

432 Class 15 Slides

github.com/THOMASELOVE/432-2018

2018-03-05

Setup

```
library(skimr); library(MASS)
library(robustbase); library(quantreg)
library(lmtest); library(sandwich)
library(boot); library(broom)
library(rms)
library(tidyverse)

decim <- function(x, k) format(round(x, k), nsmall=k)
```

Today's Materials

- 0 Comments on Quiz 1
 - 1 Robust Linear Regression Methods with Huber weights
 - 2 Robust Linear Regression with bisquare weights (biweights)
 - 3 Bounded Influence Regression & Least Trimmed Squares
 - 4 Penalized Least Squares using `ols` in the `rms` package
 - 5 Quantile Regression on the Median
-
- Looking forward to Thursday

Project 1 Groups

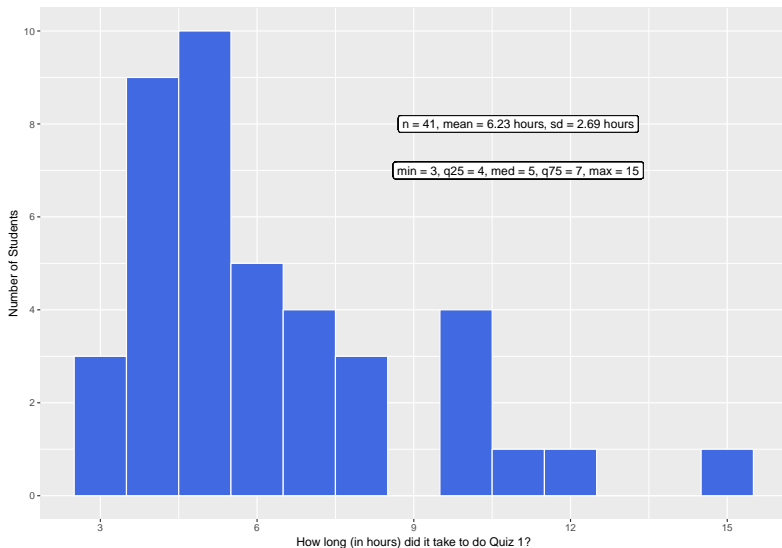
Project 1 Groups for Discussion in Class on Thursday

Group	Names
-------	-------

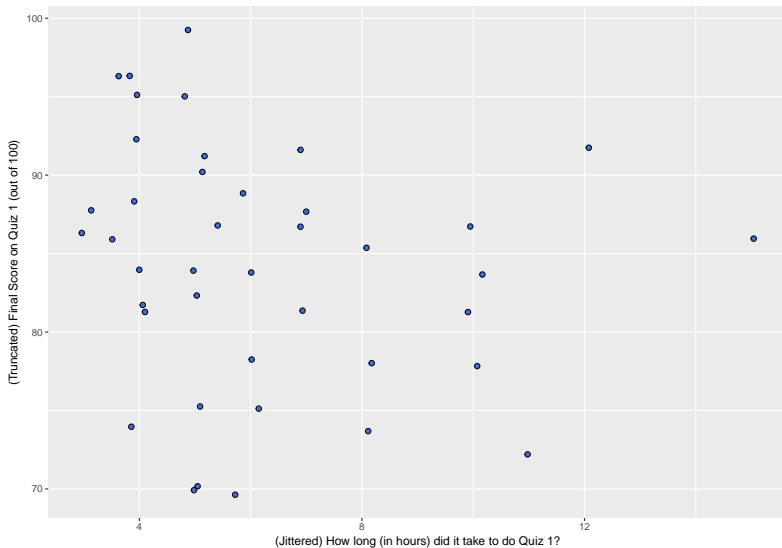
- | | |
|----|--|
| 1 | Laura Baldassari, Jenny Feng, Maher Kazimi, Satyakam Mishra, Vinh Trinh |
| 2 | Zainab (Albar) Albar, Dongze (Zaza) He, Nik Krieger, Andrew Shan |
| 3 | Andrew Tang, Sneha Vakamudi, Ruipeng Wei, Peter Wilkinson |
| 4 | Gwen Donley, Carli Lehr, Connor Swingle, Frances Wang |
| 5 | Ryan Honomichl, JJ Huang, Xin Xin Yu, Bilal Zonjy |
| 6 | Khaled Alayed, Kedar Mahajan, Preeti Pathak, Sarah Planchon Pope |
| 7 | Estee Cramer, Laura Cremer, Hyun Jo Kim, Roberto Martinez |
| 8 | Abhishek Deshpande, Jack McDonnell, Grace Park, Gabby Rieth |
| 9 | Haimeng Bai, Sophia Cao, Kate Dobbs, Elina Misicka |
| 10 | Vaishali (Vee) Deo, Caroline El Sanadi, Kaylee Sarna, Sandra Silva Camargo |
-

Quiz 1

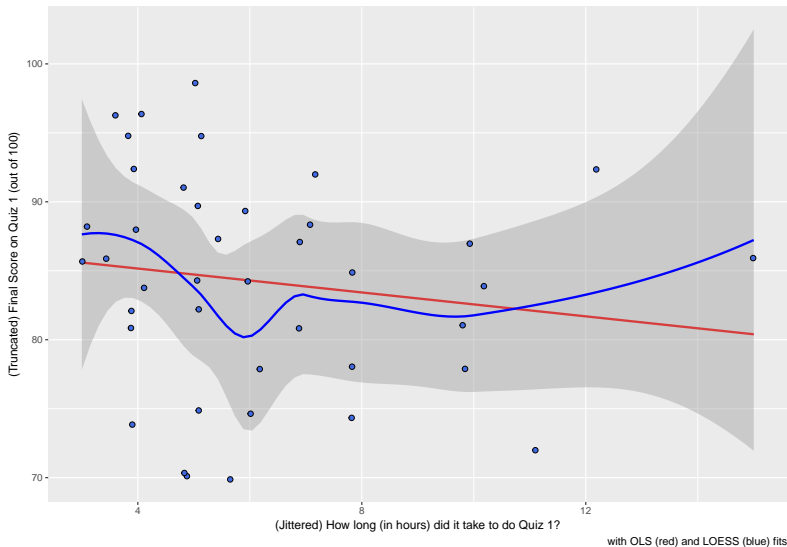
Quiz 1: How Long Did It Take?



Is Time Taken Associated with Score?



Time and Grade? (with OLS and loess fits)



OLS fit

Linear Regression Model

```
ols(formula = grade_on_quiz ~ time_on_quiz, data = q1min)
```

		Model Likelihood Ratio Test		Discrimination Indexes	
Obs	41	LR chi2	0.95	R2	0.023
sigma	7.7226	d.f.	1	R2 adj	-0.002
d.f.	39	Pr(> chi2)	0.3296	g	1.240

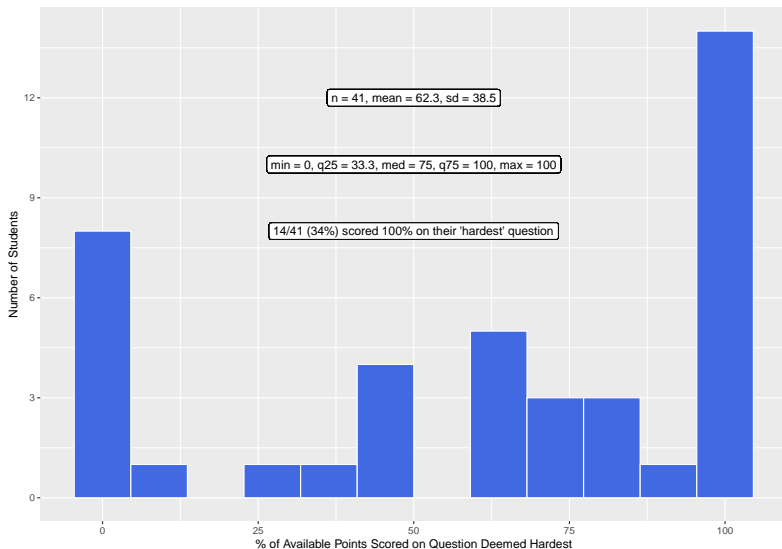
	Coef	S.E.	t	Pr(> t)
Intercept	86.8956	3.0707	28.30	<0.0001
time_on_quiz	-0.4333	0.4532	-0.96	0.3448

Quiz 1: What question was hardest?

Question	Votes	% of Available Points Awarded
Q21	14.33	a 90%, b 88%, c 85%, d 63% , e 66%
Q24	12.33	a 76%, b 68%, c 73%, d 66%
Q19	8.67	39%
Q23	1.17	22%
Q01	1	90%
Q22	1	98%
Q25	1	88%
Q28	1	54%
Q18	0.5	85%
Q15	0	59%
Q29	0	63%

I'll discuss Questions 15, 19, 21, 23, 24, 28 and 29 in class today.

Scoring on the questions you thought were hardest?



Q21

For each row, identify the model-fitting approach that most directly leads to the specified summary.

Model-fitting options are: `ols`, `lrm`, `glm` with binomial, none of these

- a. C statistic
- b. Adjusted R^2
- c. odds ratio estimate showing effect of increasing a quantitative predictor by 1 unit, while holding the others constant
- d. odds ratio estimate showing effect of increasing a quantitative predictor from the 25th to the 75th percentile of its distribution, while holding the others constant
- e. estimate of effect on a quantitative outcome associated with changing a binary predictor from 0 to 1
- f. Nagelkerke R-squared

Q24

R-square: 0.212, Adjusted R-square: 0.204

	index.orig	training	test	optimism	index.corrected
R-square	0.212	0.224	0.199	0.023	0.189

Consider the model summaries above. Identify the value in the columns that matches the description for each row.

- a. % of variation explained using this model for the data used to fit this model
- b. estimate of % of variation that will be explained by this model in a new data set
- c. value that is plotted in a best subsets analysis
- d. square of correlation between observed and predicted baby weights

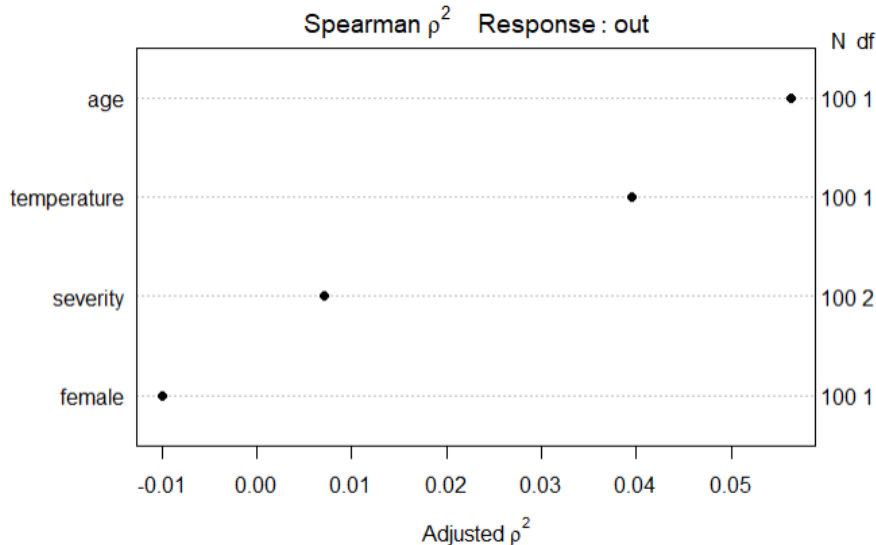
Q19

Linear Regression: must include four predictors: age (in years), female (1 = female, 0 = male), severity (three categories: High, Medium, Low) and temperature (in degrees Celsius).

Examine the Spearman rho-squared plot shown on the next slide. Suppose you are permitted to spend an additional **three** degrees of freedom. Which model?

- a. `ols(outcome ~ rcs(age, 3) + temperature + rcs(severity, 3) + female %ia% age, data = data_19)`
- b. `ols(outcome ~ rcs(age, 4) + rcs(temperature, 3) + severity + female, data = data_19)`
- c. `ols(outcome ~ female*age + temperature*severity, data = data_19)`
- d. `ols(outcome ~ rcs(age, 3) + rcs(temperature, 5) + severity + female, data = data_19)`
- e. None of these models are appropriate.

Q19 Spearman Plot



Q23 (beginning)

Variable	gestation	mom.age	mom.ht	mom.wt	dad.age	dad.ht
VIF	1.00	1.63	1.59	1.58	1.64	13.578

A child's birth weight depends on many things, among them the parents' genetic makeup, gestation period, and mother's activities during pregnancy. Suppose we are fitting a model using the `data_23` data set (which I have provided to you, **so you should take a look at it**) to predict a child's weight at birth using seven continuous predictors: gestation period (in days), mother's age (in years), height (in inches) and (pre-pregnancy) weight (in pounds); and father's age, height and weight. We run a full model containing main effects of all predictors, and obtain the variance inflation factors shown above.

Q23 (actual question)

Which of the following statements best describes the most appropriate next step to take in light of this output and your examination of the `data_23` data set?

- a. We should calculate and examine the leverage values for our observations.
- b. We should draw pictures, and study the distributions of the predictors in our model.
- c. We should drop the `dad.wt` variable and fit a new model.
- d. We should drop the `dad.ht` variable and fit a new model.
- e. We definitely need to drop one of the predictors about the father's size, but we're not sure which one.
- f. We should calculate and examine the Cook's distance for our observations.
- g. There isn't a substantial problem here. We can confidently make predictions using this model.









Q23 (skim)

Skim summary statistics

n obs: 1236

n variables: 8

Variable type: numeric

variable	missing	complete	n	mean	sd	p0	p25	median	p75	p100	hist
baby.wt	0	1236	1236	119.58	18.24	55	108.75	120	131	176	
dad.age	0	1236	1236	30.74	8.52	18	25	29	35	99	
dad.ht	0	1236	1236	81.67	14.28	60	70	73	99	99	
dad.wt	0	1236	1236	505.4	406.69	110	165	190	999	999	
gestation	0	1236	1236	286.91	75.16	148	272	280	288	999	
mom.age	0	1236	1236	27.37	6.46	15	23	26	31	99	
mom.ht	0	1236	1236	64.67	5.26	53	62	64	66	99	
mom.wt	0	1236	1236	153.98	147.87	87	115	126	140	999	

Q28 and Q29

were the “best subsets” questions.

It seems that most people who got them wrong miscounted. When the plot says “# of predictors, including the intercept” then a model with $x = 4$ is model containing four predictors, including the intercept, or a model with intercept PLUS THREE actually interesting predictors.

Q15

Suppose you are reviewing an academic paper and you have the four options listed below. In “How to be a Modern Scientist”, Jeff Leek suggests that there is a #1 way to be a jerk reviewer. Which of the following recommendation decisions could be made by someone who was actively TRYING TO BE a jerk reviewer? (Select any that apply.)

- ① Reject
- ② **Major revisions**
- ③ **Minor revisions**
- ④ Accept

Leek: “The #1 way to be a jerk reviewer is . . . [to ask for either major or minor revisions] . . . even if you think the paper is uninteresting and you wouldn’t accept it even if they did everything you said.”

General Comments on Quiz 1

I've probably said most of what should be said. 8-10 people scored within each of the following ranges:

- 90-99 (A: congratulations!)
- 87-89 (A-/B+, no problems)
- 84-86 (B+/B, near the median, no problems)
- 78-83 (a reasonable B grade, room to improve)
- below 78 (needs to improve)

If you have a complaint or question about grading on Quiz 1 after looking over the answer sketch, including the Results section, the most useful thing to do is **email it to me, or to 431-help**.

The crimestat data and an OLS fit

The crimestat data set

For each of 51 states (including the District of Columbia), we have the state's ID number, postal abbreviation and full name, as well as:

- **crime** - the violent crime rate per 100,000 people
- **poverty** - the official poverty rate (% of people living in poverty in the state/district) in 2014
- **single** - the percentage of households in the state/district led by a female householder with no spouse present and with her own children under 18 years living in the household in 2016
- **trump** - whether Donald Trump won the popular vote in the 2016 presidential election in that state/district (which we'll ignore for today)

The crimestat data set

```
crimestat <- read.csv("crimestat.csv") %>% tbl_df(
  crimestat
```

```
# A tibble: 51 x 7
```

	sid	state	crime	poverty	single	trump	state.full
	<int>	<fct>	<dbl>	<dbl>	<dbl>	<int>	<fct>
1	1	AL	427	19.2	9.02	1	Alabama
2	2	AK	636	11.4	7.63	1	Alaska
3	3	AZ	400	18.2	8.31	1	Arizona
4	4	AR	480	18.7	9.41	1	Arkansas
5	5	CA	396	16.4	7.25	0	California
6	6	CO	309	12.1	6.75	0	Colorado
7	7	CT	237	10.8	8.04	0	Connecticut
8	8	DE	489	13.0	6.52	0	Delaware
9	9	DC	1244	18.4	8.41	0	District of Colu~
10	10	FL	540	16.6	8.29	1	Florida

```
# ... with 41 more rows
```

Modeling crime with poverty and single

Our main goal will be to build a linear regression model to predict **crime** using centered versions of both **poverty** and **single**.

```
crimestat <- crimestat %>%  
  mutate(pov_c = poverty - mean(poverty),  
         single_c = single - mean(single))
```

Our original (OLS) model

```
(mod1 <- lm(crime ~ pov_c + single_c, data = crimestat))
```

Call:

```
lm(formula = crime ~ pov_c + single_c, data = crimestat)
```

Coefficients:

(Intercept)	pov_c	single_c
364.41	16.11	23.84

Significance of our coefficients?

```
tidy(mod1)
```

	term	estimate	std.error	statistic	p.value
1	(Intercept)	364.40588	22.932525	15.890351	9.475916e-21
2	pov_c	16.11462	9.615642	1.675876	1.002655e-01
3	single_c	23.84281	18.384226	1.296917	2.008596e-01

Robust Linear Regression with Huber Weights

Robust Linear Regression with Huber weights

There are several ways to do robust linear regression using M-estimation, including weighting using Huber and bisquare strategies.

- Robust linear regression here will make use of a method called iteratively re-weighted least squares (IRLS) to estimate models.
- M-estimation defines a weight function which is applied during estimation.
- The weights depend on the residuals and the residuals depend on the weights, so an iterative process is required.

We'll fit the model, using the default weighting choice: what are called Huber weights, where observations with small residuals get a weight of 1, and the larger the residual, the smaller the weight.

Our robust model (using MASS::rlm)

```
rob.huber <- rlm(crime ~ pov_c + single_c, data = crimestat)
```

Summary of the robust (Huber weights) model

```
tidy(rob.huber)
```

	term	estimate	std.error	statistic
1	(Intercept)	343.79816	13.130938	26.182300
2	pov_c	11.90977	5.505822	2.163123
3	single_c	30.98679	10.526627	2.943658

Now, *both* predictors appear to have estimates that exceed twice their standard error. So this is a very different result than ordinary least squares gave us.

Glance at the robust model (vs. OLS)

```
glance(mod1)
```

	r.squared	adj.r.squared	sigma	statistic	p.value	df
1	0.196879	0.1634156	163.771	5.883417	0.005184941	3

	logLik	AIC	BIC	deviance	df.residual
1	-330.8419	669.6837	677.411	1287405	48

```
glance(rob.huber)
```

	sigma	converged	logLik	AIC	BIC	deviance
1	59.14497	TRUE	-331.3785	670.7569	678.4842	1314784

Understanding the Huber weights a bit

Let's augment the data with results from this model, including the weights used.

```
crime_with_huber <- augment(rob.huber, crimestat) %>%  
  mutate(w = rob.huber$w) %>% arrange(w) %>% tbl_df  
  
head(crime_with_huber, 3)
```

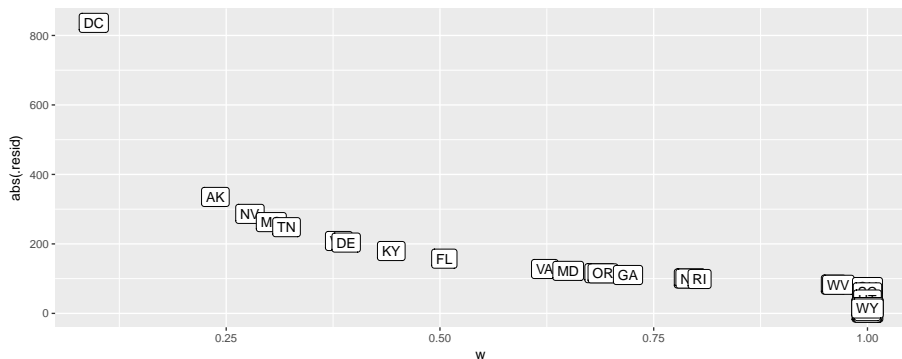
A tibble: 3 x 15

	sid	state	crime	poverty	single	trump	state.full	pov_c
	<int>	<fct>	<dbl>	<dbl>	<dbl>	<int>	<fct>	<dbl>
1	9	DC	1244	18.4	8.41	0	District o~	3.53
2	2	AK	636	11.4	7.63	1	Alaska	-3.47
3	29	NV	636	15.4	7.66	0	Nevada	0.527

... with 7 more variables: single_c <dbl>, .fitted <dbl>,
.se.fit <dbl>, .resid <dbl>, .hat <dbl>, .sigma <dbl>,
w <dbl>

Are cases with large residuals down-weighted?

```
ggplot(crime_with_huber, aes(x = w, y = abs(.resid))) +  
  geom_label(aes(label = state))
```



Conclusions from the Plot of Weights

- The district of Columbia will be down-weighted the most, followed by Alaska and then Nevada and Mississippi.
- But many of the observations will have a weight of 1.
- In ordinary least squares, all observations would have weight 1.
- So the more cases in the robust regression that have a weight close to one, the closer the results of the OLS and robust procedures will be.

summary(rob.huber)

```
Call: rlm(formula = crime ~ pov_c + single_c, data = crimestat)
```

Residuals:

Min	1Q	Median	3Q	Max
-262.751	-45.641	1.762	36.732	836.244

Coefficients:

	Value	Std. Error	t value
(Intercept)	343.7982	13.1309	26.1823
pov_c	11.9098	5.5058	2.1631
single_c	30.9868	10.5266	2.9437

Residual standard error: 59.14 on 48 degrees of freedom

Robust Linear Regression with the bisquare weighting function

Robust Linear Regression with the biweight

As mentioned there are several possible weighting functions - we'll next try the biweight, also called the bisquare or Tukey's bisquare, in which all cases with a non-zero residual get down-weighted at least a little. Here is the resulting fit...

```
(rob.biweight <- rlm(crime ~ pov_c + single_c,  
                     data = crimestat, psi = psi.bisquare))
```

Call:

```
rlm(formula = crime ~ pov_c + single_c, data = crimestat, psi  
Converged in 13 iterations
```

Coefficients:

(Intercept)	pov_c	single_c
336.17015	10.31578	34.70765

Degrees of freedom: 51 total; 48 residual

Scale estimate: 67.3

Coefficients and Standard Errors

```
tidy(rob.biweight)
```

	term	estimate	std.error	statistic
1	(Intercept)	336.17015	12.673297	26.525864
2	pov_c	10.31578	5.313932	1.941271
3	single_c	34.70765	10.159752	3.416191

Understanding the biweights weights a bit

Let's augment the data, as above

```
crime_with_biweights <- augment(rob.biweight, crimestat) %>%  
  mutate(w = rob.biweight$w) %>% arrange(w) %>% tbl_df  
  
head(crime_with_biweights, 3)
```

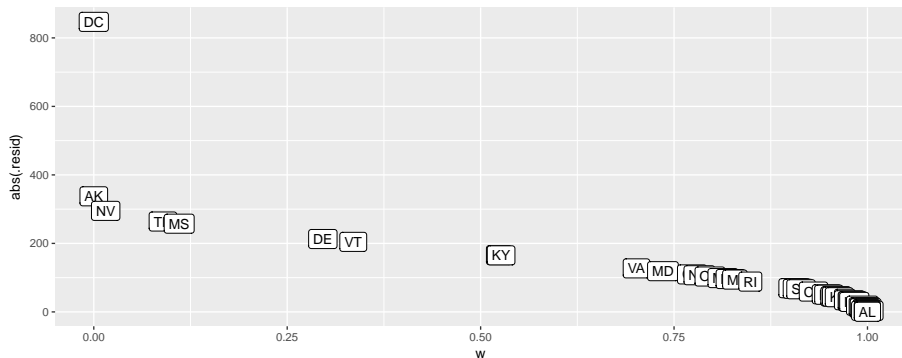
A tibble: 3 x 13

	sid	state	crime	poverty	single	trump	state.full	pov_c
	<int>	<fct>	<dbl>	<dbl>	<dbl>	<int>	<fct>	<dbl>
1	2	AK	636	11.4	7.63	1	Alaska	-3.47
2	9	DC	1244	18.4	8.41	0	District o~	3.53
3	29	NV	636	15.4	7.66	0	Nevada	0.527

... with 5 more variables: single_c <dbl>, .fitted <dbl>,
.se.fit <dbl>, .resid <dbl>, w <dbl>

Relationship of Weights and Residuals

```
ggplot(crime_with_biweights, aes(x = w, y = abs(.resid))) +  
  geom_label(aes(label = state))
```



Conclusions from the biweights plot

Again, cases with large residuals (in absolute value) are down-weighted generally, but here, Alaska and Washington DC receive no weight at all in fitting the final model.

- We can see that the weight given to DC and Alaska is dramatically lower (in fact it is zero) using the bisquare weighting function than the Huber weighting function and the parameter estimates from these two different weighting methods differ.
- The maximum weight (here, for Alabama) for any state using the biweight is still slightly smaller than 1.

summary(rob.biweight)

Call: `rlm(formula = crime ~ pov_c + single_c, data = crimestat`

Residuals:

Min	1Q	Median	3Q	Max
-257.58	-40.53	8.01	45.30	846.81

Coefficients:

	Value	Std. Error	t value
(Intercept)	336.1702	12.6733	26.5259
pov_c	10.3158	5.3139	1.9413
single_c	34.7077	10.1598	3.4162

Residual standard error: 67.27 on 48 degrees of freedom

Comparing OLS and the two weighting schemes

```
glance(mod1) # OLS
```

	r.squared	adj.r.squared	sigma	statistic	p.value	df
1	0.196879	0.1634156	163.771	5.883417	0.005184941	3

	logLik	AIC	BIC	deviance	df.residual
1	-330.8419	669.6837	677.411	1287405	48

```
glance(rob.biweight) # biweights
```

	sigma	converged	logLik	AIC	BIC	deviance
1	67.2749	TRUE	-331.8601	671.7201	679.4474	1339850

```
glance(rob.huber) # Huber weights
```

	sigma	converged	logLik	AIC	BIC	deviance
1	59.14497	TRUE	-331.3785	670.7569	678.4842	1314784

Bounded-Influence Regression

Bounded-Influence Regression and Least-Trimmed Squares

Under certain circumstances, M-estimators can be vulnerable to high-leverage observations, and so, bounded-influence estimators, like least-trimmed squares (LTS) regression have been proposed. The biweight that we have discussed is often fitted as part of what is called an MM-estimation procedure, by using an LTS estimate as a starting point.

The `ltsReg` function, which is part of the `robustbase` package (Note: **not** the `ltsreg` function from MASS) is what I use below to fit a least-trimmed squares model. The LTS approach minimizes the sum of the h smallest squared residuals, where h is greater than $n/2$, and by default is taken to be $(n + p + 1)/2$.

Least Trimmed Squares Model

```
lts1 <- ltsReg(crime ~ pov_c + single_c, data = crimestat)
```

Summarizing the LTS model

```
summary(lts1)$coeff
```

	Estimate	Std. Error	t value	Pr(> t)
Intercept	339.14817	11.616766	29.194715	1.601245e-29
pov_c	16.99322	4.973459	3.416781	1.418337e-03
single_c	24.99819	9.136683	2.736024	9.073473e-03

MM estimation

Specifying the argument `method="MM"` to `rlm` requests bisquare estimates with start values determined by a preliminary bounded-influence regression, as follows...

```
rob.MM <- rlm(crime ~ pov_c + single_c,  
              data = crimestat, method = "MM")  
  
glance(rob.MM)
```

	sigma	converged	logLik	AIC	BIC	deviance
1	75.79447	TRUE	-331.8072	671.6144	679.3418	1337077

summary(rob.MM)

```
Call: rlm(formula = crime ~ pov_c + single_c, data = crimestat)
```

Residuals:

Min	1Q	Median	3Q	Max
-252.412	-41.097	8.696	47.140	847.128

Coefficients:

	Value	Std. Error	t value
(Intercept)	336.3928	13.1930	25.4979
pov_c	10.5579	5.5318	1.9086
single_c	32.7754	10.5763	3.0989

Residual standard error: 75.79 on 48 degrees of freedom

Penalized Least Squares

Penalized Least Squares with rms

We can apply a penalty to least squares directly through the `ols` function in the `rms` package.

```
d <- datadist(crimestat)
options(datadist = "d")
pls <- ols(crime ~ pov_c + single_c, penalty = 1,
           data = crimestat, x=T, y = T)
```

The pls fit

Linear Regression Model

```
ols(formula = crime ~ pov_c + single_c, data = crimestat, x =  
y = T, penalty = 1)
```

		Model Likelihood		Discrimination	
		Ratio Test		Indexes	
Obs	51	LR chi2	11.18	R2	0.197
sigma	159.1209	d.f.	1.946198	R2 adj	0.164
d.f.	48.0538	Pr(> chi2)	0.0035	g	89.298

Residuals

Min	1Q	Median	3Q	Max
-284.24	-65.93	-16.68	15.66	807.01

How to Choose the Penalty in Penalized Least Squares?

The problem here is how to choose the penalty - and that's a subject I'll essentially skip today. The most common approach (that we've seen with the lasso) is cross-validation.

Meanwhile, what do we conclude about the fit here from AIC and BIC?

```
AIC(pls); BIC(pls)
```

d.f.

669.5781

d.f.

677.2014

Quantile Regression (on the Median)

Quantile Regression on the Median

We can use the `rq` function in the `quantreg` package to model the **median** of our outcome (violent crime rate) on the basis of our predictors, rather than the mean, as is the case in ordinary least squares.

```
rob.quan <- rq(crime ~ pov_c + single_c, data = crimestat)

glance(rob.quan)
```

	tau	logLik	AIC	BIC	df.residual
1	0.5	-315.7569	637.5138	643.3093	48

summary(rob.quan)

```
Call: rq(formula = crime ~ pov_c + single_c, data = crimestat)
```

```
tau: [1] 0.5
```

Coefficients:

	coefficients	lower bd	upper bd
(Intercept)	344.75658	336.94534	366.23603
pov_c	10.54757	3.06714	28.95962
single_c	32.27249	4.45889	48.18925

Estimating a different quantile ($\tau = 0.70$)

In fact, if we like, we can estimate any quantile by specifying the τ parameter (here $\tau = 0.5$, by default, so we estimate the median.)

```
(rob.quan70 <- rq(crime ~ pov_c + single_c, tau = 0.70,  
                  data = crimestat))
```

Call:

```
rq(formula = crime ~ pov_c + single_c, tau = 0.7, data = crimestat)
```

Coefficients:

(Intercept)	pov_c	single_c
379.72818	19.30376	32.15827

Degrees of freedom: 51 total; 48 residual

Conclusions

Comparing Five of the Models

Estimating the Mean

	Fit	Intercept CI	pov_c CI	single_c CI
	OLS	(318.6, 410.2)	(-3.13, 35.35)	(-12.92, 60.60)
	Robust (Huber)	(320.0, 367.6)	(0.89, 22.93)	(9.93, 52.05)
	Robust (biweight)	(310.7, 361.5)	(-0.30, 20.94)	(14.39, 55.03)
	Robust (MM)	(310.0, 362.8)	(-0.50, 21.62)	(11.62, 53.94)

Note: CIs estimated for OLS and Robust methods as point estimate \pm 2 standard errors

Estimating the Median

	Fit	Intercept CI	pov_c CI	single_c CI
	Quantile (Median) Reg	(336.9, 366.2)	(3.07, 28.96)	(4.46, 48.19)

Comparing AIC and BIC

Fit	AIC	BIC
OLS	669.7	677.4
Robust (Huber)	670.8	678.5
Robust (biweight)	671.7	679.4
Robust (MM)	671.6	679.3
Quantile (median)	637.5	643.3

Some General Thoughts

- ① When comparing the results of a regular OLS regression and a robust regression for a data set which displays outliers, if the results are very different, you will most likely want to use the results from the robust regression.
 - Large differences suggest that the model parameters are being highly influenced by outliers.
- ② Different weighting functions have advantages and drawbacks.
 - Huber weights can have difficulties with really severe outliers.
 - Bisquare weights can have difficulties converging or may yield multiple solutions.
 - Quantile regression approaches have some nice properties, but describe medians (or other quantiles) rather than means.

Project 1 Groups for Discussion in Class on Thursday

Group	Names
-------	-------

- | | |
|----|--|
| 1 | Laura Baldassari, Jenny Feng, Maher Kazimi, Satyakam Mishra, Vinh Trinh |
| 2 | Zainab (Albar) Albar, Dongze (Zaza) He, Nik Krieger, Andrew Shan |
| 3 | Andrew Tang, Sneha Vakamudi, Ruipeng Wei, Peter Wilkinson |
| 4 | Gwen Donley, Carli Lehr, Connor Swingle, Frances Wang |
| 5 | Ryan Honomichl, JJ Huang, Xin Xin Yu, Bilal Zonjy |
| 6 | Khaled Alayed, Kedar Mahajan, Preeti Pathak, Sarah Planchon Pope |
| 7 | Estee Cramer, Laura Cremer, Hyun Jo Kim, Roberto Martinez |
| 8 | Abhishek Deshpande, Jack McDonnell, Grace Park, Gabby Rieth |
| 9 | Haimeng Bai, Sophia Cao, Kate Dobbs, Elina Misicka |
| 10 | Vaishali (Vee) Deo, Caroline El Sanadi, Kaylee Sarna, Sandra Silva Camargo |
-