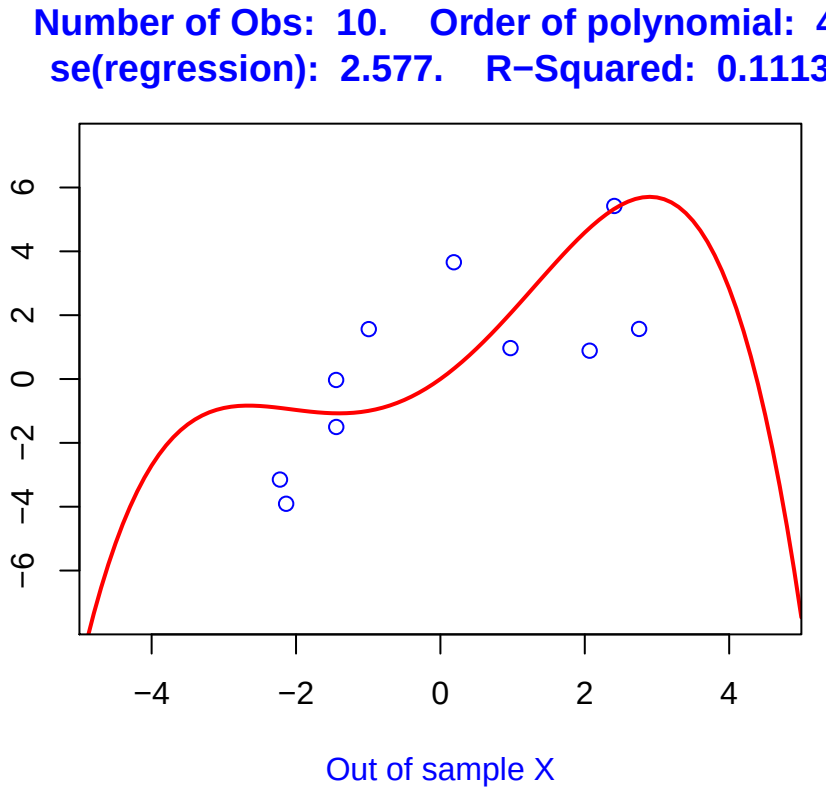
Above is the fit from a cubic specification of x; that is, we estimated:

$$Y = \hat{\beta}_1 x + \hat{\beta}_2 x^2 + \hat{\beta}_3 x^3 + \hat{\varepsilon}$$

How many polynomials can we add and still find $\hat{\beta}$?

What will happen to the fit in and out of sample as we add polynomials?

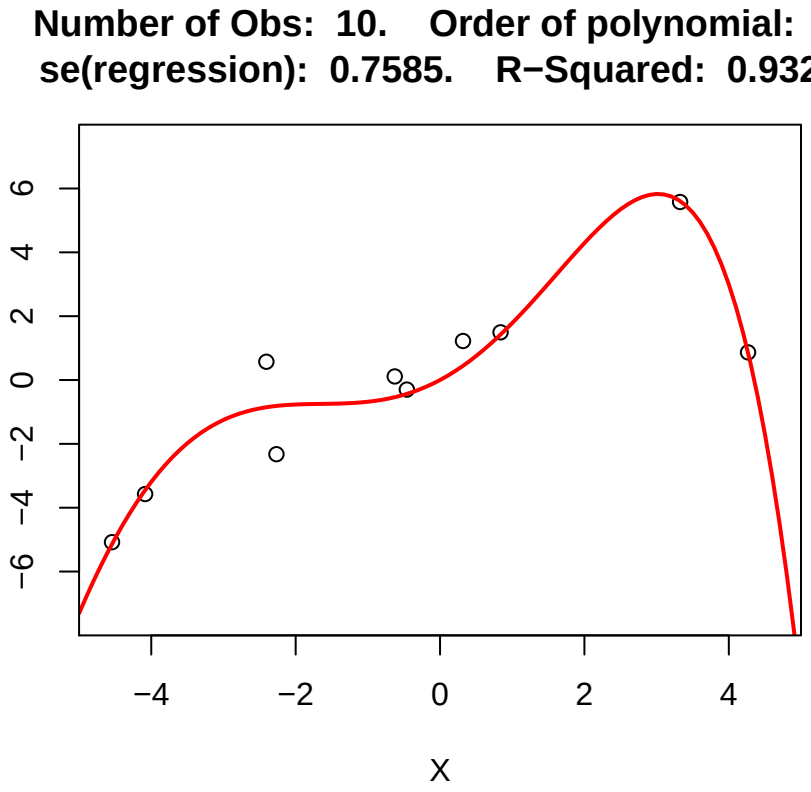


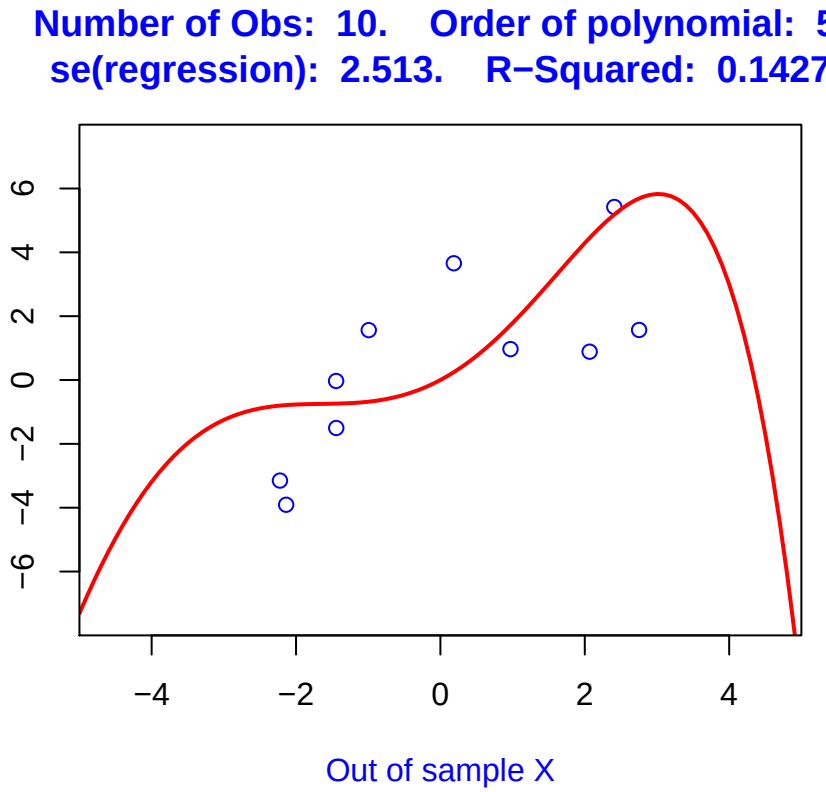


Above is the fit from a quartic specification of x; that is, we estimated:

$$Y = \hat{\beta}_1 x + \hat{\beta}_2 x^2 + \hat{\beta}_3 x^3 + \hat{\beta}_4 x^4 + \hat{\varepsilon}$$

On the left, we see the model finds a curious non-linearity, by which low and high x suppress y, but middle values of x increase y. Do you trust this finding?

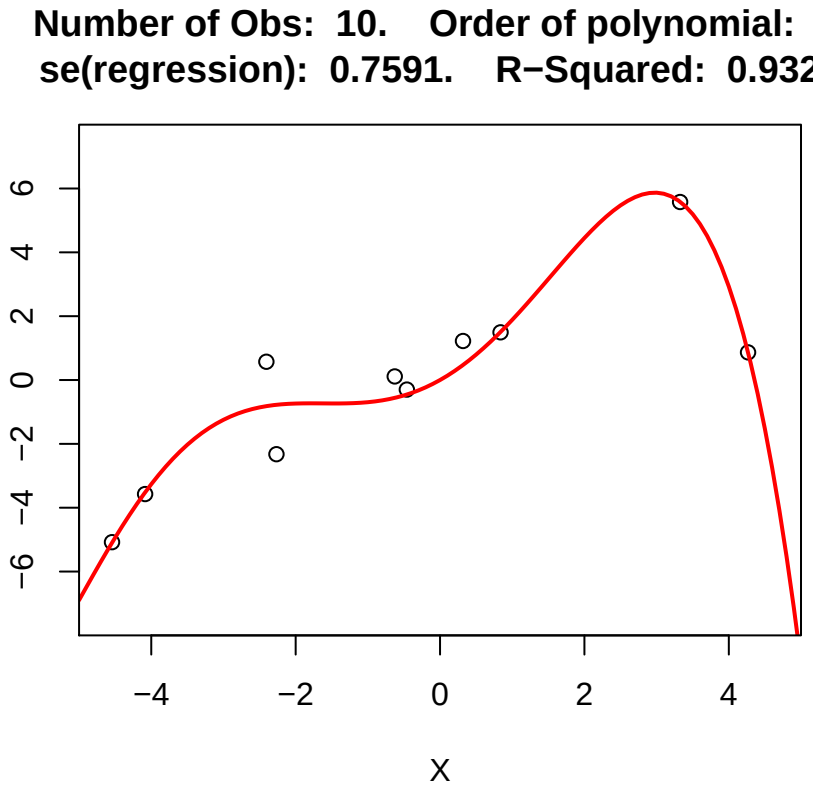


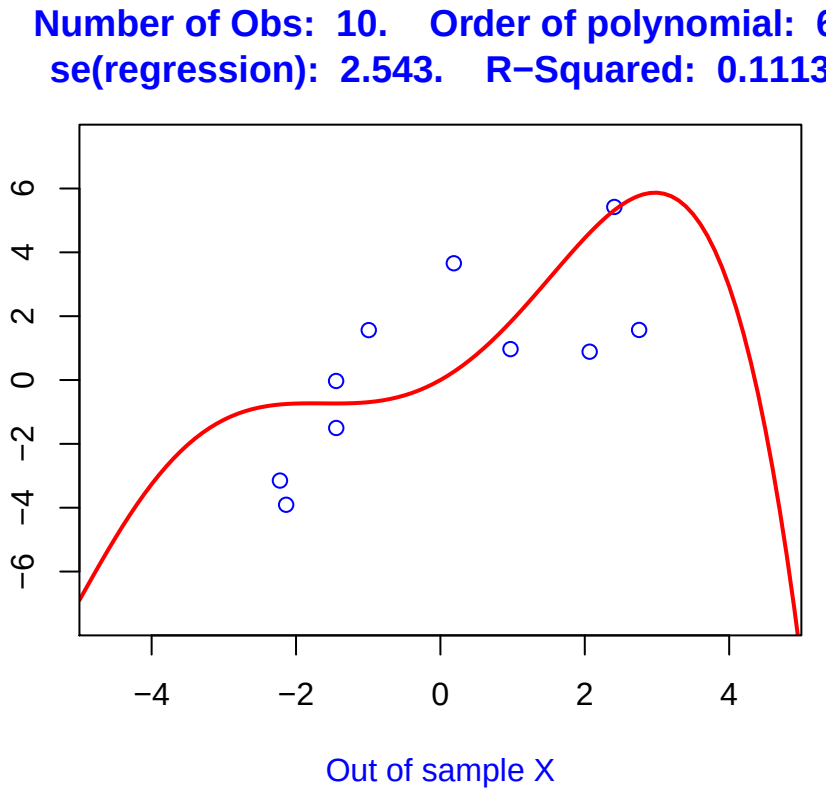


In small samples,
outliers can easily create the illusion of complex curves relating x and y

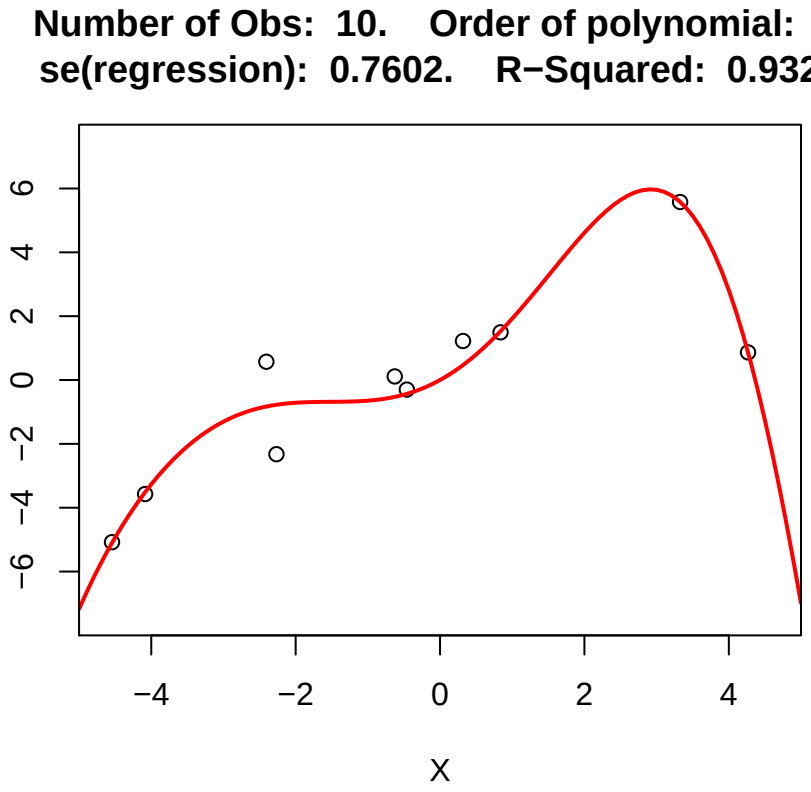
We need *a lot* of data to discern if such curves are more than spurious

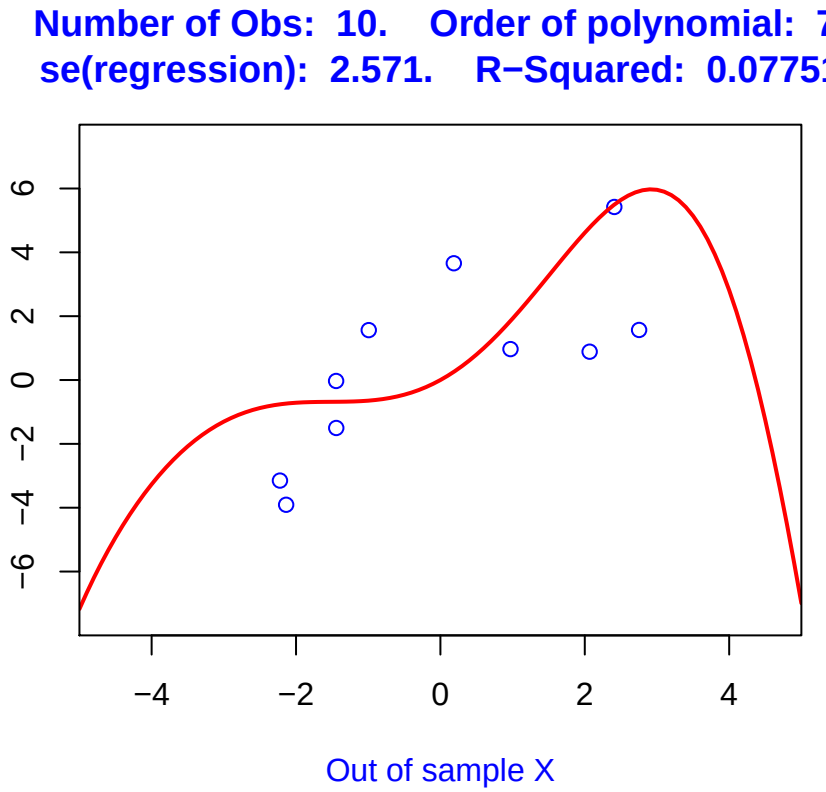
(And so we probably need a strong theory, too, to justify the data collection)



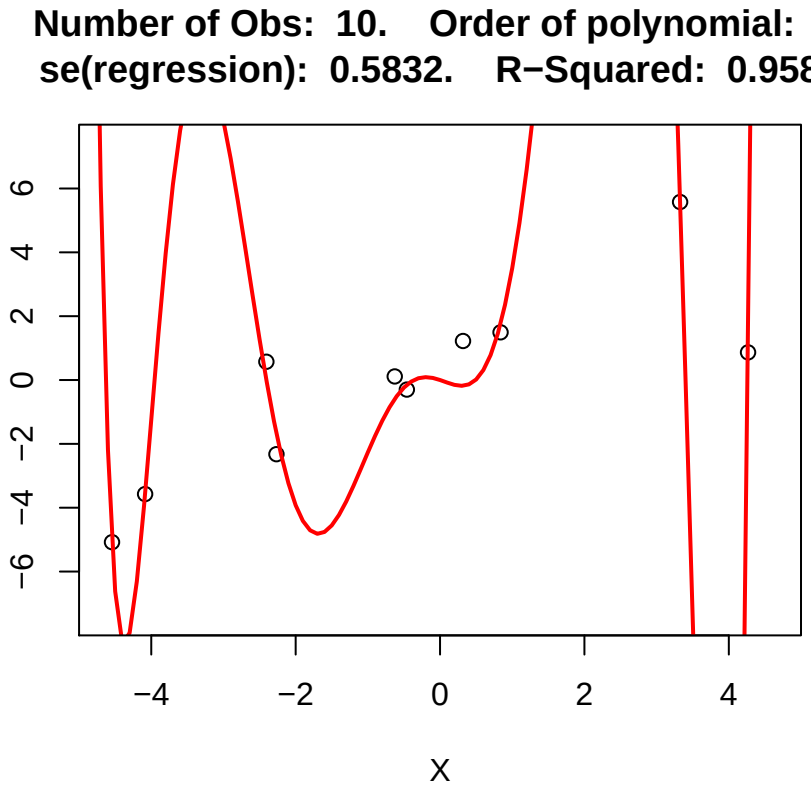


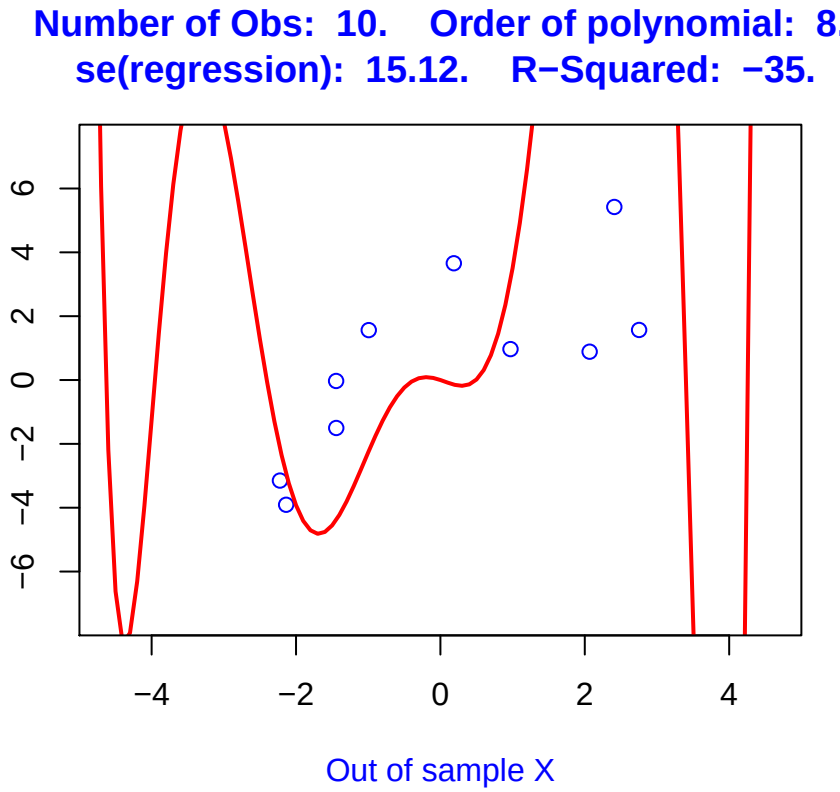
What happens as we approach a tenth-order polynomial?





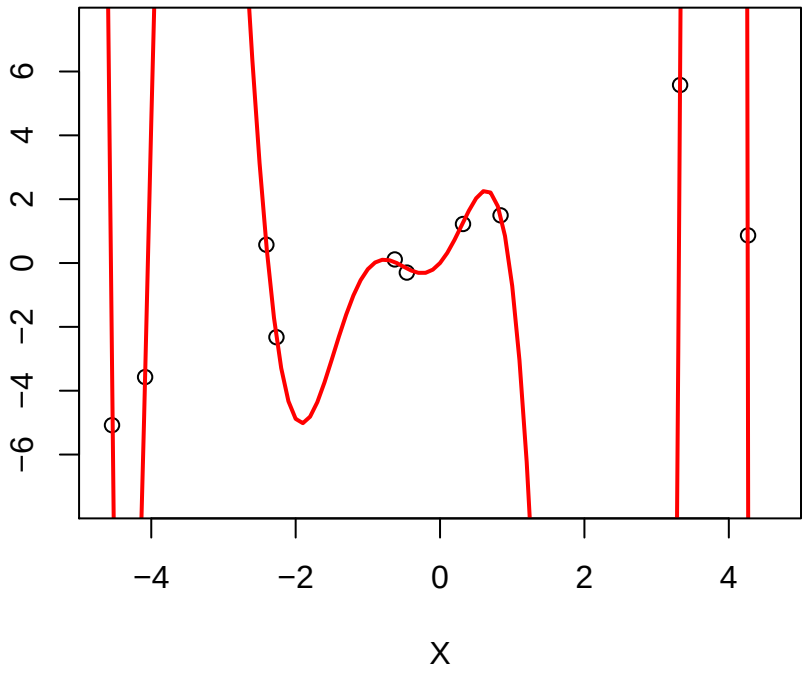
What happens as we approach a tenth-order polynomial?



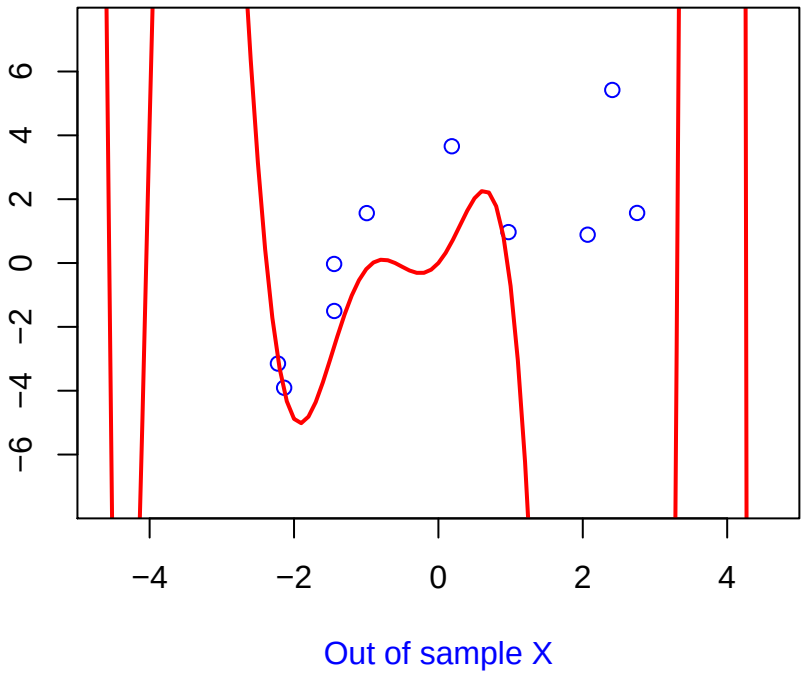


What happens as we approach a tenth-order polynomial?

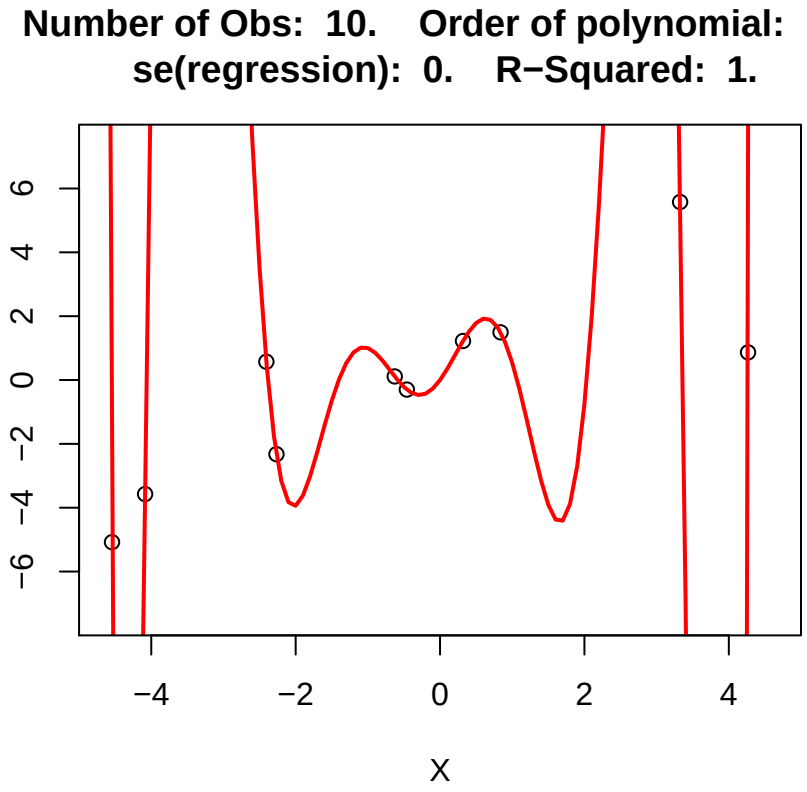
Number of Obs: 10. Order of polynomial: 9.
se(regression): 0.05322. R−Squared: 0.9997.

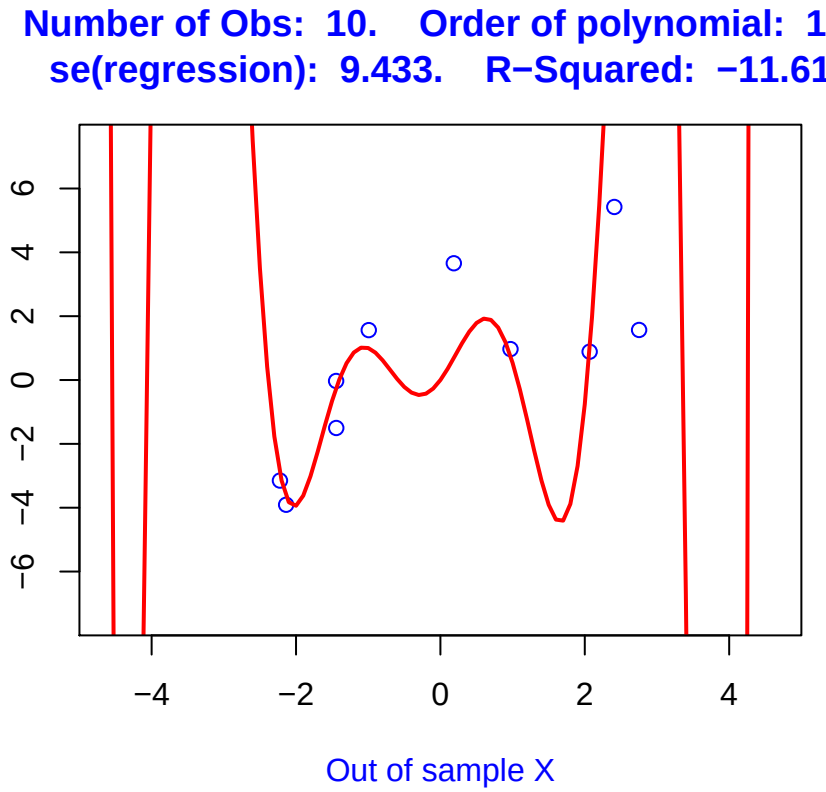


Number of Obs: 10. Order of polynomial: 9.
se(regression): 46.64. R−Squared: −384.0.



What happens as we approach a tenth-order polynomial?

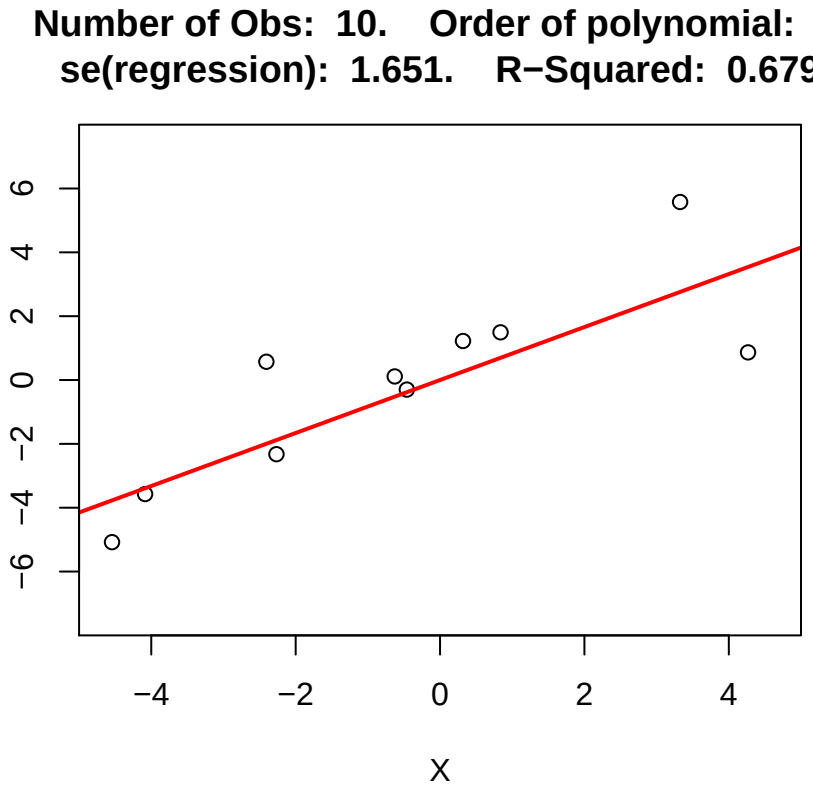


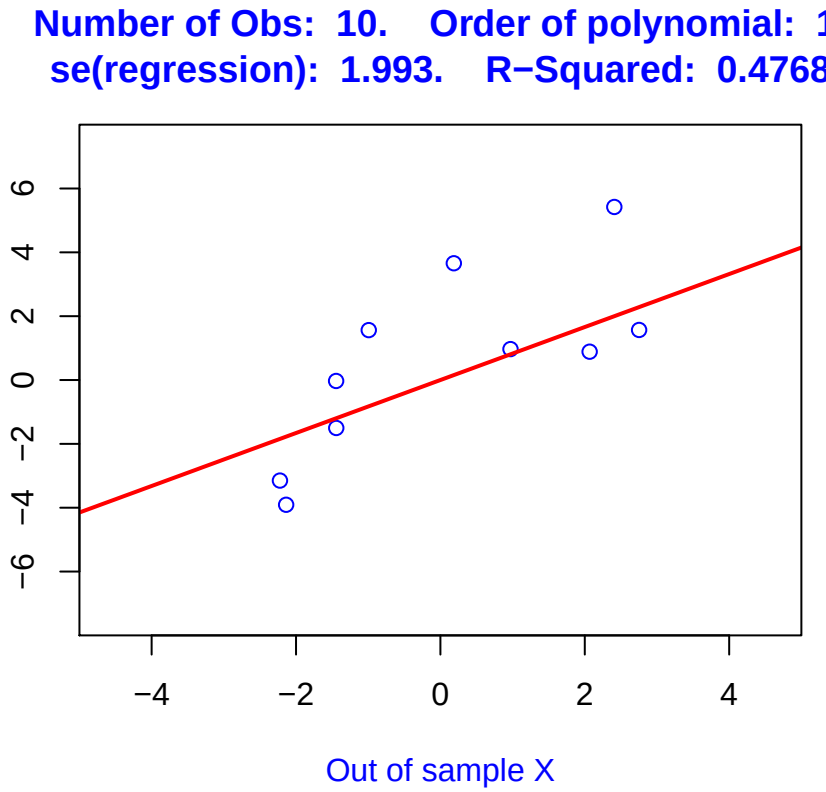


When the number of parameters in the model equals the number of observations, least squares is able to fit a line through every datapoint

This means the fit will be "perfect": no error

And out of sample, it will be completely useless,
and worse than guessing that y simply equals its sample mean in every case

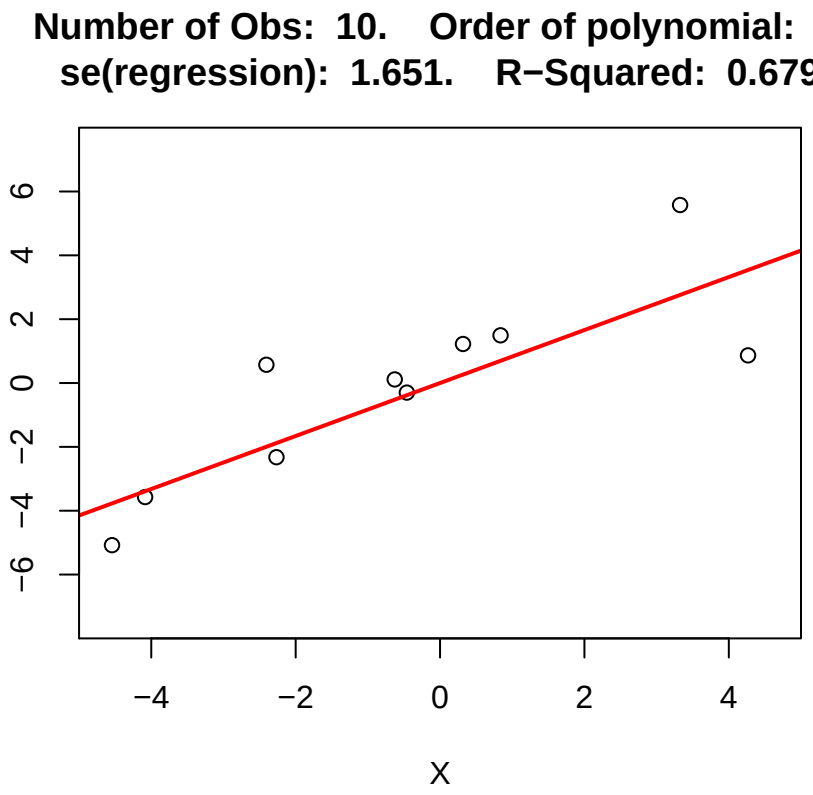


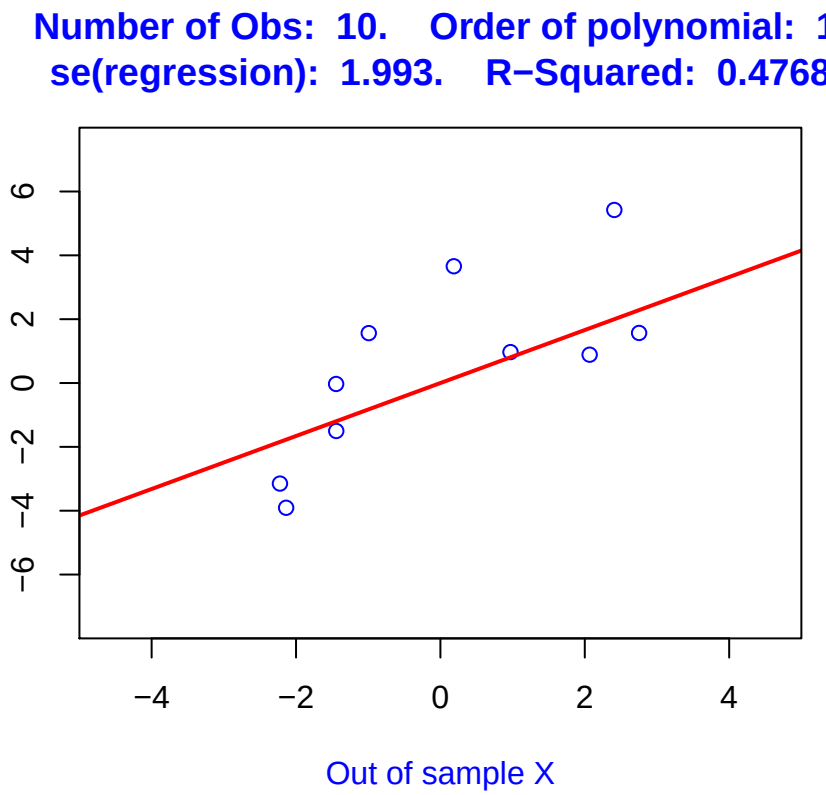


Two lessons:

1. Beware curve-fitting:

Significance tests may suggest adding polynomials,
but that isn't enough to justify their inclusion





2. Beware good fits in-sample unless they fit well out of sample, too

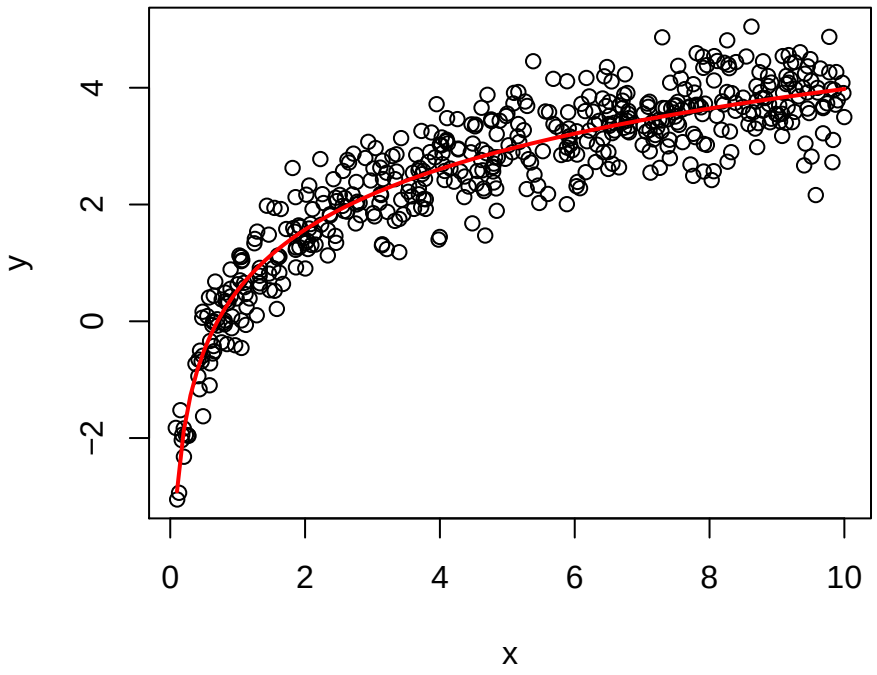
Every goodness of fit measure has an out-of-sample counterpart

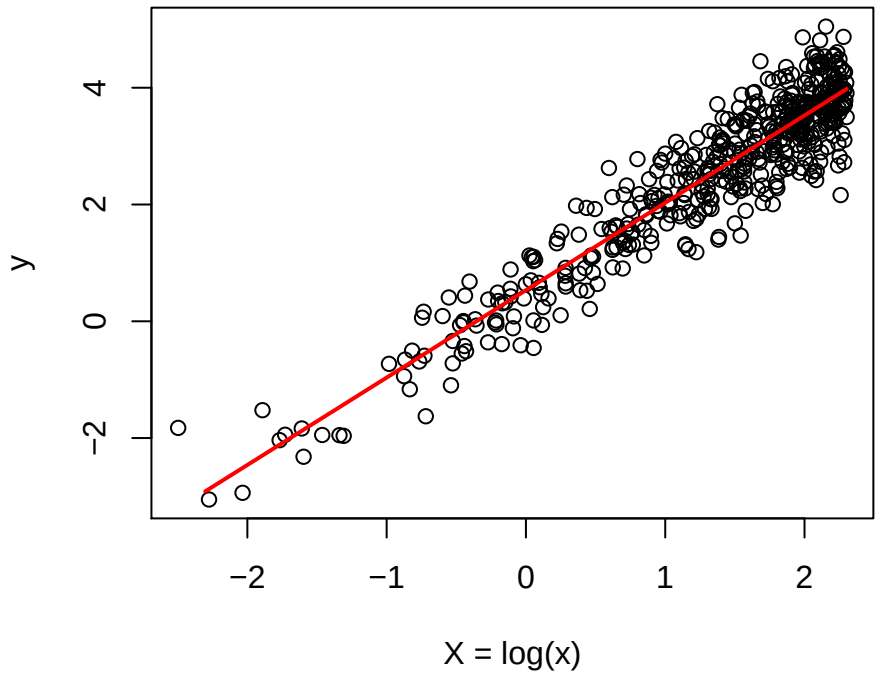
Out of sample goodness of fit is *much more important* than in sample

More Transformations of covariates

Suppose $X_1 = \log(x_1)$. Then,

$$Y = \beta_0 + \beta_1 \log(x_1) + \beta_2 X_2 + \ldots + \varepsilon$$





These are the same regression

Left is on the original, untransformed scale

Right is on the transformed, log scale

Log transformations for effects that diminish in per unit potency as x increases

Transformations of covariates

We could keep going, combining transformations and interactions to get very nonlinear models

Suppose $X_1 = x_1 \times x_2$ and $X_2 = x_2$. Then,

$$Y = \beta_0 + \beta_1 x_1 \times x_2 + \beta_2 x_2 + \ldots + \varepsilon$$

Transformations of covariates

We could keep going, combining transformations and interactions to get very nonlinear models

Suppose $X_1 = x_1 \times x_2$ and $X_2 = x_2$. Then,

$$Y = \beta_0 + \beta_1 x_1 \times x_2 + \beta_2 x_2 + \ldots + \varepsilon$$

Suppose $X_1 = x_1 \times \log(x_2) \times \sqrt{x_3}$ and $X_2 = x_2/(x_1 + x_2^2)$. Then,

$$Y = \beta_0 + \beta_1 (x_1 \times \log(x_2) \times \sqrt{x_3}) + \beta_2 x_2/(x_1 + x_2^2) + \ldots + \varepsilon$$

Without strong theoretical support,
these models will be silly and produce unreliable results with little predictive power

Transformations of the response variable

We could even replace Y

This is a good idea if you think Y is not a linear function of the regressors, but $g(y)$ *is* a linear function of them

Usually, this is the case for counts, e.g., or money.

Raising a government budget from \$1 million to \$10 million may be "just as hard" as raising it from \$10 to \$100 million

If X's affect the order of magnitude of Y, you should log Y

Transformations of the response variable

If $Y = \log(y)$, then,

$$\log(y) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \varepsilon$$

Now all X's have diminishing effect

In fact, level changes in X yield *percentage* changes in Y

If you log both X and Y, then % changes in X cause % changes in Y

Transformations of the response variable

What if Y is bounded but continuous?

Suppose it ranges between 0 and 1 (but doesn't include these values)?

Then we need to "stretch it out" to range from $-\infty$ to ∞

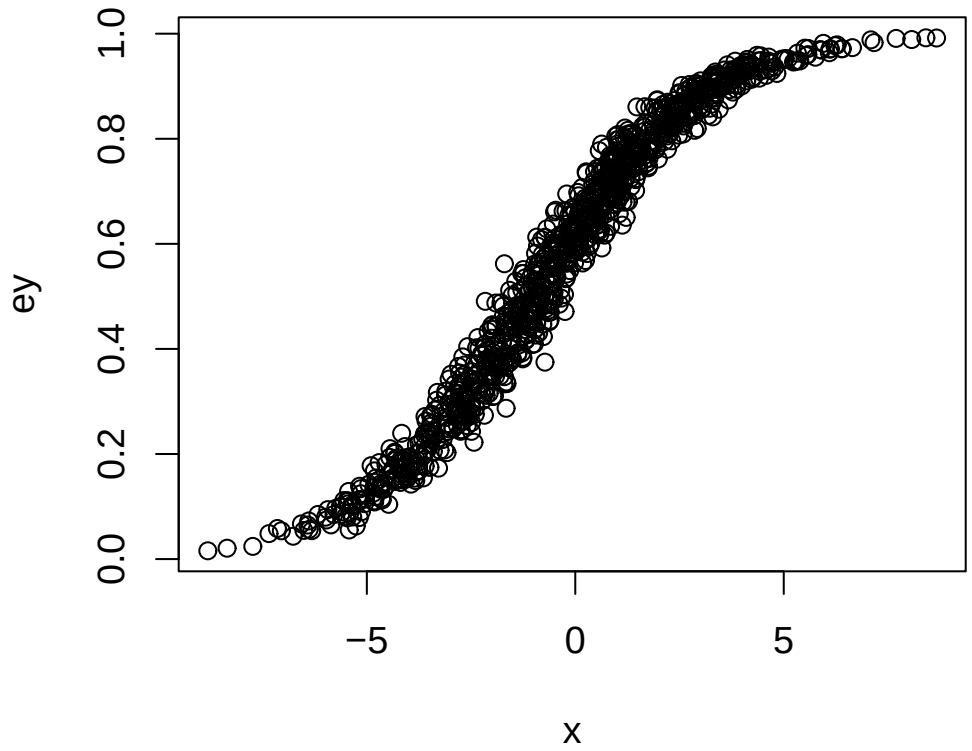
The logit transformation does this: $Y = \log(y/(1-y))$.

$$\log\left(\frac{y}{1-y}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \varepsilon$$

This model doesn't work if the original data include 0s or 1s

It is not "the logit model", which is a non-linear model of 0s and 1s only

It is a linear regression with a logit transformed response variable



Note the curve is S-shaped.

We could rescale it to work for any bounds, not just $(0, 1)$

That is, just transform y to $y^* = \frac{y-a}{b-a}$,
then run the regression of $\text{logit}(y^*)$ on your covariates

This works generally, with the caveat that for any bounds (a, b),
none of the data can be exactly a or b

Transformations of the response variable

All of the above models are examples of linear regression

They are all linear in the parameters

They can all be estimated by least squares

Transformations of the response variable

All of the above models are examples of linear regression

They are all linear in the parameters

They can all be estimated by least squares

What specifications *can't* we estimate?

We can't use LS to estimate models that are *non-linear* in the parameters

Example of a model that is non-linear in the parameters:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_1 \beta_2 X_3 + \ldots + \varepsilon$$

No amount of algebra can turn the above into a linear model

There are advanced methods to deal with this, e.g., non-linear least squares

Doesn't come up that often,
because so many specifications *are* linear in the parameters

Lots of flexibility hidden in linear regression

So just what is linear regression again?

We have expanded linear regression to encompass any model
for unbounded continuous functions $f(\cdot)$ and arbitrary functions $g_k(\cdot)$:

$$y_i = \beta_0 + \beta_1 x_{1i} + \ldots + \varepsilon_i$$

So just what is linear regression again?

We have expanded linear regression to encompass any model
for unbounded continuous functions $f(\cdot)$ and arbitrary functions $g_k(\cdot)$:

$$y_i = \beta_0 + \beta_1 x_{1i} + \ldots + \varepsilon_i$$

$$f(y_i) = \beta_0 + \beta_1 g_1(x_{1i}) + \ldots + \varepsilon_i$$

So just what is linear regression again?

We have expanded linear regression to encompass any model
for unbounded continuous functions $f(\cdot)$ and arbitrary functions $g_k(\cdot)$:

$$y_i = \beta_0 + \beta_1 x_{1i} + \ldots + \varepsilon_i$$

$$f(y_i) = \beta_0 + \beta_1 g_1(x_{1i}) + \ldots + \varepsilon_i$$

$$f(y_i) = \beta_0 + \sum_{k=1}^{K} \beta_k g_k(x_{ki}) + \ldots + \varepsilon_i$$

"Linearity" in linear regression just refers to the fact that
the effect of a one unit change in $g(x_k)$ on $f(y)$ is β_k.

But the relationship between x_k and y themselves could be very non-linear,
as a result of $f(\cdot)$ and $g_k(\cdot)$.

Interpreting models with transformed covariates or responses

Various methods:

- For logged response, $\hat{\beta}$ is the % change in Y for a level change in X
- For logged covariate, $\hat{\beta}$ is the level change in Y for a % changes in X
- For both sides logged, $\hat{\beta}$ is the % change in Y for % changes in X, known as the elasticity of Y with respect to X
- For polynomial coefficients, make a plot of $\hat{Y}$ as X varies: don't try to interpret each coefficient separately!
- For interaction like $X \times Z$, make one or more plots of $\hat{Y}$ under different combinations of X and Z: don't try to interpret the coefficients on X, Z, and $X \times Z$ separately!

Interpreting models with transformed covariates or responses

So how do we get $\hat{Y}$ for these models?

Could do it by hand fairly easily

But what if we want confidence intervals around $\hat{Y}$ too?

Use `predict()`.

Key is setting `newdata` input correctly

If `lm()` did the transformations and interactions

ie, you used the `I()` command in the model formula

and you used `*` or `:` to make the interactions

Then `predict()` will construct interactions and polynomials from their base terms as needed

If not, you need to give predict properly constructed interactions and polynomials

What we've learned about linear regression so far

Now we know how to:

1. Specify a regression model
2. Estimate that model
3. Interpret our findings

What we've learned about linear regression so far

Now we know how to:

1. Specify a regression model
2. Estimate that model
3. Interpret our findings

But how do we know if our findings are any good?

That we used the right specification?

That our model explained the data well or poorly?

Need to learn one more skill:

4. Select models with good fit

Running example: Life Expectancy

We will consider a simple linear regression today

The data are 138 countries observed in 1985

The response variable is average life expectancy in years

We consider four covariates:

gdpcap85	Per capita GDP in 1985, thousands of international dollars
school	Average years of education
civlib5	$\text{low} = 1$ to $\text{high} = 7$ scale of civil liberties
wartime	Percent of recent history spent at war

Data are from Barro & Lee

Example contrived for pedagogical use, not a serious effort at demography

Running example: Life Expectancy

```
lm(formula = lifeexp ~ gdpcap85 + school + civlib5 + wartime,
    na.action = "na.omit")

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 52.9671697  3.0379356  17.435   <2e-16 ***
gdpcap85     0.0003216  0.0002551   1.260    0.211
school       2.3866392  0.4102266   5.818    9e-08 ***
civlib5     -0.7522863  0.4861266  -1.548    0.125
wartime      1.0936902  5.5354286   0.198    0.844
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.43 on 90 degrees of freedom
Multiple R-Squared: 0.7509,     Adjusted R-squared: 0.7399
F-statistic: 67.84 on 4 and 90 DF,  p-value: < 2.2e-16
```

Running example: Life Expectancy

Note that initially, only education has a significant effect

Surprising?

Perhaps we have badly fit the model?

Two questions in model fitting

Question 1: Are the relationships I think I see really there?

Question 2: Do they explain much of what is happening?

Somewhat overlapping questions

Often helpful to tackle them together

How? There are many ways to answer these questions

You will need to use several to avoid being fooled

Are the relationships I think I see really there?

Look to:

- standard errors & associated tests
- patterns in the residuals
- warning signs for outliers

Standard errors and associated tests

You already know the t-test

But what if you want to test more than one parameter at a time?

F-test: joint tests of whether some or all the parameters of the model are 0

Recall the analysis of variance breakdown of what a regression "explains":

Total Sum of Squares	$\sum_{i=1}^{n} (y_i - \bar{y})^2$	$(\mathbf{y} - \bar{y})' (\mathbf{y} - \bar{y})$
Residual Sum of Squares	$\sum_{i=1}^{n} (y_i - \hat{y}_i)^2$	$\boldsymbol{\varepsilon}' \boldsymbol{\varepsilon}$
Regression Sum of Squares	$\sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2$	$(\mathbf{X}\boldsymbol{\beta} - \bar{y})' (\mathbf{X}\boldsymbol{\beta} - \bar{y})$

Recall that $R^2 =$ Regression Sum of Squares/Total Sum of Squares

or the proportion of variance explained by the model

Standard errors and associated tests

Suppose our "null" hypothesis is $\beta_1 = \beta_2 = 0$.

This hypothesis imposes restrictions on $q = 2$ parameters of a model with k parameters total estimated on n observations

The F-test associated with this null is:

$$
\begin{aligned}
F_0 &= \frac{\text{Regrssion Sum of Squares}/q}{\text{Residual Sum of Squares}/(n-k-1)} \\
&= \frac{n-k-1}{q} \times \frac{R^2_{\text{unrestricted}} - R^2_{\text{restricted}}}{1 - R^2_{\text{unrestricted}}}
\end{aligned}
$$

Under the null, F_0 is follows an F distribution with $q, n-k-1$ degrees of freedom

We can reject this null even when we can't be sure which of β_1 or β_2 is non-zero

We don't know which is explaining variance in Y, but we know that together, they explained enough variance to reject the null

Standard errors and associated tests

A special case of the F-test asks whether *every* β is zero simultaneously.

In other words, do any of our covariates explain anything?

Reported as part of the regression summary:

```
F-statistic:  67.84 on 4 and 90 DF, p-value:  < 2.2e-16
```

In this case, we could choose to reject the null that *every* coefficient is 0

This is the same as saying some parameter(s) is/are not zero (but which?)

Seldom interesting. Virtually always rejected

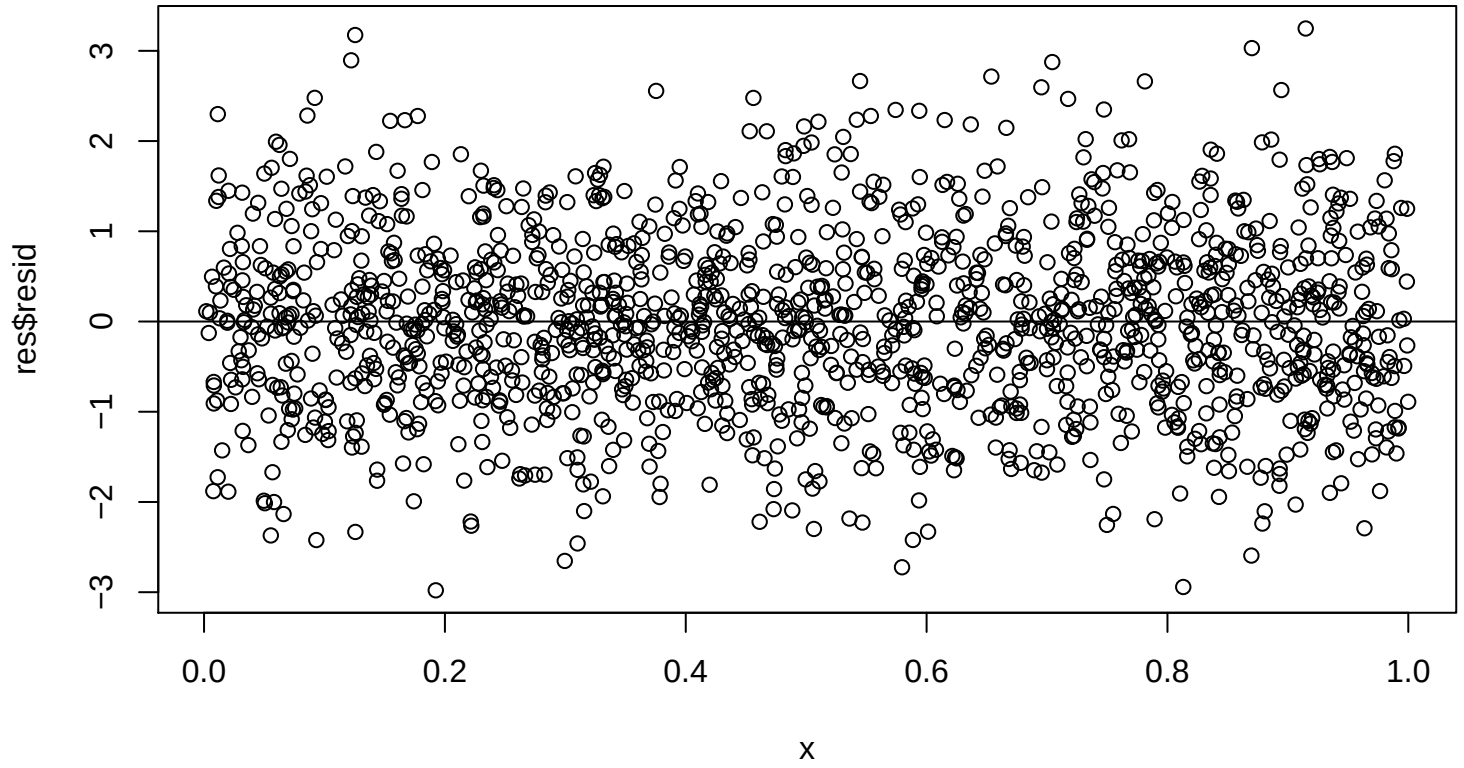
Useful in rare cases where we *want* a regression to find null results;
e.g., if there shouldn't be a relationship between y and a set of covariates

Examining residuals

We noted earlier that residuals from a least squares regression should have no pattern

In particular, the variance in ε_i should not be a function of Y or X

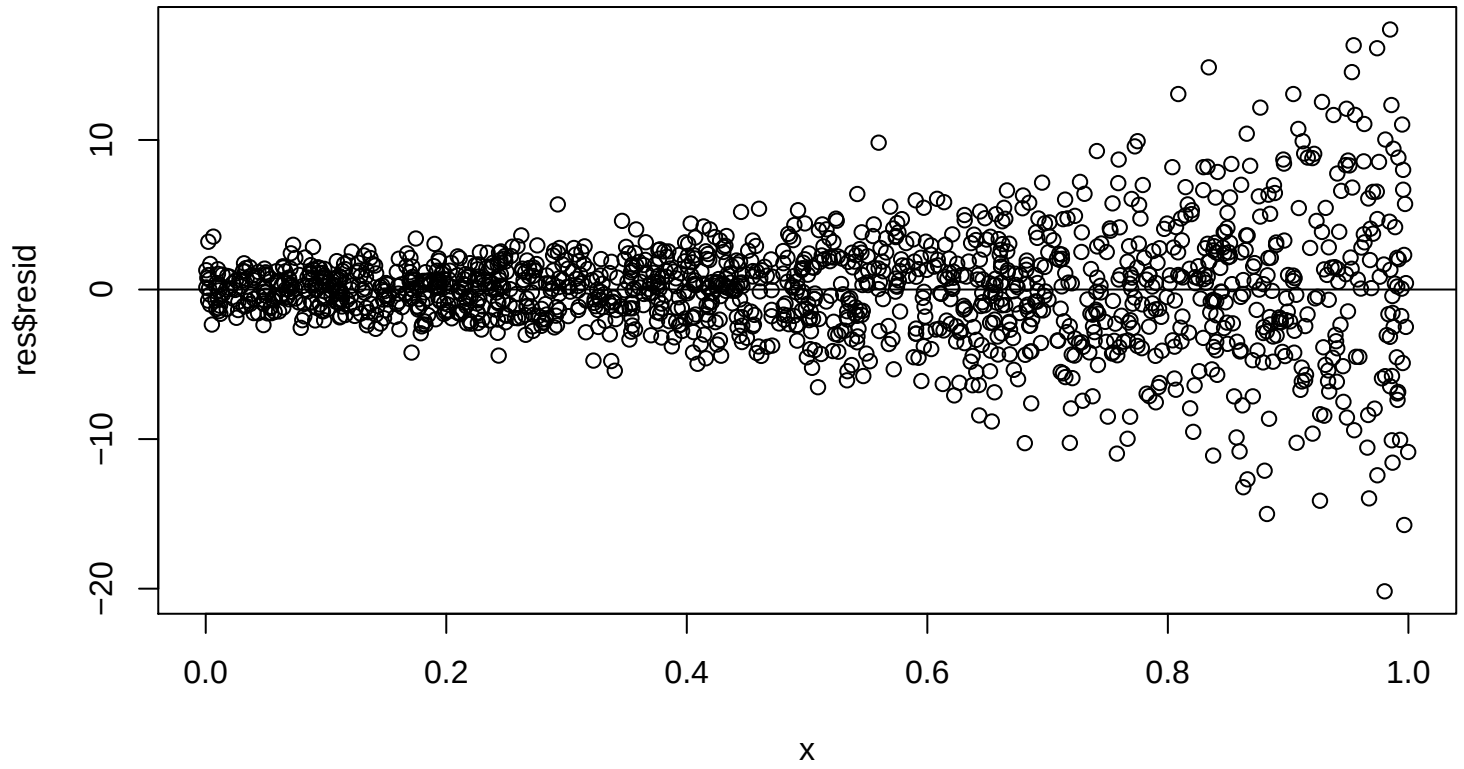
Examining residuals for heteroskedasticity



This is what we hope to see: white noise

No relationship between X (or Y) and the mean or variance of the residuals

Examining residuals for heteroskedasticity



Often, however, we find a megaphone shape, or some other change in variance

So why do we care? So what do we do about it?

Weighted least squares

Besides biasing SEs, heteroskedasticity makes LS inefficient. Why?

Weighted least squares

Besides biasing SEs, heteroskedasticity makes LS inefficient. Why?

Observations with higher variance in errors contain less information

Weighted least squares

Besides biasing SEs, heteroskedasticity makes LS inefficient. Why?

Observations with higher variance in errors contain less information

Observations with lower variance tend to be very close to the LS line

Weighted least squares

Besides biasing SEs, heteroskedasticity makes LS inefficient. Why?

Observations with higher variance in errors contain less information

Observations with lower variance tend to be very close to the LS line

We'll get better (more efficient) estimates if we give more *weight* to the latter

Weighted least squares

Besides biasing SEs, heteroskedasticity makes LS inefficient. Why?

Observations with higher variance in errors contain less information

Observations with lower variance tend to be very close to the LS line

We'll get better (more efficient) estimates if we give more *weight* to the latter

Needed:

- A measure of $\mathrm{Var}(\varepsilon_i) = \sigma_i^2$ for each observation

Weighted least squares

Besides biasing SEs, heteroskedasticity makes LS inefficient. Why?

Observations with higher variance in errors contain less information

Observations with lower variance tend to be very close to the LS line

We'll get better (more efficient) estimates if we give more *weight* to the latter

Needed:

- A measure of $\mathrm{Var}(\varepsilon_i) = \sigma_i^2$ for each observation
- A method for weighting observations in least squares estimation

Weighted least squares

Besides biasing SEs, heteroskedasticity makes LS inefficient. Why?

Observations with higher variance in errors contain less information

Observations with lower variance tend to be very close to the LS line

We'll get better (more efficient) estimates if we give more *weight* to the latter

Needed:

- A measure of $\mathrm{Var}(\varepsilon_i) = \sigma_i^2$ for each observation
- A method for weighting observations in least squares estimation
- In particular, we want to minimize the weighted sum of squares

$$\sum_{i=1}^{n} w_i \varepsilon_i^2 = \boldsymbol{\varepsilon}' \mathbf{W} \boldsymbol{\varepsilon}$$

where $\mathbf{W}$ is a diagonal matrix with weights w_i on the diagonal

Weighted least squares

The solution is simple: just add weight terms to the estimator

$$\hat{\boldsymbol{\beta}}_{\mathrm{WLS}} = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y}$$

where $\mathbf{W}$ is a diagonal matrix with weights w_i on the diagonal

Weighted least squares

The solution is simple: just add weight terms to the estimator

$$\hat{\boldsymbol{\beta}}_{\mathrm{WLS}} = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y}$$

where $\mathbf{W}$ is a diagonal matrix with weights w_i on the diagonal

What are the weights? They are (proportional to) the standard error for each y_i

Weighted least squares

The solution is simple: just add weight terms to the estimator

$$\hat{\boldsymbol{\beta}}_{\mathrm{WLS}} = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y}$$

where $\mathbf{W}$ is a diagonal matrix with weights w_i on the diagonal

What are the weights? They are (proportional to) the standard error for each y_i

Ideally, the weights are defined such that

$$\begin{aligned} \varepsilon_i &\sim \mathcal{N}(0, \sigma_i^2) \\ \sigma_i^2 &= 1/w_i^2 \end{aligned}$$

Weighted least squares

The solution is simple: just add weight terms to the estimator

$$\hat{\boldsymbol{\beta}}_{\mathrm{WLS}} = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y}$$

where $\mathbf{W}$ is a diagonal matrix with weights w_i on the diagonal

What are the weights? They are (proportional to) the standard error for each y_i

Ideally, the weights are defined such that

$$\begin{aligned} \varepsilon_i &\sim \mathcal{N}(0, \sigma_i^2) \\ \sigma_i^2 &= 1/w_i^2 \end{aligned}$$

$\rightarrow$ the larger the weight w_i, the smaller the variance of ε_i, the more information in i

In `R`, just add the argument `lm(...,weights)`

Weighted least squares

Hypothetical example: Modeling school spending

- Small districts have more variable budgets

  A single shock (a charitable gift; a new disabled student) can shift the mean

- Big districts have stable budgets

  Shocks average out over many schools

- In a model of district spending per pupil, small districts should get lower weight

- Signal to noise ratio higher in large districts

Adjusting standard errors for heteroskedasticity

Suppose we diagnose, or suspect, heteroskedasticity, but have no weights

We cannot use WLS, and thus rely on the less efficient LS estimates

Adjusting standard errors for heteroskedasticity

Suppose we diagnose, or suspect, heteroskedasticity, but have no weights

We cannot use WLS, and thus rely on the less efficient LS estimates

But we can try to get standard errors approaching the WLS se's

Recall the standard errors from LS are the square roots of the diagonal elements of

$$\hat{\mathrm{V}}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}^2(\mathbf{X}'\mathbf{X})^{-1}$$

So if σ^2 varies by i, these will be wrong.

Adjusting standard errors for heteroskedasticity

Suppose we diagnose, or suspect, heteroskedasticity, but have no weights

We cannot use WLS, and thus rely on the less efficient LS estimates

But we can try to get standard errors approaching the WLS se's

Recall the standard errors from LS are the square roots of the diagonal elements of

$$\hat{\mathrm{V}}(\hat{\boldsymbol{\beta}}) = \hat{\sigma}^2(\mathbf{X}'\mathbf{X})^{-1}$$

So if σ^2 varies by i, these will be wrong.

A "heteroskedasticity robust" formula for the Var-Cov matrix is:

$$\hat{\mathrm{V}}(\hat{\boldsymbol{\beta}}) = (\mathbf{X}'\mathbf{X})^{-1}(\sum_i \hat{\varepsilon}_i x_i' x_i)(\mathbf{X}'\mathbf{X})^{-1}$$

Adjusting standard errors for heteroskedasticity

$$\hat{\mathrm{V}}(\hat{\boldsymbol{\beta}}) = (\mathbf{X}'\mathbf{X})^{-1}(\sum_i \hat{\varepsilon}_i x_i' x_i)(\mathbf{X}'\mathbf{X})^{-1}$$

SE's calculated from this equation are known by many names:

- White standard errors
- robust standard errors
- sandwich standard errors
- heteroskedastcity consistent standard errors

Always a second best approach.

But very popular (used in most poli sci papers that use LS)

Adjusting standard errors for heteroskedasticity

To get robust standard errors in R:

```
res <- lm(lifeexp~gdpcap85+school+civlib5+wartime,
          na.action="na.omit")

vc <- hccm(res)                    # library car must be loaded

se.corrected <- sqrt(diag(vc))
    # se's are the square roots of diagonal elements of
    # the variance-covariance matrix of the parameters


# Note:  you'll need to calculate your own t-tests & CIs
```

If the robust SEs are very similar to the usual SEs, don't bother reporting them

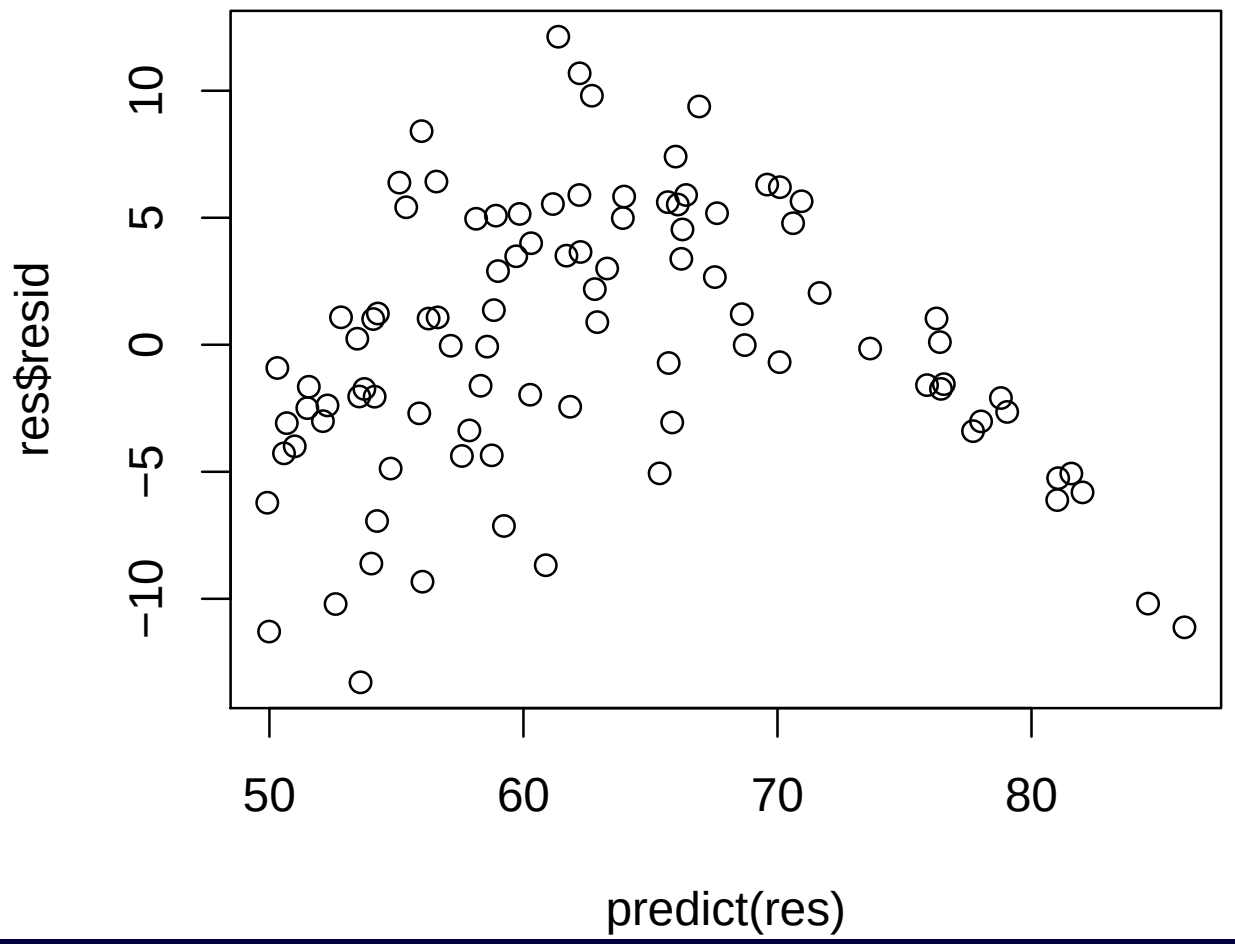
(Catch-22: If robust se's are very different, LS is inefficient anyway,
and you should look for weights or ways to model the heteroskedasticity)

The life expectancy example

Is heteroskedasticity a problem in our example?

Let's look at the residuals

Life expectancy residuals against fitted Y



Whoa! That's not (just) heteroskedasticity!

The residuals appear correlated with the fit itself. (Why is that bad?)

Other patterns in the residuals

Non-constant variance isn't the only pattern we could see in residuals

What if the residuals don't appear to have mean 0 for all levels of X?

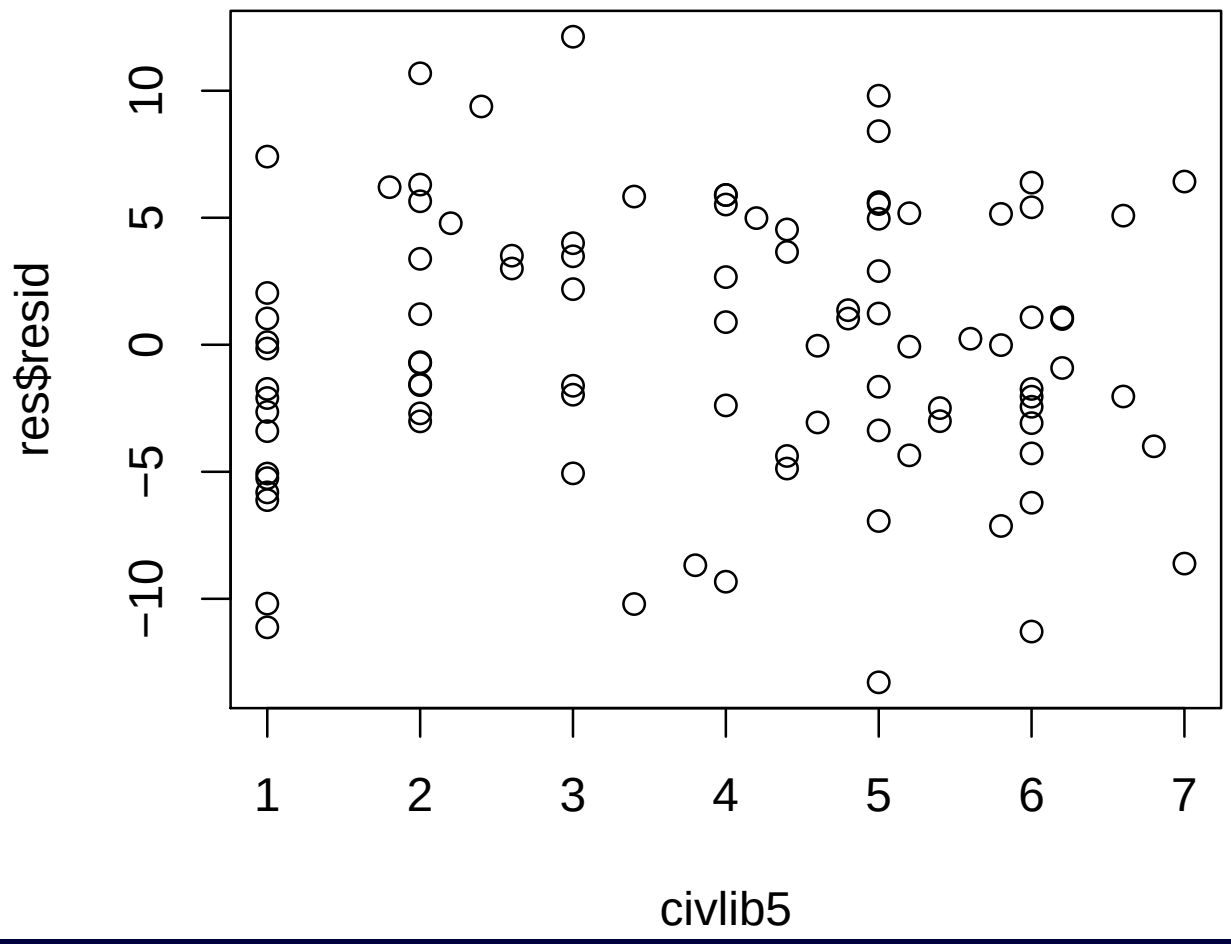
E.g., what if they are rising in X, or show curvature?

This is evidence of *misspecification*

In particular, curvature in the residuals indicates missing transformations of X

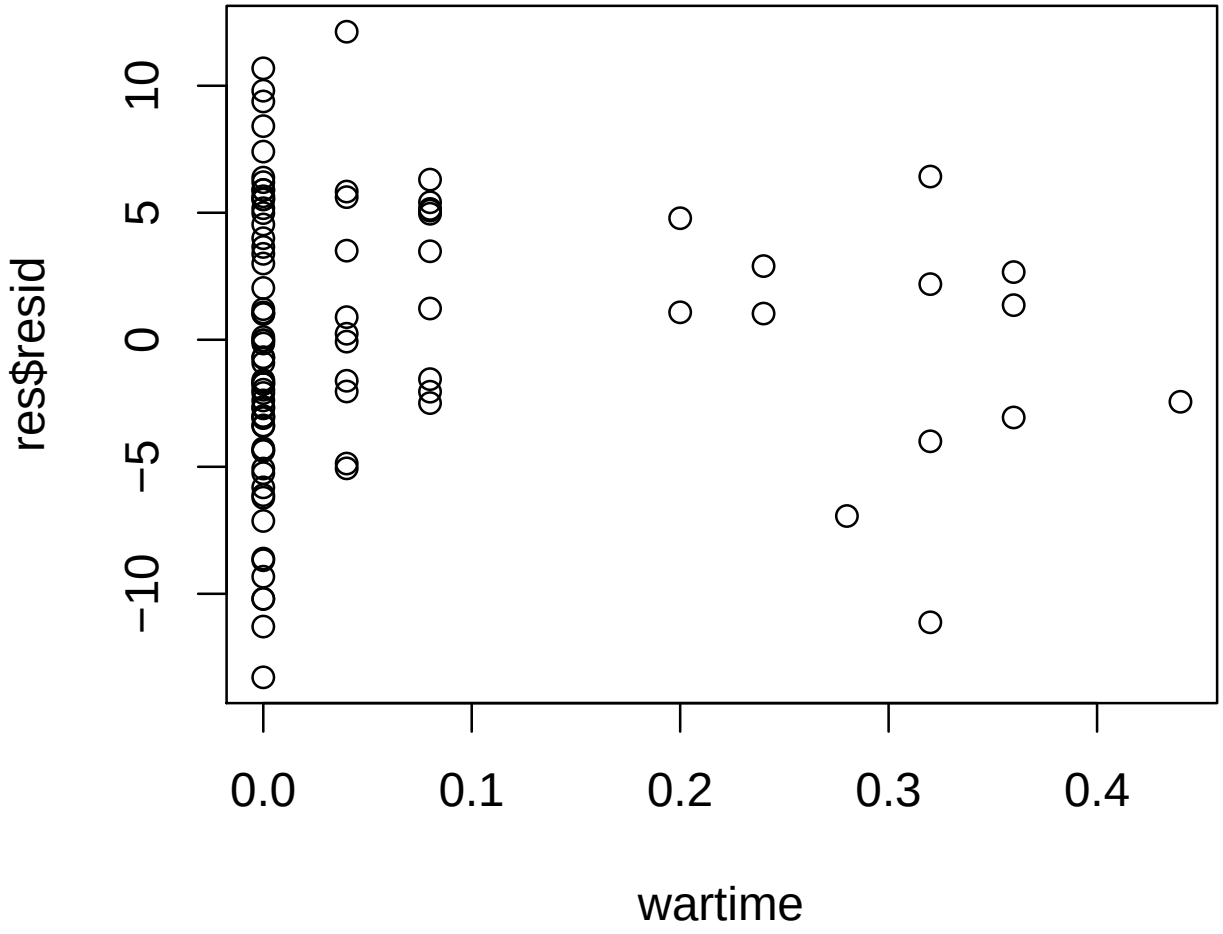
Let's plot our life expectancy residuals against each covariate

Life expectancy residuals against Civil Liberties



Not much sign of a pattern in the mean or variance here.

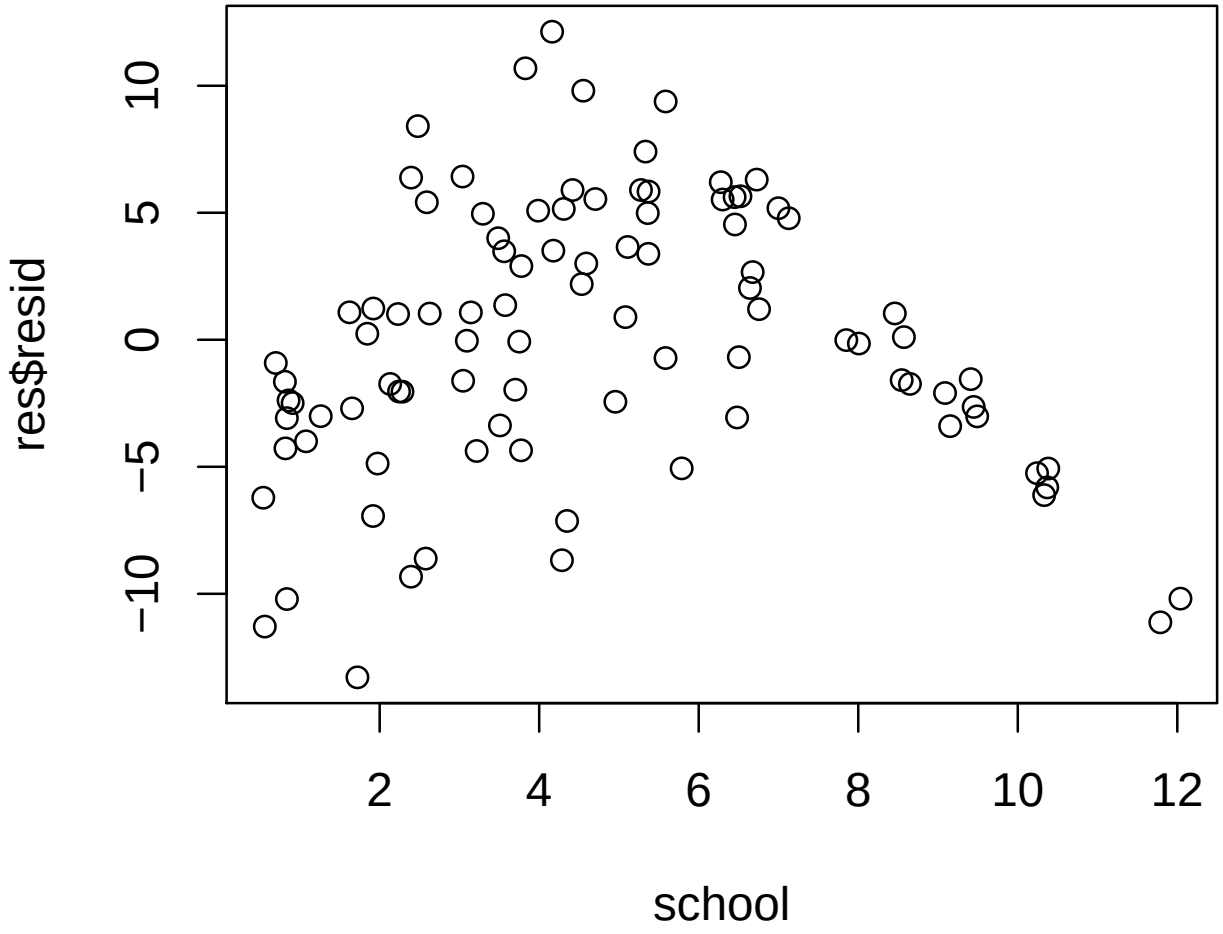
Life expectancy residuals against Wartime



This looks like heteroskedasticity, but is it?

Perhaps there just are too few high war cases to see the full spread

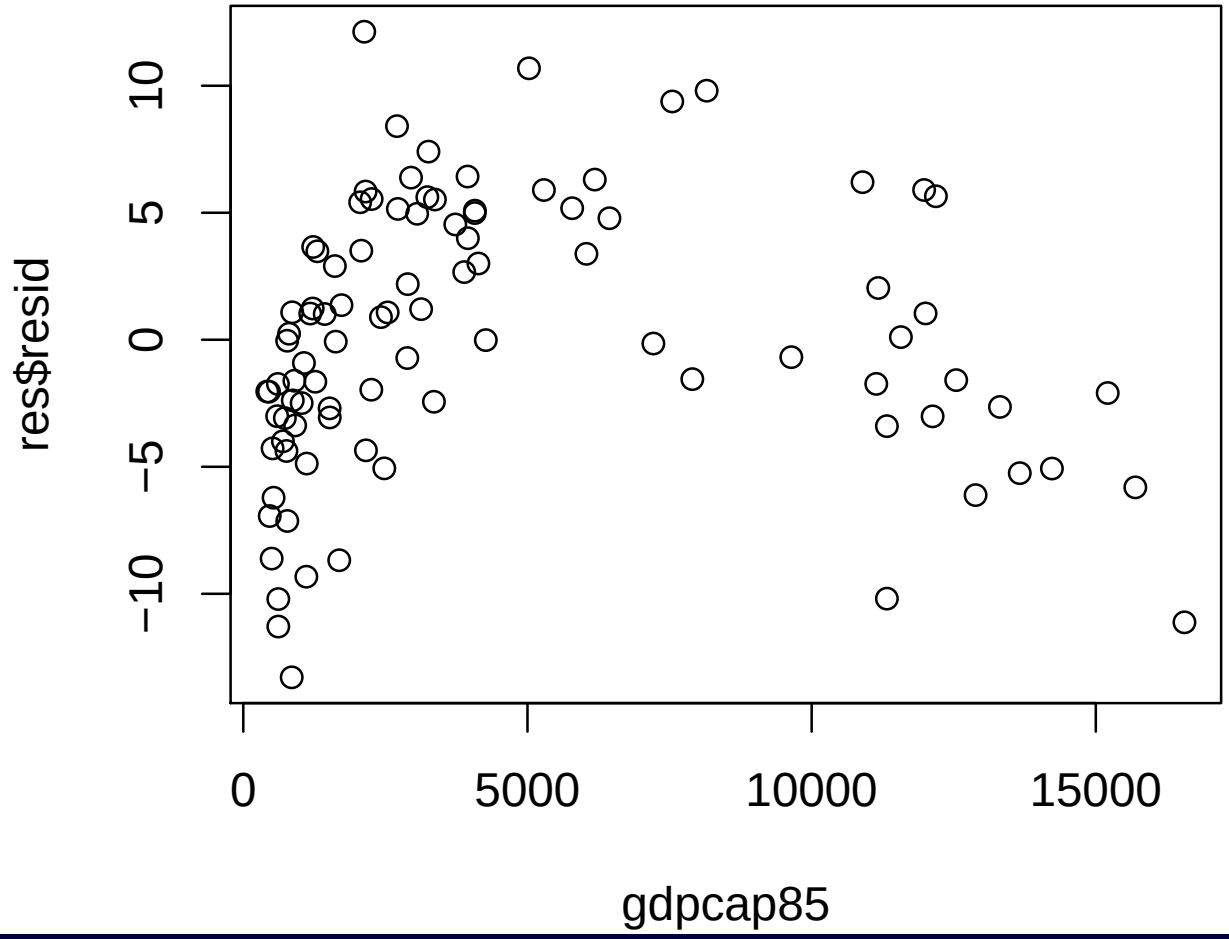
Life expectancy residuals against School



Clearly something amiss. What does this plot show?

How might we fix it?

Life expectancy residuals against GDP



Remember that GDP wasn't even significant.

But there *does* seem to be a relation between GDP and the residual

So why didn't this show up in our model? What did we miss?

Other patterns in the residuals

A pattern in the mean of the residuals is evidence of specification bias

How do we eliminate the bias?

Include transformations of the regressor!

E.g., if you see a curve, try adding X^2 or $\log X$

Let's respecify the Life expectancy regression to include logs of GDP and school

Respecified model

```
lm(formula = lifeexp ~ I(log(gdpcap85)) + I(log(school)) + civlib5 +
    wartime, na.action = "na.omit")

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      13.1893     5.4439   2.423   0.0174 *
I(log(gdpcap85))  5.3730     0.6867   7.825 9.35e-12 ***
I(log(school))    6.0431     0.9040   6.685 1.88e-09 ***
civlib5          -0.1843     0.3084  -0.598   0.5517
wartime          -0.2174     3.6894  -0.059   0.9531
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.627 on 90 degrees of freedom
Multiple R-Squared: 0.8889,     Adjusted R-squared: 0.8839
F-statistic:   180 on 4 and 90 DF,  p-value: < 2.2e-16
```

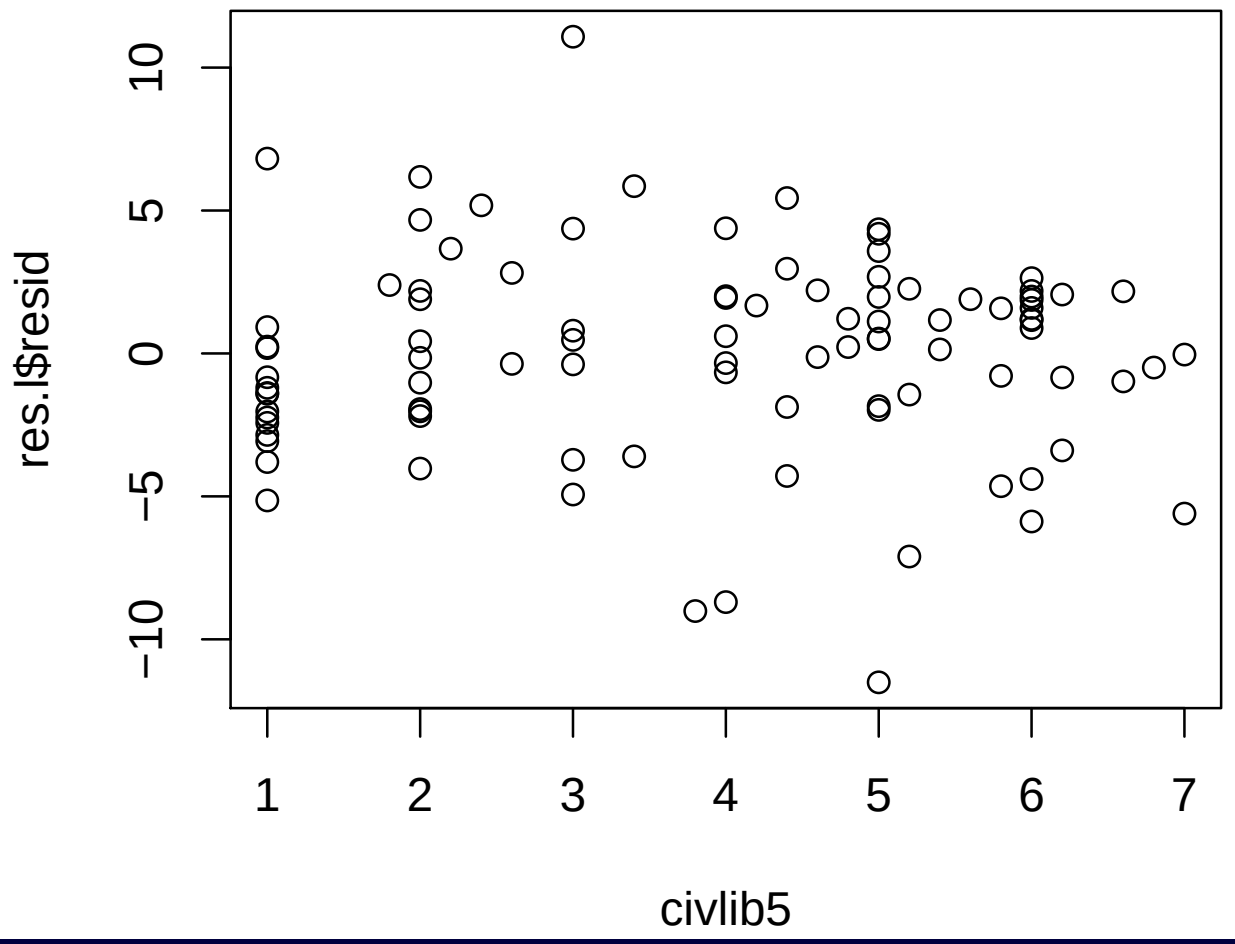
Respecified model: residuals

The se's indicate a much better fit regarding GDP and schools

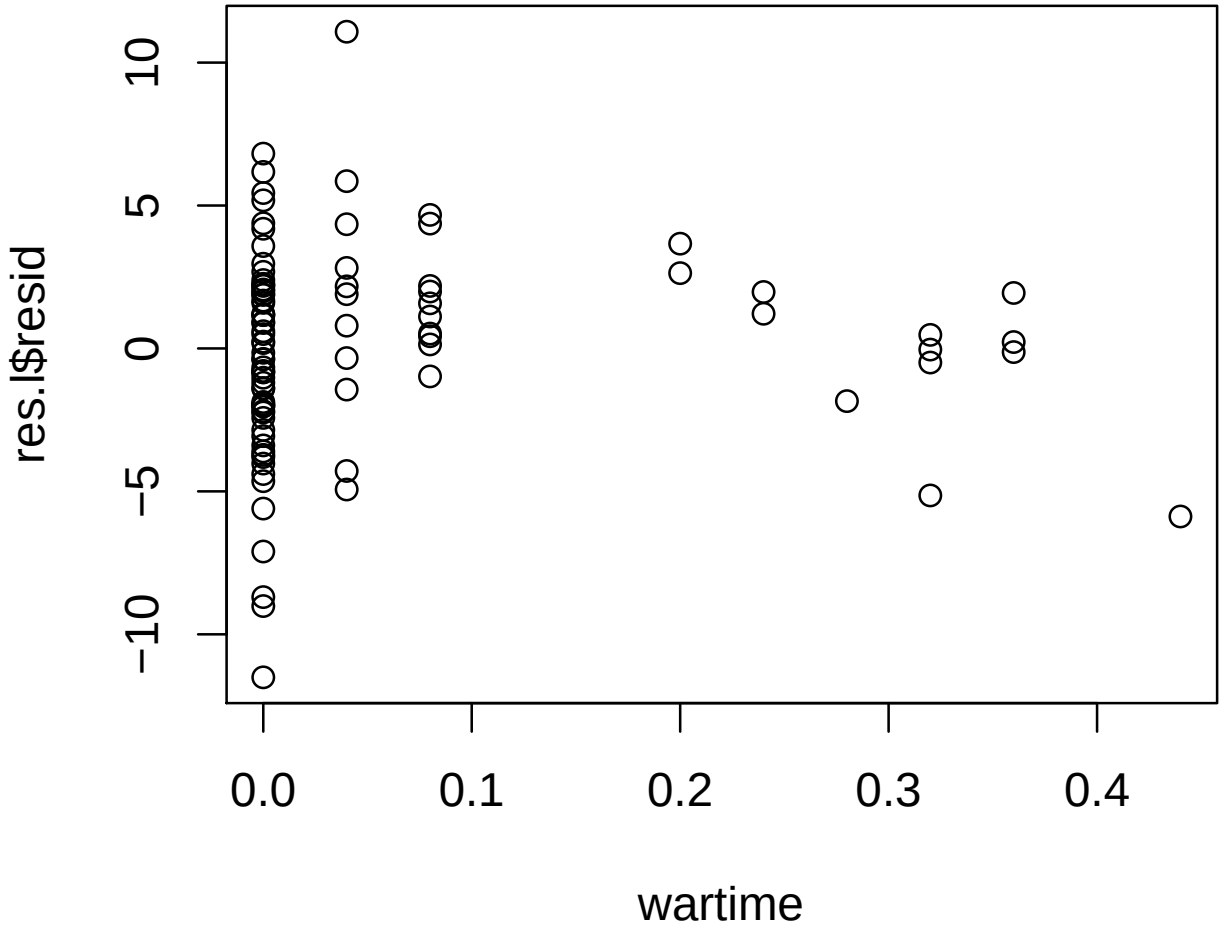
But was this the "right" transformation?

Let's run through the residual diagnostics again,
to see if the new specification eliminates the patterns we found

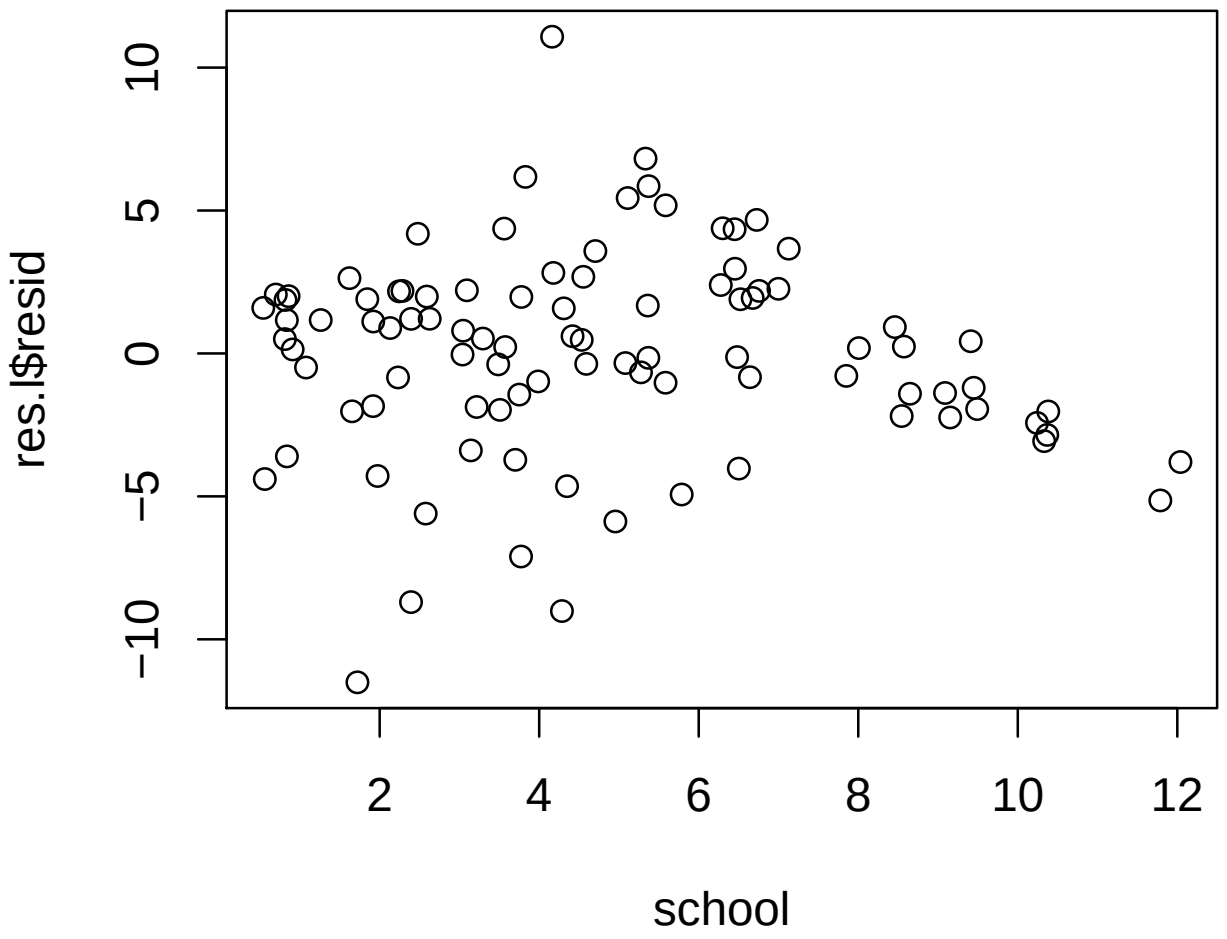
Life expectancy residuals against Civil Liberties



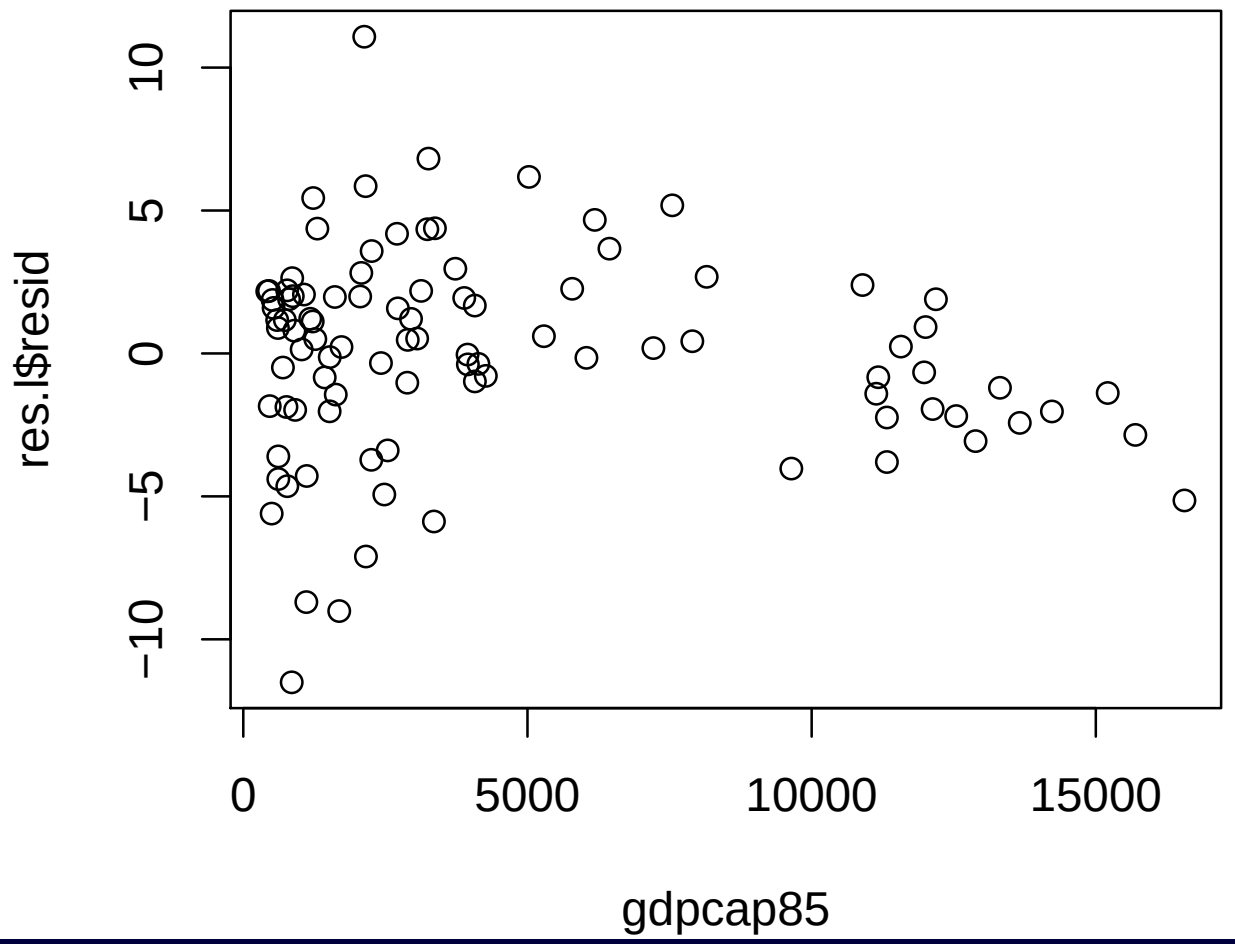
Life expectancy residuals against Wartime



Life expectancy residuals against School



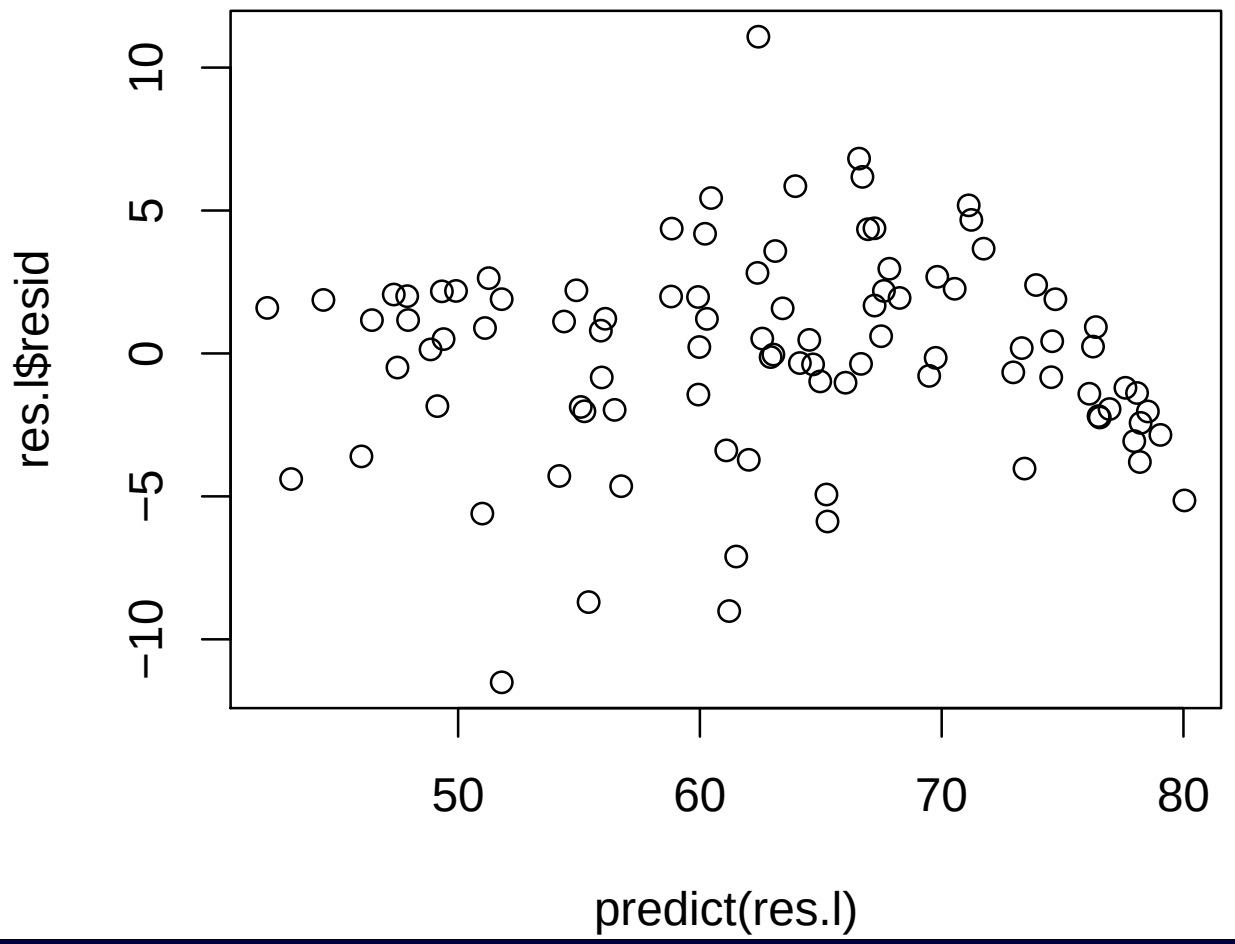
Life expectancy residuals against GDP



Much better overall, though perhaps a bit of underfitting at the high end

Let's look at residuals against $\hat{Y}$

Life expectancy residuals against fitted Y



A pattern still shows up at the highest fitted values

Possibly a result of limits to medical technology

Further testing of transformations of X, and perhaps Y, might help

Is heteroskedasticity messing up the standard errors?

There may still be some (mild) heteroskedasticity here

We can check the robust SEs:

```
vc <- hccm(res.l)
se.corrected <- sqrt(diag(vc))
```

	Est	SE	Robust SE
Intercept	13.19	5.44	5.38
log(gdpcap85)	5.37	0.69	0.67
log(school)	6.04	0.90	0.84
civlib5	−0.18	0.31	0.29
wartime	−0.22	3.69	3.78

In this case, it makes little difference which SEs we use

Caveat: robust SEs are not perfect fixes, and may be wrong in small samples

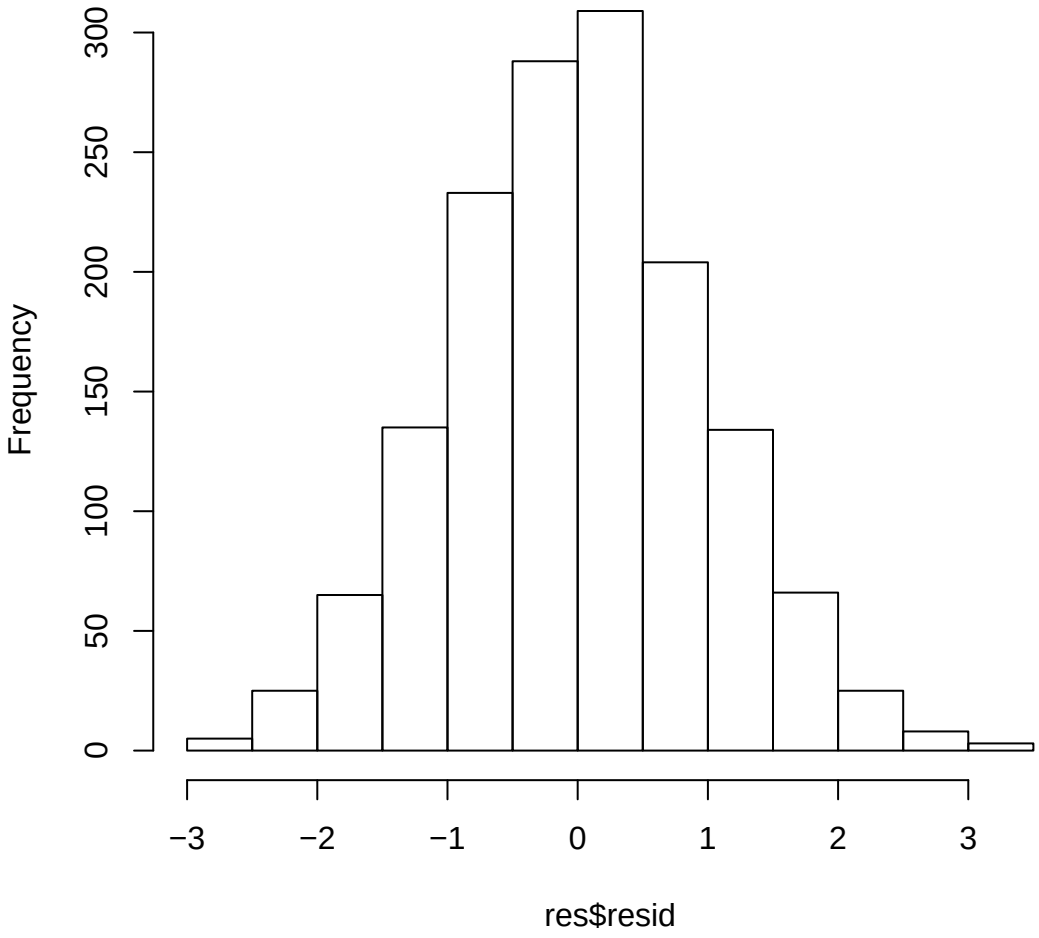
What about transforming the response?

How do you know when to transform Y, instead of X?

Theory: if you think all covariates should have diminishing effects as Y rises

Evidence: Histogram of residuals shows non-Normality amenable to a transformative "fix"

Other patterns in the residuals



Histograms of the residuals are also useful

The histogram should look approximately Normal

If it appears skewed, e.g., to the right, try logging Y

Does my model explain much of what is happening?

The "coefficient of determination", R^2

The standard error of the regression, σ^2

Out of sample tests

Cross-validation

Much ado about R^2

Many regression tables in journals report R^2

R^2 is the proportion of variance in Y explained by the regression model

$$R^2 = \frac{\text{Explained sum of squares}}{\text{Total sum of squares}} = 1 - \frac{\text{Residual sum of squares}}{\text{Total sum of squares}}$$

$$= \frac{\sum \hat{y}^2}{\sum y^2} = 1 - \frac{\sum \hat{\varepsilon}^2}{\sum y^2}$$

R^2 is bounded by 0 and 1

$R^2 = 0$: model has no explanatory power

$R^2 = 1$: model perfectly predicts every obs without error

Conventional wisdom: higher R^2 is "better"

King ("How not to lie with statistics") on what R^2 does & doesn't say

- R^2 shows the *proportion of variance explained* by the covariates, compared to a model with no covariates

King ("How not to lie with statistics") on what R^2 does & doesn't say

- R^2 shows the *proportion of variance explained* by the covariates, compared to a model with no covariates

- R^2 indirectly reports the scatter around the regression line
  ($\hat{\sigma}^2$ directly reports the amount of scatter)

King ("How not to lie with statistics") on what R^2 does & doesn't say

- R^2 shows the *proportion of variance explained* by the covariates, compared to a model with no covariates

- R^2 indirectly reports the scatter around the regression line
  ($\hat{\sigma}^2$ directly reports the amount of scatter)

- R^2s comparable only for models with same observations and response variable
  (In the life expectancy example, it's good when adding substantively interesting variables raises R^2 noticably)

King ("How not to lie with statistics") on what R^2 does & doesn't say

- R^2 shows the *proportion of variance explained* by the covariates, compared to a model with no covariates

- R^2 indirectly reports the scatter around the regression line
  ($\hat{\sigma}^2$ directly reports the amount of scatter)

- R^2s comparable only for models with same observations and response variable
  (In the life expectancy example, it's good when adding substantively interesting variables raises R^2 noticably)

- Maximizing R^2 is usually perverse. More covariates always raise R^2.

King ("How not to lie with statistics") on what R^2 does & doesn't say

- R^2 shows the *proportion of variance explained* by the covariates, compared to a model with no covariates

- R^2 indirectly reports the scatter around the regression line
  ($\hat{\sigma}^2$ directly reports the amount of scatter)

- R^2s comparable only for models with same observations and response variable
  (In the life expectancy example, it's good when adding substantively interesting variables raises R^2 noticably)

- Maximizing R^2 is usually perverse. More covariates always raise R^2.

- The most useful model is seldom the one with the highest R^2.
  (Uninteresting high R^2 models:
  Y regressed on itself. Vote choice regressed on vote intention.)

King ("How not to lie with statistics") on what R^2 does & doesn't say

- R^2 shows the *proportion of variance explained* by the covariates, compared to a model with no covariates

- R^2 indirectly reports the scatter around the regression line
  ($\hat{\sigma}^2$ directly reports the amount of scatter)

- R^2s comparable only for models with same observations and response variable
  (In the life expectancy example, it's good when adding substantively interesting variables raises R^2 noticably)

- Maximizing R^2 is usually perverse. More covariates always raise R^2.

- The most useful model is seldom the one with the highest R^2.
  (Uninteresting high R^2 models:
  Y regressed on itself. Vote choice regressed on vote intention.)

- R^2 is not an "estimate". (It can't be significant or non-significant.)

King ("How not to lie with statistics") on what R^2 does & doesn't say

- R^2 shows the *proportion of variance explained* by the covariates, compared to a model with no covariates
- R^2 indirectly reports the scatter around the regression line
  ($\hat{\sigma}^2$ directly reports the amount of scatter)
- R^2s comparable only for models with same observations and response variable
  (In the life expectancy example, it's good when adding substantively interesting variables raises R^2 noticably)
- Maximizing R^2 is usually perverse. More covariates always raise R^2.
- The most useful model is seldom the one with the highest R^2.
  (Uninteresting high R^2 models:
  Y regressed on itself. Vote choice regressed on vote intention.)
- R^2 is not an "estimate". (It can't be significant or non-significant.)
- R^2 seldom of substantive interest, unlike $\hat{\sigma}^2$, or the se of $\hat{\beta}$.

The standard error of the regression

$\hat{\sigma}$ is at least as useful to report as R^2

It tells us how much the fitted values, $\hat{y}$, miss the true y on average

Thus it is on the scale of y. Easy to work into a substantive conclusion

How is this a measure of goodness of fit?

As σ goes down, we make better predictions

$\hat{y}$ gets closer to the true y

Fitting the life expectancy model

Covariates	Goodness of Fit R^2	ser ($\hat{\sigma}$)
GDP, Sch, CL, War	0.75	5.43
log(GDP), log(Sch), CL, War	0.88	3.63

R^2 is a useful statistic here (same data & response)

What does it mean?

What does the $\hat{\sigma}$ mean?

Is the second model a better fit?

More persuasive tests of fit

The most persuasive evidence of model fit is successful prediction

Suppose we estimate a model of presidential approval ratings, using data over 1950—1995.

More persuasive tests of fit

The most persuasive evidence of model fit is successful prediction

Suppose we estimate a model of presidential approval ratings, using data over 1950—1995.

We could ask: How well does the model fit the 1950—1995 data?

Could also ask: How well does the model *predict* 1996—2005 data?

Out-of-sample tests are powerful checks against spurious findings

If we intend the model to apply to the out of sample data, this is a better test

More persuasive tests of fit

The most persuasive evidence of model fit is successful prediction

Suppose we estimate a model of presidential approval ratings, using data over 1950—1995.

We could ask: How well does the model fit the 1950—1995 data?

Could also ask: How well does the model *predict* 1996—2005 data?

Out-of-sample tests are powerful checks against spurious findings

If we intend the model to apply to the out of sample data, this is a better test

The usual caveat applies: the best fitting model is not necessarily the most interesting

But if our model fits as well to the out of sample data as the in sample, much greater confidence we have found something real

(Both samples could have outliers, heteroskedasticity, or specification error, so we still have to check for these)

Out of sample goodness of fit

How to do an out-of-sample test:

1. Fit the model on the training sample, $\{\mathbf{X}_{\text{training}}, \mathbf{y}_{\text{training}}\}$, obtain $\hat{\boldsymbol{\beta}}_{\text{training}}$

Out of sample goodness of fit

How to do an out-of-sample test:

1. Fit the model on the training sample, $\{\mathbf{X}_{\text{training}}, \mathbf{y}_{\text{training}}\}$, obtain $\hat{\boldsymbol{\beta}}_{\text{training}}$

2. Calculate the fitted $\hat{\mathbf{y}}_{\text{test}}$ for the test sample, $\{\mathbf{X}_{\text{test}}, \mathbf{y}_{\text{test}}\}$

Out of sample goodness of fit

How to do an out-of-sample test:

1. Fit the model on the training sample, $\{\mathbf{X}_{\text{training}}, \mathbf{y}_{\text{training}}\}$, obtain $\hat{\boldsymbol{\beta}}_{\text{training}}$
2. Calculate the fitted $\hat{\mathbf{y}}_{\text{test}}$ for the test sample, $\{\mathbf{X}_{\text{test}}, \mathbf{y}_{\text{test}}\}$
3. Compare fit for training sample to our ability to predict test sample.

Out of sample goodness of fit

How to do an out-of-sample test:

1. Fit the model on the training sample, $\{\mathbf{X}_{\text{training}}, \mathbf{y}_{\text{training}}\}$, obtain $\hat{\boldsymbol{\beta}}_{\text{training}}$

2. Calculate the fitted $\hat{\mathbf{y}}_{\text{test}}$ for the test sample, $\{\mathbf{X}_{\text{test}}, \mathbf{y}_{\text{test}}\}$

3. Compare fit for training sample to our ability to predict test sample.

   For example, compare $\hat{\sigma}_{\text{training}}$, the standard error of the residuals from the training regression, to the standard error of the "residuals" from the test predictions:

$$\text{std dev}(\mathbf{y}_{\text{test}} - \mathbf{X}_{\text{test}}\hat{\boldsymbol{\beta}}_{\text{training}})$$

   (This is the average prediction error)

   A very good predictive model will have about the same error out of sample as in sample. (Rare to find)

Cross-validation

What if we're short on data, and can't afford to sequester half of it for an out of sample GoF test?

Cross-validation

What if we're short on data, and can't afford to sequester half of it for an out of sample GoF test? Not a problem: just use *cross-validation*

1. Select all but $1/k$th of the data, $\{\mathbf{y}_{\text{training}}, \mathbf{X}_{\text{training}}\}$.
   Leave-one-out cross validation ($k = n$) is especially good.

Cross-validation

What if we're short on data, and can't afford to sequester half of it for an out of sample GoF test? Not a problem: just use *cross-validation*

1. Select all but $1/k$th of the data, $\{\mathbf{y}_{\text{training}}, \mathbf{X}_{\text{training}}\}$.
   Leave-one-out cross validation ($k = n$) is especially good.

2. Regress $\mathbf{y}_{\text{training}}$ on $\mathbf{X}_{\text{training}}$ to obtain β_{training}

Cross-validation

What if we're short on data, and can't afford to sequester half of it for an out of sample GoF test? Not a problem: just use *cross-validation*

1. Select all but $1/k$th of the data, $\{\mathbf{y}_{\text{training}}, \mathbf{X}_{\text{training}}\}$.
   Leave-one-out cross validation ($k = n$) is especially good.

2. Regress $\mathbf{y}_{\text{training}}$ on $\mathbf{X}_{\text{training}}$ to obtain β_{training}

3. Use $\hat{\beta}_{\text{training}}$ and $\mathbf{X}_{\text{test}}$ to form predictions $\hat{\mathbf{y}}_{\text{test}}$

Cross-validation

What if we're short on data, and can't afford to sequester half of it for an out of sample GoF test? Not a problem: just use *cross-validation*

1. Select all but $1/k$th of the data, $\{\mathbf{y}_{\text{training}}, \mathbf{X}_{\text{training}}\}$.
   Leave-one-out cross validation ($k = n$) is especially good.

2. Regress $\mathbf{y}_{\text{training}}$ on $\mathbf{X}_{\text{training}}$ to obtain β_{training}

3. Use $\hat{\boldsymbol{\beta}}_{\text{training}}$ and $\mathbf{X}_{\text{test}}$ to form predictions $\hat{\mathbf{y}}_{\text{test}}$

4. Compare $\hat{\mathbf{y}}_{\text{test}}$ with the true $\mathbf{y}_{\text{test}}$, by calculating the average prediction error:

$$\text{std dev}(\mathbf{y}_{\text{test}} - \mathbf{X}_{\text{test}}\hat{\boldsymbol{\beta}}_{\text{training}})$$

Cross-validation

What if we're short on data, and can't afford to sequester half of it for an out of sample GoF test? Not a problem: just use *cross-validation*

1. Select all but $1/k$th of the data, $\{\mathbf{y}_{\text{training}}, \mathbf{X}_{\text{training}}\}$.
   Leave-one-out cross validation ($k = n$) is especially good.

2. Regress $\mathbf{y}_{\text{training}}$ on $\mathbf{X}_{\text{training}}$ to obtain β_{training}

3. Use $\hat{\boldsymbol{\beta}}_{\text{training}}$ and $\mathbf{X}_{\text{test}}$ to form predictions $\hat{\mathbf{y}}_{\text{test}}$

4. Compare $\hat{\mathbf{y}}_{\text{test}}$ with the true $\mathbf{y}_{\text{test}}$, by calculating the average prediction error:

$$\text{std dev}(\mathbf{y}_{\text{test}} - \mathbf{X}_{\text{test}}\hat{\boldsymbol{\beta}}_{\text{training}})$$

5. Repeat steps 1–4 k times, and average the squared average prediction errors. The square root is the k-fold cross-validation predictive error.

The best predictive model, according to cross-validation, will minimize this average predictive error.

Cross-validation

R can do cross-validation for you:

```
library(boot)
data <- data.frame(lifeexp,gdpcap85,school,civlib5,wartime,
                   row.names=country)
data <- na.omit(data)

res.glm1 <- glm(lifeexp~gdpcap85+school+civlib5+wartime,
                na.action="na.omit")
              # Note the use of glm rather than lm

cv.err <- cv.glm(data,res.glm1,k=5)
              # This runs the cross-validation

cv.err$delta
              # Report the CV prediction error

# And repeat for each model
```

Cross-validation

Covariates	R^2	Goodness of Fit ser ($\hat{\sigma}$)	k-fold CV error
GDP, Sch, CL, War	0.75	5.43	32.3
log(GDP), log(Sch), CL, War	0.88	3.63	13.8

The CV prediction error is much higher than $\hat{\sigma}$

Not a surprise

When we try to predict "new" data from a model,
we usually find it works less well

Getting the specification "right" makes a *huge* difference for correct prediction

Missing life expectancy by 32 years is horrific.
Missing by 14 is just okay.

Thinking we would only miss by 3.63 was overconfident.

Cross-validation

Covariates	R^2	Goodness of Fit ser ($\hat{\sigma}$)	k-fold CV error
GDP, Sch, CL, War	0.75	5.43	32.3
log(GDP), log(Sch), CL, War	0.88	3.63	13.8
log(GDP), log(Sch), log(CL), War, War^2	0.90	3.48	12.1

Could we do better? Try to eliminate remaining curvature in the residuals?

Try a model with curves in each covariate

(Why did I use the square of War?)

A slight improvement. Shows up in CV, so it might even be real.

But getting GDP and School right was crucial