Gov 50: 12. Linear Regression (II)

Matthew Blackwell

Harvard University

Fall 2018

1. Today's agenda

2. Model fit

3. Multiple predictors

1/ Today's agenda

Logistics

- Midterm evaluation results.
- My office hours rescheduled to tomorrow 10:30am-12:00pm.
- DataCamp 4 due tonight.
- HW 3 due next Thursday
  - Extra credit worth a good amount of post-curve grade (3%)

Where are we? Where are going?

- Trying to get good predictions of some variable.
- Last time: how to use **linear regression** to predict outcomes using another variable.
- Now: assess model fit and use more than 1 variable to predict.

Why do we care about prediction?

- Prediction is broadly across different fields.
- Policy:
  - Can policymakers predict where crime is likely occur in a city to deploy police resources?
  - Can a school district predict which students will drop out of school to target counseling interventions?
- Business:
  - Can Amazon predict what product a customer is going to buy based on their past purchases (Amazon)?
  - Can Netflix/YouTube/Spotify predict what movies/TV show/song a person will like based on what they have viewed/listened to in the past?
- Linear regression often used to do these predictions, but how well does our model predict the data?

2/ Model fit

Presidential popularity and the midterms

- Does popularity of the president or recent changes in the economy better predict midterm election outcomes?

Name	Description
`year`	midterm election year
`president`	name of president
`party`	Democrat or Republican
`approval`	Gallup approval rating at midterms
`seat.change`	change in the number of House seat's for the president's party
`rdi.change`	change in real disposable income over the year before midterms

Loading the data

```
midterms <- read.csv("data/midterms.csv")
head(midterms)
```

```
##   year  president party approval seat.change
## 1 1946     Truman     D       33         -55
## 2 1950     Truman     D       39         -29
## 3 1954 Eisenhower     R       61          -4
## 4 1958 Eisenhower     R       57         -47
## 5 1962    Kennedy     D       61          -4
## 6 1966    Johnson     D       44         -47
##   rdi.change
## 1         NA
## 2        8.0
## 3        0.2
## 4       -0.8
## 5        4.1
## 6        3.2
```

Fitting the approval model

```r
fit.app <- lm(seat.change ~ approval, data = midterms)
fit.app
```

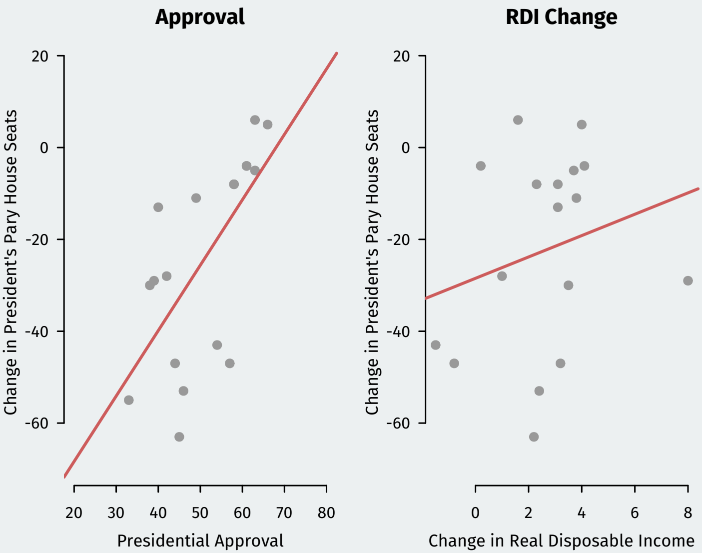
```
##
## Call:
## lm(formula = seat.change ~ approval, data = midterms)
##
## Coefficients:
## (Intercept)     approval
##      -96.84         1.42
```

Fitting the income model

```r
fit.rdi <- lm(seat.change ~ rdi.change, data = midterms)
fit.rdi
```

```
##
## Call:
## lm(formula = seat.change ~ rdi.change, data = midterms)
##
## Coefficients:
## (Intercept)   rdi.change
##      -28.48         2.33
```

Comparing models



Model fit

- How well does the model "fit the data"?
  - More specifically, how well does the model predict the outcome variable in the data?
- **Coefficient of determination** or R^2 ("R-squared"):
  - Prediction error just using the mean of Y: **Total sum of squares**

$$\text{TSS} = \sum_{i=1}^{n} (Y_i - \overline{Y})^2$$

  - Prediction error with the model: **Sum of squared residuals**

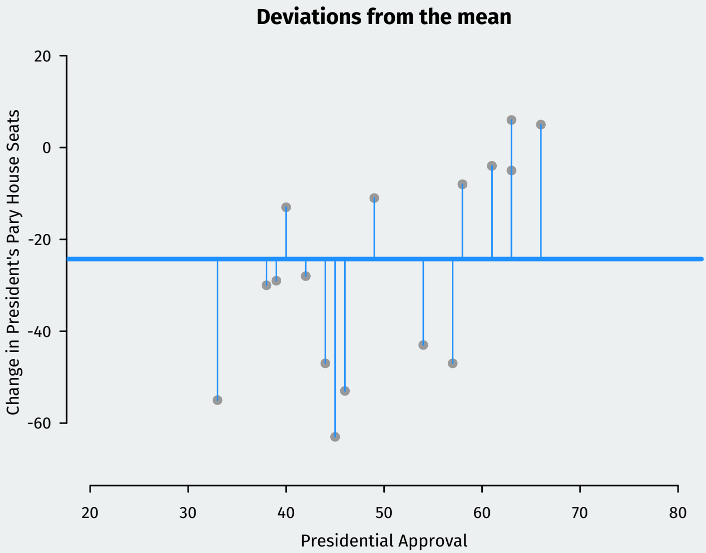
$$\text{SSR} = \sum_{i=1}^{n} \hat{\epsilon}_i^2$$

  - **Proportional reduction in error** how much of the prediction error is eliminated by using the model:

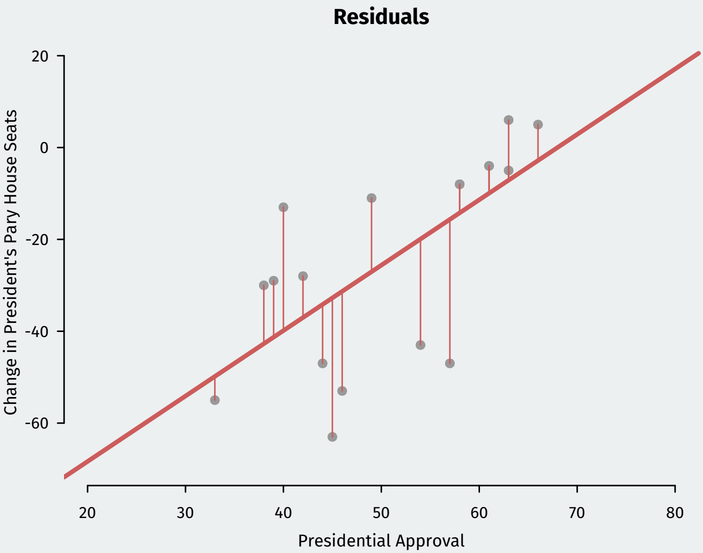
$$R^2 = \frac{TSS - SSR}{TSS}$$

- Roughly: proportion of the variation in Y_i "explained by" X_i

Total SS vs SSR



Total SS vs SSR



Model fit in R

- To access R^2 from the `lm()` output, first pass it to the `summary()` function:

```
fit.app.sum <- summary(fit.app)
fit.app.sum$r.squared
```

```
## [1] 0.431
```

- Compare to the fit using change in income:

```
fit.rdi.sum <- summary(fit.rdi)
fit.rdi.sum$r.squared
```

```
## [1] 0.0544
```

- Which does a better job predicting midterm election outcomes?

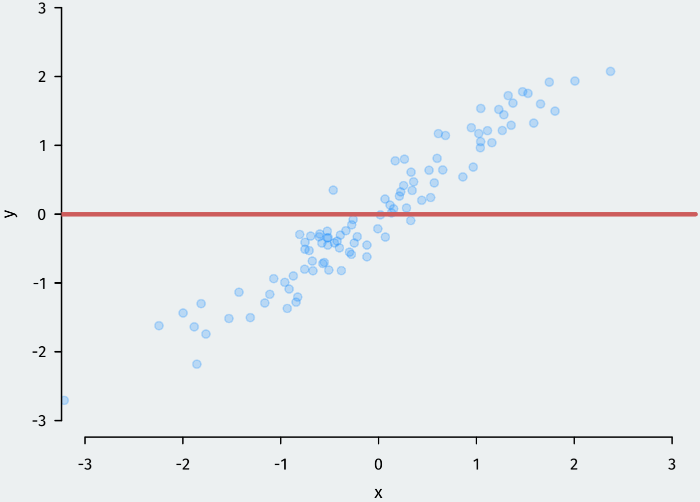
Fake data, better fit

- Little hard to see what's happening in that example.
- Let's look at fake variables x and y:

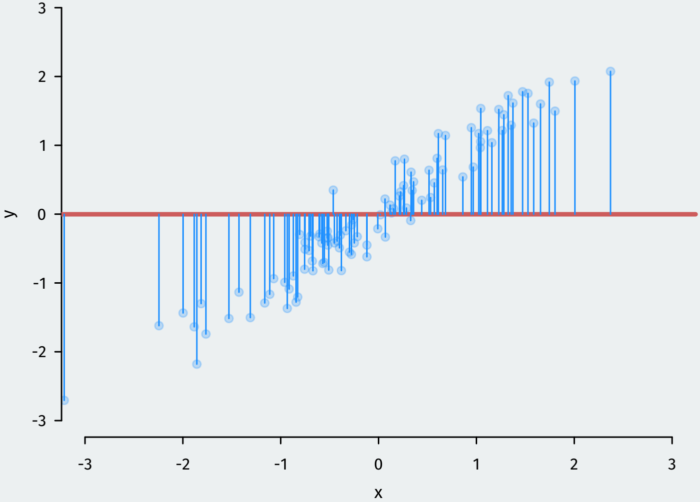
```
fit.x <- lm(y ~ x)
```

- Very good model fit: $R^2 \approx 0.95$

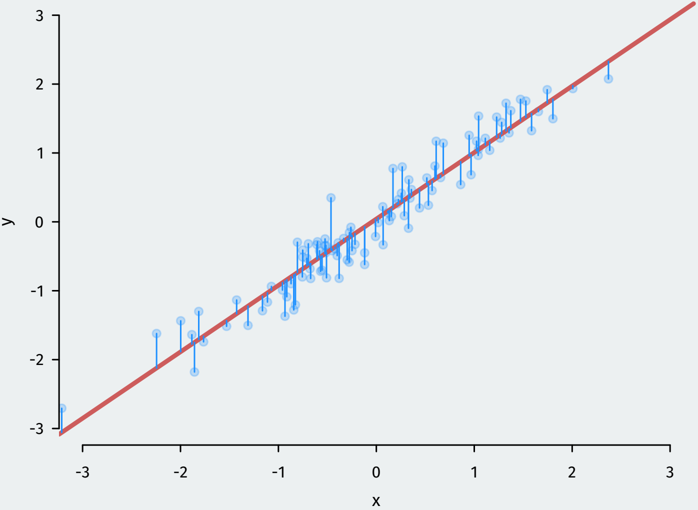
Fake data, better fit



Fake data, better fit

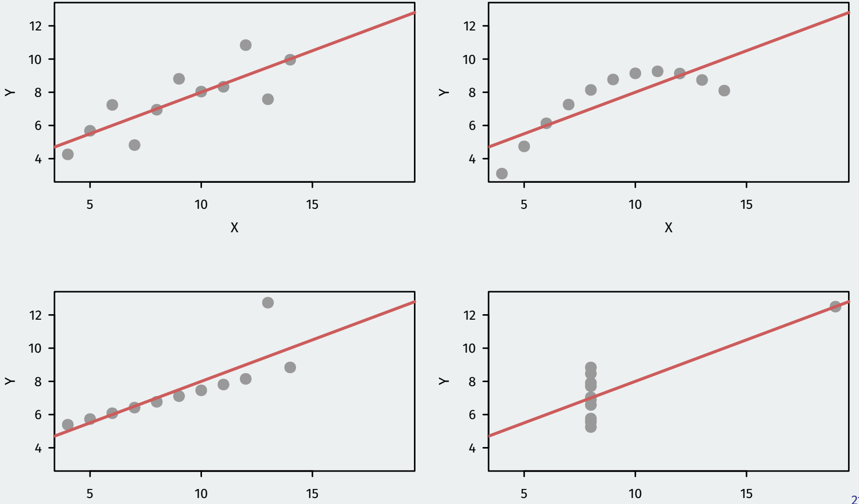


Fake data, better fit



Is R-squared useful?

- Can be very misleading. Each of these samples have the same R^2 even though they are vastly different:



Overfitting

- **In-sample fit**: how well your estimated model helps predict the data used to estimate the model.
  - R^2 is a measure of in-sample fit.
- **Out-of-sample fit**: how well your estimated model help predict outcomes outside of the sample used to fit the model.
- **Overfitting**: OLS and other statistical procedures designed to predict in-sample outcomes really well, but may do really poorly out of sample.
  - Example: predicting winner of Democratic presidential primary with gender of the candidate.
  - Until 2016, gender of the canidate was a **perfect** predictor of who wins the primary.
  - Prediction for 2016 based on this: Bernie Sanders as Dem. nominee.
  - Bad out-of-sample prediction due to overfitting!
- Could waste tons of governmental or corporate resources with a bad prediction model!

Avoiding overfitting

- Several procedure exist to guard against overfitting.
- **Cross validation** is the most popular:
  - Randomly choose half the sample to set aside (**test set**)
  - Estimate the coefficients with the remaining half of the units (**training set**)
  - Assess the model fit on the held out test set.
  - Switch the test and training set and repeat, average the results.
- Congrats, you know machine learning/artificial intelligence!

3/ Multiple predictors

Multiple predictors

- What if we want to predict Y as a function of many variables?

$$Y_i = \alpha + \beta_1 X_{i1} + \beta_2 X_{i2} + \cdots + \beta_k X_{ik} + \epsilon_i$$

- Why might we do this?
  - Better predictions!
  - Better interpretation: β_1 is the effect of X_1 holding all other independent variables constant. (**ceteris paribus**)
- With midterms data:

$$\texttt{seat.change}_i = \alpha + \beta_1 \texttt{approval}_i + \beta_2 \texttt{rdi.change}_i + \epsilon_i$$

Multiple regression in R

```r
mult.fit <- lm(seat.change ~ approval + rdi.change,
               data = midterms)
mult.fit
```

```
##
## Call:
## lm(formula = seat.change ~ approval + rdi.change, data = midterms)
##
## Coefficients:
## (Intercept)     approval   rdi.change
##     -117.17         1.61         4.21
```

- $\widehat{\alpha} =$ -117.2: average seat change president has 0% approval and no change in income levels.
- $\widehat{\beta}_1 =$ 1.61: average increase in seat change for additional percentage point of approval, **holding RDI change fixed**
- $\widehat{\beta}_2 =$ 4.213: average increase in seat change for each additional percentage point increase of RDI, **holding approval fixed**

Least squares with multiple regression

- How do we estimate the coefficients?
- The same exact way as before: minimize prediction error!
- Residuals (aka prediction error) with multiple predictors:

$$\hat{\epsilon}_i = Y_i - \widehat{\alpha} - \widehat{\beta}_1 X_{i1} - \widehat{\beta}_2 X_{i2}$$

- Find the coefficients that minimizes the **sum of the squared residuals**:

$$\text{SSR} = \sum_{i=1}^{n} \hat{\epsilon}_i^2 = (Y_i - \widehat{\alpha} - \widehat{\beta}_1 X_{i1} - \widehat{\beta}_2 X_{i2})^2$$

Model fit with multiple predictors

- R^2 mechanically increases when you add a variables to the regression.
  - But this could be overfitting!!
- Solution: penalize regression models with more variables.
  - Occam's razor: **simpler models are preferred**
- Adjusted R^2: lowers regular R^2 for each additional covariate.
  - If the added covariates doesn't help predict, adjusted R^2 goes down!

```r
summary(mult.fit)$adj.r.squared
```

```
## [1] 0.458
```

```r
summary(mult.fit)$r.squared
```

```
## [1] 0.526
```

On deck

- Next week:
  - How can we use regression for **causal inference**?
  - Allowing for different slopes for different groups of observations.
  - Allowing for non-linear effects!