# Unit 12 Pre-Class Excercise

w203: Statistics for Data Science

*Adam Yang*

```r
load("elects.Rdata")
```

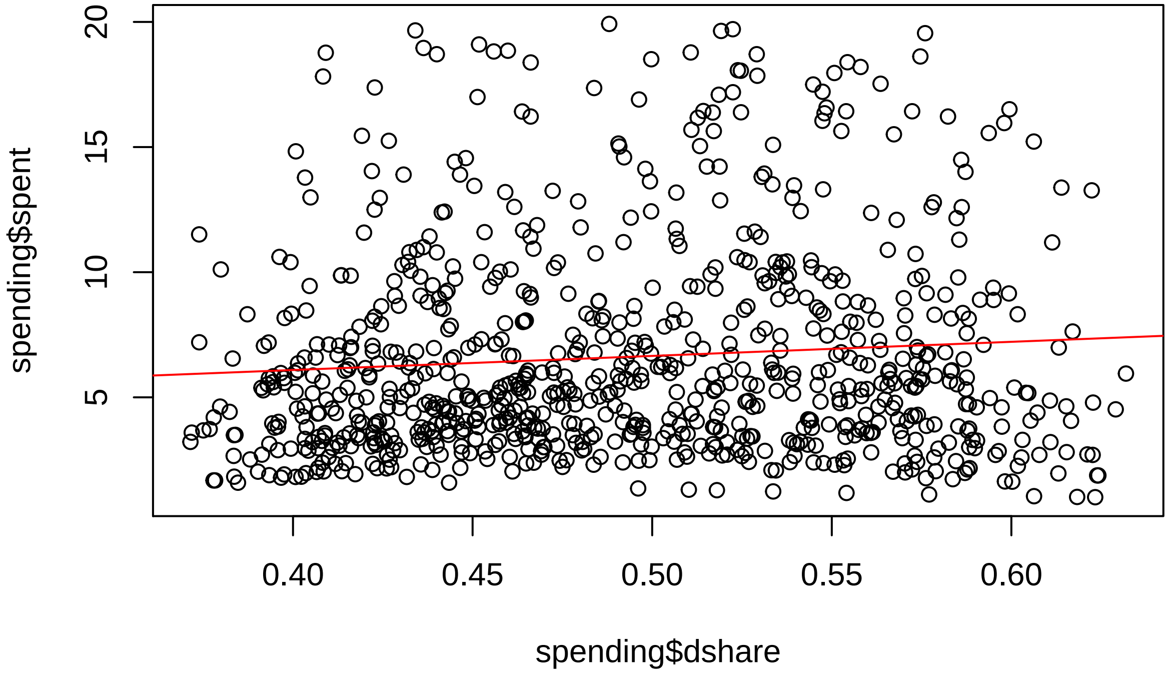
The `spending` data frame contains simulated data on election results and campaign spending in the U.S. The `dshare` and `rshare` variables capture the vote shares for the Democratic and Republican parties respectively, while `spent` measures total spending on advertisements in millions of dollars for each race. For this exercise, think of spending as the outcome variable and the vote share variables as predictors.

```r
head(spending)
```

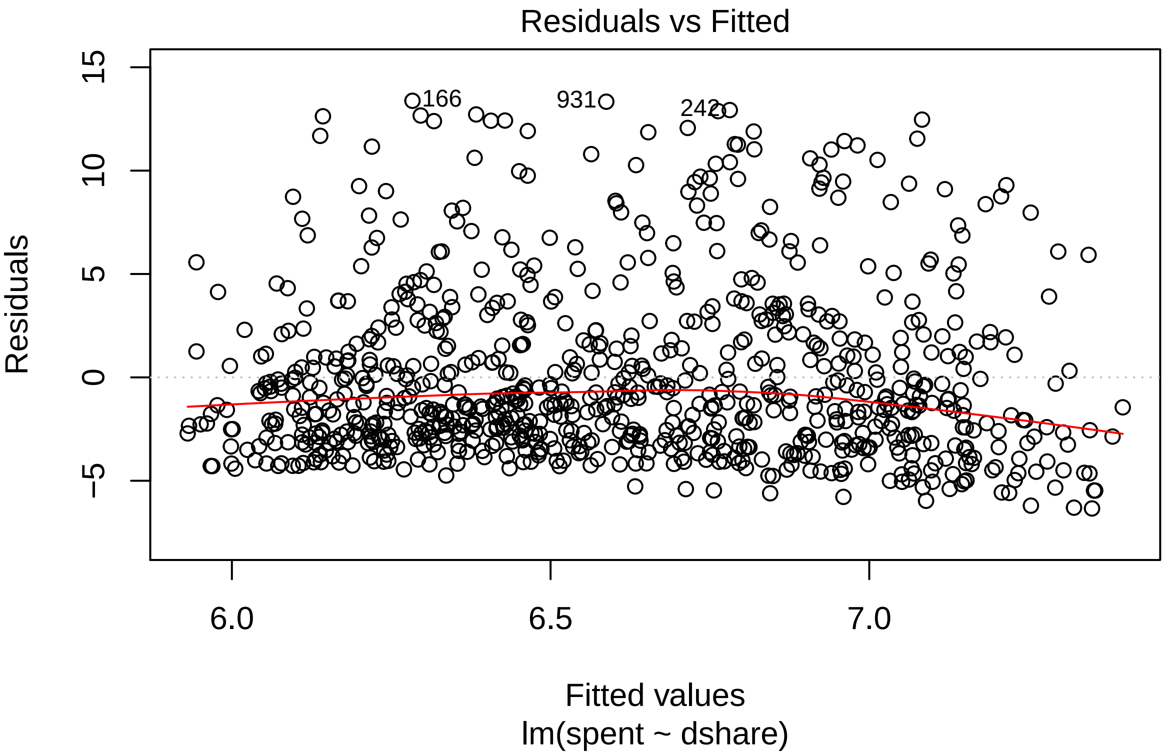
```
##      dshare    rshare spent
## 1 0.4573411 0.5426589  3.24
## 2 0.4969932 0.5030068  5.65
## 3 0.4222159 0.5777841  6.84
## 4 0.4220736 0.5779264  8.06
## 5 0.4173364 0.5826636  1.93
## 6 0.4272072 0.5727928  6.82
```

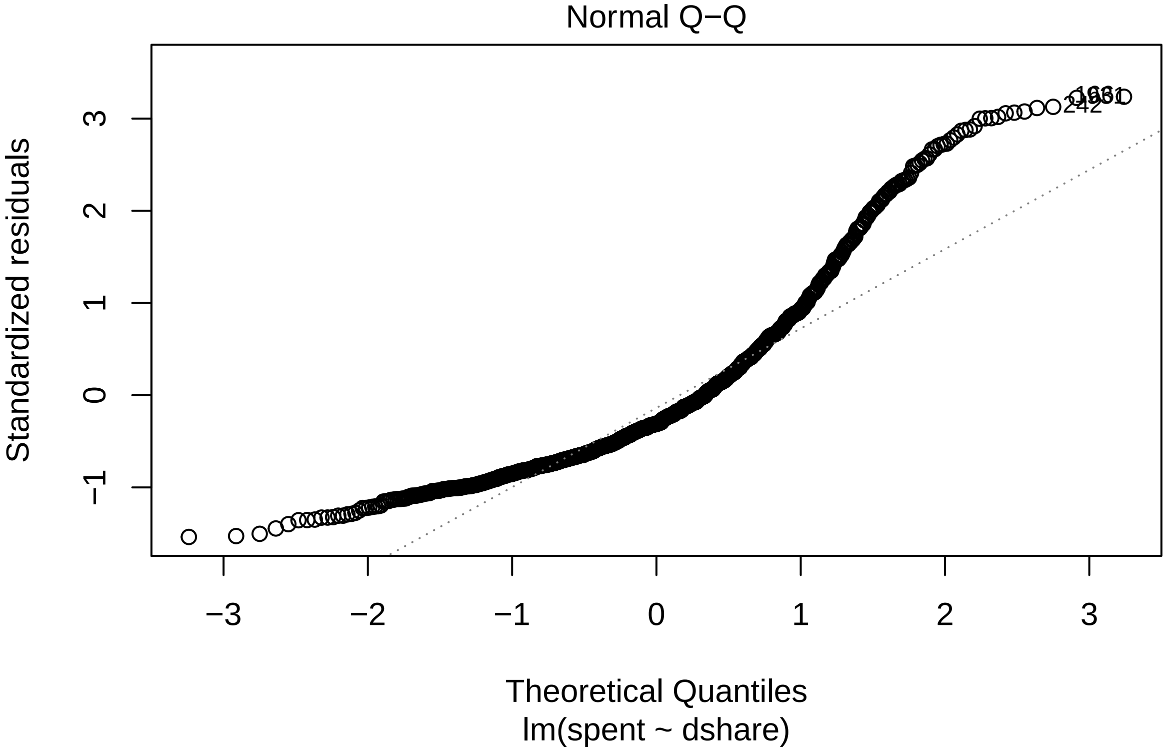
1. Is there a linear relationship between campaign spending and democratic vote share? Generate a scatter plot with a regression line. What does this scatter plot suggest about the appropriateness of the classical linear model assumptions in this case?

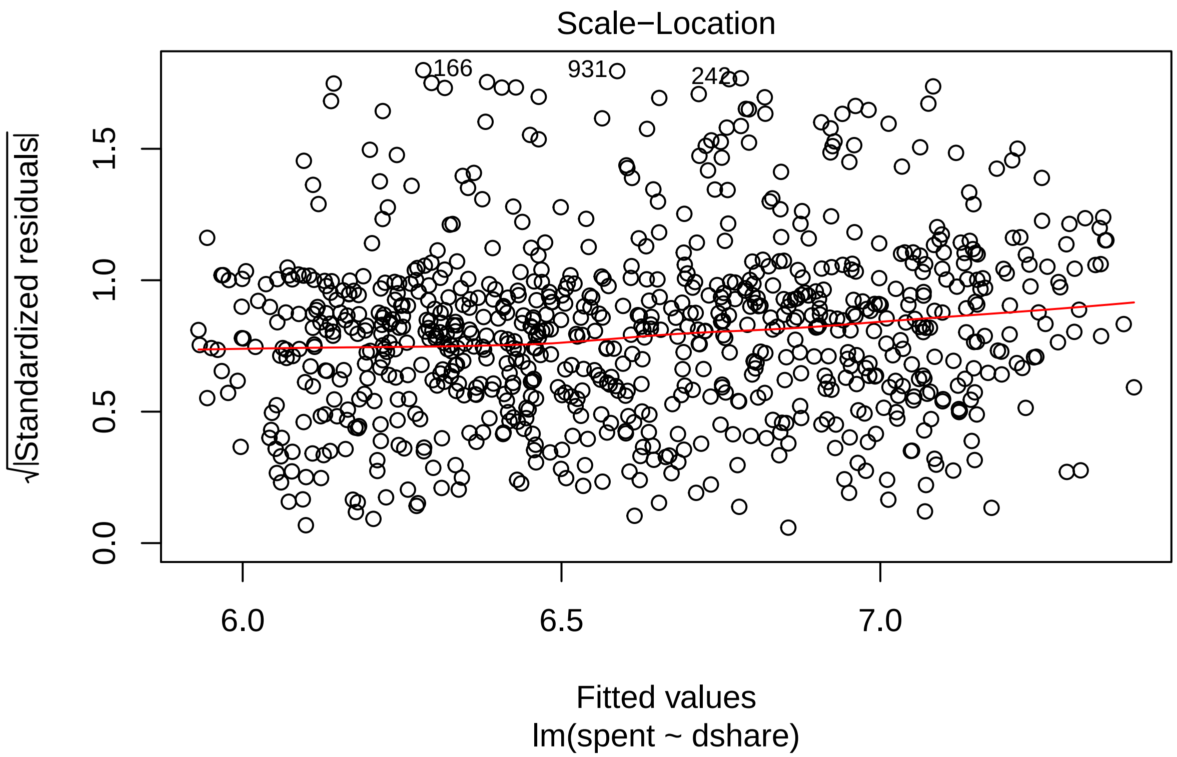
```r
model1 <- lm(spent~dshare, data = spending)
plot(spending$dshare,spending$spent)
abline(model1, col = "red")
```

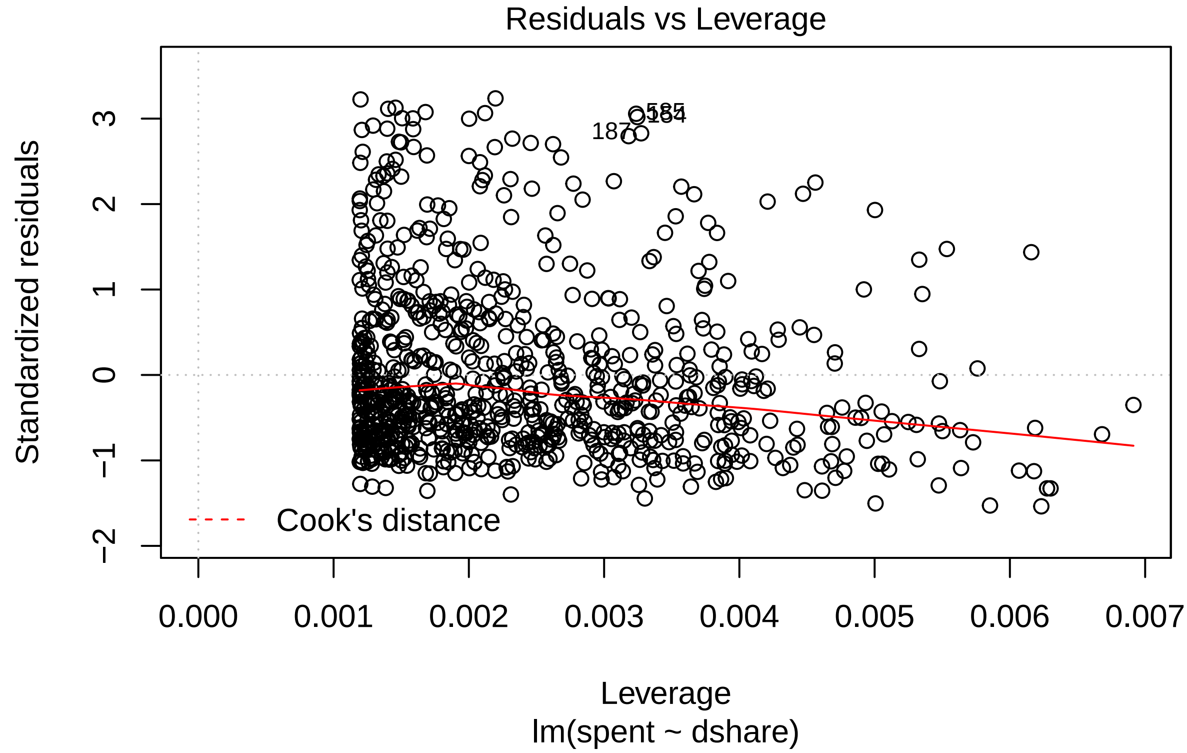


```r
plot(model1)
```


Residuals vs Fitted
Residuals
15
10
5
0
−5
166
931
242
6.0
6.5
7.0
Fitted values
lm(spent ~ dshare)


Normal Q−Q
Standardized residuals
3
2
1
0
−1
166
931
242
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(spent ~ dshare)





```
summary(model1)$r.squared
```

```
## [1] 0.007480978
```

```
model1$coefficients
```

```
## (Intercept)      dshare
##     3.838259    5.633058
```

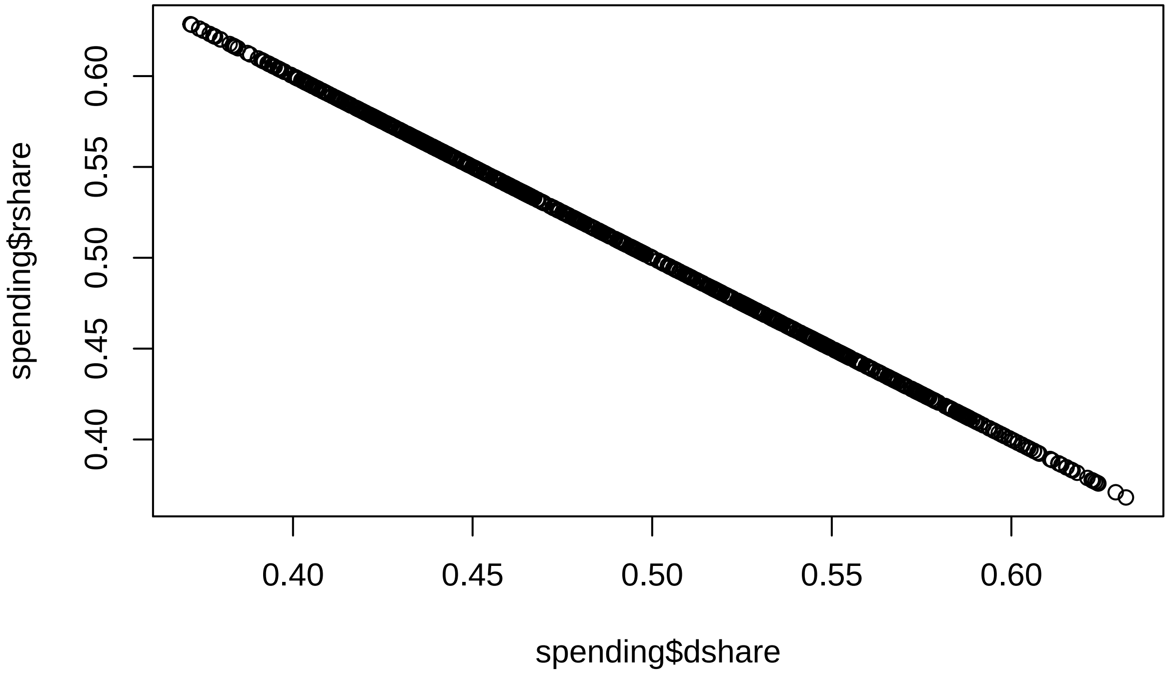
It looks like a lot of the variation in `spent` cannot be explained by dshare alone. The residuals vs fitted value plot shows a lot of variance as well as a slight curve, suggesting that the residuals are not completely unbiased. The Q-Q plot is very non-linear, which is driven by the positive skew in the `spent` variable. A log transform on `spent` can fix this issue.

2. If we want to improve this model, we have two options - add more variables or transform the variables we have. Let's try to add another variable first. Note that the two vote share variables do not sum to one. This could be due to the existence of a third political party or to error in measuring votes. To begin with, what assumption would be violated if the two variables did sum to one?

```
sum <- spending$dshare + spending$rshare
paste("The average of dshare + rshare is", mean(sum))
```

```
## [1] "The average of dshare + rshare is 1"
```

```
plot(spending$dshare, spending$rshare)
```



As shown above, contrary to what the question says, `dshare` and `rshare` does in fact add up to 1 in this data set. Therefore, dshare and rshare violate the "no perfect multicolinearity" assumtion. We should not do a multivariable regression with `dshare` and `rshare` in this case.

3. Since the two variables don't sum exactly to one, we should be able to include republican vote share as a second predictor. But before doing any coding, ask yourself what adding republican voteshare will do to the precision of our estimate on the effect of democratic vote share. Explain why this effect makes sense.

Contrary to what the question says, `dshare` and `rshare` does actually sum exactly to one. Therefore we cannot include republican vote share as a second predictor. However, if we pretend that they do not sum exactly to one, we can utilize the following equations to figure out the impact of adding republican voteshare:

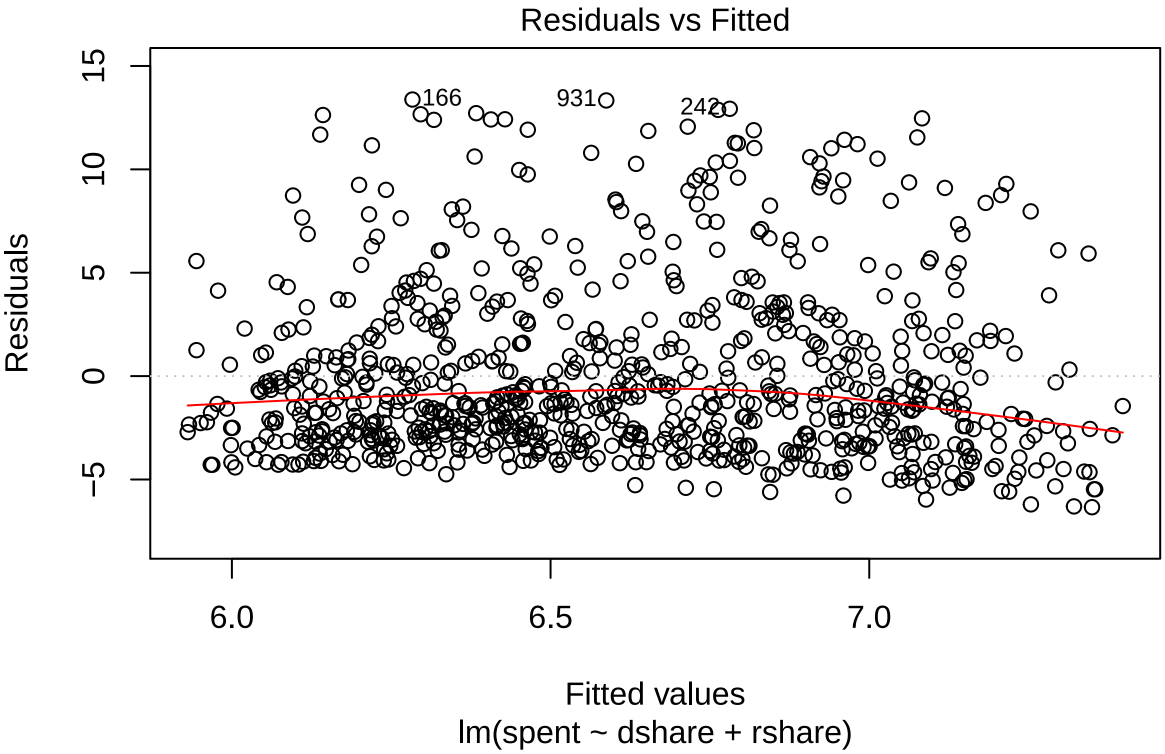
$$spent = \beta_0 + \beta_1 dshare + \beta_2 rshare + u$$

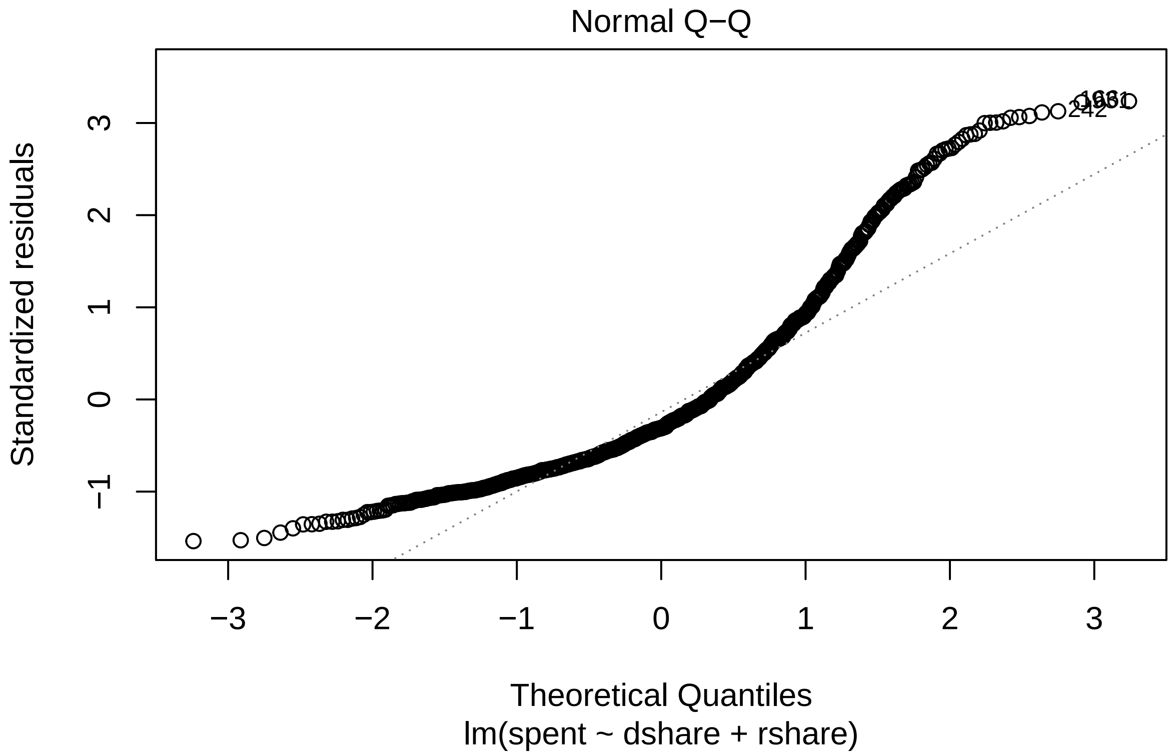
$$rshare = \alpha_0 + \alpha_1 dshare + r_1$$

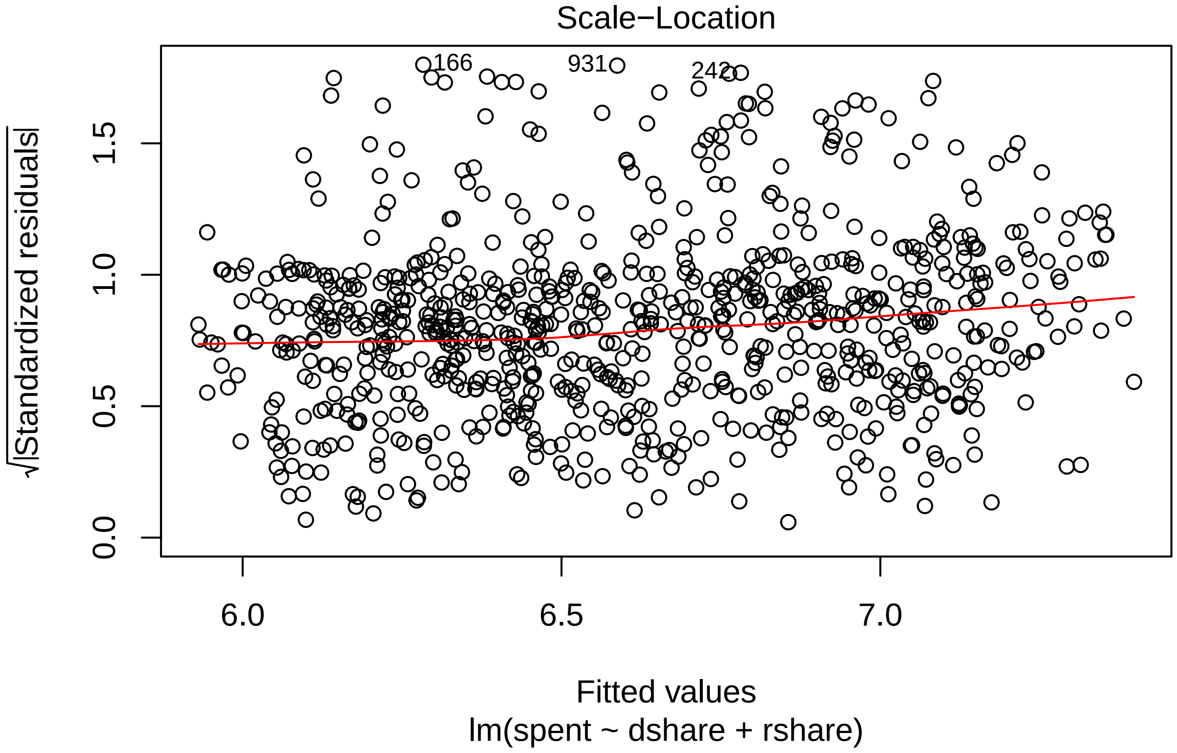
In the first equation, we would assume that as **rshare** increases, **spent** would also increase so $\beta_2$ is probably positive. In the second equation, we would assume that as **dshare** increases, **rshare** would decrease because dshare would take away from rshare. Therefore, $\alpha_1$ would be negative. Since $\beta_1$ is positive, and $\alpha_1\beta_2$ is negative, it means the lack of rshare was making our $\beta_1$ slope coefficient seem smaller than it was. We should see a higher $\beta_1$ after we introduce **rshare** into the model.

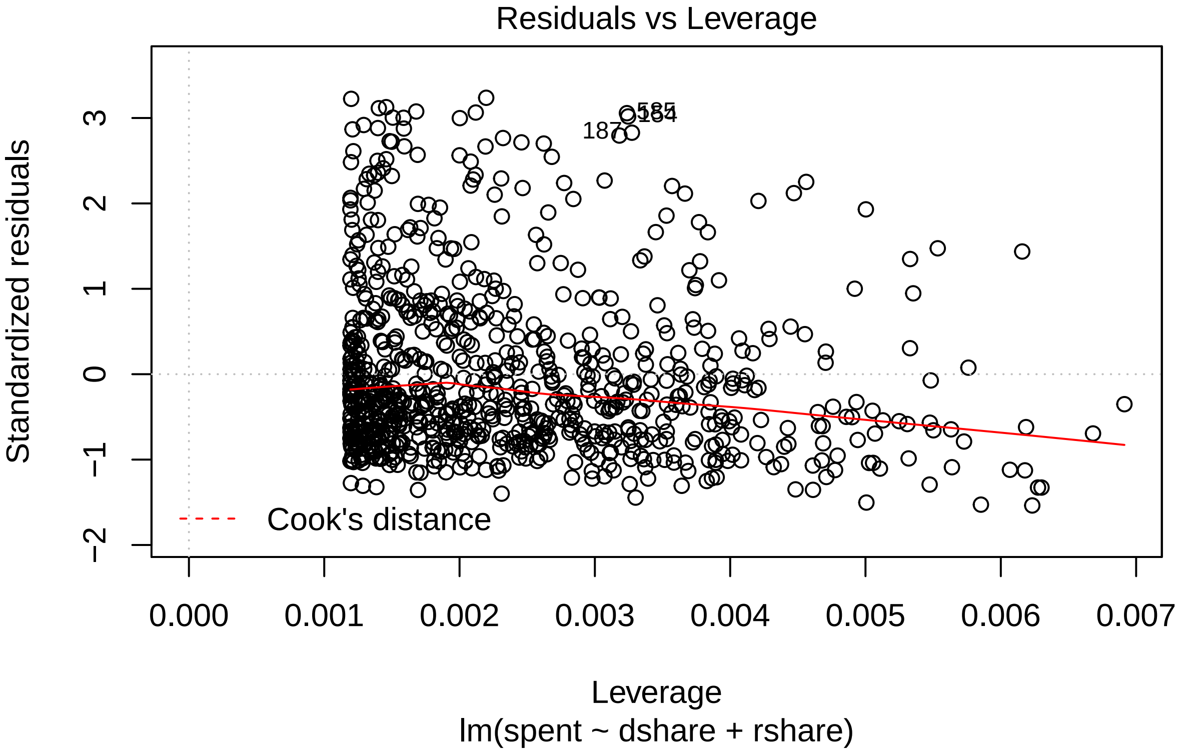
4. Now generate a model that predicts campaign spending using both democratic and republican vote shares. What does the result tell you about the practical implications of very highly correlated predictors?

```
model2 <- lm(spent~dshare+rshare, data = spending)
plot(model2)
```




Normal Q−Q
Standardized residuals
−1
0
1
2
3
166
931
242
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(spent ~ dshare + rshare)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
166
931
242
6.0
6.5
7.0
Fitted values
lm(spent ~ dshare + rshare)



```
summary(model2)$r.squared
```

```
## [1] 0.007480978
```

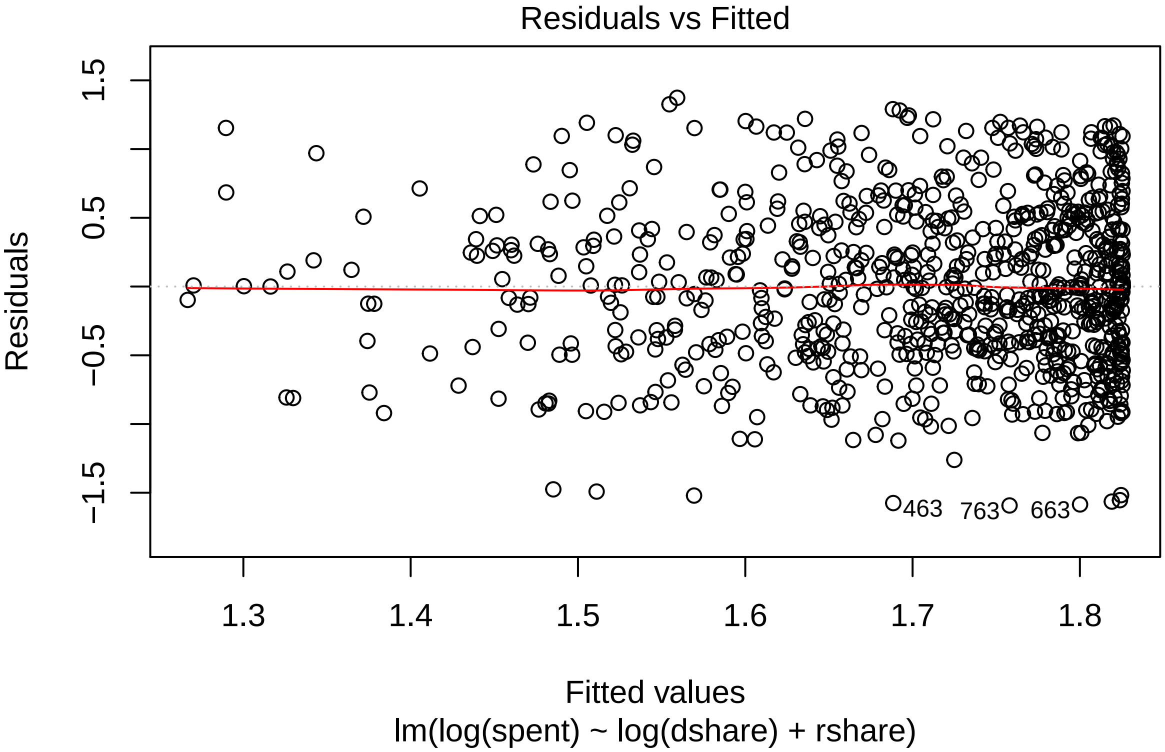
```
model2$coefficients
```

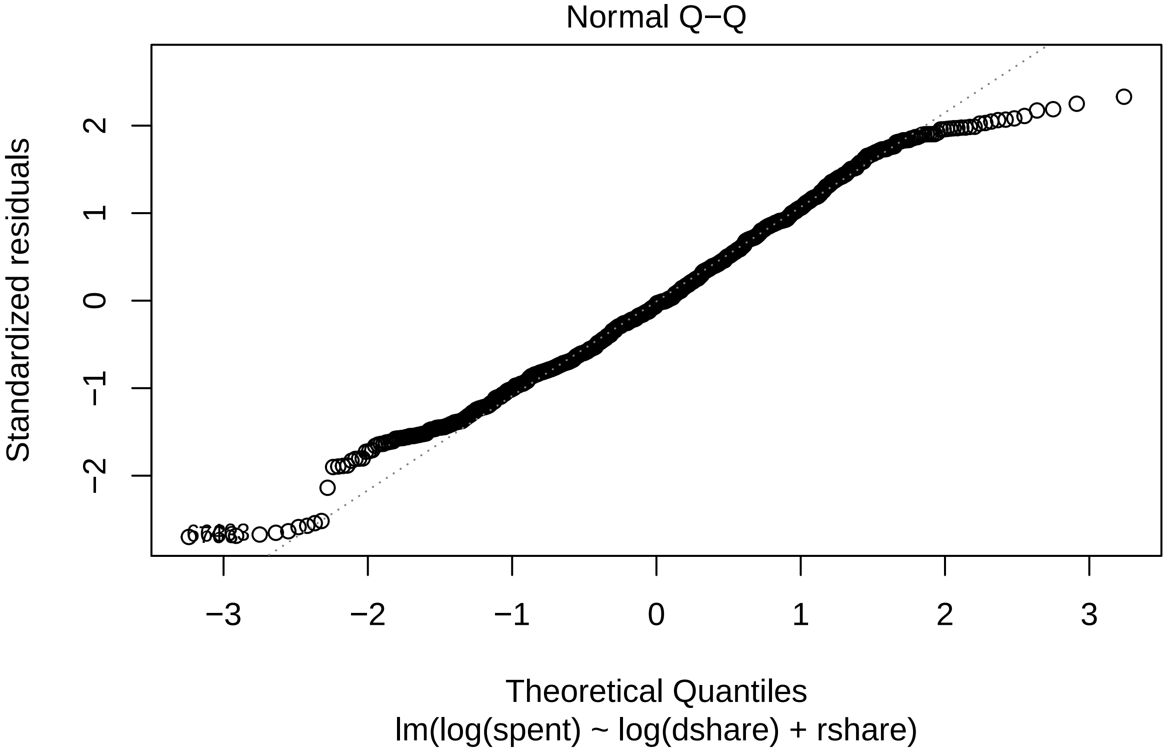
```
## (Intercept)      dshare      rshare 
##    3.838259    5.633058          NA
```

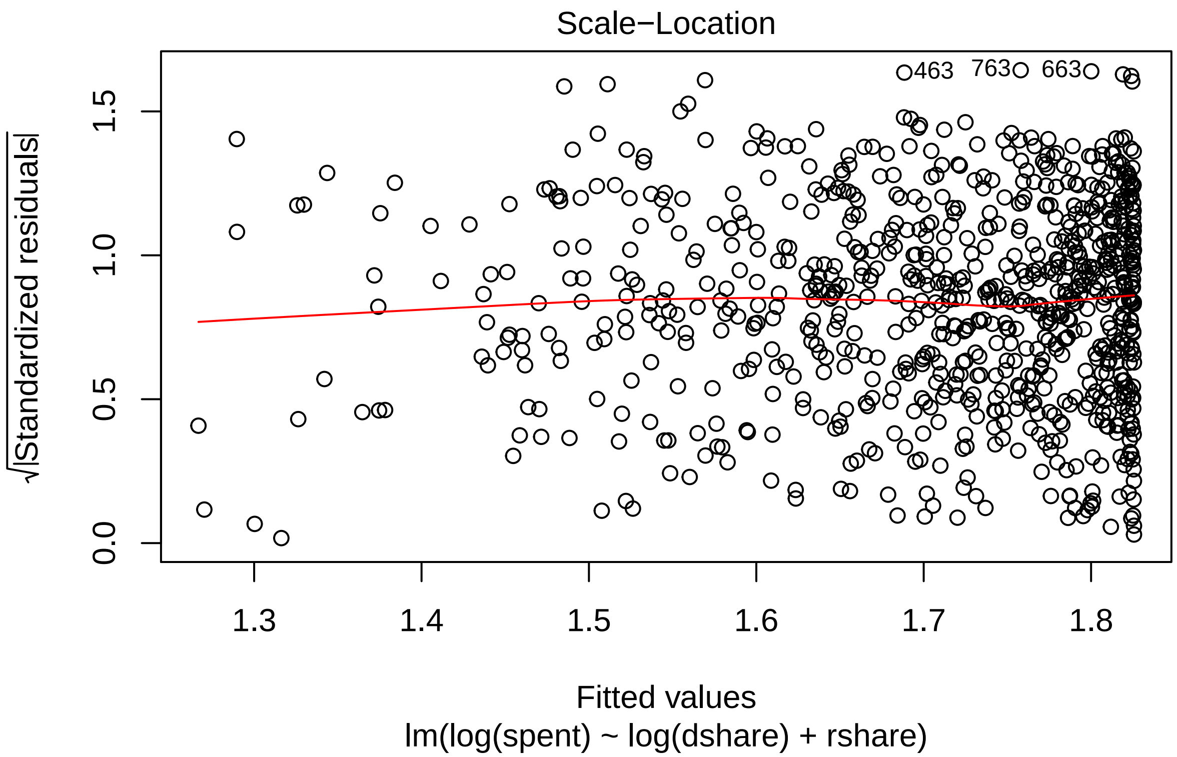
Since `rshare` and `dshare` are perfectly colinear, we could not build a model with both variables. However, if they were not perfectly colinear, but in fact imperfectly colinear, then the OLS would remain unbiased but with an increased variance and covariance. As a result of increased variance and covariance, it would become more difficult to obtain a precise estimation of the coefficients, making the OLS less practical.

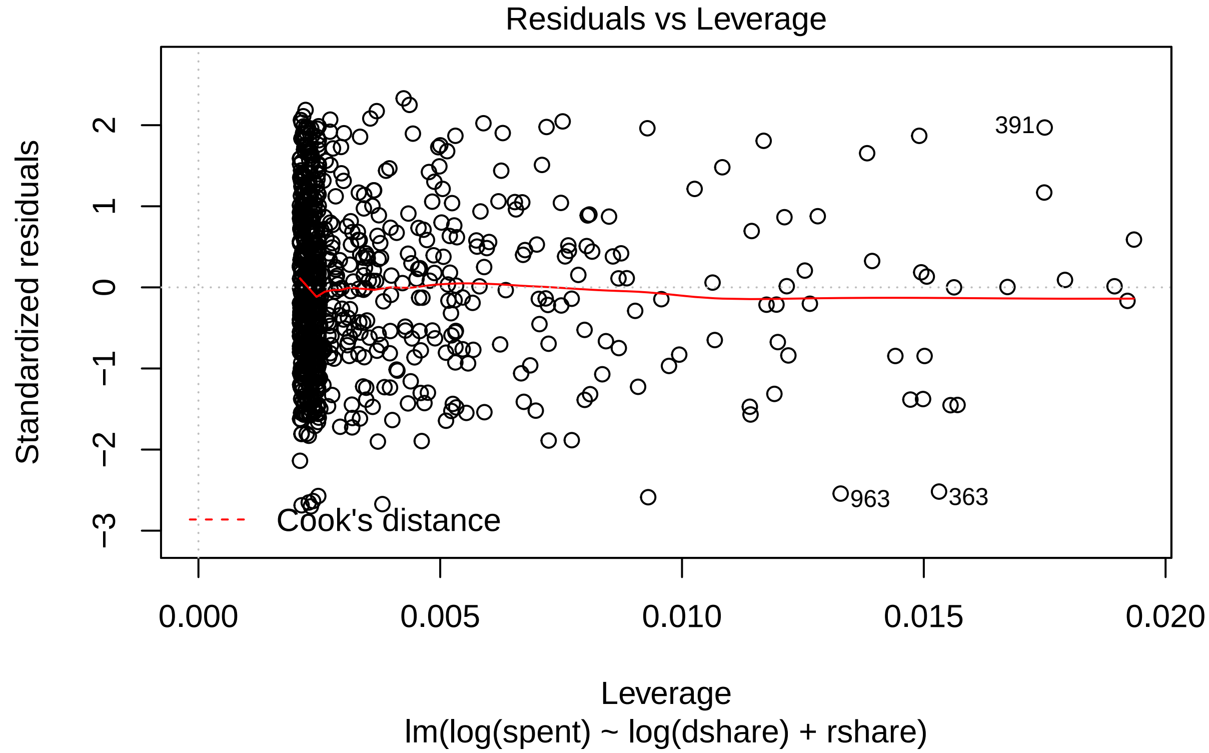
5. Finally, what is the transformation of variables that might solve the problems identified thus far? Perform that transformation and describe the results.

```
model3 <- lm(log(spent)~log(dshare)+rshare, data = spending)
plot(model3)
```


Residuals vs Fitted
Residuals
1.5
0.5
−0.5
−1.5
463
763
663
1.3
1.4
1.5
1.6
1.7
1.8
Fitted values
lm(log(spent) ~ log(dshare) + rshare)


Normal Q−Q
Standardized residuals
2
1
0
−1
−2
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(log(spent) ~ log(dshare) + rshare)





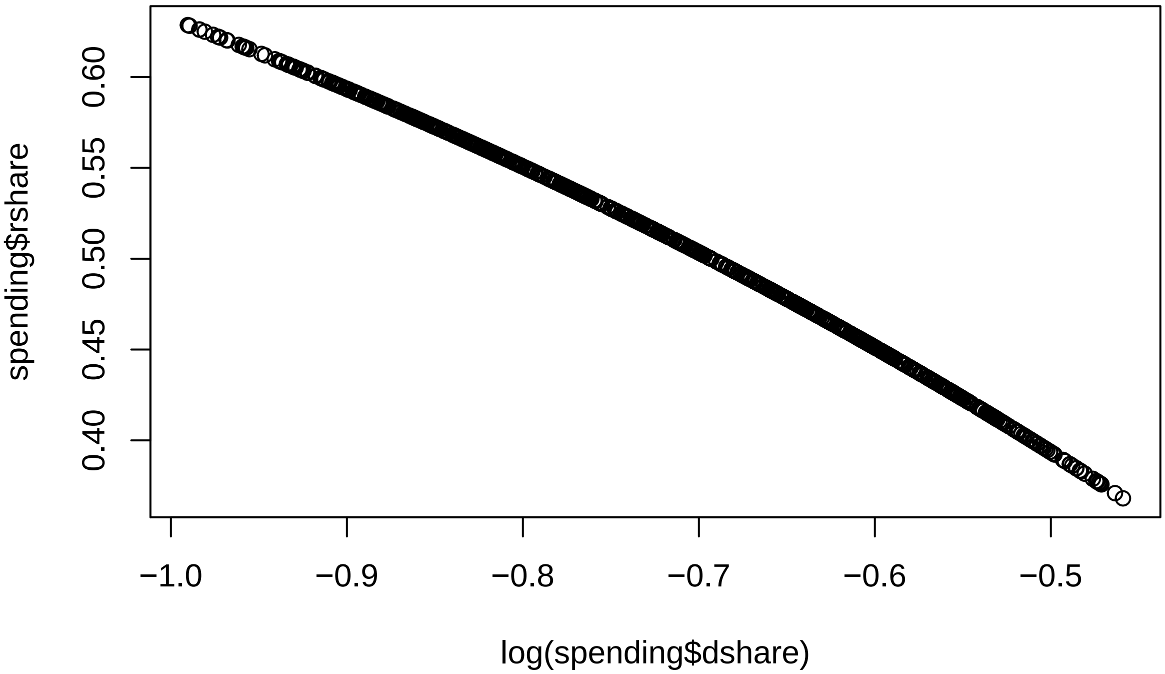
```
summary(model3)$r.squared
```

```
## [1] 0.03710242
```

```
model3$coefficients
```

```
## (Intercept) log(dshare)      rshare
##   -2.273188   13.551917   26.984341
```

```
plot(log(spending$dshare),spending$rshare)
```



First, I did a logrithmic transformation on `spent` to get rid of the positive skew as much as possible. Then I did a logrithmic transformation on `dshare` to force it to be less linear with `rshare`. However, `dshare` and `rshare` are perfectly co-linear so I don't think this is a wise choice. I think it would be best to only use one of these variables.