Linear Regression

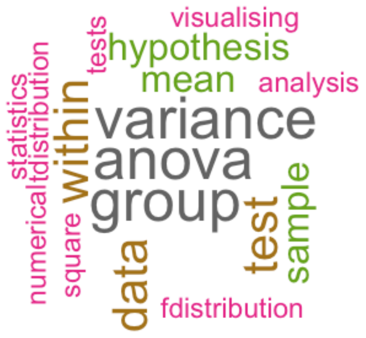
Computational Mathematics and Statistics

Jason Bryer, Ph.D.

March 4, 2025

One Minute Paper Results

What was the most important thing you learned during this class?



What important question remains unanswered for you?

SAT Scores

We will use the SAT data for 162 students which includes their verbal and math scores. We will model math from verbal. Recall that the linear model can be expressed as:

$$y = mx + b$$

Or alternatively as:

$$y = b_1 x + b_0$$

Where m (or b_1) is the slope and b (or b_0) is the intercept. Therefore, we wish to model:

$$SAT_{math} = b_1 SAT_{verbal} + b_0$$

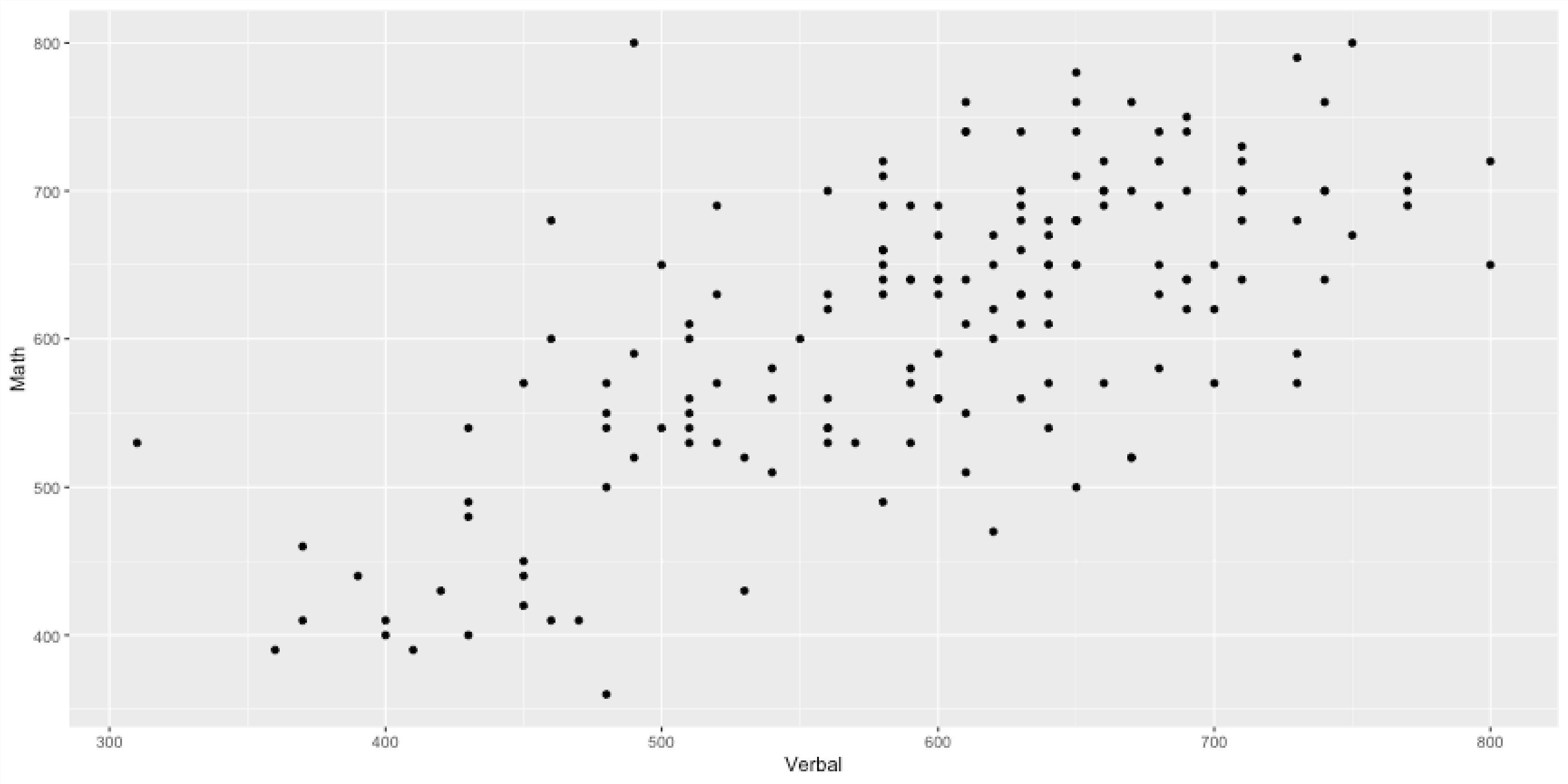
Data Prep

To begin, we read in the CSV file and convert the `Verbal` and `Math` columns to integers. The data file uses `.` (i.e. a period) to denote missing values. The `as.integer` function will automatically convert those to `NA` (the indicator for a missing value in R). Finally, we use the `complete.cases` eliminate any rows with any missing values.

```
sat <- read.csv('../course_data/SAT_scores.csv', stringsAsFactors=FALSE)
names(sat) <- c('Verbal','Math','Sex')
sat$Verbal <- as.integer(sat$Verbal)
sat$Math <- as.integer(sat$Math)
sat <- sat[complete.cases(sat),]
```

Scatter Plot

The first step is to draw a scatter plot. We see that the relationship appears to be fairly linear.



Descriptive Statistics

Next, we will calculate the means and standard deviations.

```
( verbalMean <- mean(sat$Verbal) )
```

```
## [1] 596.2963
```

```
( mathMean <- mean(sat$Math) )
```

```
## [1] 612.0988
```

```
( verbalSD <- sd(sat$Verbal) )
```

```
## [1] 99.5199
```

```
( mathSD <- sd(sat$Math) )
```

```
## [1] 98.13435
```

```
( n <- nrow(sat) )
```

```
## [1] 162
```

Correlation

The population correlation, rho, is defined as $\rho_{xy} = \frac{\sigma_{xy}}{\sigma_x \sigma_y}$ where the numerator is the *covariance* of *x* and *y* and the denominator is the product of the two standard deviations.

The sample correlation is calculated as $r_{xy} = \frac{Cov_{xy}}{s_x s_y}$

The covariates is calculated as $Cov_{xy} = \frac{\sum_{i=1}^{n} \left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right)}{n-1}$

```
(cov.xy <- sum( (sat$Verbal - verbalMean) * (sat$Math - mathMean) ) / (n - 1))
```

```
## [1] 6686.082
```

```
cov(sat$Verbal, sat$Math)
```

```
## [1] 6686.082
```

Correlation (cont.)

$$r_{xy} = \frac{\frac{\sum_{i=1}^{n}\left(X_i - \overline{X}\right)\left(Y_i - \overline{Y}\right)}{n-1}}{s_x s_y}$$

```
cov.xy / (verbalSD * mathSD)
```

```
## [1] 0.6846061
```

```
cor(sat$Verbal, sat$Math)
```

```
## [1] 0.6846061
```

http://bcdudek.net/rectangles

DAV 5300 Spring 2025

z-Scores

Calcualte z-scores (standard scores) for the verbal and math scores.

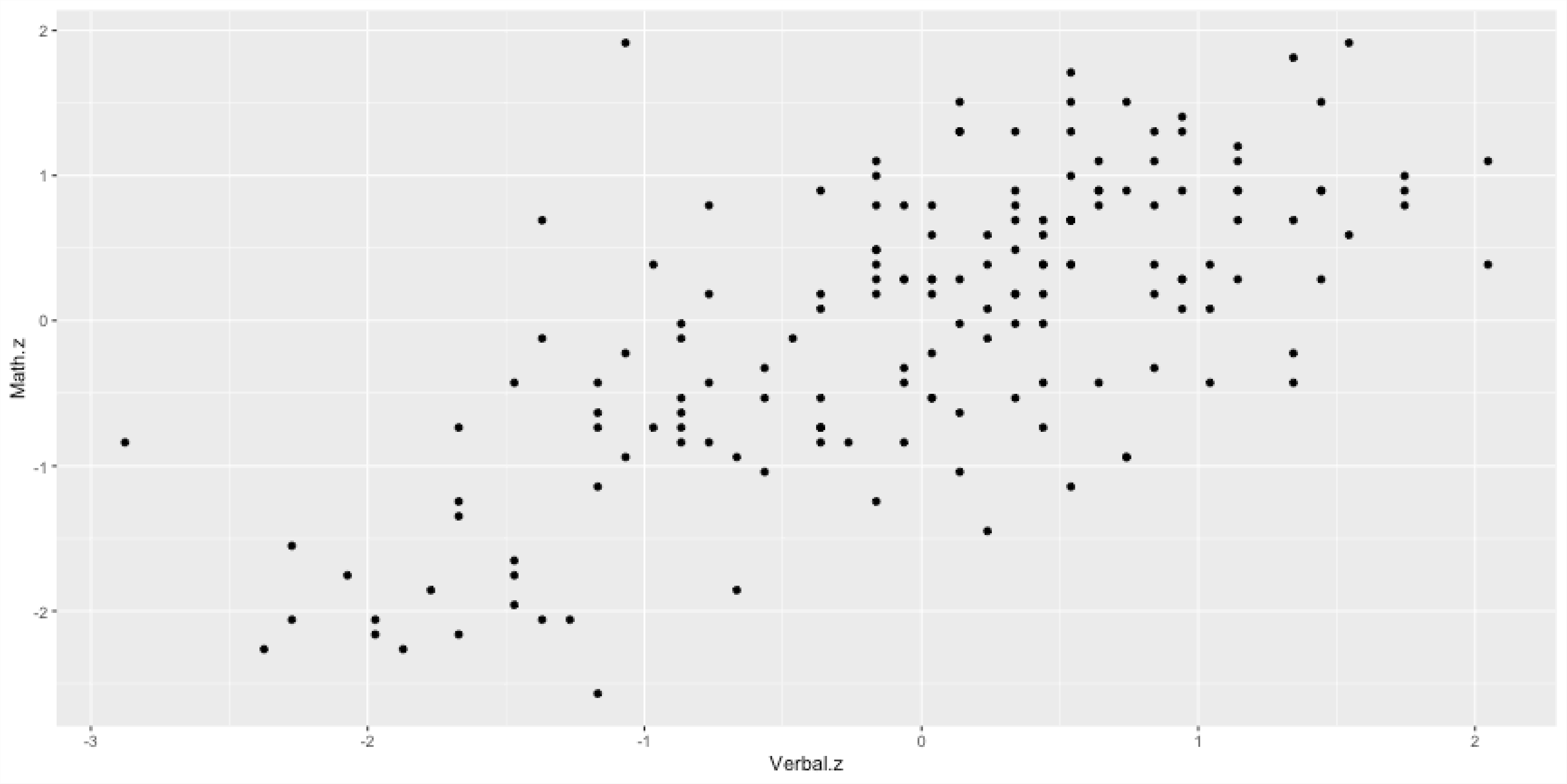
$$z = \frac{y - \overline{y}}{s}$$

```
sat$Verbal.z <- (sat$Verbal - verbalMean) / verbalSD
sat$Math.z <- (sat$Math - mathMean) / mathSD
head(sat)
```

```
##   Verbal Math Sex    Verbal.z      Math.z
## 1    450  450   F -1.47002058 -1.65180456
## 2    640  540   F  0.43914539 -0.73469449
## 3    590  570   M -0.06326671 -0.42899113
## 4    400  400   M -1.97243268 -2.16131016
## 5    600  590   M  0.03721571 -0.22518889
## 6    610  610   M  0.13769813 -0.02138665
```

Scatter Plot of z-Scores

Scatter plot of z-scores. Note that the pattern is the same but the scales on the x- and y-axes are different.



Correlation

Calculate the correlation manually using the z-score formula:

$$r = \frac{\sum z_x z_y}{n - 1}$$

```
r <- sum( sat$Verbal.z * sat$Math.z ) / ( n - 1 )
r
```

```
## [1] 0.6846061
```

Or the `cor` function in R is probably simplier.

```
cor(sat$Verbal, sat$Math)
```

```
## [1] 0.6846061
```

And to show that the units don't matter, calculate the correlation with the z-scores.

```
cor(sat$Verbal.z, sat$Math.z)
```

```
## [1] 0.6846061
```

Calculate the slope.

$$m = r\frac{S_y}{S_x} = r\frac{S_{math}}{S_{verbal}}$$

```
m <- r * (mathSD / verbalSD)
m
```

```
## [1] 0.6750748
```

Calculate the intercept

Recall that the point where the mean of x and mean of y intersect will be on the line of best fit). Therefore,

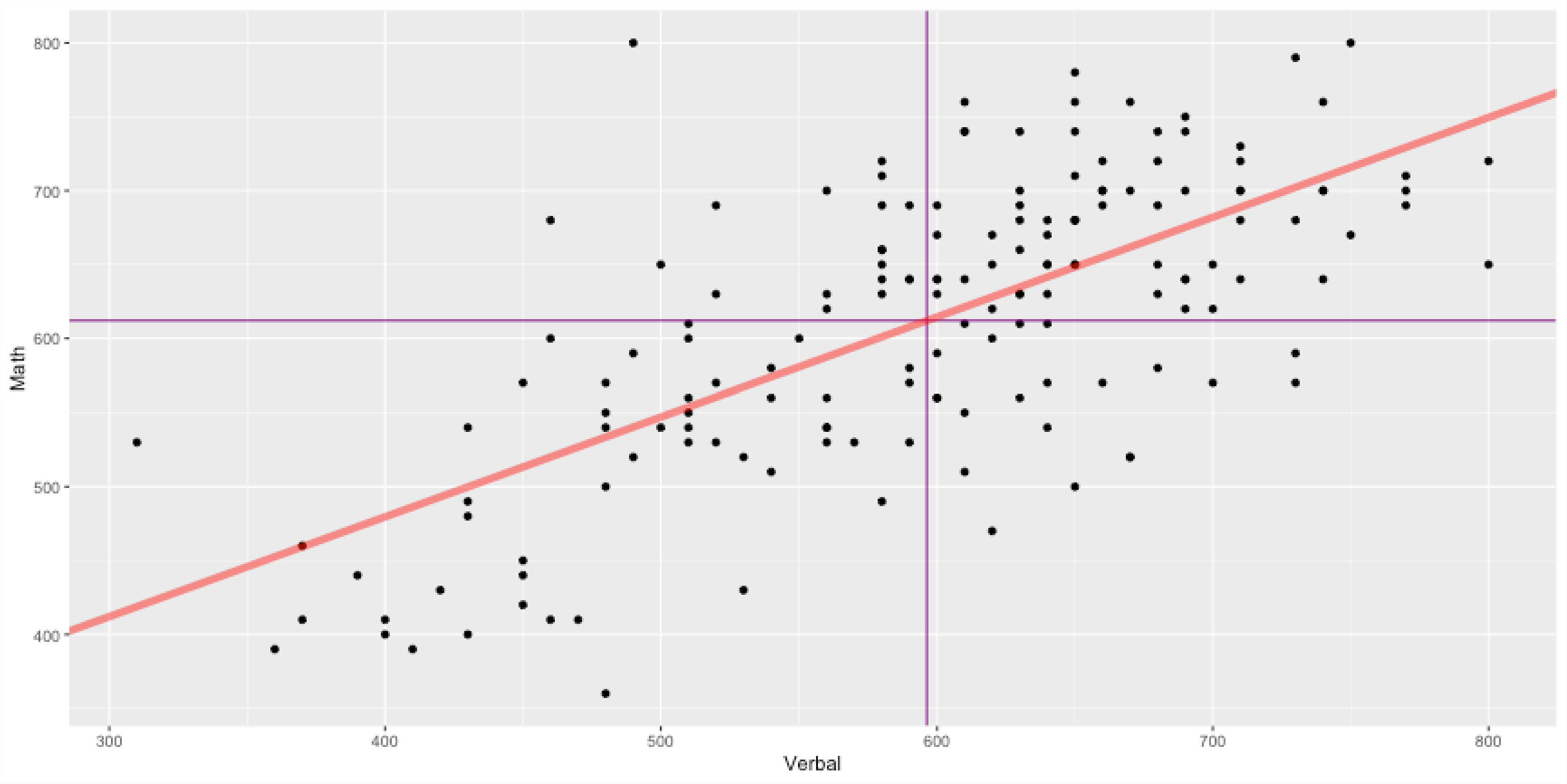
$$b = \overline{y} - m\overline{x} = \overline{SAT_{math}} - m\overline{SAT_{verbal}}$$

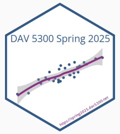
```
b <- mathMean - m * verbalMean
b
```

```
## [1] 209.5542
```

Scatter Plot with Regression Line

We can now add the regression line to the scatter plot. The vertical and horizontal lines represent the mean Verbal and Math SAT scores, respectively.





Examine the Residuals

To examine the residuals, we first need to calculate the predicted values of y (Math scores in this example).

```r
sat$Math.predicted <- m * sat$Verbal + b
sat$Math.predicted.z <- m * sat$Verbal.z + 0
head(sat, n=4)
```

```
##   Verbal Math Sex    Verbal.z     Math.z Math.predicted Math.predicted.z
## 1    450  450   F -1.47002058 -1.6518046       513.3378      -0.99237384
## 2    640  540   F  0.43914539 -0.7346945       641.6020       0.29645598
## 3    590  570   M -0.06326671 -0.4289911       607.8483      -0.04270976
## 4    400  400   M -1.97243268 -2.1613102       479.5841      -1.33153958
```

Examine the Residuals (cont.)

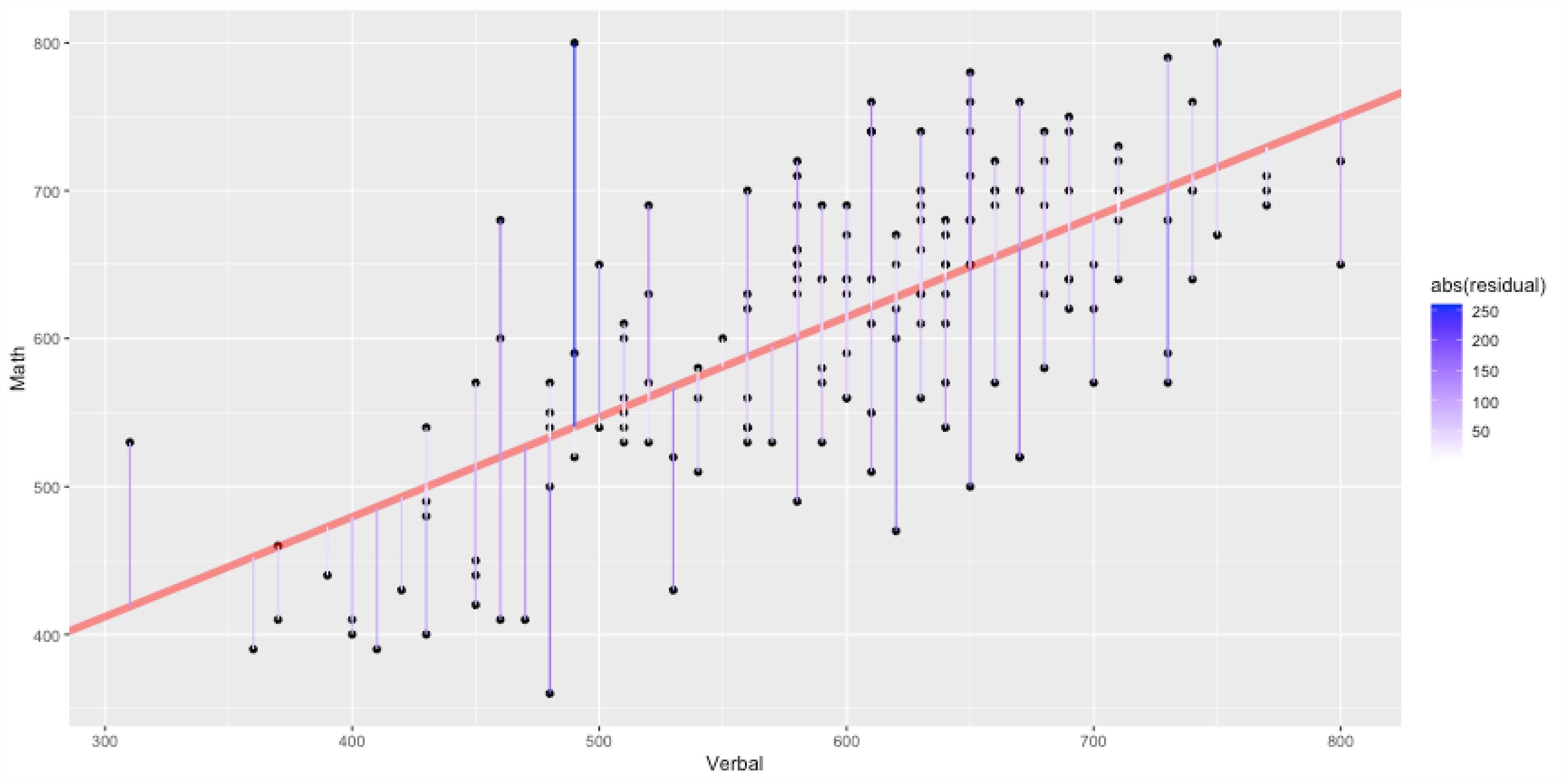
The residuals are simply the difference between the observed and predicted values.

```
sat$residual <- sat$Math - sat$Math.predicted
sat$residual.z <- sat$Math.z - sat$Math.predicted.z
head(sat, n=4)
```

```
##   Verbal Math Sex    Verbal.z     Math.z Math.predicted Math.predicted.z   residual residual.z
## 1    450  450   F -1.47002058 -1.6518046       513.3378      -0.99237384  -63.33782 -0.6594307
## 2    640  540   F  0.43914539 -0.7346945       641.6020       0.29645598 -101.60203 -1.0311505
## 3    590  570   M -0.06326671 -0.4289911       607.8483      -0.04270976  -37.84829 -0.3862814
## 4    400  400   M -1.97243268 -2.1613102       479.5841      -1.33153958  -79.58408 -0.8297706
```

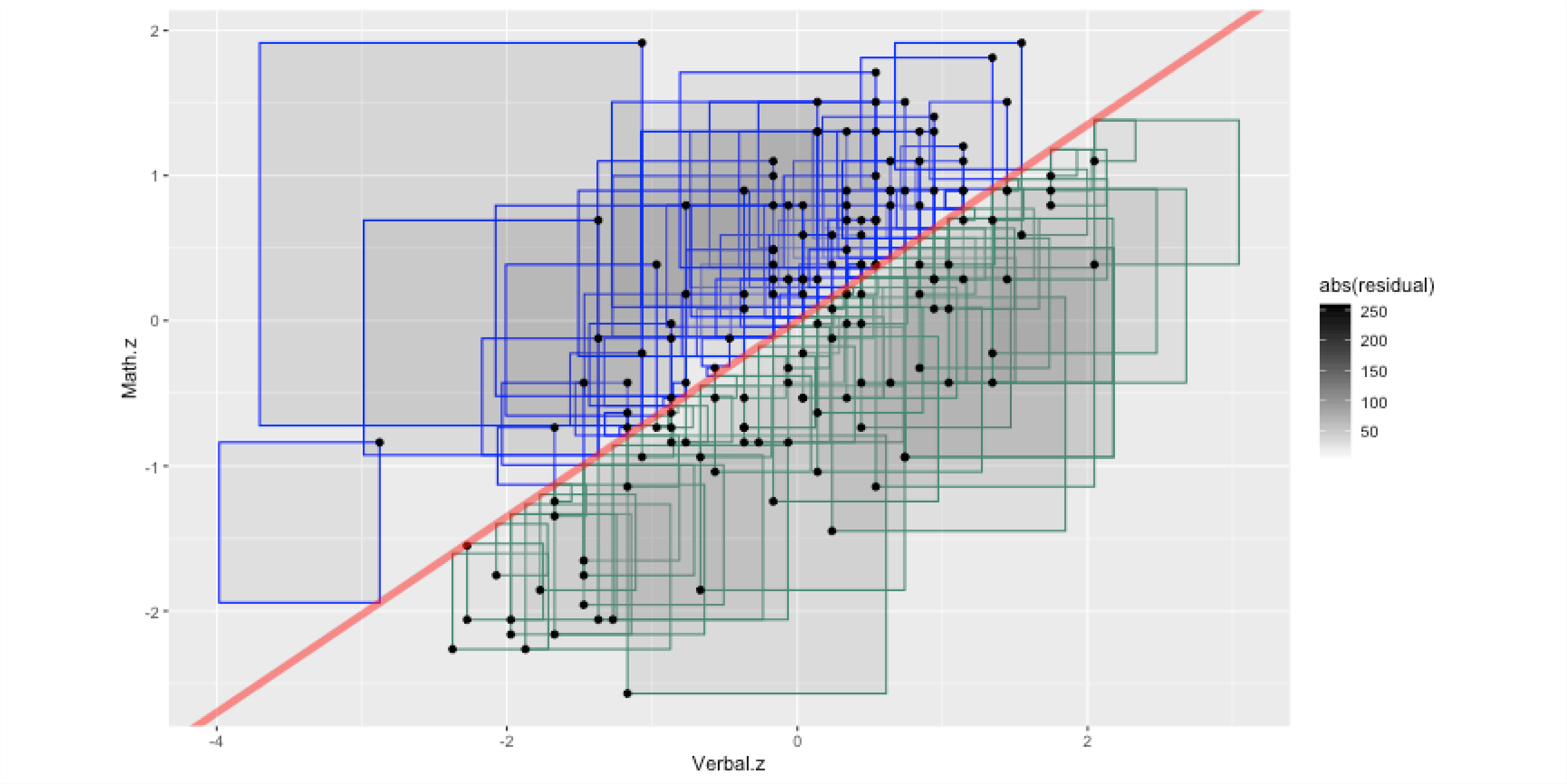
Scatter Plot with Residuals

Plot our regression line with lines representing the residuals. The line of best fit minimizes the residuals.



Scatter Plot with Residuals

Using the z-scores ensures that a 1-unit change in the *x*-axis is the same as a 1-unit change in the *y*-axis. This makes it easiert to plot the residuals as squares.



Minimizing Sum of Squared Residuals

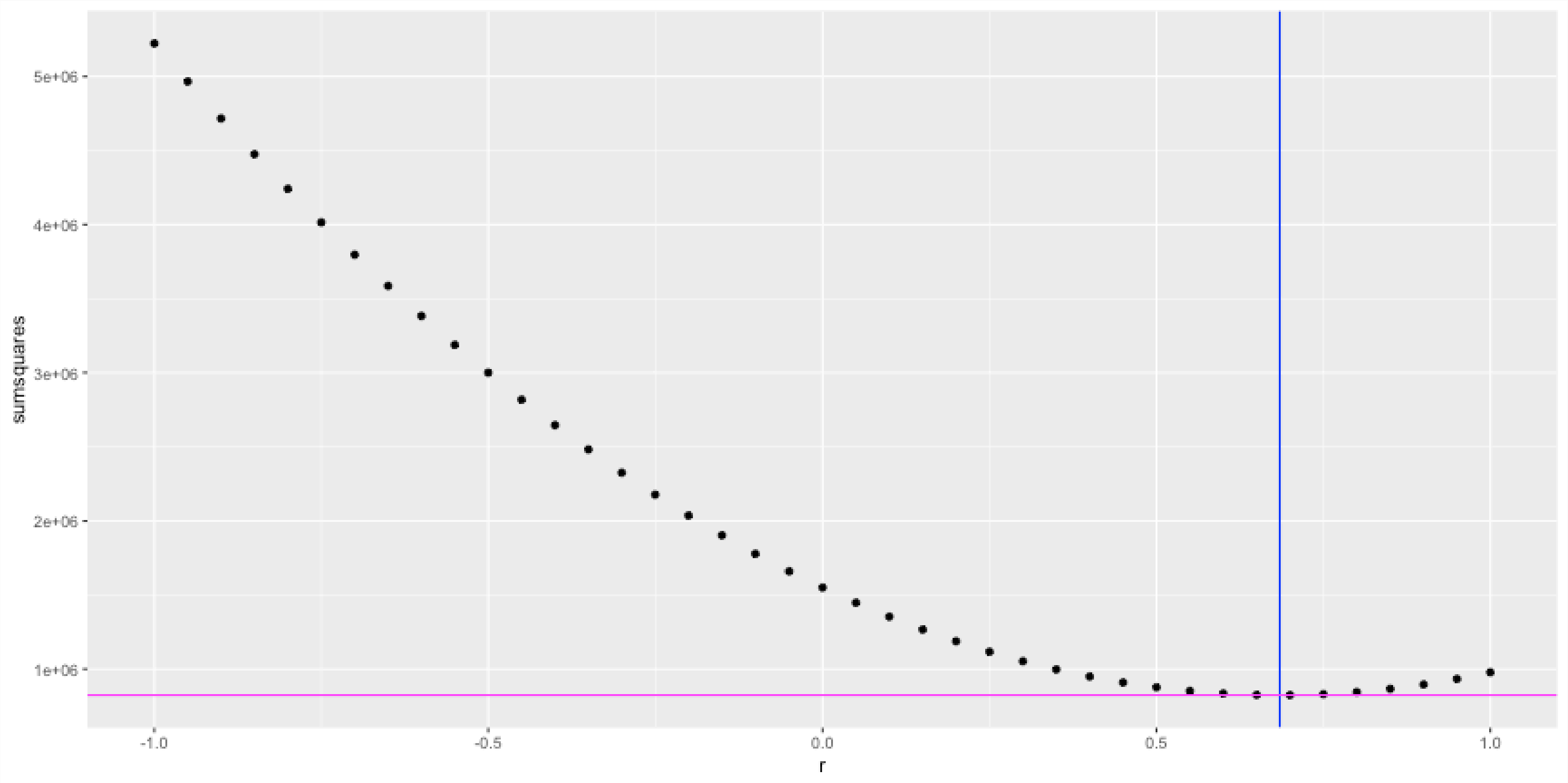
What does it mean to minimize the sum of squared residuals?

To show that $m = r\frac{S_y}{S_x}$ minimizes the sum of squared residuals, this loop will calculate the sum of squared residuals for varying values of between -1 and 1.

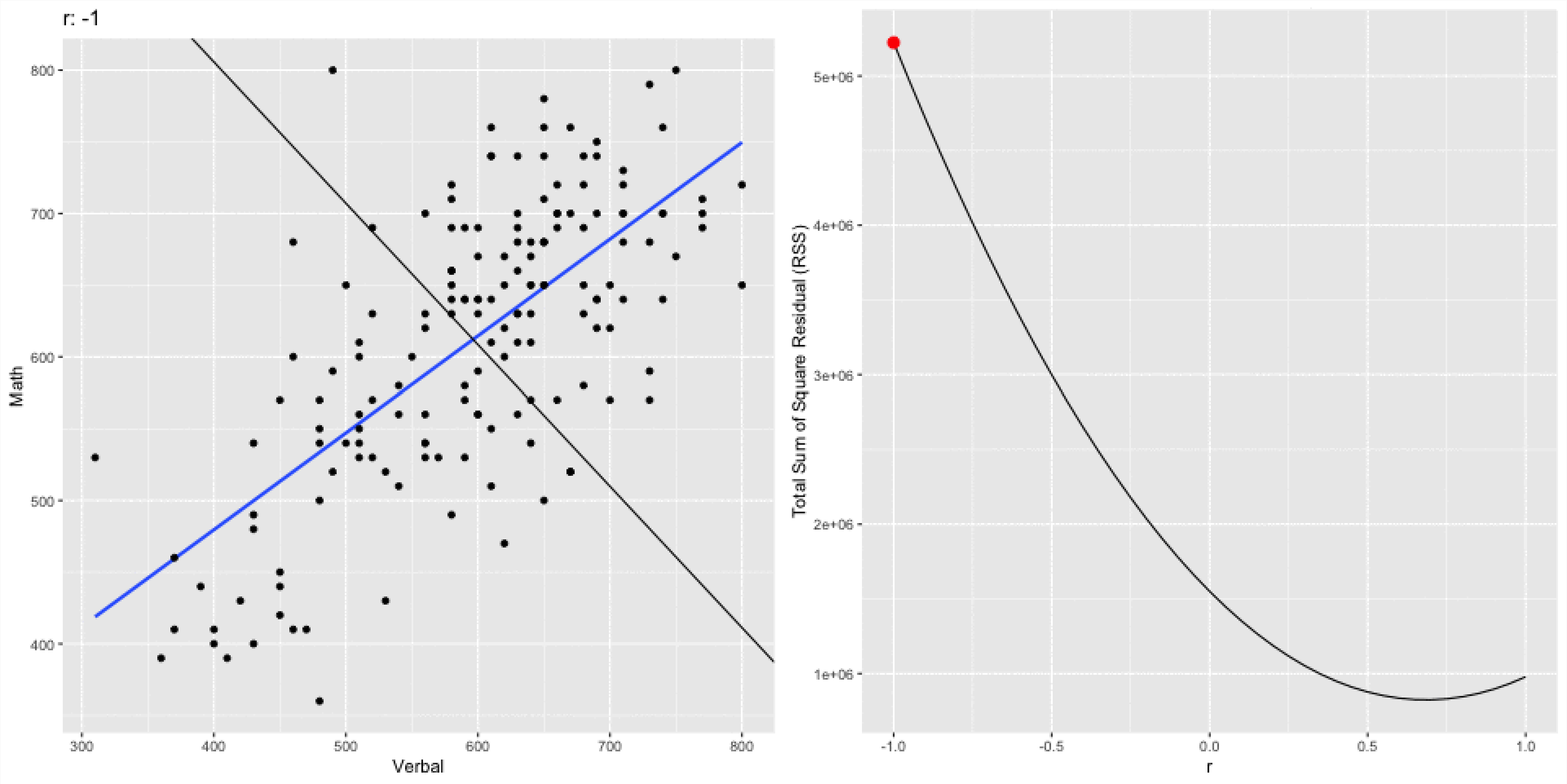
```
results <- data.frame(r=seq(-1, 1, by=.05),
                      m=as.numeric(NA),
                      b=as.numeric(NA),
                      sumsquares=as.numeric(NA))
for(i in 1:nrow(results)) {
    results[i,]$m <- results[i,]$r * (mathSD / verbalSD)
    results[i,]$b <-  mathMean - results[i,]$m * verbalMean
    predicted <- results[i,]$m * sat$Verbal + results[i,]$b
    residual <- sat$Math - predicted
    sumsquares <- sum(residual^2)
    results[i,]$sumsquares <- sum(residual^2)
}
```

Minimizing the Sum of Squared Residuals

Plot the sum of squared residuals for different slopes (i.e. r's). The vertical line corresponds to the r (slope) calcluated above and the horizontal line corresponds the sum of squared residuals for that r. This should have the smallest sum of squared residuals.



Regression Line with RSS

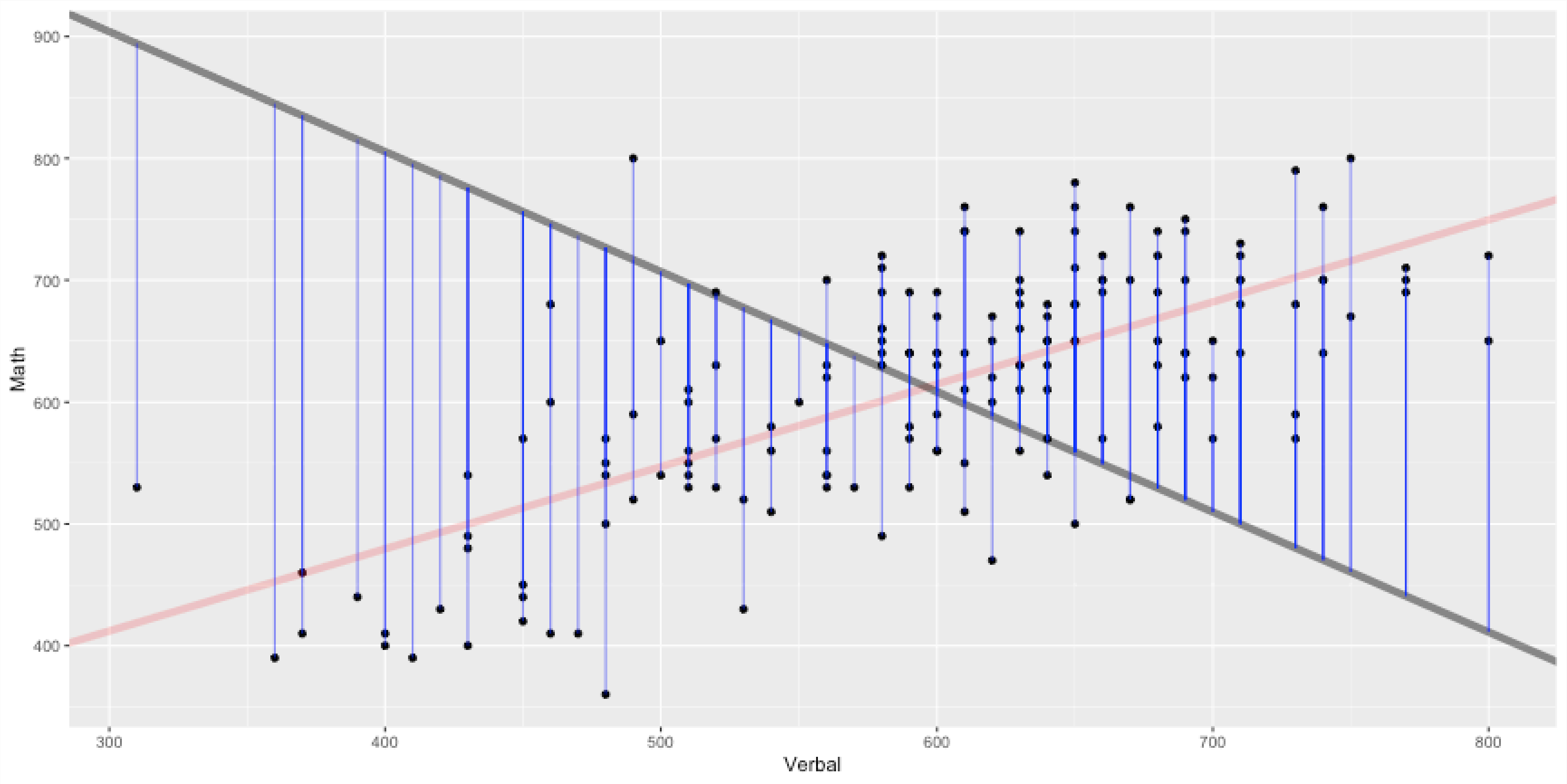


Example of a "bad" model

To exemplify how the residuals change, the following scatter plot picks one of the "bad" models and plot that regression line with the original, best fitting line. Take particular note how the residuals would be less if they ended on the red line (i.e. the better fitting model). This is particularly evident on the far left and far right, but is true across the entire range of values.

```
b.bad <- results[1,]$b
m.bad <- results[1,]$m
sat$predicted.bad <- m.bad * sat$Verbal + b.bad
```

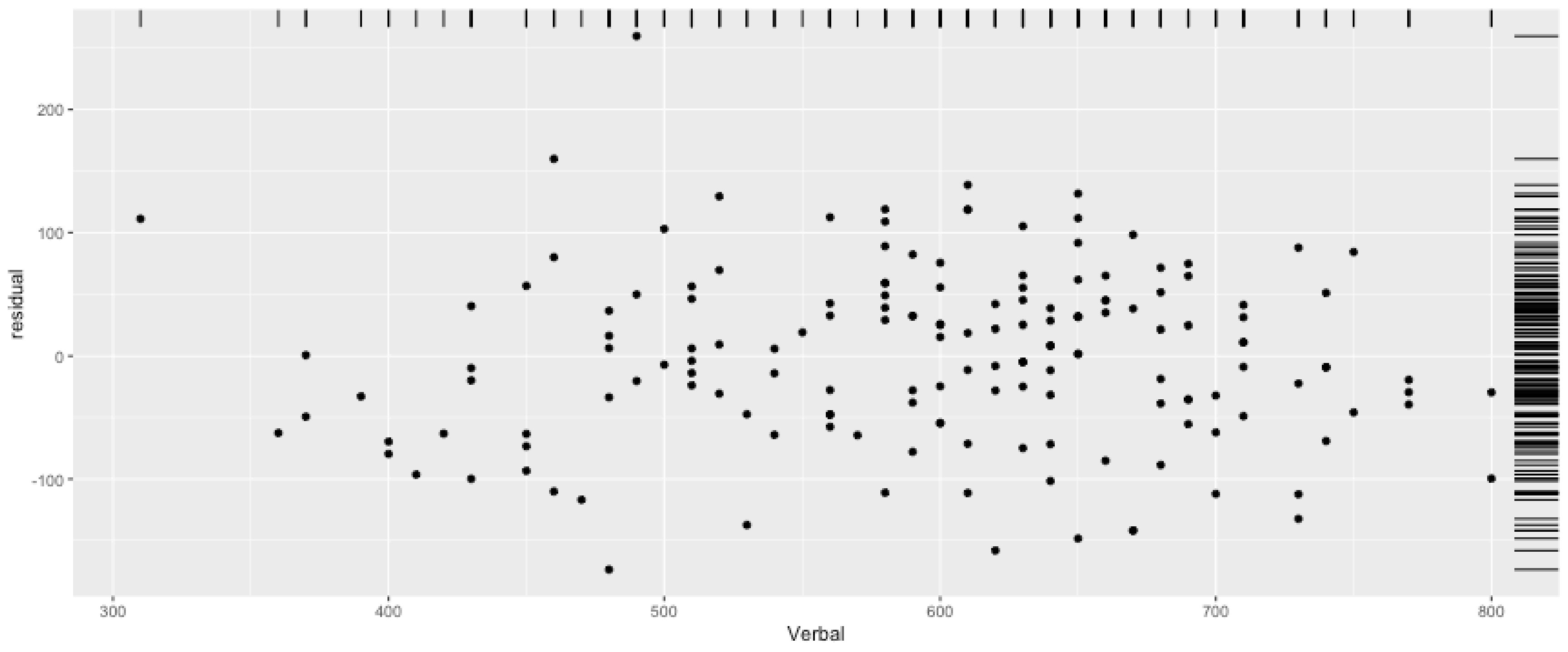
Example of a "bad" model



Residual Plot

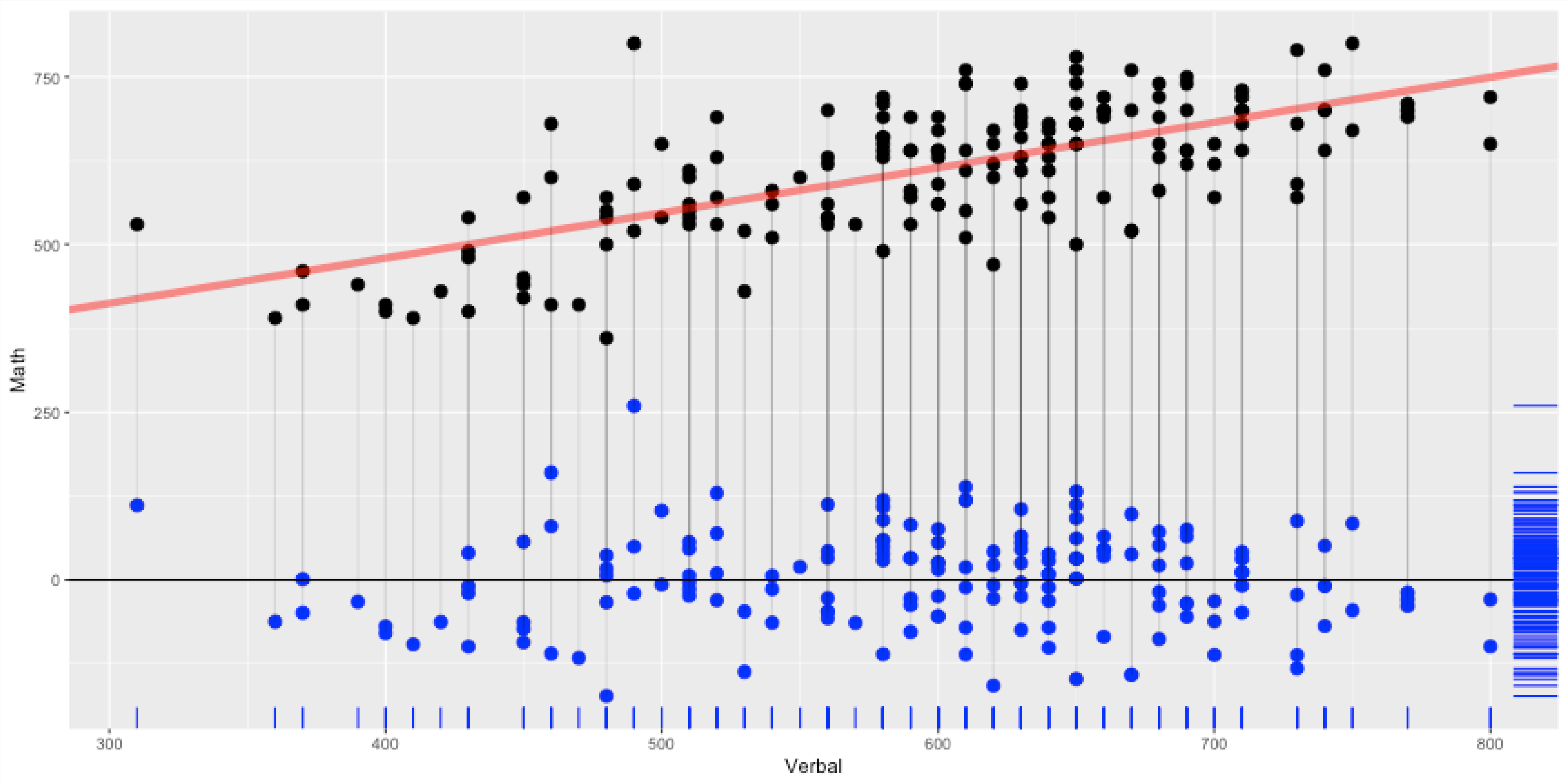
Next, we'll plot the residuals with the independent variable. In this plot we expect to see no pattern, bending, or clustering if the model fits well. The rug plot on the right and top given an indication of the distribution. Below, we will also examine the histogram of residuals.

```
ggplot(sat, aes(x=Verbal, y=residual)) + geom_point() + geom_rug(sides='rt')
```



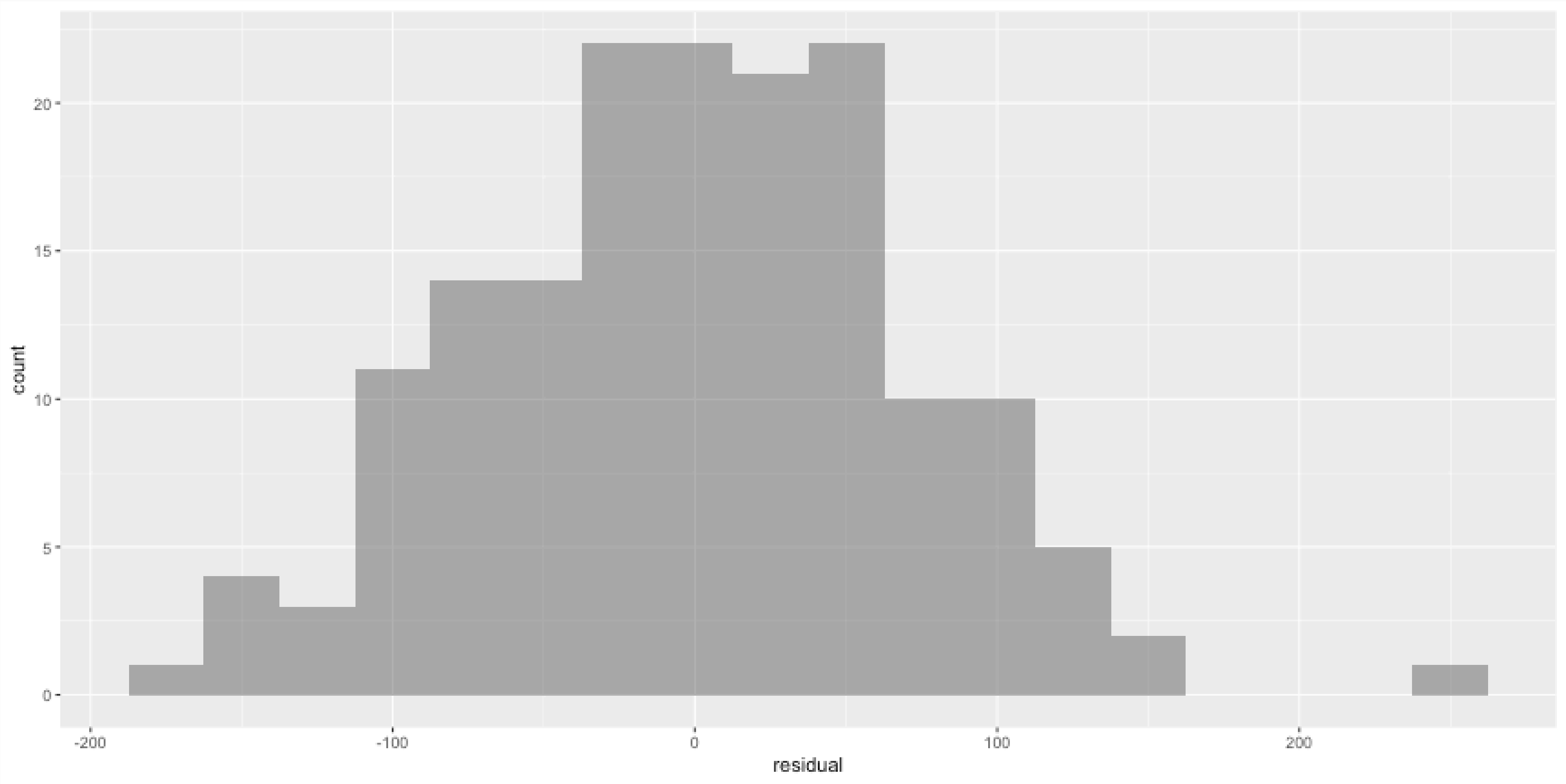
Scatter and Residual Plot, Together

In an attempt to show the relationship between the predicted value and the residuals, this figures combines both the basic scatter plot with the residuals. Each Math score is connected with the corresponding residual point.



Histogram of residuals

```r
ggplot(sat, aes(x=residual)) + geom_histogram(alpha=.5, binwidth=25)
```



Calculate R^2

```r
r ^ 2
```

```
## [1] 0.4686855
```

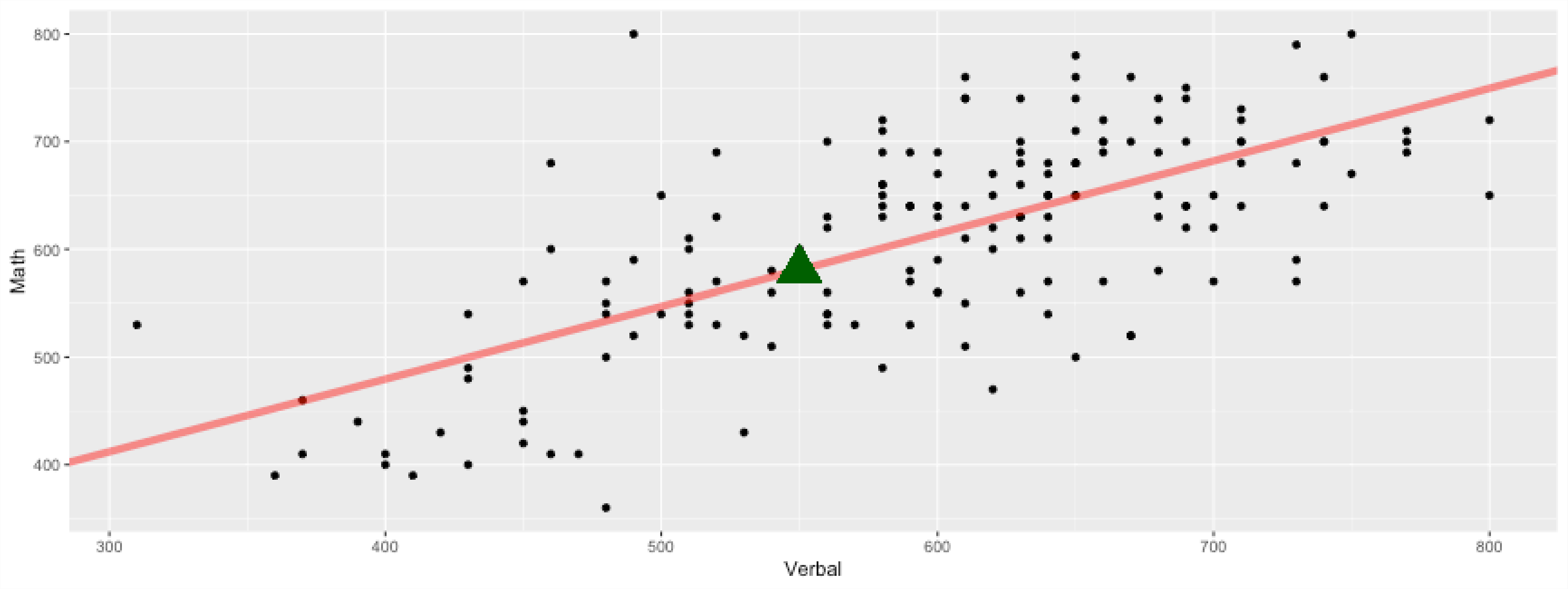
This model accounts for 46.9% of the variance math score predicted from verbal score.

Prediction

Now we can predict Math scores from new Verbal.

```r
newX <- 550
(newY <- newX * m + b)
```

```
## [1] 580.8453
```



Using R's built in function for linear modeling

The `lm` function in R will calculate everything above for us in one command.

```
sat.lm <- lm(Math ~ Verbal, data=sat)
summary(sat.lm)
```

```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -173.590  -47.596    1.158   45.086  259.659
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 209.55417   34.34935   6.101 7.66e-09 ***
## Verbal        0.67507    0.05682  11.880  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 71.75 on 160 degrees of freedom
## Multiple R-squared:  0.4687,    Adjusted R-squared:  0.4654
## F-statistic: 141.1 on 1 and 160 DF,  p-value: < 2.2e-16
```

Predicted Values, Revisited

We can get the predicted values and residuals from the `lm` function

```
sat.lm.predicted <- predict(sat.lm)
sat.lm.residuals <- resid(sat.lm)
```

Confirm that they are the same as what we calculated above.

```
head(cbind(sat.lm.predicted,
           sat$Math.predicted), n=4)
```

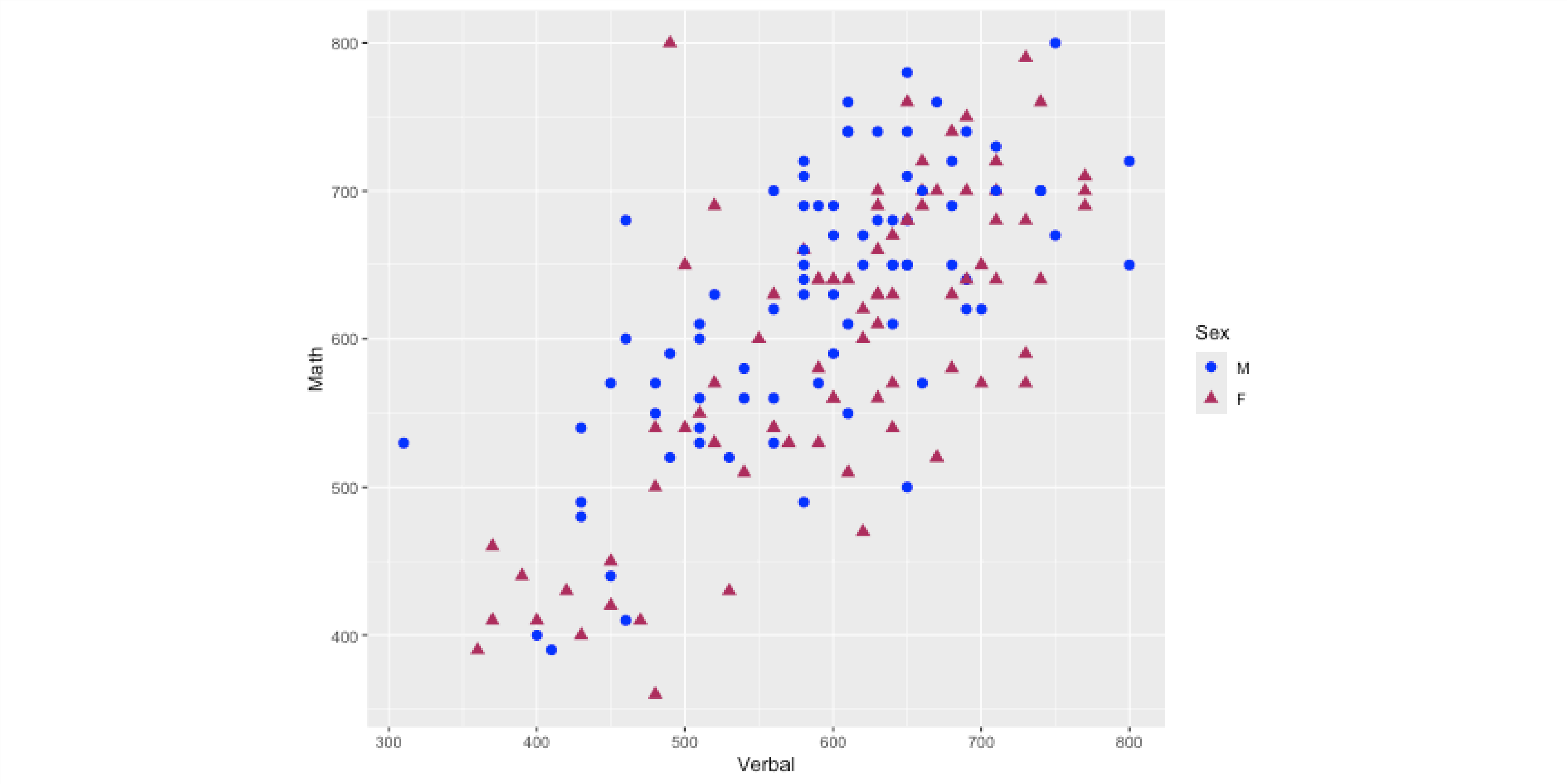
```
##   sat.lm.predicted
## 1         513.3378 513.3378
## 2         641.6020 641.6020
## 3         607.8483 607.8483
## 4         479.5841 479.5841
```

```
head(cbind(sat.lm.residuals,
           sat$residual), n=4)
```

```
##   sat.lm.residuals
## 1        -63.33782  -63.33782
## 2       -101.60203 -101.60203
## 3        -37.84829  -37.84829
## 4        -79.58408  -79.58408
```

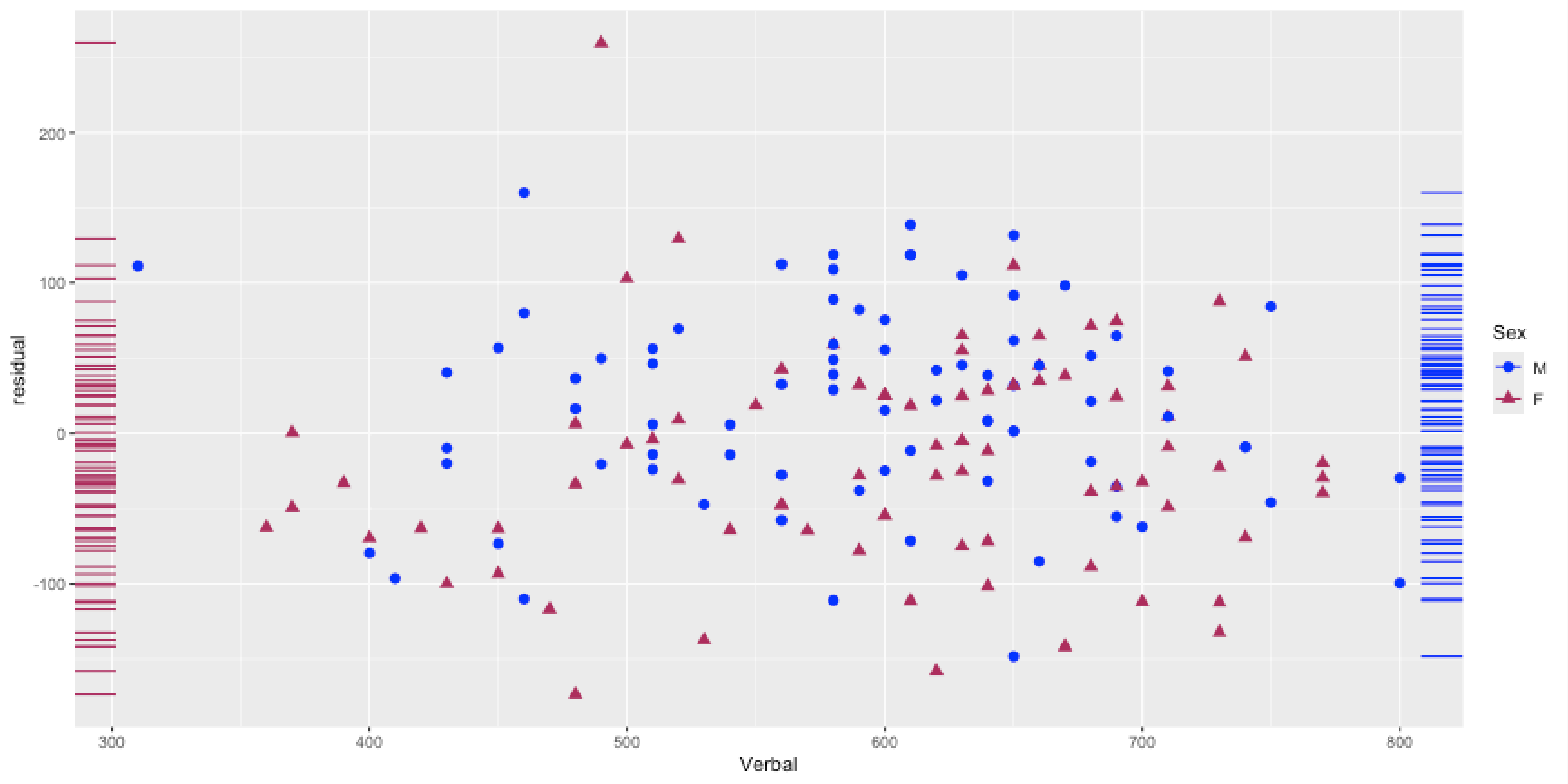
Residuals - Implications for Grouping Variables

First, let's look at the scatter plot but with a sex indicator.



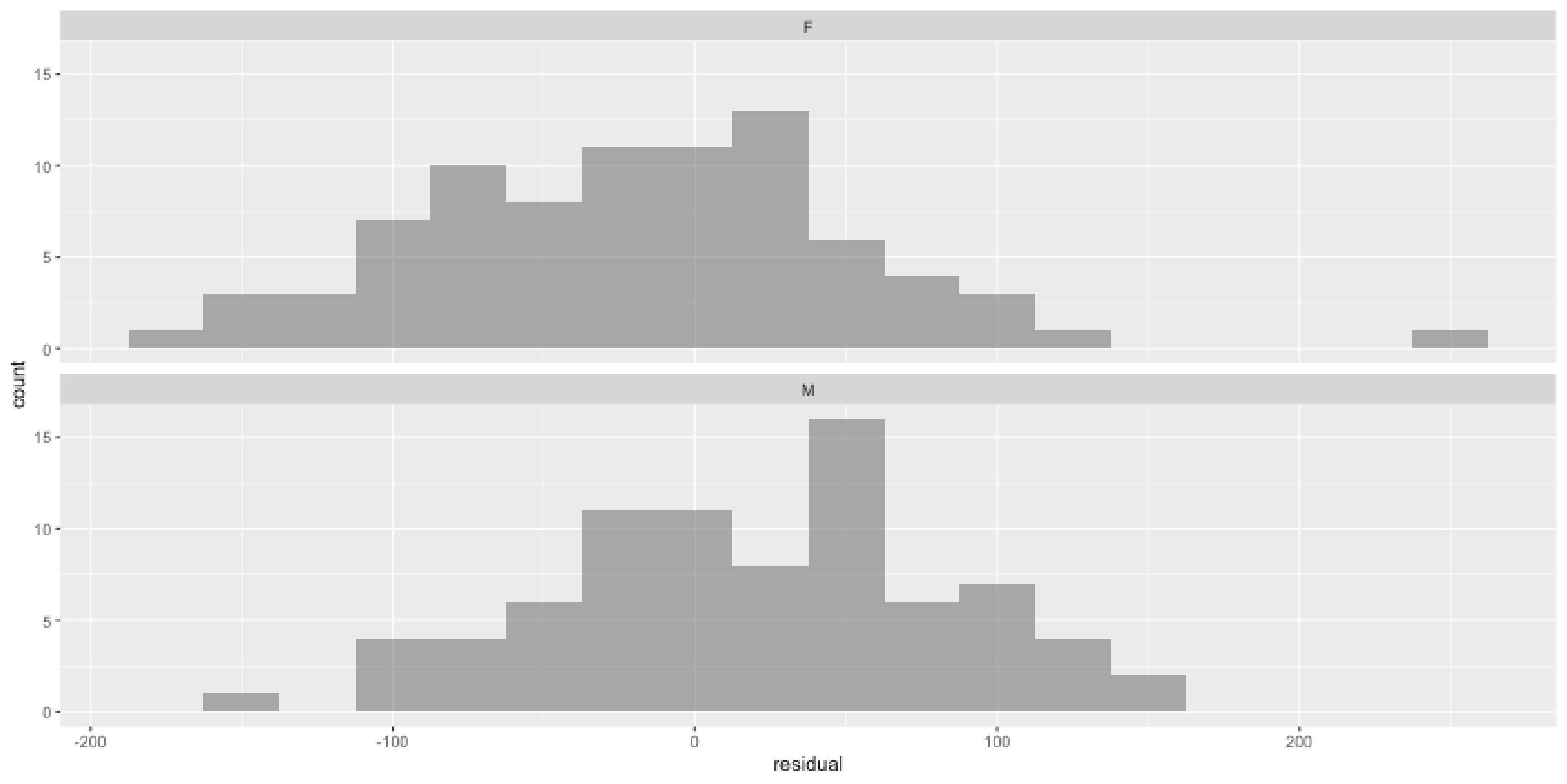
Residual Plot by Sex

And also the residual plot with an indicator for sex.



Histograms

The histograms also show that the distribution are different across sex.



Grouping Variable

Upon careful examination of these two figures, there is some indication there may be a difference between sexes. In the scatter plot, it appears that there is a cluster of males towoards the top left and a cluster of females towards the right. The residual plot also shows a cluster of males on the upper left of the cluster as well as a cluster of females to the lower right. Perhaps estimating two separate models would be more appropriate.

To start, we create two data frames for each sex.

```
sat.male <- sat[sat$Sex == 'M',]
sat.female <- sat[sat$Sex == 'F',]
```

Descriptive Statistics

Calculate the mean for Math and Verbal for both males and females.

```
(male.verbal.mean <- mean(sat.male$Verbal))
```

```
## [1] 590.375
```

```
(male.math.mean <- mean(sat.male$Math))
```

```
## [1] 626.875
```

```
(female.verbal.mean <- mean(sat.female$Verbal))
```

```
## [1] 602.0732
```

```
(female.math.mean <- mean(sat.female$Math))
```

```
## [1] 597.6829
```

Two Regression Models

Estimate two linear models for each sex.

```
sat.male.lm <- lm(Math ~ Verbal,
                  data=sat.male)
sat.male.lm
```

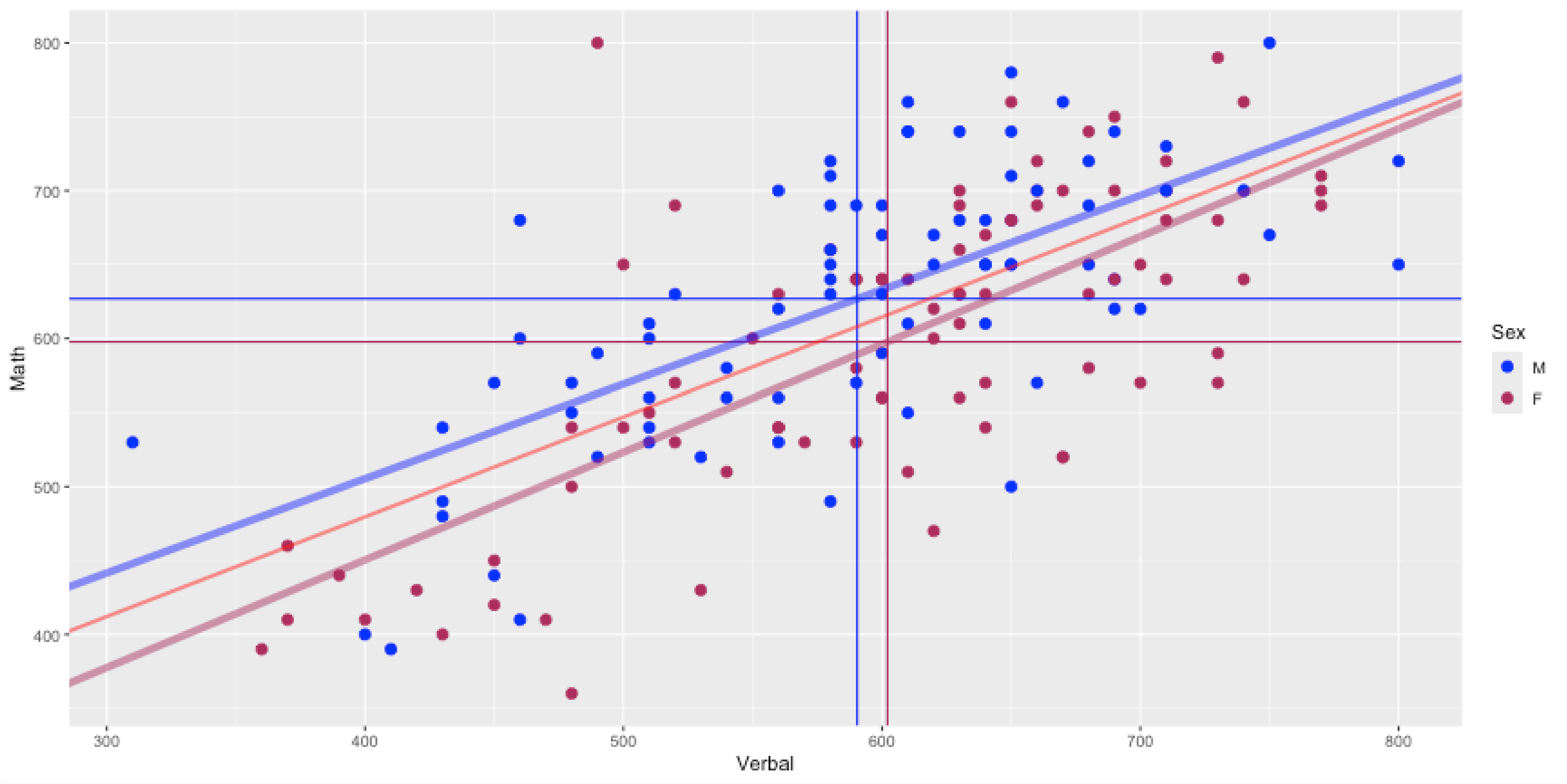
```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat.male)
##
## Coefficients:
## (Intercept)       Verbal
##    250.1452       0.6381
```

```
sat.female.lm <- lm(Math ~ Verbal,
                    data=sat.female)
sat.female.lm
```

```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat.female)
##
## Coefficients:
## (Intercept)       Verbal
##    158.9965       0.7286
```

Two Regression Models Visualized

We do in fact find that the intercepts and slopes are both fairly different. The figure below adds the regression lines to the scatter plot.



R^2

Let's compare the R^2 for the three models.

```
cor(sat$Verbal, sat$Math) ^ 2
```

```
## [1] 0.4686855
```

```
cor(sat.male$Verbal, sat.male$Math) ^ 2
```

```
## [1] 0.4710744
```

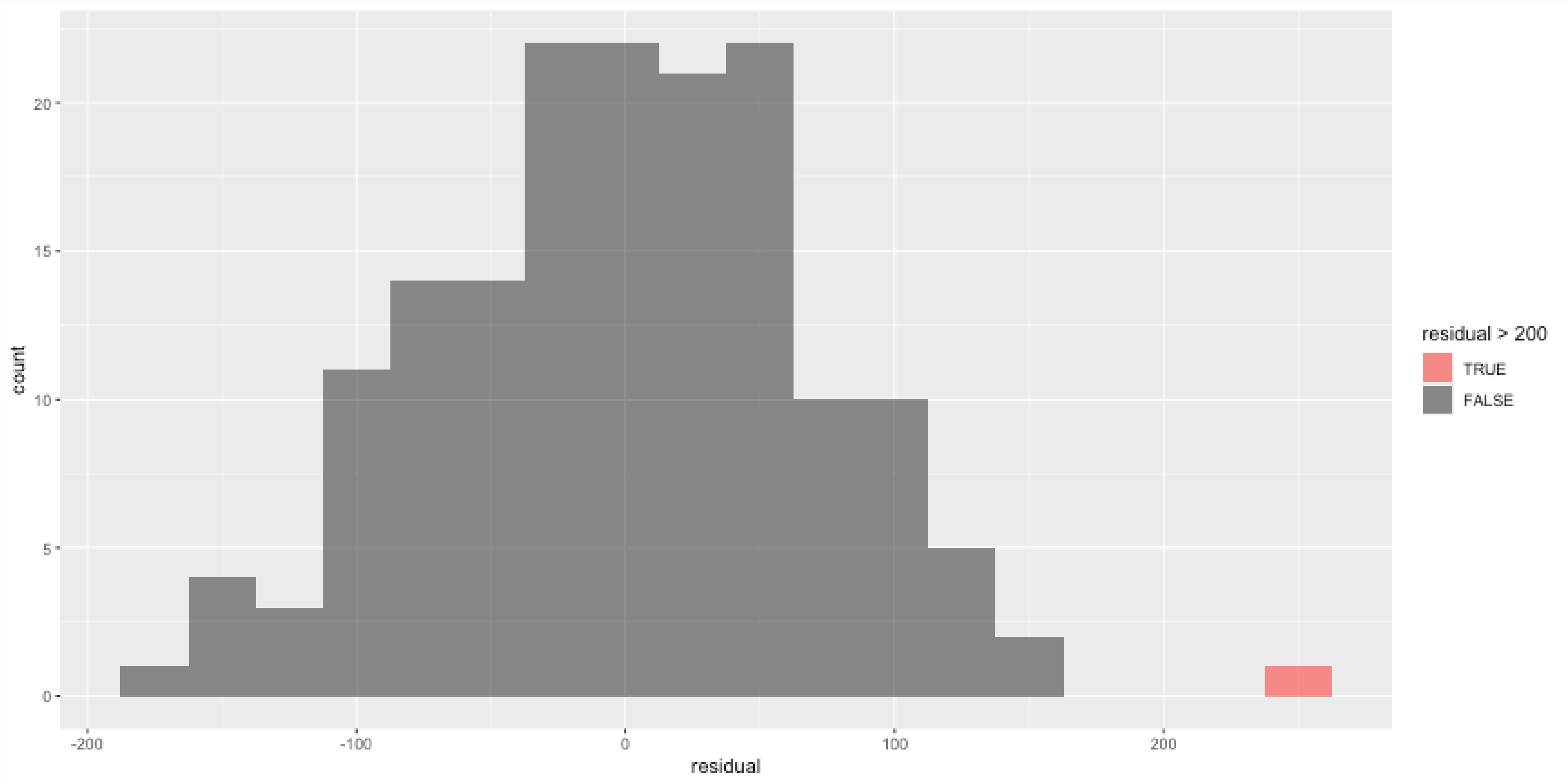
```
cor(sat.female$Verbal, sat.female$Math) ^ 2
```

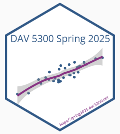
```
## [1] 0.5137626
```

The R^2 for the full model accounts for approximately 46.9% of the variance. By estimating separate models for each sex we can account for 47.1% and 51.4% of the variance for males and females, respectively.

Examining Possible Outliers

Re-examining the histogram of residuals, there is one data point with a residual higher than the rest. This is a possible outlier. In this section we'll examine how that outlier may impact our linear model.



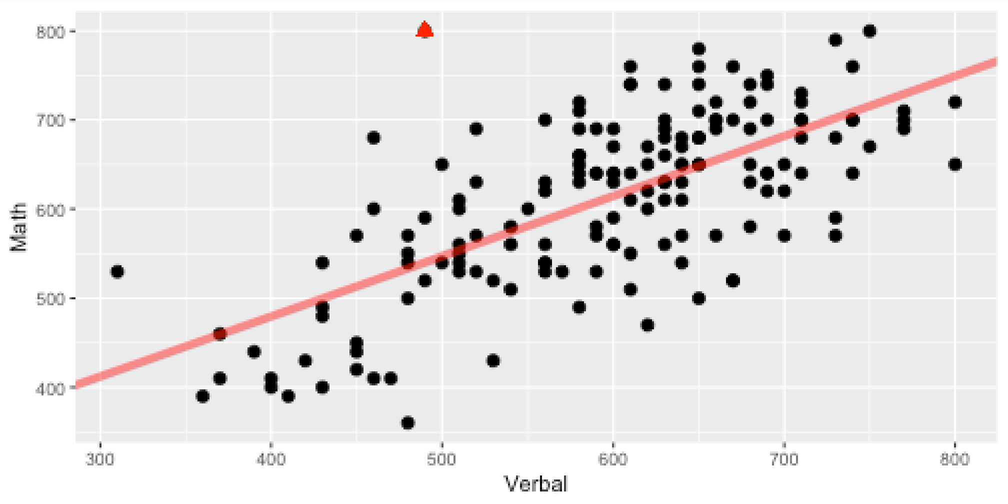


Possible Outlier

We can extract that record from our data frame. We can also highlight that point on the scatter plot.

```
sat.outlier <- sat[sat$residual > 200,]
sat.outlier
```

```
##      Verbal Math Sex   Verbal.z   Math.z Math.predicted Math.predicted.z residual residual.z predicted.bad
## 162    490  800   F -1.068091 1.914735       540.3408       -0.7210412 259.6592   2.635776      716.9152
```



Possible Outlier (cont.)

We see that excluding this point changes model slightly. With the outlier included we can account for 45.5% of the variance and by excluding it we can account for 47.9% of the variance. Although excluding this point improves our model, this is an insufficient enough reason to do so. Further explanation is necessary.

```r
(sat.lm <- lm(Math ~ Verbal, data=sat))
```

```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat)
##
## Coefficients:
## (Intercept)       Verbal
##    209.5542       0.6751
```

```r
(sat.lm2 <- lm(Math ~ Verbal,
               data=sat[sat$residual < 200,]))
```

```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat[sat$residual
##
## Coefficients:
## (Intercept)       Verbal
##    197.4697       0.6926
```

R^2 with and without the outlier

```
summary(sat.lm)$r.squared
```

```
## [1] 0.4686855
```

```
summary(sat.lm2)$r.squared
```

```
## [1] 0.5013288
```

More outliers

For the following two examples, we will add outliers to examine how they would effect our models. In the first example, we will add an outlier that is close to our fitted model (i.e. a small residual) but lies far away from the cluster of points. As we can see below, this single point increases our R^2 by more than 5%.

```r
outX <- 1200
outY <- 1150
sat.outlier <- rbind(sat[,c('Verbal','Math')], c(Verbal=outX, Math=outY))
```

Regression Models

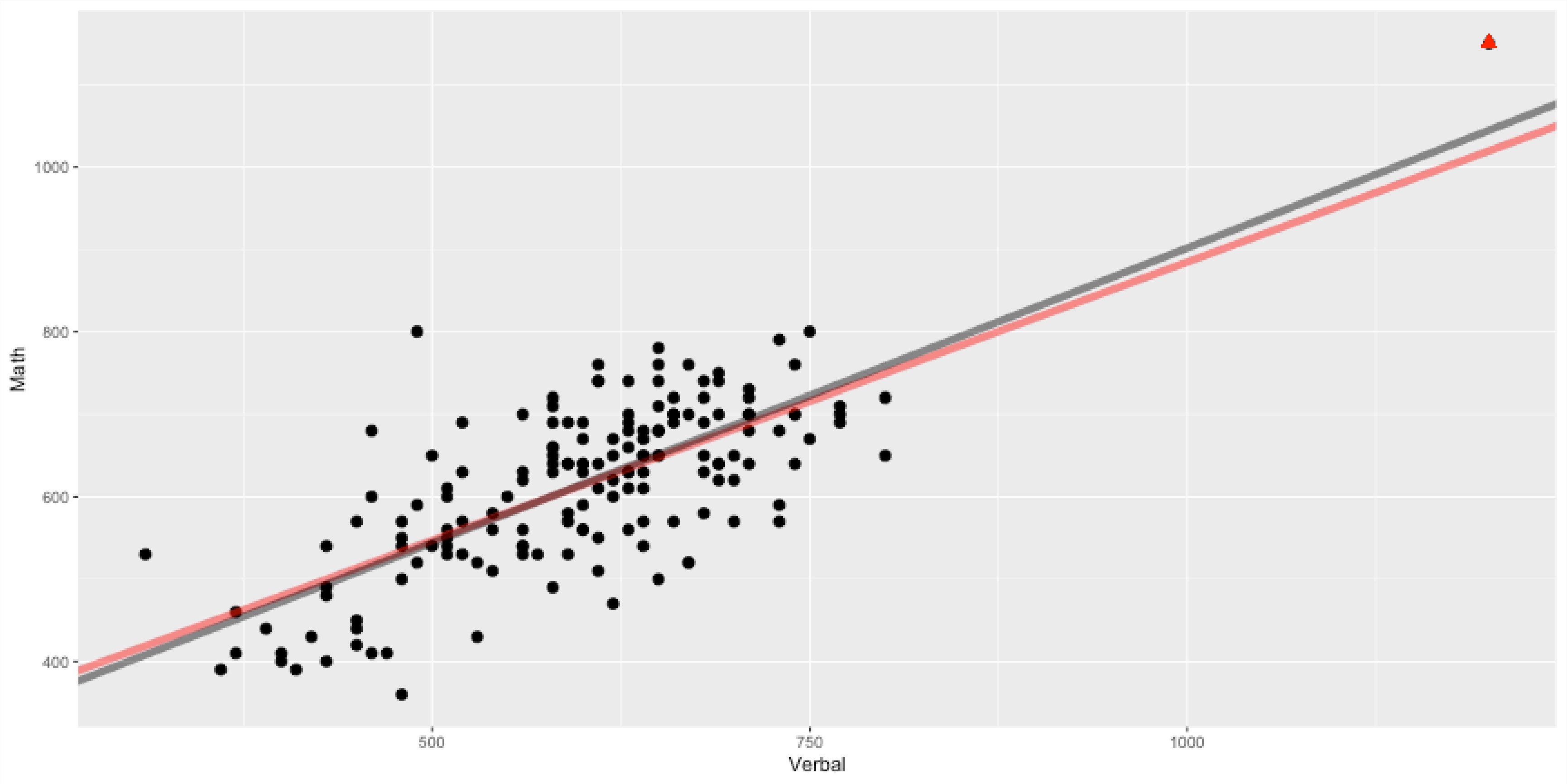
```
(sat.lm <- lm(Math ~ Verbal,
              data=sat))
```

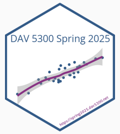
```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat)
##
## Coefficients:
## (Intercept)       Verbal
##    209.5542       0.6751
```

```
(sat.lm2 <- lm(Math ~ Verbal,
               data=sat.outlier))
```

```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat.outlier)
##
## Coefficients:
## (Intercept)       Verbal
##     186.372        0.715
```

Scatter Plot





R^2

```r
summary(sat.lm)$r.squared
```

```
## [1] 0.4686855
```

```r
summary(sat.lm2)$r.squared
```

```
## [1] 0.5443222
```

Outliers

Outliers can have the opposite effect too. In this example, our R^2 is decreased by almost 16%.

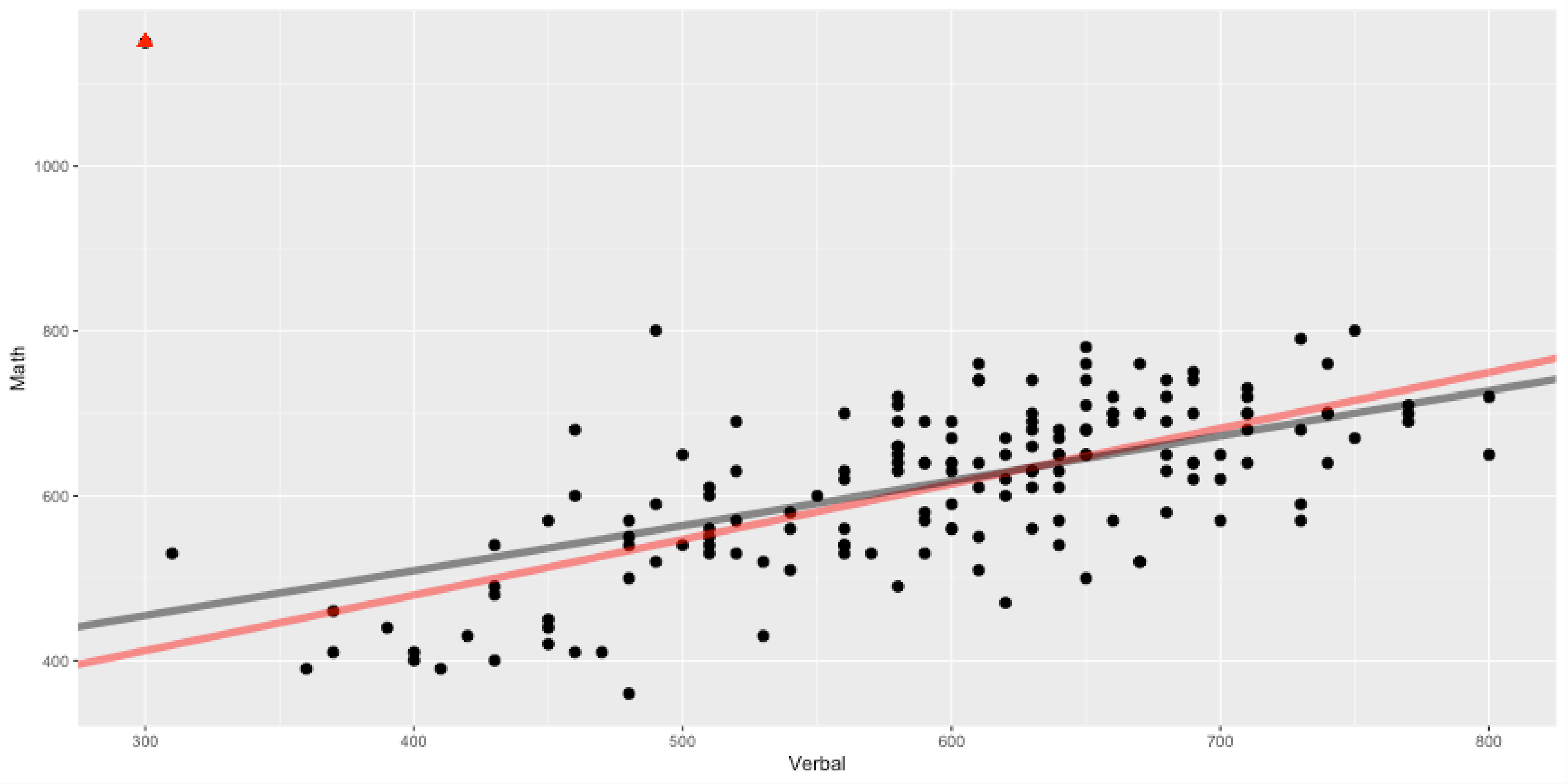
```r
outX <- 300
outY <- 1150
sat.outlier <- rbind(sat[,c('Verbal','Math')], c(Verbal=outX, Math=outY))
```

```r
(sat.lm <- lm(Math ~ Verbal,
              data=sat))
```

```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat)
##
## Coefficients:
## (Intercept)       Verbal
##    209.5542       0.6751
```

```r
(sat.lm2 <- lm(Math ~ Verbal,
               data=sat.outlier))
```

```
##
## Call:
## lm(formula = Math ~ Verbal, data = sat.outlier)
##
## Coefficients:
## (Intercept)       Verbal
##    290.8915       0.5459
```



1000
800
600
400
Math
300
400
500
600
700
800
Verbal

R^2

```
summary(sat.lm)$r.squared
```

```
## [1] 0.4686855
```

```
summary(sat.lm2)$r.squared
```

```
## [1] 0.2726476
```

NYS Report Card

NYS publishes data for each school in the state. We will look at the grade 8 math scores for 2012 and 2013. 2013 was the first year the tests were aligned with the Common Core Standards. There was a lot of press about how the passing rates for most schools dropped. Two questions we wish to answer:

1. Did the passing rates drop in a predictable manner?
2. Were the drops different for charter and public schools?

```
load('../course_data/NYSReportCard-Grade7Math.Rda')
names(reportCard)
```

```
##  [1] "BEDSCODE"      "School"        "NumTested2012" "Mean2012"      "Pass2012"      "Charter"       "GradeSubject
##  [8] "County"        "BOCES"         "NumTested2013" "Mean2013"      "Pass2013"
```

`reportCard` Data Frame

Show 3 entries   Search:

BEDSCODE	School	NumTested2012	Mean2012	Pass2012	Charter	GradeSubject	County	BOCES	NumTested2013	Mean2013	Pa
010100010020	NORTH ALBANY ACADEMY	47	649	13	false	Grade 7 Math	Albany	BOCES ALBANY-SCHOH-SCHENECTADY-SARAT	45	268	
010100010030	WILLIAM S HACKETT MIDDLE SCHOOL	212	652	30	false	Grade 7 Math	Albany	BOCES ALBANY-SCHOH-SCHENECTADY-SARAT	250	279	
010100010045	STEPHEN AND HARRIET MYERS MIDDLE SCHOOL	262	670	50	false	Grade 7 Math	Albany	BOCES ALBANY-SCHOH-SCHENECTADY-SARAT	256	284	

Showing 1 to 3 of 1,362 entries

Previous 1 2 3 4 5 … 454 Next

Descriptive Statistics

```
summary(reportCard$Pass2012)
```

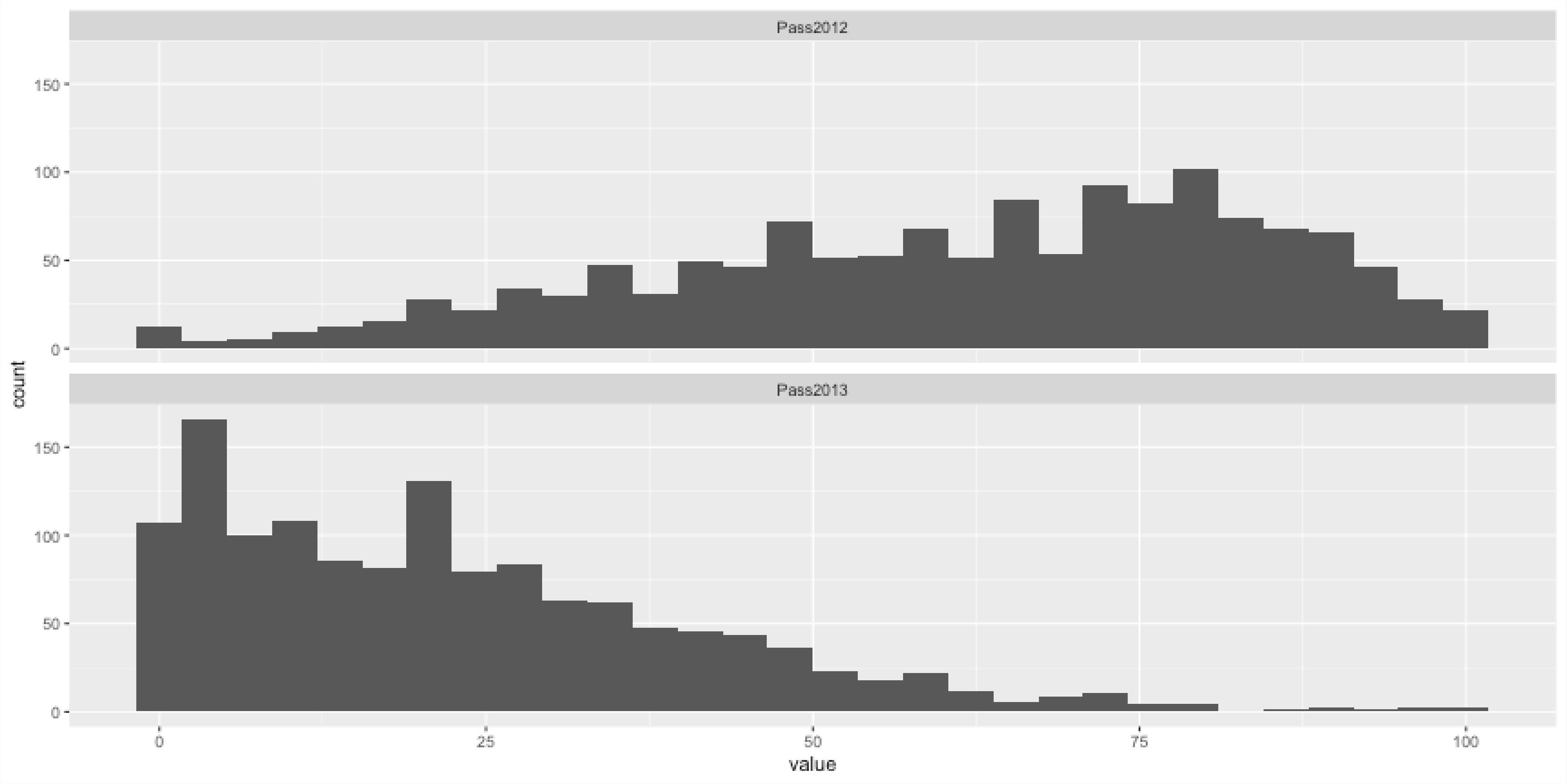
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    0.00   46.00   65.00   61.73   80.00  100.00
```

```
summary(reportCard$Pass2013)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    0.00    7.00   20.00   22.83   33.00   99.00
```

Histograms

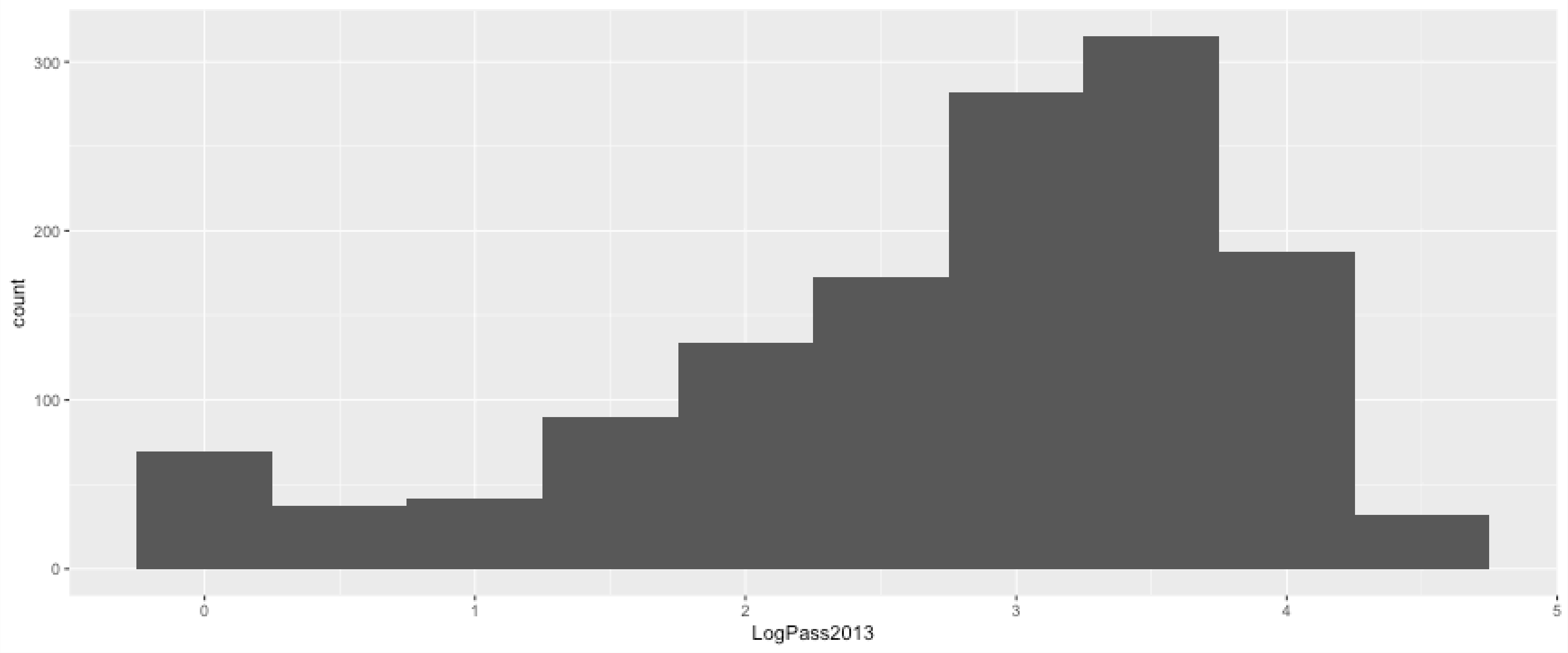
```
melted <- melt(reportCard[,c('Pass2012', 'Pass2013')])
ggplot(melted, aes(x=value)) + geom_histogram() + facet_wrap(~ variable, ncol=1)
```



Log Transformation

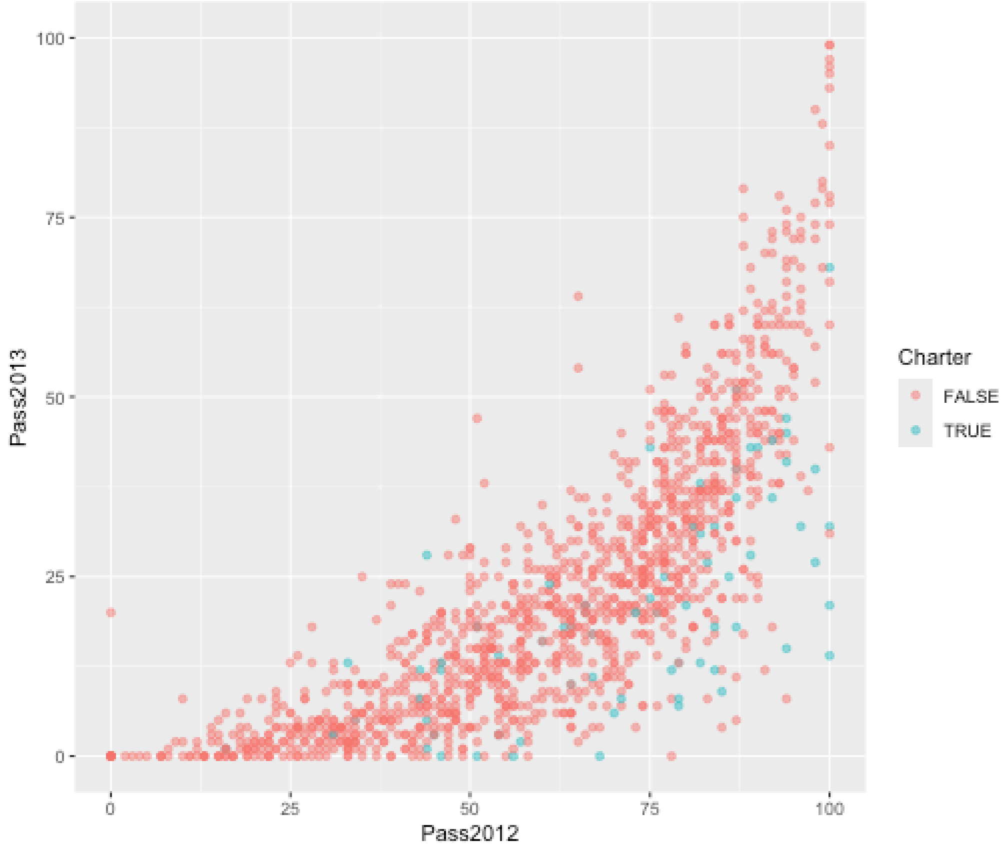
Since the distribution of the 2013 passing rates is skewed, we can log transfor that variable to get a more reasonably normal distribution.

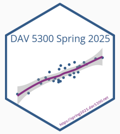
```
reportCard$LogPass2013 <- log(reportCard$Pass2013 + 1)
ggplot(reportCard, aes(x=LogPass2013)) + geom_histogram(binwidth=0.5)
```



Scatter Plot

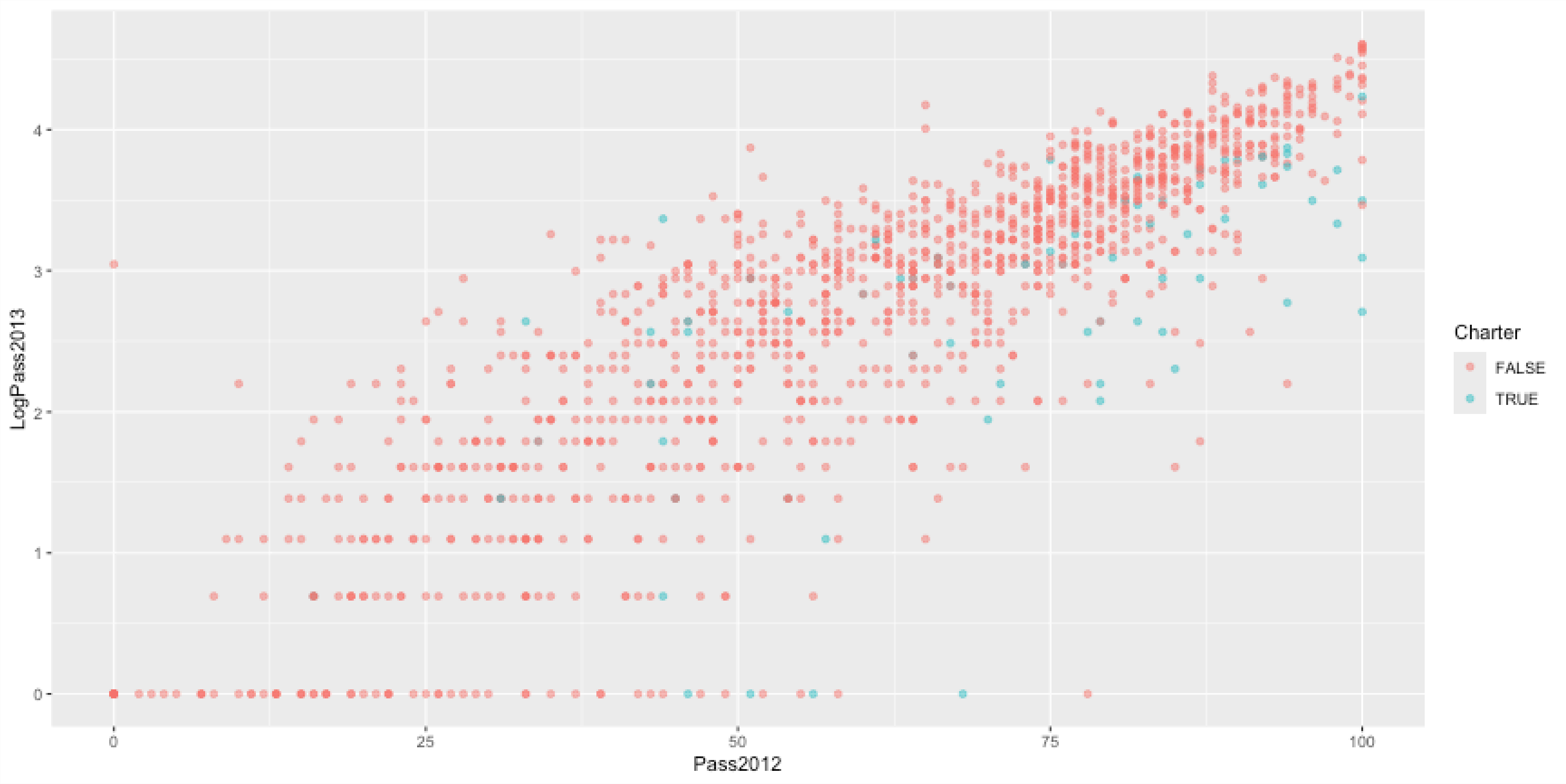
```
ggplot(reportCard, aes(x=Pass2012, y=Pass2013, color=Charter)) +
    geom_point(alpha=0.5) + coord_equal() + ylim(c(0,100)) + xlim(c(0,100))
```

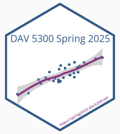




Scatter Plot (log transform)

```r
ggplot(reportCard, aes(x=Pass2012, y=LogPass2013, color=Charter)) +
    geom_point(alpha=0.5) + xlim(c(0,100)) + ylim(c(0, log(101)))
```





Correlation

```
cor.test(reportCard$Pass2012, reportCard$Pass2013)
```

```
##
##     Pearson's product-moment correlation
##
## data:  reportCard$Pass2012 and reportCard$Pass2013
## t = 47.166, df = 1360, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.7667526 0.8071276
## sample estimates:
##       cor
## 0.7877848
```

Correlation (log transform)

```
cor.test(reportCard$Pass2012, reportCard$LogPass2013)
```

```
##
##      Pearson's product-moment correlation
##
## data:  reportCard$Pass2012 and reportCard$LogPass2013
## t = 56.499, df = 1360, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.8207912 0.8525925
## sample estimates:
##       cor
## 0.8373991
```

Linear Regression

```
lm.out <- lm(Pass2013 ~ Pass2012, data=reportCard)
summary(lm.out)
```

```
##
## Call:
## lm(formula = Pass2013 ~ Pass2012, data = reportCard)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -35.484  -6.878  -0.478   5.965  51.675
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -16.68965    0.89378  -18.67   <2e-16 ***
## Pass2012      0.64014    0.01357   47.17   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.49 on 1360 degrees of freedom
## Multiple R-squared:  0.6206,	Adjusted R-squared:  0.6203
## F-statistic:  2225 on 1 and 1360 DF,  p-value: < 2.2e-16
```

Linear Regression (log transform)

```
lm.log.out <- lm(LogPass2013 ~ Pass2012, data=reportCard)
summary(lm.log.out)
```

```
##
## Call:
## lm(formula = LogPass2013 ~ Pass2012, data = reportCard)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3880 -0.2531  0.0776  0.3461  2.7368
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.307692   0.046030   6.685 3.37e-11 ***
## Pass2012    0.039491   0.000699  56.499  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5915 on 1360 degrees of freedom
## Multiple R-squared:  0.7012,    Adjusted R-squared:  0.701
## F-statistic:  3192 on 1 and 1360 DF,  p-value: < 2.2e-16
```

Did the passing rates drop in a predictable manner?

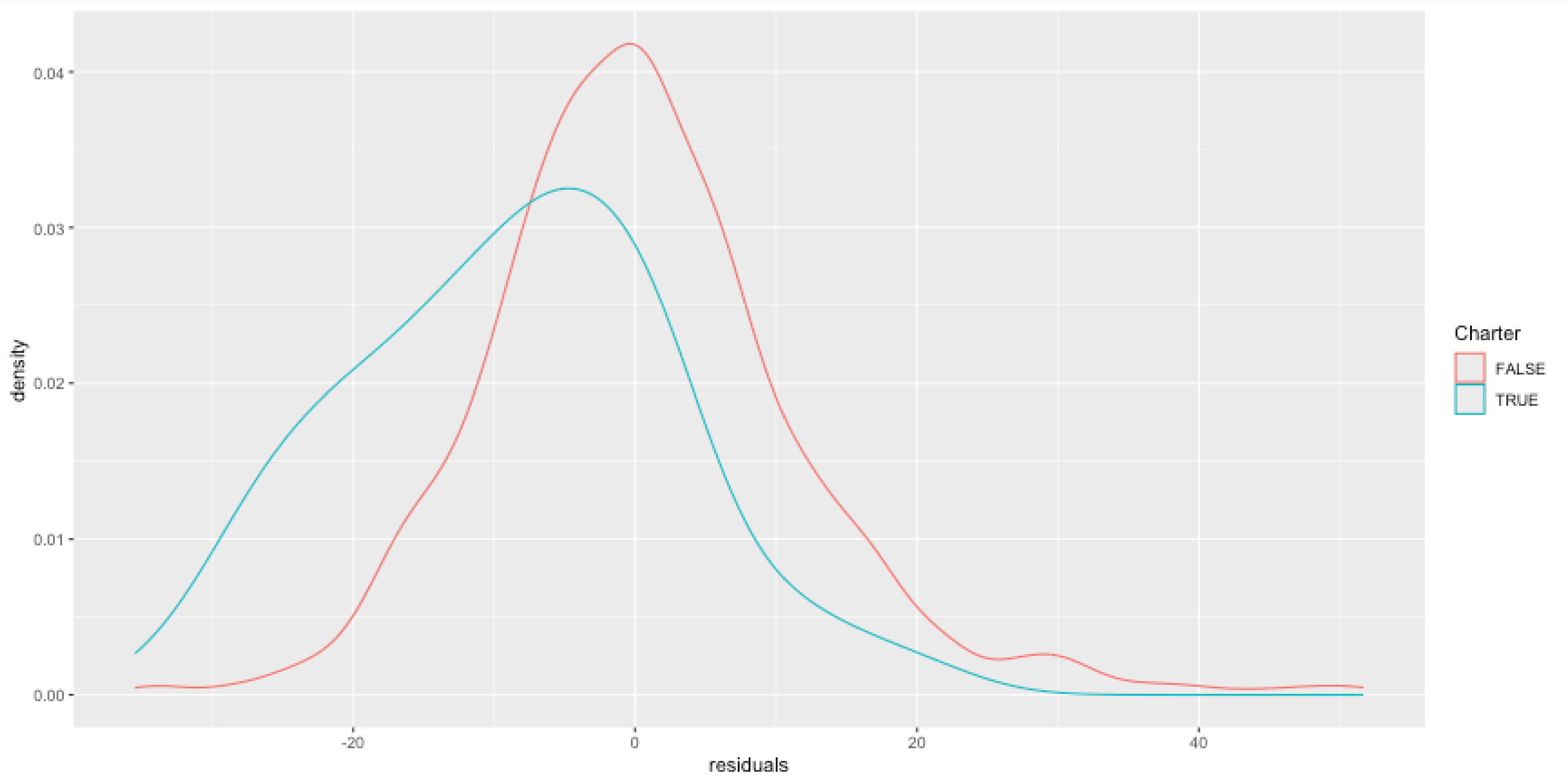
Yes! Whether we log tranform the data or not, the correlations are statistically significant with regression models with R^2 creater than 62%.

To answer the second question, whether the drops were different for public and charter schools, we'll look at the residuals.

```
reportCard$residuals <- resid(lm.out)
reportCard$residualsLog <- resid(lm.log.out)
```

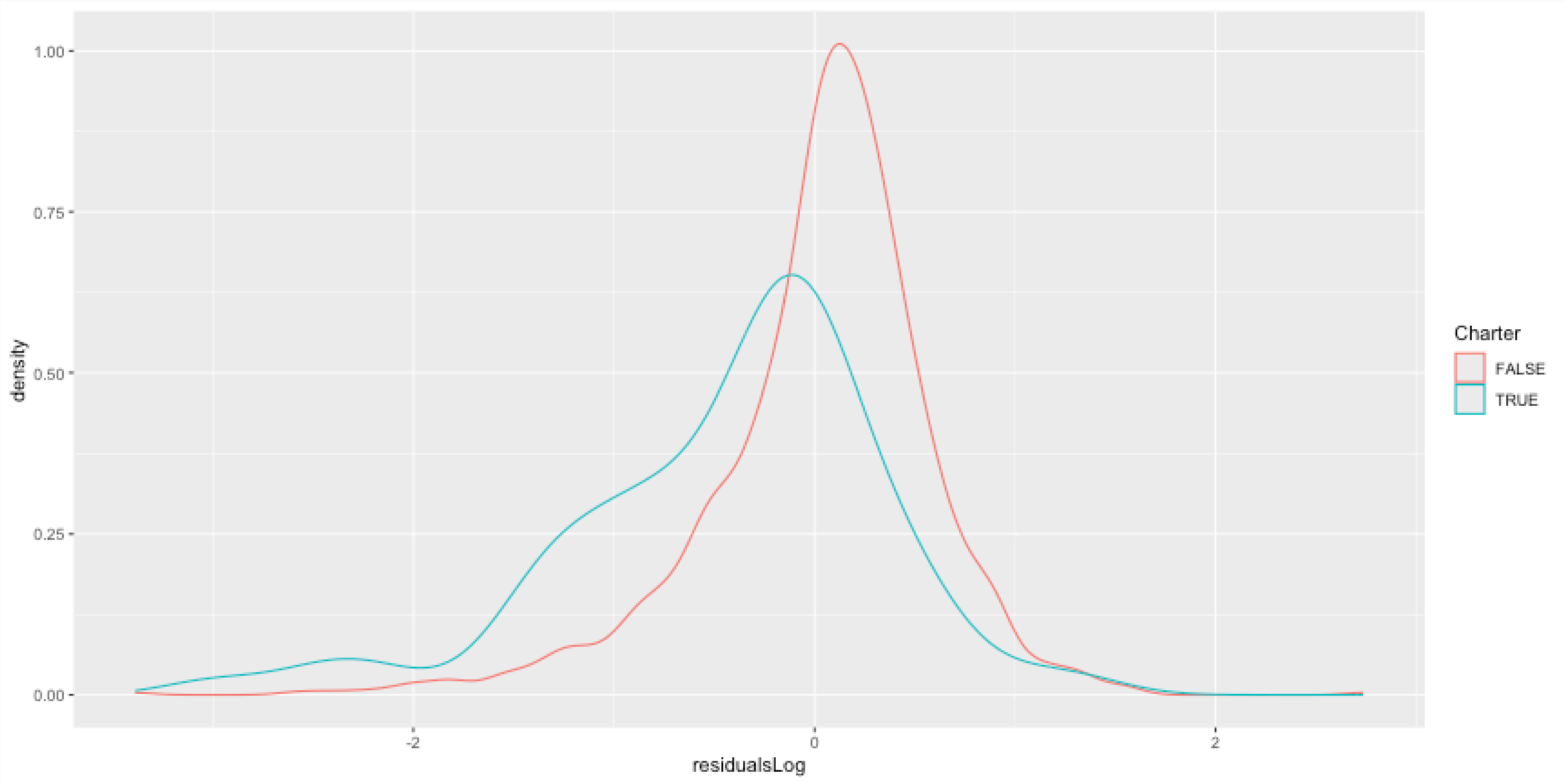
Distribution of Residuals

```
ggplot(reportCard, aes(x=residuals, color=Charter)) + geom_density()
```



Distribution of Residuals

```
ggplot(reportCard, aes(x=residualsLog, color=Charter)) + geom_density()
```



Null Hypothesis Testing

H_0: There is no difference in the residuals between charter and public schools.

H_A: There is a difference in the residuals between charter and public schools.

```
t.test(residuals ~ Charter, data=reportCard)
```

```
##
##      Welch Two Sample t-test
##
## data:  residuals by Charter
## t = 6.5751, df = 77.633, p-value = 5.091e-09
## alternative hypothesis: true difference in means between group FALSE and group TRUE is not equal to 0
## 95 percent confidence interval:
##   6.411064 11.980002
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            0.479356           -8.716177
```

Null Hypothesis Testing (log transform)

```
t.test(residualsLog ~ Charter, data=reportCard)
```

```
##
##      Welch Two Sample t-test
##
## data:  residualsLog by Charter
## t = 4.7957, df = 74.136, p-value = 8.161e-06
## alternative hypothesis: true difference in means between group FALSE and group TRUE is not equal to 0
## 95 percent confidence interval:
##  0.2642811 0.6399761
## sample estimates:
## mean in group FALSE  mean in group TRUE
##          0.02356911         -0.42855946
```

Polynomial Models (e.g. Quadratic)

It is possible to fit quatric models fairly easily in R, say of the following form:

$$y = b_1 x^2 + b_2 x + b_0$$

```
quad.out <- lm(Pass2013 ~ I(Pass2012^2) + Pass2012, data=reportCard)
summary(quad.out)$r.squared
```

```
## [1] 0.7065206
```

```
summary(lm.out)$r.squared
```

```
## [1] 0.6206049
```

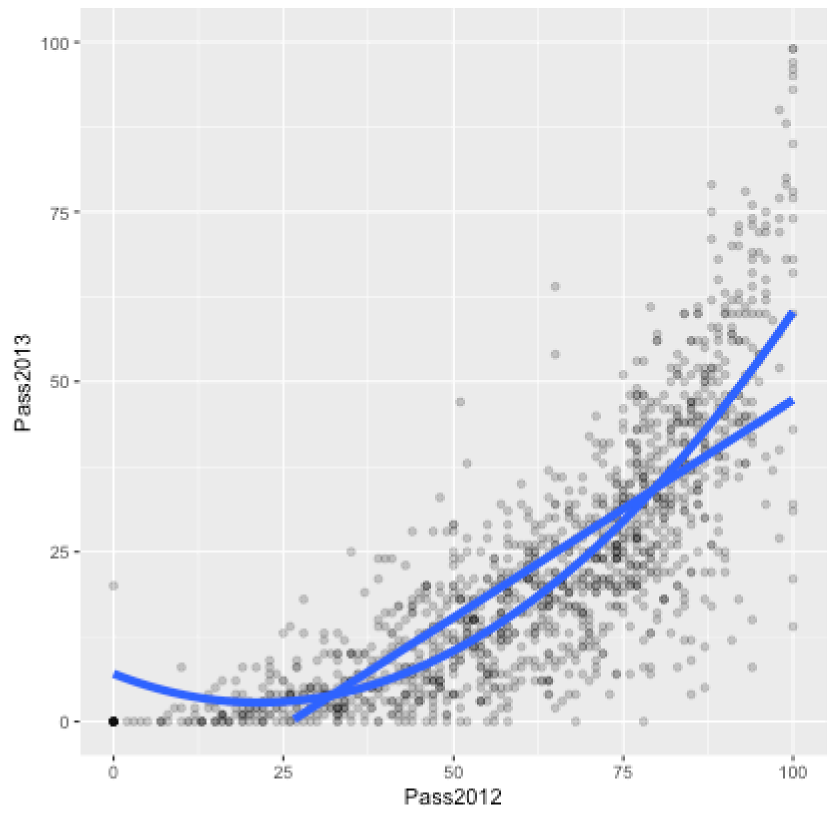
Quadratic Model

```
summary(quad.out)
```

```
##
## Call:
## lm(formula = Pass2013 ~ I(Pass2012^2) + Pass2012, data = reportCard)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -46.258  -4.906  -0.507   5.430  43.509
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    7.0466153  1.4263773   4.940 8.77e-07 ***
## I(Pass2012^2)  0.0092937  0.0004659  19.946  < 2e-16 ***
## Pass2012      -0.3972481  0.0533631  -7.444 1.72e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.11 on 1359 degrees of freedom
## Multiple R-squared:  0.7065,    Adjusted R-squared:  0.7061
## F-statistic:  1636 on 2 and 1359 DF,  p-value: < 2.2e-16
```

Scatter Plot

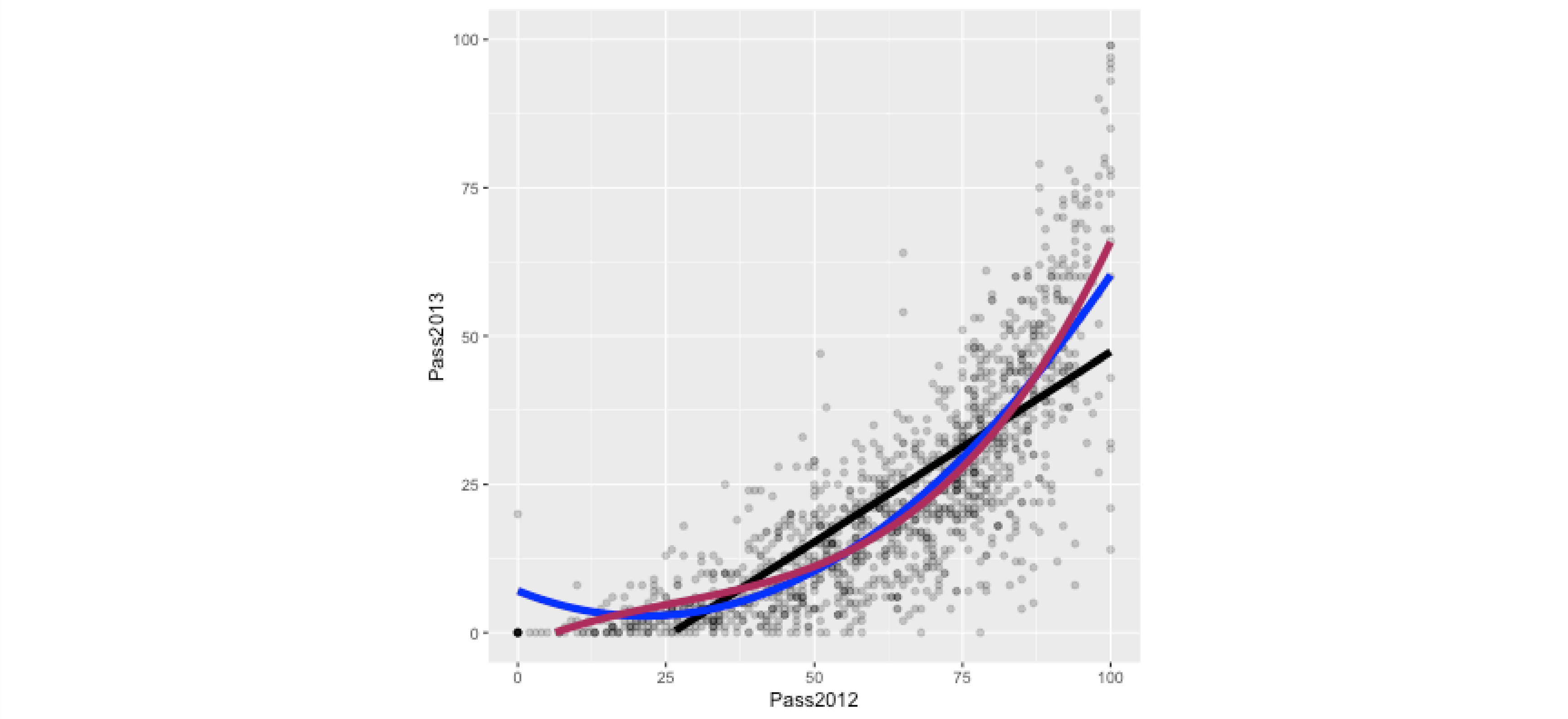
```r
ggplot(reportCard, aes(x=Pass2012, y=Pass2013)) + geom_point(alpha=0.2) +
    geom_smooth(method='lm', formula=y ~ x, size=2, se=FALSE) +
    geom_smooth(method='lm', formula=y ~ I(x^2) + x, size=2, se=FALSE) +
    coord_equal() + ylim(c(0,100)) + xlim(c(0,100))
```



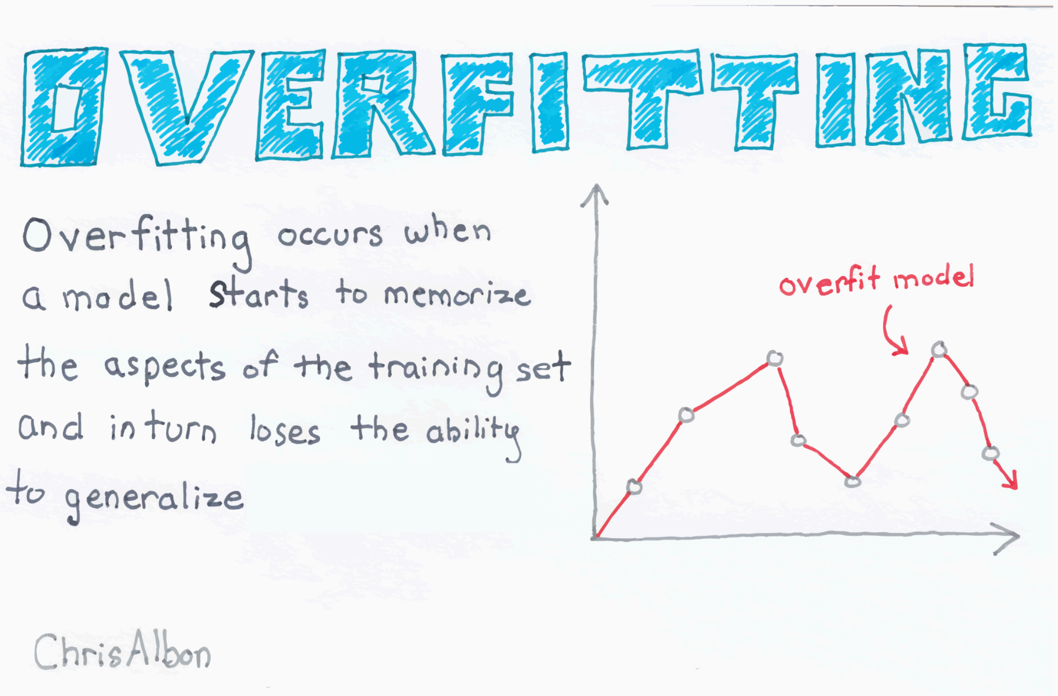
Let's go crazy, cubic!

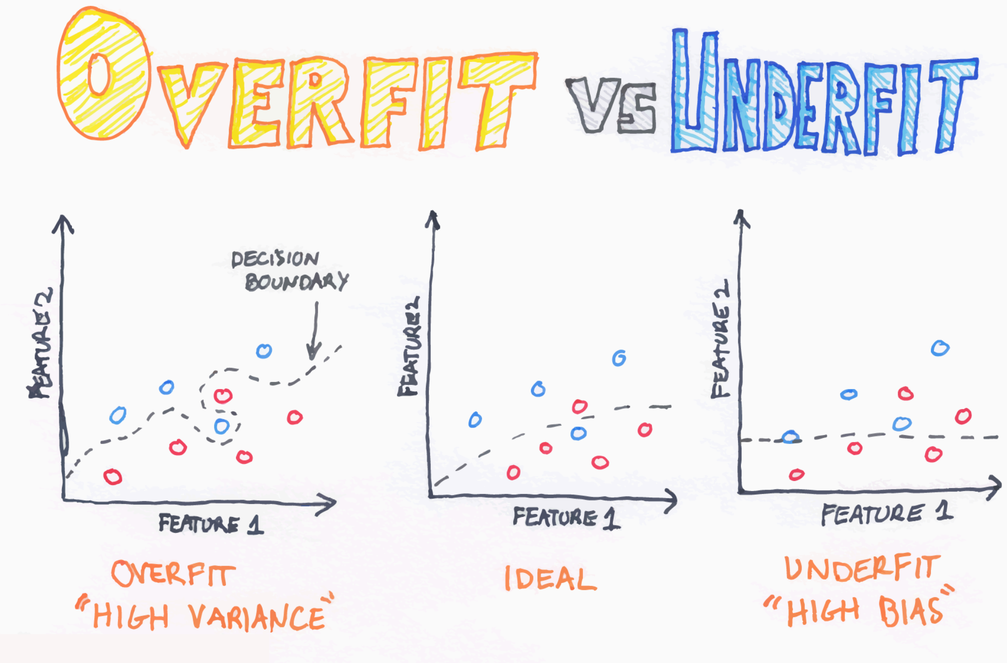
```
cube.out <- lm(Pass2013 ~ I(Pass2012^3) + I(Pass2012^2) + Pass2012, data=reportCard)
summary(cube.out)$r.squared
```

```
## [1] 0.7168206
```



Be careful of overfitting...

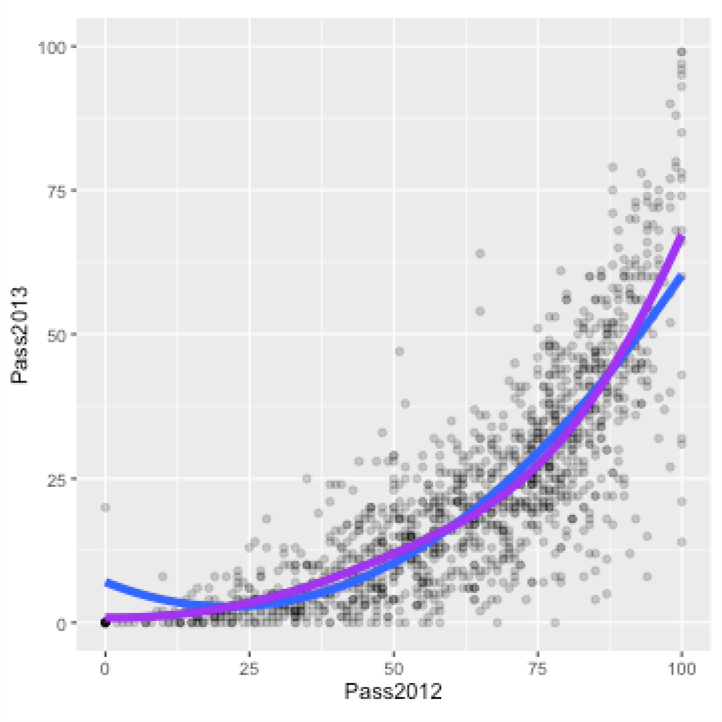




Source: Chris Albon @chrisalbon MachineLearningFlashCards.com

Loess Regression

```
ggplot(reportCard, aes(x=Pass2012, y=Pass2013)) + geom_point(alpha=0.2) +
    geom_smooth(method='lm', formula=y~poly(x,2,raw=TRUE), size=2, se=FALSE) +
    geom_smooth(method='loess', formula = y ~ x, size=2, se=FALSE, color = 'purple') +
    coord_equal() + ylim(c(0,100)) + xlim(c(0,100))
```




```
library('VisualStats')
library('ShinyDemo')
shiny_demo('loess', package = 'VisualStats')
```

See this site for more info:

https://visualstats.bryer.org/loess.html

Shiny App

```r
shiny::runGitHub('NYSchools','jbryer',subdir='NYSReportCard')
```

See also the Github repository for more information: https://github.com/jbryer/NYSchools

One Minute Paper

1. What was the most important thing you learned during this class?
2. What important question remains unanswered for you?

https://forms.gle/sTwKB3HivjtbafBb7