Lab 2: US Presidential Election Data

June 27, 2016

Today we will be reviewing various data sets having to do with the Mariners and US presidential elections. Our goals for this week are

- Review concepts about regression
- Introduce the ‘lm‘ function
- Examining the effect of outliers

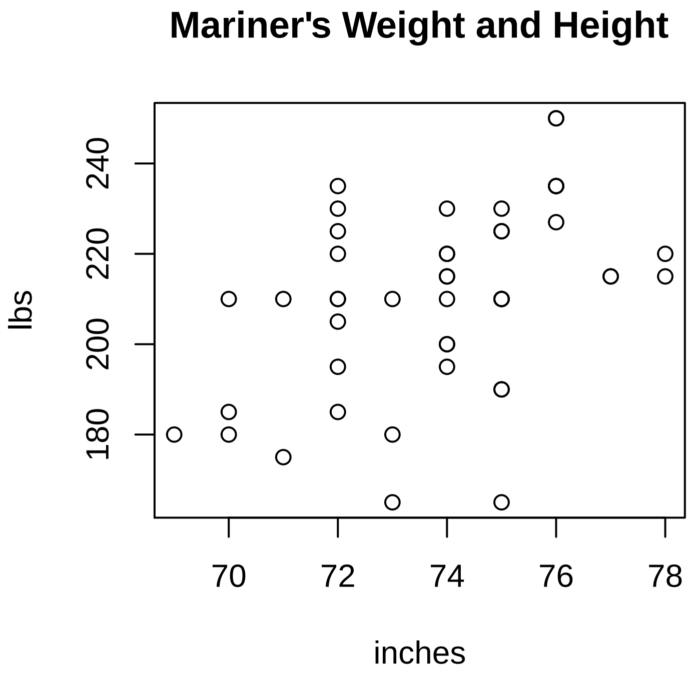
1 Best Fitting Line: Seattle Mariners

To review a few points about regression, we'll consider the weight and height of the Mariner's roster. First, let's read in the data and plot what it looks like.

```
# If you want to pull the file straight from my website, run this
mariners <- read.csv("http://www.stat.washington.edu/~ysamwang/notes/mariners.csv")

# If you have the file locally, run this
mariners <- read.csv("mariners.csv")
head(mariners)
```

```
##               Name  Wt Height
## 1    Jonathan Aro 235     72
## 2  Joaquin Benoit 250     76
## 3    Steve Cishek 215     78
## 4       Ryan Cook 215     74
## 5      Edwin Diaz 165     75
## 6 Charlie Furbush 215     77
```

```
plot(mariners$Height, mariners$Wt, main = "Mariner's Weight and Height",
     xlab = "inches", ylab = "lbs")
```



Suppose I am interested in the line which best describes the relationship between height (x variable) and weight (y variable) for the current Mariners roster. Thus, my **population** of interest is the current Mariners roster. Thus, in this case, I can actually calculate my **parameters** of interest, the a and b which minimize the sum of squared residuals, because I have access to the entire population (note this is typically not the case).

```
# Using the formulas from class
b <- cov(mariners$Wt, mariners$Height) / var(mariners$Height)
a <- mean(mariners$Wt) - b * mean(mariners$Height)

# Population parameters
a

## [1] -104.4259

b

## [1] 4.262324
```

So our regression equation would be

$$Weight_i = -104.43 + 4.26 \times Height + \epsilon_i \tag{1}$$

Questions

- How should we interpret these parameters?

Using these values, we can create predictions for each player's weight based on their height. We can also calculate the residual and check that the sum of the residuals is 0 as we claimed in class.

```
y.hat <- a + b * mariners$Height
residual <- mariners$Wt - y.hat
```

```
sum(residual)

## [1] -1.591616e-12
```

Now let's check to see that these values of a and b actually mimimize the sum of squared errors

$$SSE = \sum_i (y_i - \hat{y}_i)^2 \tag{2}$$

To do this, let's first calculate the SSE for our current estimates of a and b

```
sum(residual^2)

## [1] 15250.27
```

Now let's take a quick eyeball at the plot, and select a value for a and b (pretend you don't know the actual values we just calculated). I've filled in a guess, but you should change the code to your own values for `a.guess` and `b.guess`

```
a.guess <- -110
b.guess <- 5
y.hat.guess <- a.guess + b.guess * mariners$Height
residual.guess <- mariners$Wt - y.hat.guess
sum(residual.guess^2)

## [1] 125224
```

Questions

- What is the SSE for your "guessed" values of a and b?
- Is it less than the SSE for the least squares values of a and b?

However, let's suppose I didn't have data for the full roster, but instead I needed to gather it myself. I ask Scott Servais, the Mariners Manager, and he says I can get the data from the players. However, since they're in the middle of the season and he doesn't want to distract the players, he says I can only ask 10 of the players, not the entire team. So I randomly select 10 players and get the following data.

To simulate this hypothetical situation happen, we first use the `sample` function which picks 10 random numbers between 1 and 46 (the number of players on the roster). Note that `c(1:46)` is shorthand for a vector containing all whole numbers between 1 and 46.

```
players <- sample(c(1:46), size = 10)

# Set of players we selected. This is will be our sample
players

##  [1]  6 16  4 32 42 11 14 25  9 44

mariners[players, ]

##                  Name  Wt Height
## 6   Charlie Furbush 215     77
## 16       Vidal Nuno 210     71
## 4         Ryan Cook 215     74
## 32 Patrick Kivlehan 215     74
## 42   Shawn O'Malley 175     71
## 11     Wade LeBlanc 210     75
## 14       Wade Miley 220     72
## 25        Tony Zych 190     75
## 9   Hisashi Iwakuma 210     75
## 44    Stefen Romero 220     74
```

We then fit a regression to the data from the 10 players selected. The 10 players that we would select is our **sample**, and the $\hat{a}$ and $\hat{b}$ we would get from only measuring 10 players are **statistics** which describe our sample.

```
b.hat <- cov(mariners$Wt[players], mariners$Height[players]) / var(mariners$Height[players])
a.hat <- mean(mariners$Wt[players]) - b * mean(mariners$Height[players])

# The statistics we calculate from our sample
b.hat

## [1] 2.261905

a.hat

## [1] -106.5595
```

Questions

- Try this out yourself by running the code. You will get a different answer because your sample will probably be different from mine.
- How do these values differ from our parameters calculated above?

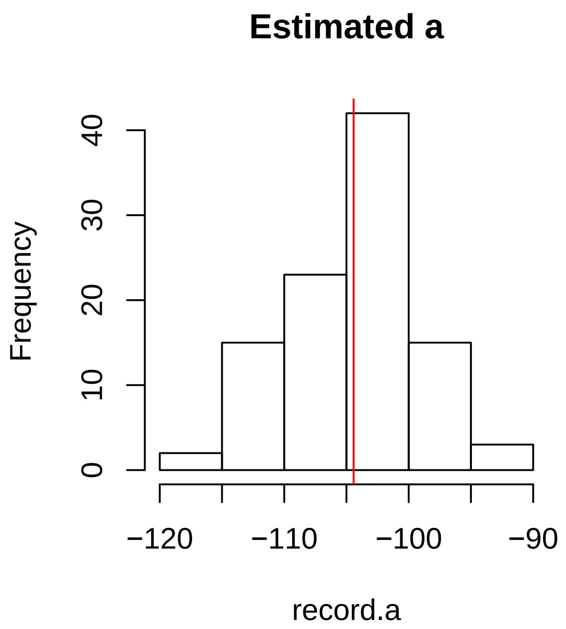
Let's see how these values differ as we take many random samples. To do this, we will use a for loop which repeats a block of code. Each time it repeats the block, it sets an index variable (in this case i) to the next value in the specified vector. We will repeat this procedure 100 times. We also create two vectors (record.a and record.b) to record the estimates values of $\hat{a}$ and $\hat{b}$ for each sample

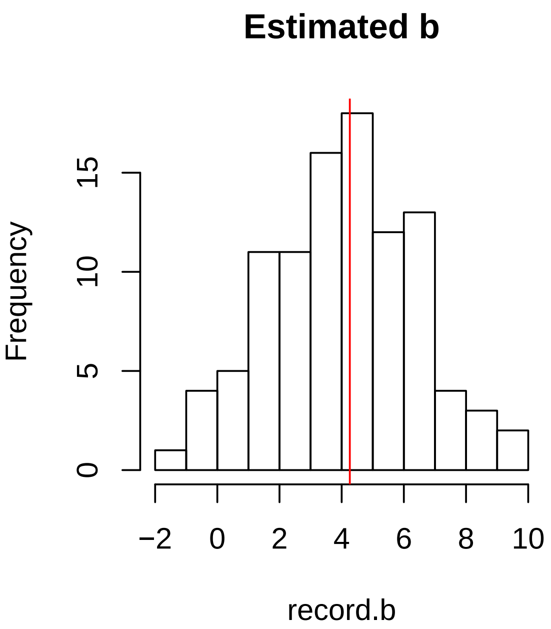
```
sample.size <- 100
record.a <- rep(0, sample.size)
record.b <- rep(0, sample.size)
for(i in c(1:sample.size)){

  # Set of players we selected. This is will be our sample
  players <- sample(c(1:dim(mariners)[1]), size = 10)

  # calculate the statistics
  b.hat <- cov(mariners$Wt[players], mariners$Height[players]) /
    var(mariners$Height[players])

  a.hat <- mean(mariners$Wt[players]) - b * mean(mariners$Height[players])

  # record the statistics we calculate from our sample
  record.b[i] <- b.hat
  record.a[i] <- a.hat
}
```

We can plot the distribution of the estimated $\hat{a}$ and $\hat{b}$ values and see that they vary with each sample around the true value of a and b we calculated above. The parameter values are indicated with the red vertical lines in the plots below.

```
par(mfrow = c(1,2))
hist(record.a, main = "Estimated a")
abline(v = a, col = "red")
```

```
hist(record.b, main = "Estimated b")
abline(v = b, col = "red")
```





We can see that each random sample we take gives us a good estimate of the true values of a and b, but $\hat{a}$ and $\hat{b}$ are different each time.

2 Linear Models for Democratic Voting% in US Presidential Elections

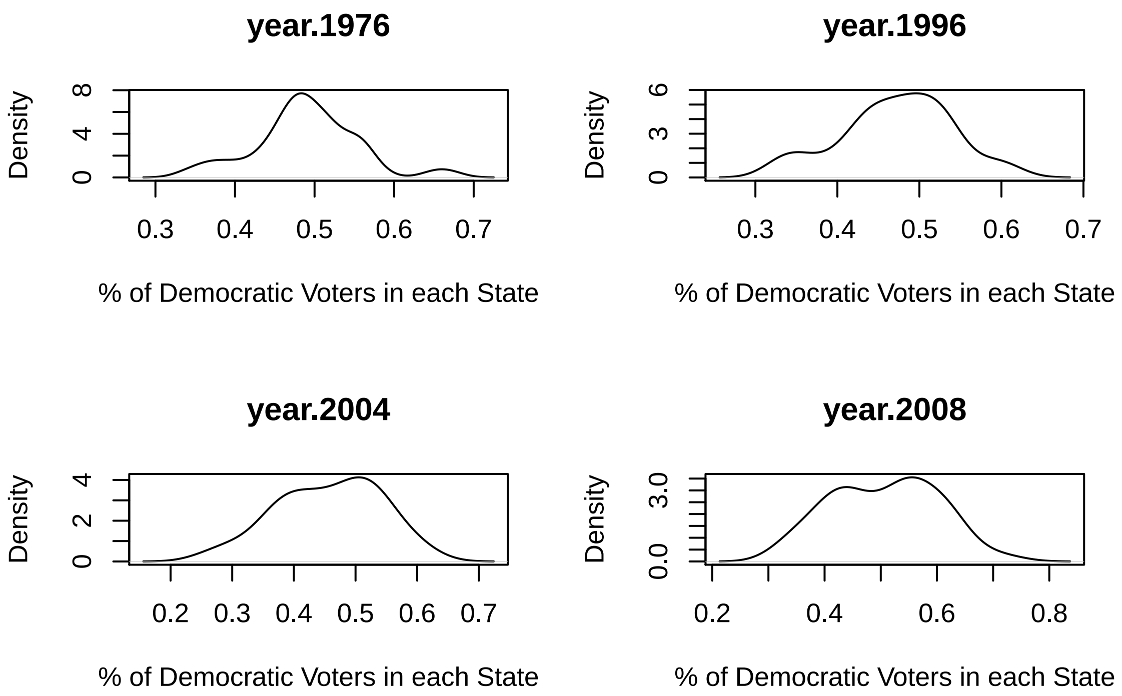
First, we will look at US presidential election data by state over the past few election cycles. Let's read in the data and take a look

```
# To grab the file directly off my website
election <- read.csv("http://www.stat.washington.edu/~ysamwang/notes/election4.csv")

# If you've downloaded the file locally
election <- read.csv("election4.csv")
head(election)

##       States year.2008 year.2004 year.1996 year.1976
## 1    Alabama      0.39      0.37      0.43      0.56
## 2     Alaska      0.38      0.36      0.33      0.36
## 3    Arizona      0.45      0.44      0.47      0.40
## 4   Arkansas      0.39      0.45      0.54      0.65
## 5 California      0.61      0.54      0.51      0.48
## 6   Colorado      0.54      0.47      0.44      0.43
```

The dataset contains the percentage of Democratic votes over the past few election cycles by state (courtesy of Wikipedia). We have data from 2008, 2004, 1996, 1976 titled year.2008, year.2004, year.1996, and year.1976 respsectively

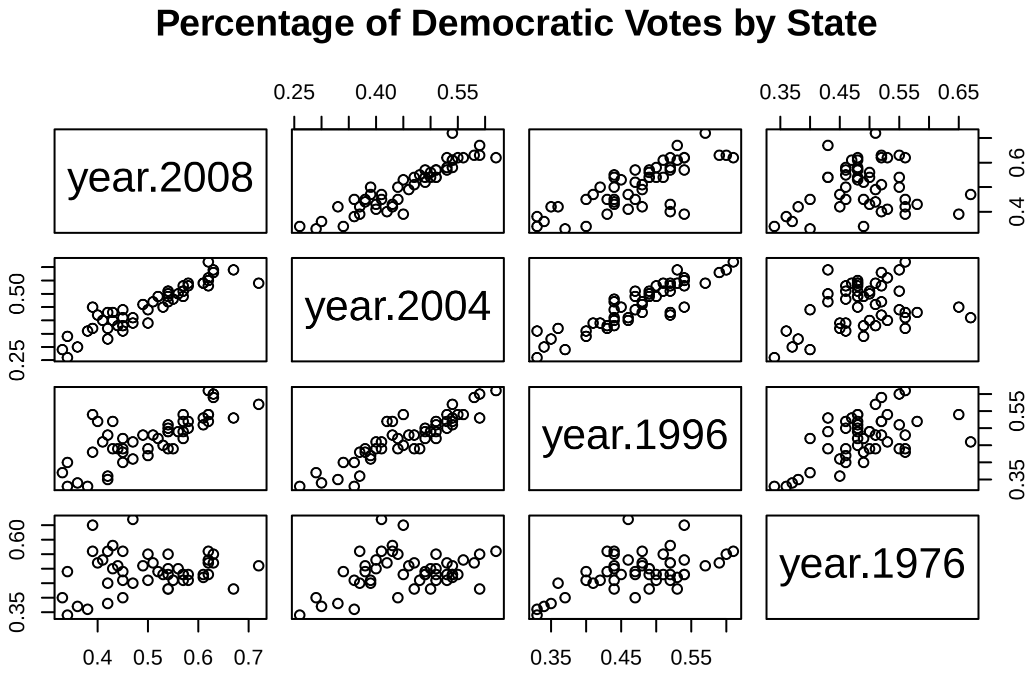
Let's take a look at our data first. First, let's consider the univariate (marginal) distributions.

```
par(mfrow = c(2,2))
for(i in 5:2){
  plot(density(election[,i]), main = names(election)[i], xlab = "% of Democratic Voters in each State")
}
```



In the plot below, each point represents a state in a given election cycle. Each plot represents two cycles, where one year is the X and the other year is the Y.

```
pairs(election[,2:5], main = "Percentage of Democratic Votes by State")
```



Questions

- What can we see from the univariate data? How would you describe the shape of each distribution? Are the uimodal or bimodal? Skewed?
- How does the distribution change over time? Does it seem like polarization by state has increased or decreased?
- Now viewing the joint distribution, what does the relationship of 2004 to 2008 to look like? What about 1996 and 2008? What about 1976 and 2008? Which relationships are stronger and which are weaker? Does this make sense?

Let's check the correlation to confirm what we can saw visually. In the matrix below, we can see the correlation between the percentage of democratic votes in each state from year to year. Notice that the matrix is symmetric because correlation(x,y) = correlation(y,x).

```
round(cor(election[,2:5]),3)
```

```
##           year.2008 year.2004 year.1996 year.1976
## year.2008     1.000     0.919     0.759     0.145
## year.2004     0.919     1.000     0.899     0.322
## year.1996     0.759     0.899     1.000     0.548
## year.1976     0.145     0.322     0.548     1.000
```

Questions

- Does this match with what you saw visually?
- Which correlations are the strongest? Which are the weakest?
- The correlation of each year with itself is 1, does this make sense?

3 The `lm` function

Consider the simple linear model

$$Y_i = \beta_0 + \beta_1 \times X_i$$

where Y_i and X_i are the % of democratic voters in 2008 and % of democratic voters in 2004 of the i^{th} state respectively, for $i = 1, \ldots, 50$ (not including DC). β_0 is the intercept, β_1 is the slope. We might expect the % of Democratic voters in 2004 to be highly indicative of the % of Democratic voters in 2008. Let's calculate the regression line for

$$2008 Democratic\% = a + b \times 2004 Democratic\% \tag{3}$$

Using the equations from class, we get can calculate $\hat{b}$ several different ways, but they all give the same value for $\hat{b}$

```
## Calculate directly
x <- election$year.2004
y <- election$year.2008
numerator <- 1 / (dim(election)[1] - 1) * sum((x - mean(x)) * (y - mean(y)))
denominator <- 1 / (dim(election)[1] - 1) * sum((x - mean(x))^2)
# beta hat
numerator / denominator

## [1] 1.031387

# Use covariance and variance
cov(x, y) / var(x)

## [1] 1.031387

# Use correlation and standard deviations
cor(x, y) * sd(y) / sd(x)

## [1] 1.031387

# save the value
b.hat <- cor(x, y) * sd(y) / sd(x)

# calculate a hat
a.hat <- mean(y) - b.hat * mean(x)
a.hat

## [1] 0.03486892
```

We can also use the `lm` function (lm stands for linear model) to do all the work for us. Let's take the output of lm and assign it to the variable `reression.model`. Inside the `lm` function, we've specified the formula we want the function to fit. The response variable (y) is on the left side of the ∼ (it should be located next to the number 1 on your keyboard). On the right hand side of the tilde, we put the explanatory variable. We also specify the data frame which contains the data of interest.

```
regression.model <- lm(year.2008 ~ year.2004, data = election)
```

We can get the fitted coefficients ($\hat{a}$ and $\hat{b}$) from the `regression.model` object by using `$coeff`. The first value is the y-intercept, and the second value is the coefficient on our explanatory variable (year.2004), which is denotes by $\hat{b}$ in the equation above. We can see that the values returned by `lm` are the same as the values we calculated above

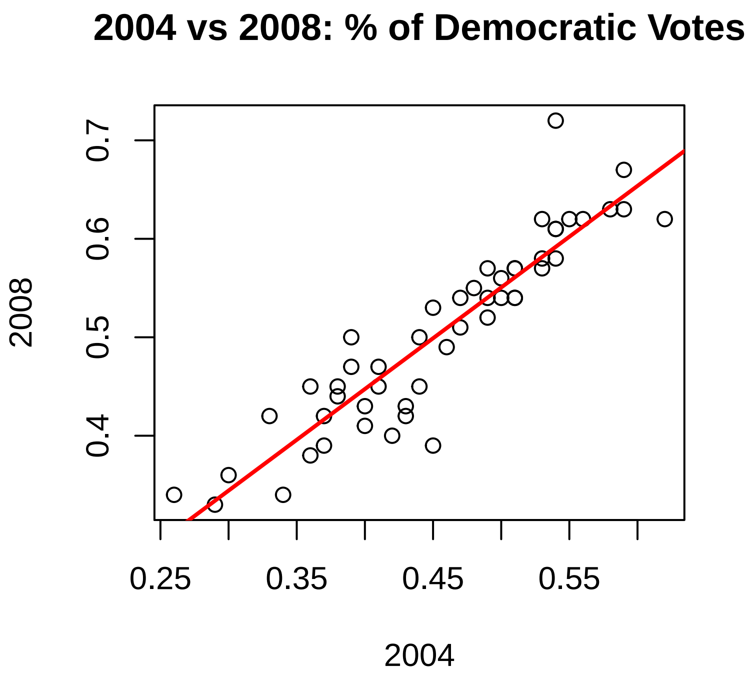
```
regression.model$coeff
```

```
## (Intercept)   year.2004
##  0.03486892  1.03138651
```

```
a.hat <- regression.model$coeff[1]
b.hat <- regression.model$coeff[2]
```

Let's take a look at the observed values and the predicted values. To plot the line, we use the `abline` command which plots a line given the y-intercept (specified by the argument `a`) and the slope (specified by the argument `b`). It looks like the model fits relatively well.

```
plot(election$year.2004, election$year.2008, main ="2004 vs 2008: % of Democratic Votes",
     xlab = "2004", ylab = "2008")
abline(a = a.hat, b = b.hat,
       col = "red", lwd = 2)
```



Questions

- What observations are you able to draw from the plot?
- Does the line fit well? Does the relationship look mostly linear?
- Are there any outliers?

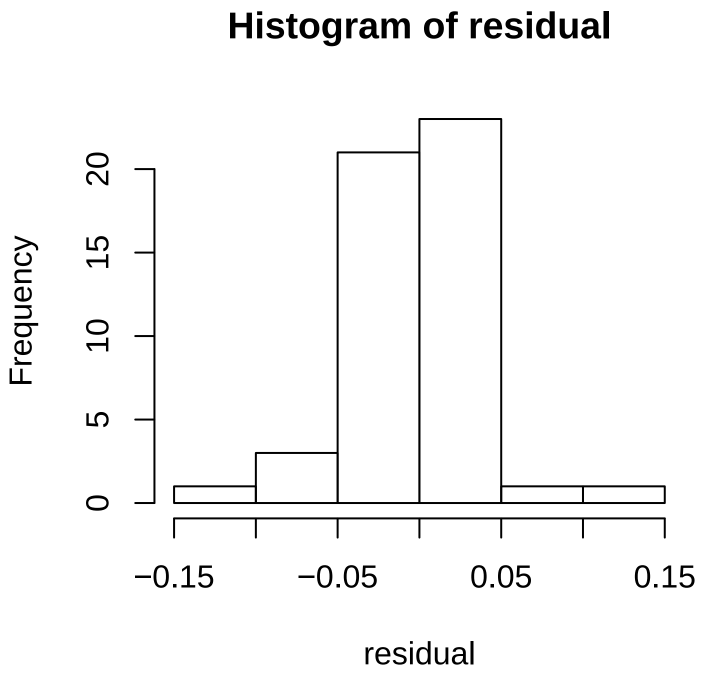
As mentioned in class, the sum of the residuals should be 0. Let's check to make sure

```
# Calculate y.hat
y.hat <- a.hat + b.hat * election$year.2004
residual <- election$year.2008 - y.hat
sum(residual)
```

```
## [1] 2.664535e-15
```

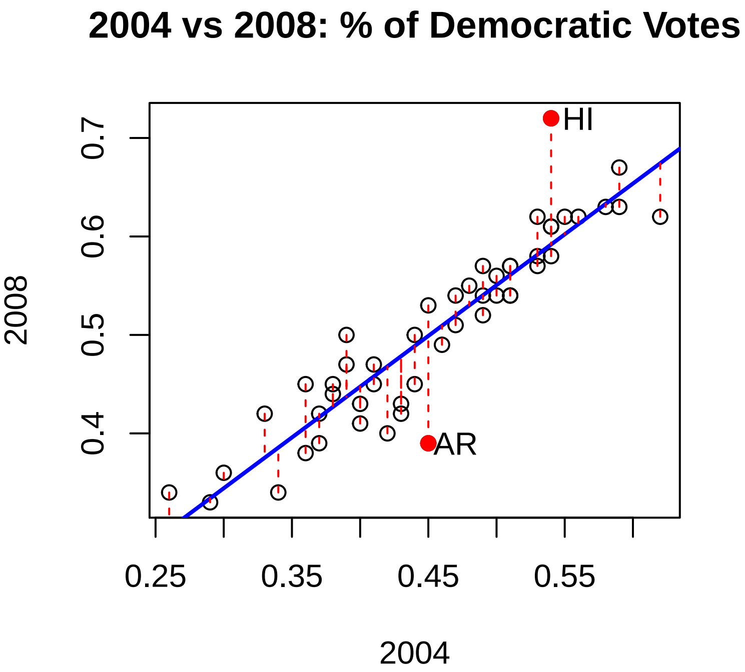
We can also take alook at the distribution of the residuals.

```
hist(residual)
```



It looks like there are two specific points with a large residuals.

```
plot(election$year.2004, election$year.2008,
     main ="2004 vs 2008: % of Democratic Votes",
     xlab = "2004", ylab = "2008")
abline(a = a.hat, b = b.hat,
       col = "blue", lwd = 2)

segments(x0 = election$year.2004, y0 = election$year.2008,
         x1 = election$year.2004, y1 = y.hat, col = "red", lty = 2)
points(election$year.2004[c(4,11)], election$year.2008[c(4,11)],
       col = "red", pch = 19)
text(election$year.2004[c(4,11)] + .02, election$year.2008[c(4,11)],
     labels = c("AR", "HI"))
```



Turns out those two states are Hawaii and Arkansas. In 2008, Hawaii had a much higher percentage of democratic votes than we would've predicted based on the 2004 election. Remember that 2008 was Obama vs McCain.

Questions

- Does this make sense with what you know about the election?
- What is Obama's homestate? Recall Hillary Clinton (who lost the Democratic noination in 2008) was the former first lady of Arkansas.

There's a very useful function in `R` called `summary`, which we've already seen from last lab. We can also use "summary" to our `regression.model` which gives us more information than just the raw output. Notice that it gives estimates for the coefficients, as well as standard errors for the coefficients. Recall in class that we said the estimated $\hat{a}$ and $\hat{b}$ are just estimates (statistics) of a parameter. The standard errors are rough estimates of how much our estimates might change if we took another sample. Recall the excercise above where we took samples of 10 Mariners players, and each sample gave a different result. The standard error is an estimate of the standard deviation of the histograms we were able to plot.

```
summary(regression.model)
```

```
##
## Call:
## lm(formula = year.2008 ~ year.2004, data = election)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.108993 -0.020013  0.001635  0.019595  0.128182
##
## Coefficients:
```

Figure 1: Confusing butterfly ballot

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.03487    0.02965   1.176    0.245
## year.2004    1.03139    0.06388  16.145   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03776 on 48 degrees of freedom
## Multiple R-squared:  0.8445,Adjusted R-squared:  0.8413
## F-statistic: 260.7 on 1 and 48 DF,  p-value: < 2.2e-16
```

4 Hanging Chads and Butterfly Ballots

In the 2000 US Presidential election with George Bush vs Al Gore, the entire election was decided by the state of Florida which itself was decided by less than 600 votes (a margin of .009%). In particular, Palm Beach county used a butterfly ballot which was widely criticized for its confusing design. Many speculated that this may have caused a large number of voters who intended to vote for Al Gore to vote for Pat Buchanan (Reform Party) instead.

We would expect that the number of registered voters in 2000 who belonged to the Reform party should be a pretty good predictor of how many people ended up voting for Pat Buchanan.

We have combined county vote data from Wikipedia with data from the Florida Division of Elections on the party affiliation of the registered voters in 2000. The variable `Buch.Votes` is the number of votes cast for Pat Buchannan and `Reg.Reform` is the number of registered reform party voters. `Total.Reg` is the total number of registered voters in that county.

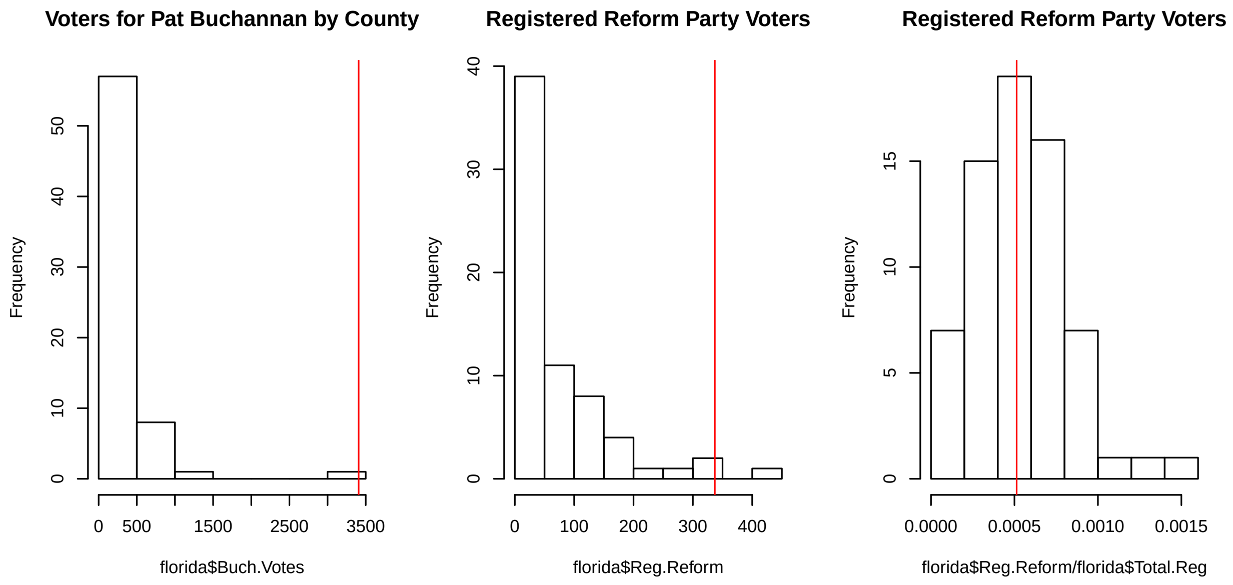
```
# To grab the file directly off my website
florida <- read.csv("http://www.stat.washington.edu/~ysamwang/notes/FL.csv")

# If you've downloaded the file locally
florida <- read.csv("FL.csv")
head(florida)
```

```
##     County Reg.Dem Reg.Rep Reg.Reform Total.Reg Buch.Votes
## 1 Alachua   64135   34319         91    120867        263
## 2    Baker   10261    1684          4     12352         73
## 3      Bay   44209   34286         55     92749        268
## 4 Bradford    9639    2832          3     13547         45
## 5  Brevard  107840  131427        148    283680        570
## 6  Broward  456789  266829        332    887764        795
```

First, let's take a look at the marginal distributions of registered reform party voters and votes for the reform party candidate Pat Buchannan. The red line in the plots below indicate the values for Palm County.

```
par(mfrow = c(1,3))

# Histogram of number of votes for Pat Buchannan
hist(florida$Buch.Votes, main =  "Voters for Pat Buchannan by County")
abline(v = florida$Buch.Votes[50], col = "red")

# Histogram of total registered reform party voters
```

```r
hist(florida$Reg.Reform, main = "Registered Reform Party Voters")
abline(v = florida$Reg.Reform[50], col = "red")

# Normalize for the number of total registered voters
hist(florida$Reg.Reform/florida$Total.Reg, main = "Registered Reform Party Voters")
abline(v = florida$Reg.Reform[50]/florida$Total.Reg[50], col = "red")
```



Questions

- Describe the distributions above?
- Does the Palm County seem like an outlier for the number of registered reform party candidates? What about for the number of votes for Pat Buchannan? What about when we normalize for the number of registered voters in each county?

We can use a regression to describe the number of voters we would've expected for Pat Buchannan in any given county, based only on the number of reform party voters. Again, we use the `lm` command.

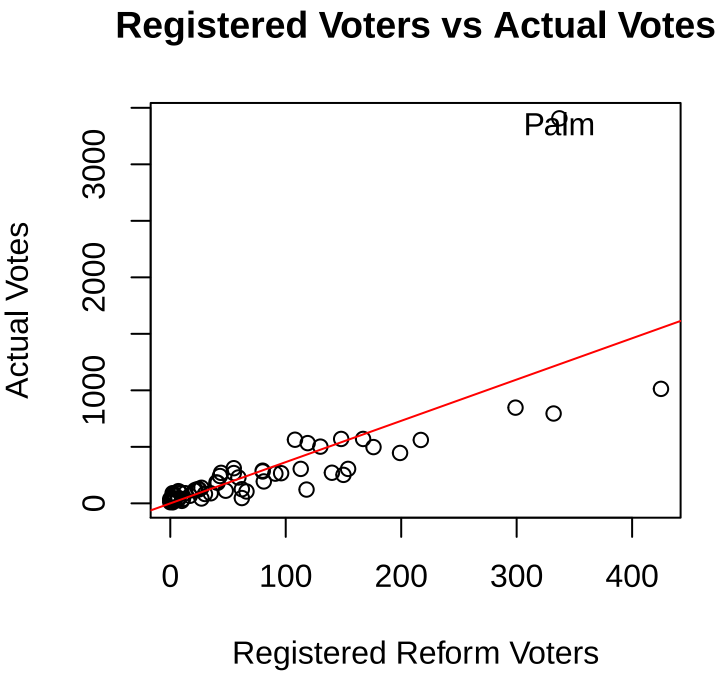
```r
florida.regression = lm(Buch.Votes~Reg.Reform, data = florida)
summary(florida.regression)

##
## Call:
## lm(formula = Buch.Votes ~ Reg.Reform, data = florida)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -538.89  -66.07   15.64   39.77 2176.50
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.2464    46.7415  -0.005    0.996
## Reg.Reform    3.6521     0.4099   8.909 7.16e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 302.7 on 65 degrees of freedom
## Multiple R-squared:  0.5498,Adjusted R-squared:  0.5429
## F-statistic: 79.37 on 1 and 65 DF,  p-value: 7.159e-13
```

Questions

- How would we interpret the estimated coefficients from the regression output?

  Let's view the predicted vs observed values

  ```
  plot(x = florida$Reg.Reform, y = florida$Buch.Votes,
       main = "Registered Voters vs Actual Votes",
       xlab = "Registered Reform Voters", ylab = "Actual Votes")

  text(x = florida$Reg.Reform[50], y = florida$Buch.Votes[50]-50,
       label = florida$County[50])
  abline(a = florida.regression$coefficients[1],
         b =  florida.regression$coefficients[2], col = "red")
  ```

  We can calculate predicted values.

  ```
  a.hat <- florida.regression$coeff[1]
  b.hat <- florida.regression$coeff[2]
  y.hat <- a.hat + b.hat * florida$Reg.Reform
  ```

Questions

- – Does Palm County appear to be an outlier in the joint distribution?
- – Based on the number of registered voters belonging to the reform party in Palm County, what is the fitted the number of actual votes for Pat Buchanan to be?
- – What is the residual for Palm County? (hint: Palm County is the 50th row in our data.frame)

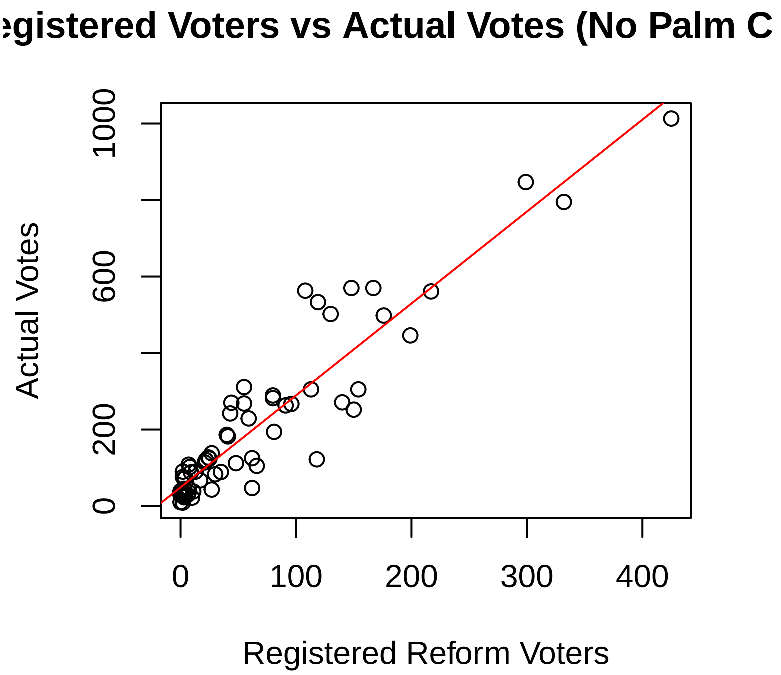
Clearly, it appears that Palm County is an outlier. Let's view the effect of Palm county on the regression and fit another model to the new data.

```
no.palm.county <- florida[-50, ]
florida.regression.no.palm <- lm(Buch.Votes~Reg.Reform, data = no.palm.county)
summary(florida.regression.no.palm)
```

```
##
## Call:
## lm(formula = Buch.Votes ~ Reg.Reform, data = no.palm.county)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -210.38  -38.58  -11.76   34.49  254.65
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  48.8089    12.4691   3.914 0.000222 ***
## Reg.Reform    2.4031     0.1164  20.648  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 80.03 on 64 degrees of freedom
## Multiple R-squared:  0.8695,Adjusted R-squared:  0.8674
## F-statistic: 426.3 on 1 and 64 DF,  p-value: < 2.2e-16
```

Let's view the predicted vs observed values without Palm county. Here we can see that the line seems to fit the data much better than before.

```
plot(x = no.palm.county$Reg.Reform,
     y = no.palm.county$Buch.Votes,
     main = "Registered Voters vs Actual Votes (No Palm County)",
     xlab = "Registered Reform Voters", ylab = "Actual Votes")


abline(a = florida.regression.no.palm$coefficients[1],
       b =  florida.regression.no.palm$coefficients[2], col = "red")
```



Questions

- How would we interpret the estimated coefficients from the regression output?
- Using this model, what is the predicted number of votes for Buchanan in Palm County? What is the prediction error? (Note this is similar, but not a residual because we did not use Palm County to fit our model)
- Compare the estimated values for this model with the estimated values of the previous model
- So which model is "correct"? The answer depends on how we define "correct," but if you had to predict the number of votes in each Florida county for the reform party candidate in this upcoming 2016 election, which model would you use? Why?