

# **Statistics 330 Paper**

Alex Jenks, Grant Mitchell, and Ethan Scott

## **Abstract**

There is no institution in modern day politics more important than elections. Regardless of whether a regime is democratic (where elections are free and fair) or autocratic, election are a staple of today's political world, providing (some) legitimacy to those who win them. Thus, one of the central puzzles in the social sciences is why some parties win more votes than others. In this paper, we fit several statistical models to predict party vote share, finding that party strength is the best predictor of a party's success in elections.

## **1 Problem and Motivation**

There is no institution in modern day politics more important than elections. Since the 1990s, almost every country in the world has switched to using elections to decide which party or candidate is in government. Even in autocratic regimes like contemporary China and Russia use elections to justify the incumbents' stays in power, even if the results are pre-determined.

Given the importance of elections, tools and variables that predict elections are valuable. In this paper, we create several statistical models that predict the vote share of elections' primary actors: political parties. In doing so, we identify several factors that strongly influence voter share. Additionally, we explore how the effect of predictors of vote share change depending on the level of democracy in the country.

### **1.1 Data Description**

To fit our models, we use data from the Varieties of Democracy database (V-Dem). V-Dem is the largest database on democracy and parties in the world, covering country-level democracy data in every country from 1789 to now and party-level data from 1970 to now. Every variable we use in our data comes from V-Dem, allowing us to cover over 1600 parties in 165 countries over the a 50 year period. For our models, our unit of analysis is a party-year observation. For example, the United States Republican Party in 2000 is an observation in our analysis.

We consider several predictors of party vote share. Regarding party factors, we include the measures of ideology, anti-pluralism (how favorably they view democracy), if the party is a pariah (i.e., and outcast), party brand continuity (new party, reformed party, same party as last election), and party strength. We also include several country factors, including, compulsory voting (none, mild enforcement, medium enforcement, and strong enforcement), election type, and quality of democracy. Because of scholar's recent focus on party strength (i.e., the party's organizational capacity to mobilize its voters), we are particularly interested in it's effect on vote share.

For our dependent variable, we use a party's raw vote percentage and multiply it by the registered voter turnout. We do this for easier comparison between cases, as we expect some parties to not only increase their raw vote share, but to encourage higher voter turnout.

## 1.2 Questions of Interest

We are interested in two questions. First, what variables best predict the a party's vote share? A branch of this question is what is the effect of party strength on party vote share?

The second question is related to the first. How does the relationship between party strength and vote share change given the countries level of democracy?

## 1.3 Regression Methods

To answer the first question, we use forward, backward, and sequential stepwise model selection to choose our predictors. After checking assumptions, re-selecting variables, and transforming our variables to meet the regression assumptions, we came to this formula:

$$[ \text{VoteShare}^{-1} = _0 + _1 \text{Party Strength} + _2 \text{Ideology} + _3 \text{Democracy Score} + _4 \text{AntiPluralism} + _5 \text{Pariah} + _6 (\text{PartyBrand Continuity} = \text{Reformed Party}) + _7 (\text{PartyBrand Continuity} = \text{Same Party}) + _8 (\text{Compulsory Voting} = \text{Mild Enforcement}) + _9 (\text{Compulsory Voting} = \text{Medium Enforcement}) + \{10\} (\text{Compulsory Voting} = \text{Strong Enforcement}) + \text{Error} ]$$

Using the same processes, we came to the following formula for the interaction model between democracy and party strength:

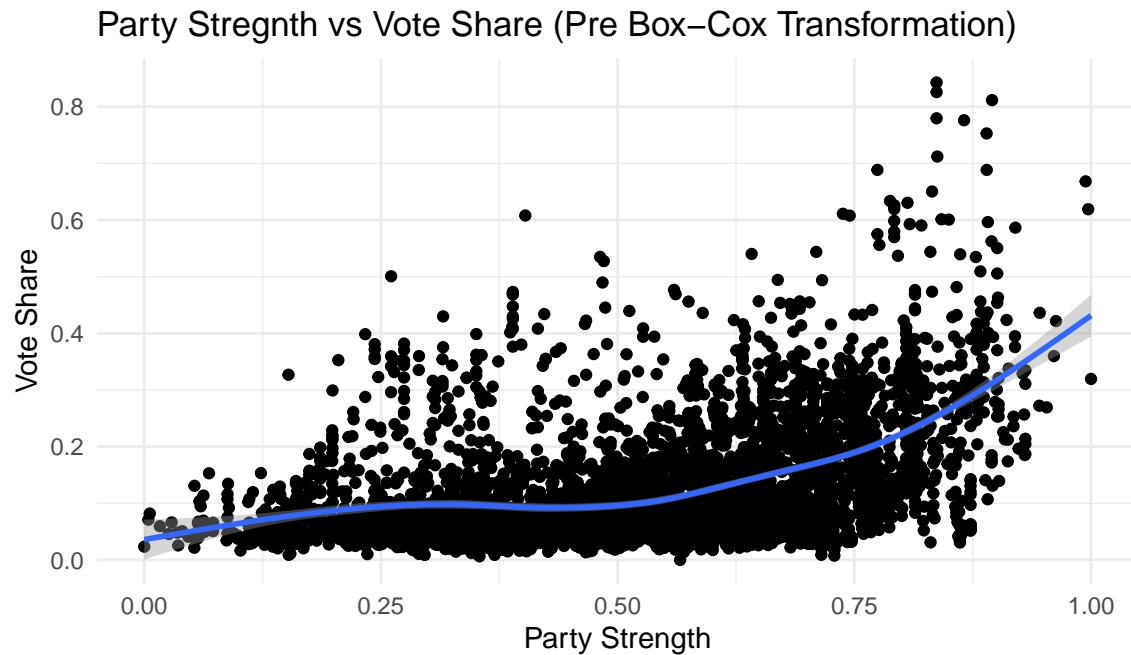
$$[ \text{VoteShare}^{-1} = _0 + _1 \text{Party Strength} + _2 \text{Ideology} + _3 \text{Democracy Score} + _4 \text{AntiPluralism} + _5 \text{Pariah} + _6 (\text{PartyBrand Continuity} = \text{Reformed Party}) + _7 (\text{PartyBrand Continuity} = \text{Same Party}) + _8 (\text{Compulsory Voting} = \text{Mild Enforcement}) + _9 (\text{Compulsory Voting} = \text{Medium Enforcement}) + \{10\} (\text{Compulsory Voting} = \text{Strong Enforcement}) + \{11\} (\text{Party Strength} \times \text{Democracy Score}) + \text{Error} ]$$

For our analysis, we report the results of three models: a baseline OLS model to compare effect of vote share, a logistical regression model using the same formula as the baseline to compare effect on control of government, and an interaction OLS model on vote share again. You can find the full results and methodology in the appendix.

## 2 Analyses, Results, and Interpretation

To start our analysis for our first question, we first look at the relationship between our main independent variable, party strength, and our outcome variable, party vote share.

```
`geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```



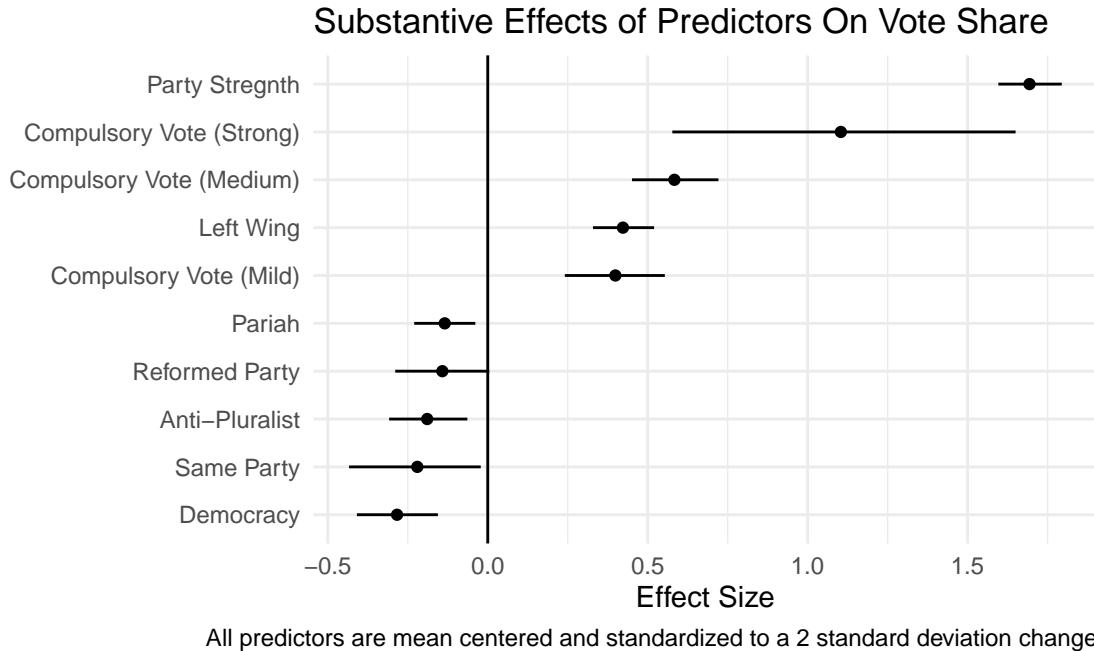
As one can see, there is strong, non-linear, and positive relationship between vote share and party strength. To more rigorously test this relationship, we fit a OLS regression.

As mentioned in the regression methods subsection, we utilize forward, backward, and sequential stepwise model selection in order to find the variables that have an effect on party vote share. The result tells us that the best predictors of party vote share are: party strength, compulsory voting, ideology, democracy score, anti-pluralism, pariah status, and party brand continuity. To remedy a violation of the normality assumption, we apply a Box Cox transformation and rerun the stepwise function and assumption checks to ensure we have a quality model. We found that this model largely met the assumptions of linearity, normality of errors, constant variance, influential observations, and multicollinearity. You can the results for our model selection and how we checked assumptions in the appendix.

Satisfied by the model, we may begin to make inferences. Below, we plot the mean effect of each of our variables and the 95% confidence interval for their respective effects. We do this so we can test the effect of each coefficient. In general, the null hypotheses is that, holding

all else constant, there is no relationship between a given predictor and the outcome variable. The alternative is that there is a relationship, holding all else constant.

For ease of interpretation, we multiply the effects by negative one, since the transformed vote share variable is difficult to interpret otherwise. In all the plots, a positive effect mean the variable is positively associated with vote share.



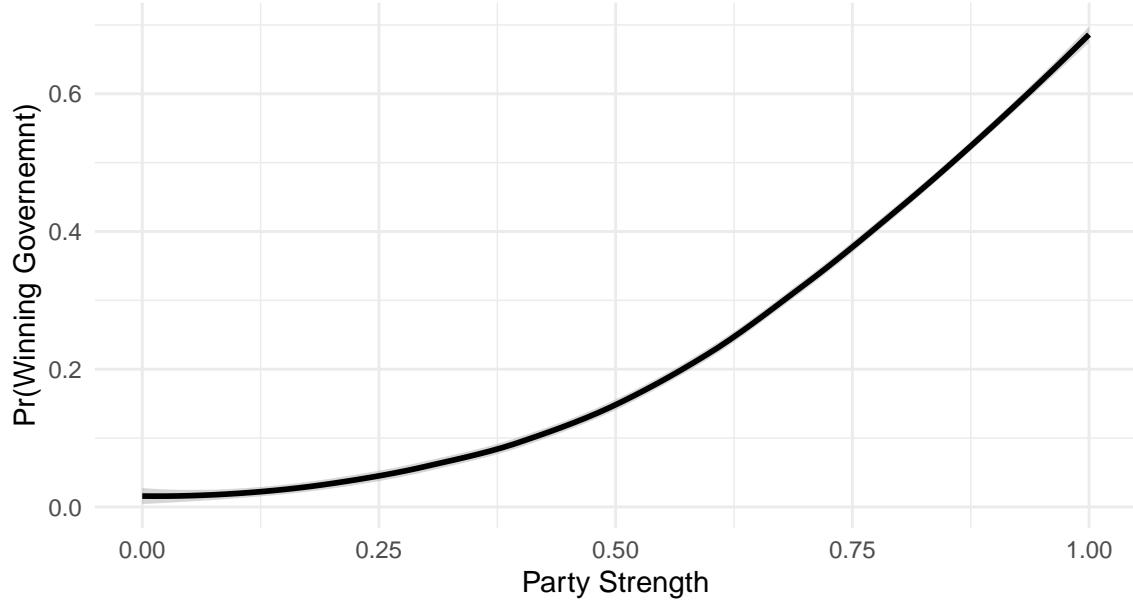
We find that party strength has the highest predicted effect on vote share of all the predictors in our model. Compulsory vote and left wing both have positive, significant effects. Meanwhile, pariah, reformed party, anti-pluralist, same party, and democracy all have negative effects on vote share.

For global diagnostics, we find that the model itself is significant, which is no surprise considering nearly all the coefficients are significant. We also find that the model predicts around 23 percent of the variance in vote share, a feat that is actually quite impressive considering what is typical of social science models.

As a robustness check, we also fit a logistical regression model predicting the probability of a party winning control of the government. Once again, party strength was the strongest predictor. The plot below shows the probability of winning control of the government give different values of party strength.

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

## Probability of Controlling Government



As shown by the previous plot, party strength is a robust predictor of party vote share, so much so that it significantly increases the probability of winning control of the government, holding all else equal. Meanwhile, party ideology, anti-pluralism, and pariah status also have significant effects of winning control of the government. We therefore reject the null hypothesis for these predictors. You can see the full results of our analysis, as well as diagnostic checks for both models, in the appendix.

For our second question, we began by plotting the vote share against party strength and added a color scale to indicate the levels of democracy for each observation.

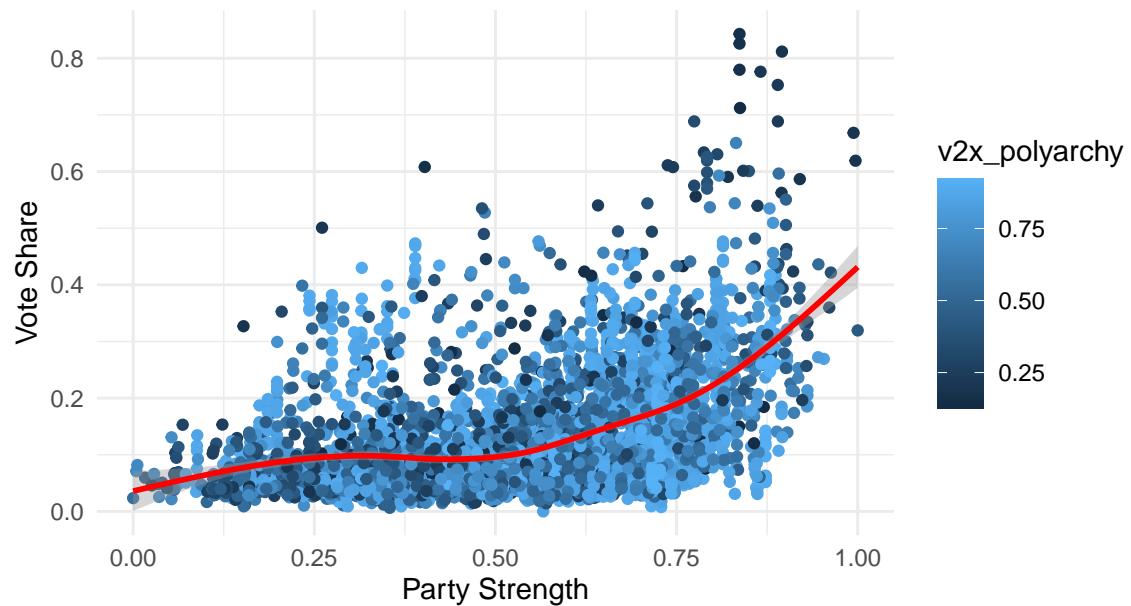
```
`geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

We suspect an interaction effect between level of democracy and party strength. It appears that as a country becomes more democratic, party strength matters less. This makes sense, since performance in government matters more in democratic systems than in autocracies. To more thoroughly investigate these results, we fit another OLS regression using the same transformed vote share outcome variable. Below is a marginal effects plot showing the coefficient size of party strength given various levels of democracy.

For this particular question, the null hypothesis is that the effect of party strength is not moderated by level of democracy. The alternative is that there is a moderating effect.

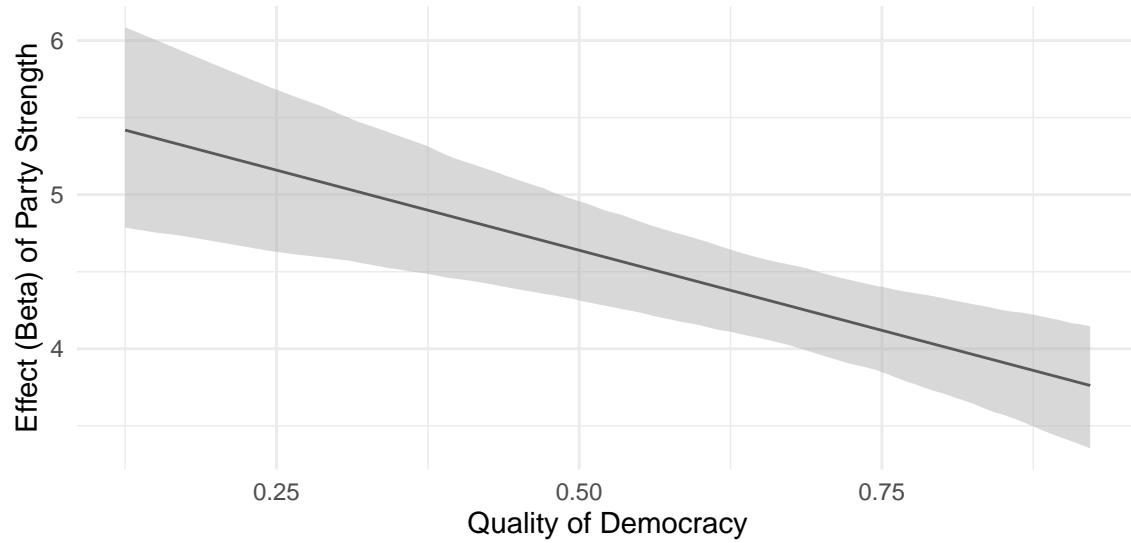
An additional plot shows the substantive effects of party strength at differing levels of democracy.

Party Strength vs Vote Share

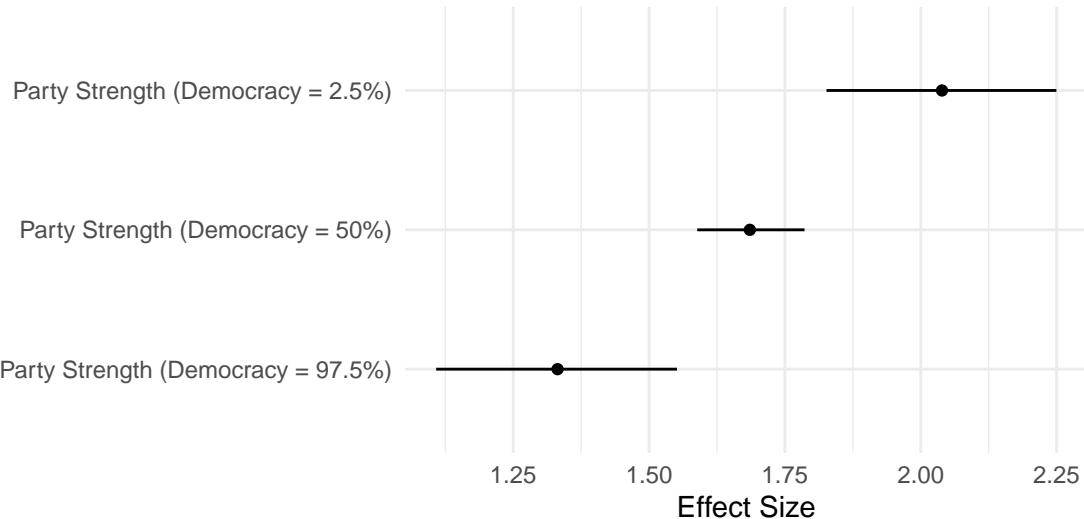


Marginal Effect of Party Strength

Conditional on Democratic Quality



## Substantive Effects of Party Strength At Different Levels of Democracy



All predictors are mean centered and standardized to a 2 standard deviation change

The effect of party strength differs significantly depending on the level of democracy in the country. Holding all else constant, as democracy increases, the effect of party strength on party vote share decreases. In countries without free and fair elections, otherwise called autocracies, party strength has a much larger effect. In democracies, the effect of party strength is diminished, probably because performance in government and meaningful accountability make elections more volatile. This interaction effect is significantly different from zero, so we reject the null hypothesis and conclude that there is a real moderating effect between party strength and level of democracy.

This model's diagnostics were slightly better. The model significance was slightly higher than the non-interaction model, while the variation explained by the model is higher than the non-interaction model, even after accounting for the number of predictors in the interaction model, around 25 percent.

As a final robustness check, we check the significance of the interaction term as a whole. After running a partial F-test, we can confirm that the interaction effect is important and should be included in our model.

### 3 Conclusions

In conclusion, our model and its results clearly indicate that party strength is an important indicator in a party's vote share, the total amount of votes a party receives in proportion to the

population. We also found that the degree of democracy existing in a country also significantly affects this turnout inversely; the more democratic a state is, the lesser the effect it has on a party's vote share, likely due to changing priorities for voters that come with free and fair elections.

There do exist a few minor weaknesses in our model. While our Box Cox transformation drastically improved the normality of the errors in our data, it is still not perfect, so there may be slight inaccuracy introduced with our analysis.

We also must acknowledge that our observations are not identically and independently distributed. The importance of this is that our observations are of political parties across the many countries in the world. Most nations have more than one party that competes for the population's votes so naturally one party's vote share is going to affect that of the others. Therefore, the observations are not independent. However, that is an often times natural consequence of using observational data, so there is not much we can do about that using the methods we are familiar with.

Finally, we also recognize that because of the nature of politics and governance, it is possible that certain parties have advantages over others that stem from winning office. For example, a party that wins office one year could potentially use the government's resources in order to increase their party vote share for the next election. Essentially, the outcome of one election may affect the predictors of the next, indicating potential reverse causality.

## 4 Contributions

Ethan wrote the conclusion, formatted the regression model into LATEX, and helped with the analysis.

Alex wrote the analysis and interpretation sections of the paper, and worked on the analysis, primary fitting the logit regression.

Grant wrote the introduction and questions sections and worked on data wrangling and the analysis and visualizations.

## APPENDIX

First, a summary of the data so you can understand what is happening

```
dat <- read.csv("~/R Projects/three30/partyinst_data_330.csv") %>%
  mutate(COW = factor(COWcode.x), leg_elect = factor(v2eltype_0), exe_elect = factor(v2elt
na.omit()
```

```

##### Here are the names of the variables we will be using:
##### COW - a numerical indicator given to each country
##### v2elvaptrn - Voting Age Turnout
##### v2elvaptrn - Registered turnout
##### leg_elect - legislative elections
##### exe_elect - executive elections
##### v2xpa_antiplural - party's anti-pluralism measure
##### v2pavote - party vote share
##### v2paseatshare - party seat share
##### v2paelcont - party brand continuity
##### v2papariah - party's pariah measure
##### v2pariglef - Party left right scale
##### elect_system - electoral system
##### v2x_polyarchy - Democracy score
##### piendla - Party institutionalization ****
##### psla - Party strength ****
##### vap_vs - Percent of the VAP turnout that a party got ****
##### reg_vs - Percent of the Registered turnout a party got ****

```

[summary\(dat\)](#)

COWcode.x	v2x_polyarchy	v2elvaptrn	v2eltrnout	v2eltype_0
Min. : 20.0	Min. : 0.1250	Min. : 13.88	Min. : 21.01	Min. : 1
1st Qu.: 211.0	1st Qu.: 0.5270	1st Qu.: 56.68	1st Qu.: 61.31	1st Qu.: 1
Median : 352.0	Median : 0.7690	Median : 68.96	Median : 73.31	Median : 1
Mean : 387.8	Mean : 0.6872	Mean : 67.28	Mean : 71.67	Mean : 1
3rd Qu.: 517.0	3rd Qu.: 0.8670	3rd Qu.: 79.22	3rd Qu.: 83.62	3rd Qu.: 1
Max. : 950.0	Max. : 0.9220	Max. : 104.31	Max. : 98.80	Max. : 1
v2eltype_6	v2elcomvot	v2elparallel	v2xpa_antiplural	
Min. : 0.0000	Min. : 0.0000	Min. : 0.00	Min. : 0.0120	
1st Qu.: 0.0000	1st Qu.: 0.0000	1st Qu.: 1.00	1st Qu.: 0.0490	
Median : 0.0000	Median : 0.0000	Median : 1.00	Median : 0.1790	
Mean : 0.2562	Mean : 0.4184	Mean : 1.04	Mean : 0.3011	
3rd Qu.: 1.0000	3rd Qu.: 1.0000	3rd Qu.: 1.00	3rd Qu.: 0.5002	
Max. : 1.0000	Max. : 3.0000	Max. : 3.00	Max. : 1.0000	
v2pavote	v2paseatshare	v2paelcont	v2papariah	
Min. : 0.000	Min. : 0.00	0:3340	Min. : -4.0880	
1st Qu.: 7.575	1st Qu.: 6.70	1: 604	1st Qu.: -0.6030	
Median : 13.100	Median : 13.00	2: 236	Median : 0.6770	
Mean : 18.367	Mean : 19.54		Mean : 0.2881	

```

3rd Qu.:26.000   3rd Qu.: 28.40          3rd Qu.: 1.5180
Max.    :91.000   Max.    :100.00          Max.    : 2.0660

piendla          psla           v2pariglef      v2pagovsup
Min.   :0.02253   Min.   :0.00000   Min.   :-4.10600  Min.   :0.000
1st Qu.:0.42357  1st Qu.:0.3836   1st Qu.:-0.96150 1st Qu.:1.000
Median  :0.54664  Median :0.5515   Median : 0.14000  Median :3.000
Mean    :0.52513  Mean   :0.5312   Mean   : 0.06453  Mean   :1.944
3rd Qu.:0.64867  3rd Qu.:0.6887   3rd Qu.: 1.23200 3rd Qu.:3.000
Max.    :0.92210  Max.   :1.00000  Max.   : 4.17400  Max.   :4.000

gov            COW        leg_elect  exe_elect  elect_system
Min.   :0.00000  211       : 116     0: 0      0:3109    0:3103
1st Qu.:0.00000  130       : 114     1:4180    1:1071    1: 438
Median  :0.00000  390       :  98      2: 606
Mean    :0.1993   380       :  90      3:  33
3rd Qu.:0.00000  325       :  83
Max.    :1.00000  740       :  83
(Other) :3596

reg_vs
Min.   :0.00000
1st Qu.:0.05238
Median  :0.09367
Mean   :0.13266
3rd Qu.:0.17898
Max.   :0.84284

```

##### The VAP vote share variable, seems suspect (fraud maybe?), so there may be some coding error.

Here is our model selections methodology:

```

## Stepwise model selection
attach(dat)

```

The following objects are masked from dat\_center:

```

COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,
v2xpa_antiplural

```

The following objects are masked from dat\_new:

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,
v2xpa_antiplural
```

The following objects are masked from dat (pos = 5):

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,
v2xpa_antiplural
```

```
base_model <- lm(reg_vs ~ 1)
full_model <- lm(reg_vs ~ elect_system + v2pariglef + psla + v2papariah + exe_elect + v2pa
car::vif(full_model)

          GVIF Df GVIF^(1/(2*Df))
elect_system    1.182357  3     1.028312
v2pariglef     1.056522  1     1.027872
psla           1.214403  1     1.102000
v2papariah     1.151020  1     1.072856
exe_elect       1.147033  1     1.070996
v2paelcont     1.176495  2     1.041472
v2xpa_antiplural 1.875132  1     1.369355
v2x_polyarchy   1.825730  1     1.351196

forward <- step(base_model,
                  direction = "forward",
                  k = 2,
                  scope = list(lower = base_model, upper = full_model))

Start:  AIC=-18443.68
reg_vs ~ 1
```

	Df	Sum of Sq	RSS	AIC
+ psla	1	9.3191	41.345	-19291
+ v2paelcont	2	1.4033	49.261	-18557
+ elect_system	3	0.7247	49.940	-18498
+ v2xpa_antiplural	1	0.1881	50.476	-18457
+ v2pariglef	1	0.0859	50.579	-18449
+ v2x_polyarchy	1	0.0608	50.604	-18447
+ v2papariah	1	0.0424	50.622	-18445
<none>			50.664	-18444
+ exe_elect	1	0.0024	50.662	-18442

Step: AIC=-19291.32

reg\_vs ~ psla

	Df	Sum of Sq	RSS	AIC
+ elect_system	3	0.85760	40.488	-19373
+ v2x_polyarchy	1	0.53999	40.805	-19344
+ v2pariglef	1	0.39644	40.949	-19330
+ v2xpa_antiplural	1	0.39597	40.949	-19330
+ v2papariah	1	0.14867	41.197	-19304
+ exe_elect	1	0.10890	41.237	-19300
+ v2paelcont	2	0.07095	41.274	-19295
<none>			41.345	-19291

Step: AIC=-19372.93

reg\_vs ~ psla + elect\_system

	Df	Sum of Sq	RSS	AIC
+ v2x_polyarchy	1	0.64001	39.848	-19438
+ v2xpa_antiplural	1	0.38050	40.107	-19410
+ v2pariglef	1	0.35380	40.134	-19408
+ v2papariah	1	0.18279	40.305	-19390
+ v2paelcont	2	0.06228	40.426	-19375
<none>			40.488	-19373
+ exe_elect	1	0.00967	40.478	-19372

Step: AIC=-19437.54

reg\_vs ~ psla + elect\_system + v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ v2pariglef	1	0.34443	39.503	-19472
+ v2paelcont	2	0.08242	39.765	-19442
+ v2papariah	1	0.05967	39.788	-19442

```

+ v2xpa_antiplural  1   0.01977 39.828 -19438
<none>                      39.848 -19438
+ exe_elect         1   0.00260 39.845 -19436

Step: AIC=-19471.82
reg_vs ~ psla + elect_system + v2x_polyarchy + v2pariglelf

          Df Sum of Sq    RSS    AIC
+ v2papariah      1  0.096469 39.407 -19480
+ v2paelcont      2  0.089314 39.414 -19477
<none>                  39.503 -19472
+ exe_elect       1  0.005852 39.498 -19470
+ v2xpa_antiplural 1  0.004288 39.499 -19470

Step: AIC=-19480.04
reg_vs ~ psla + elect_system + v2x_polyarchy + v2pariglelf + v2papariah

          Df Sum of Sq    RSS    AIC
+ v2paelcont      2  0.089198 39.318 -19486
<none>                  39.407 -19480
+ exe_elect       1  0.008068 39.399 -19479
+ v2xpa_antiplural 1  0.000046 39.407 -19478

Step: AIC=-19485.52
reg_vs ~ psla + elect_system + v2x_polyarchy + v2pariglelf + v2papariah +
v2paelcont

          Df Sum of Sq    RSS    AIC
<none>                  39.318 -19486
+ exe_elect       1  0.0064218 39.311 -19484
+ v2xpa_antiplural 1  0.0003123 39.317 -19484

back <- step(full_model,
              direction = "backward",
              k = 2,
              scope = list(lower = base_model, upper = full_model))

Start: AIC=-19482.21
reg_vs ~ elect_system + v2pariglelf + psla + v2papariah + exe_elect +
v2paelcont + v2xpa_antiplural + v2x_polyarchy

```

	Df	Sum of Sq	RSS	AIC
- v2xpa_antiplural	1	0.0001	39.311	-19484
- exe_elect	1	0.0062	39.317	-19484
<none>			39.311	-19482
- v2paelcont	2	0.0877	39.399	-19477
- v2papariah	1	0.0947	39.406	-19474
- v2x_polyarchy	1	0.3347	39.646	-19449
- v2pariglef	1	0.3850	39.696	-19444
- elect_system	3	0.8860	40.197	-19395
- psla	1	8.8698	48.181	-18634

Step: AIC=-19484.2

```
reg_vs ~ elect_system + v2pariglef + psla + v2papariah + exe_elect +
v2paelcont + v2x_polyarchy
```

	Df	Sum of Sq	RSS	AIC
- exe_elect	1	0.0064	39.318	-19486
<none>			39.311	-19484
- v2paelcont	2	0.0876	39.399	-19479
- v2papariah	1	0.0983	39.410	-19476
- v2pariglef	1	0.3922	39.704	-19445
- v2x_polyarchy	1	0.5097	39.821	-19432
- elect_system	3	0.8866	40.198	-19397
- psla	1	8.8948	48.206	-18634

Step: AIC=-19485.52

```
reg_vs ~ elect_system + v2pariglef + psla + v2papariah + v2paelcont +
v2x_polyarchy
```

	Df	Sum of Sq	RSS	AIC
<none>			39.318	-19486
- v2paelcont	2	0.0892	39.407	-19480
- v2papariah	1	0.0964	39.414	-19477
- v2pariglef	1	0.3884	39.706	-19446
- v2x_polyarchy	1	0.5048	39.823	-19434
- elect_system	3	0.9080	40.226	-19396
- psla	1	8.9672	48.285	-18629

```
both <- step(base_model,
  direction = "both",
  k = 2,
```

```
    scope = list(lower = base_model, upper = full_model))
```

Start: AIC=-18443.68

reg\_vs ~ 1

	Df	Sum of Sq	RSS	AIC
+ psla	1	9.3191	41.345	-19291
+ v2paelcont	2	1.4033	49.261	-18557
+ elect_system	3	0.7247	49.940	-18498
+ v2xpa_antiplural	1	0.1881	50.476	-18457
+ v2pariglef	1	0.0859	50.579	-18449
+ v2x_polyarchy	1	0.0608	50.604	-18447
+ v2papariah	1	0.0424	50.622	-18445
<none>		50.664		-18444
+ exe_elect	1	0.0024	50.662	-18442

Step: AIC=-19291.32

reg\_vs ~ psla

	Df	Sum of Sq	RSS	AIC
+ elect_system	3	0.8576	40.488	-19373
+ v2x_polyarchy	1	0.5400	40.805	-19344
+ v2pariglef	1	0.3964	40.949	-19330
+ v2xpa_antiplural	1	0.3960	40.949	-19330
+ v2papariah	1	0.1487	41.197	-19304
+ exe_elect	1	0.1089	41.237	-19300
+ v2paelcont	2	0.0709	41.274	-19295
<none>		41.345		-19291
- psla	1	9.3191	50.664	-18444

Step: AIC=-19372.93

reg\_vs ~ psla + elect\_system

	Df	Sum of Sq	RSS	AIC
+ v2x_polyarchy	1	0.6400	39.848	-19438
+ v2xpa_antiplural	1	0.3805	40.107	-19410
+ v2pariglef	1	0.3538	40.134	-19408
+ v2papariah	1	0.1828	40.305	-19390
+ v2paelcont	2	0.0623	40.426	-19375
<none>		40.488		-19373
+ exe_elect	1	0.0097	40.478	-19372
- elect_system	3	0.8576	41.345	-19291

- psla 1 9.4520 49.940 -18498

Step: AIC=-19437.54

reg\_vs ~ psla + elect\_system + v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ v2pariglef	1	0.3444	39.503	-19472
+ v2paelcont	2	0.0824	39.765	-19442
+ v2papariah	1	0.0597	39.788	-19442
+ v2xpa_antiplural	1	0.0198	39.828	-19438
<none>			39.848	-19438
+ exe_elect	1	0.0026	39.845	-19436
- v2x_polyarchy	1	0.6400	40.488	-19373
- elect_system	3	0.9576	40.805	-19344
- psla	1	10.0317	49.880	-18501

Step: AIC=-19471.82

reg\_vs ~ psla + elect\_system + v2x\_polyarchy + v2pariglef

	Df	Sum of Sq	RSS	AIC
+ v2papariah	1	0.0965	39.407	-19480
+ v2paelcont	2	0.0893	39.414	-19477
<none>			39.503	-19472
+ exe_elect	1	0.0059	39.498	-19470
+ v2xpa_antiplural	1	0.0043	39.499	-19470
- v2pariglef	1	0.3444	39.848	-19438
- v2x_polyarchy	1	0.6306	40.134	-19408
- elect_system	3	0.9177	40.421	-19382
- psla	1	10.3195	49.823	-18504

Step: AIC=-19480.04

reg\_vs ~ psla + elect\_system + v2x\_polyarchy + v2pariglef + v2papariah

	Df	Sum of Sq	RSS	AIC
+ v2paelcont	2	0.0892	39.318	-19486
<none>			39.407	-19480
+ exe_elect	1	0.0081	39.399	-19479
+ v2xpa_antiplural	1	0.0000	39.407	-19478
- v2papariah	1	0.0965	39.503	-19472
- v2pariglef	1	0.3812	39.788	-19442
- v2x_polyarchy	1	0.4848	39.892	-19431
- elect_system	3	0.9375	40.344	-19388
- psla	1	10.3853	49.792	-18504

Step: AIC=-19485.52  
 reg\_vs ~ psla + elect\_system + v2x\_polyarchy + v2pariglef + v2papariah +  
 v2paelcont

	Df	Sum of Sq	RSS	AIC
<none>		39.318	-19486	
+ exe_elect	1	0.0064	39.311	-19484
+ v2xpa_antiplural	1	0.0003	39.317	-19484
- v2paelcont	2	0.0892	39.407	-19480
- v2papariah	1	0.0964	39.414	-19477
- v2pariglef	1	0.3884	39.706	-19446
- v2x_polyarchy	1	0.5048	39.823	-19434
- elect_system	3	0.9080	40.226	-19396
- psla	1	8.9672	48.285	-18629

forward

Call:  
 lm(formula = reg\_vs ~ psla + elect\_system + v2x\_polyarchy + v2pariglef +  
 v2papariah + v2paelcont)

Coefficients:

	(Intercept)	psla	elect_system1	elect_system2	elect_system3
	0.027821	0.256712	0.024796	0.037849	0.054959
v2x_polyarchy		v2pariglef	v2papariah	v2paelcont1	v2paelcont2
-0.054333		0.006713	-0.003638	-0.007241	-0.018432

back

Call:  
 lm(formula = reg\_vs ~ elect\_system + v2pariglef + psla + v2papariah +  
 v2paelcont + v2x\_polyarchy)

Coefficients:

	(Intercept)	elect_system1	elect_system2	elect_system3	v2pariglef
	0.027821	0.024796	0.037849	0.054959	0.006713
psla		v2papariah	v2paelcont1	v2paelcont2	v2x_polyarchy
0.256712		-0.003638	-0.007241	-0.018432	-0.054333

```
both
```

Call:

```
lm(formula = reg_vs ~ psla + elect_system + v2x_polyarchy + v2pariglef +
v2papariah + v2paelcont)
```

Coefficients:

	(Intercept)	psla	elect_system1	elect_system2	elect_system3
0.027821	0.256712	0.024796	0.037849	0.054959	
v2x_polyarchy	v2pariglef	v2papariah	v2paelcont1	v2paelcont2	
-0.054333	0.006713	-0.003638	-0.007241	-0.018432	

```
##### The best formula from all three stepwise selections is this:
```

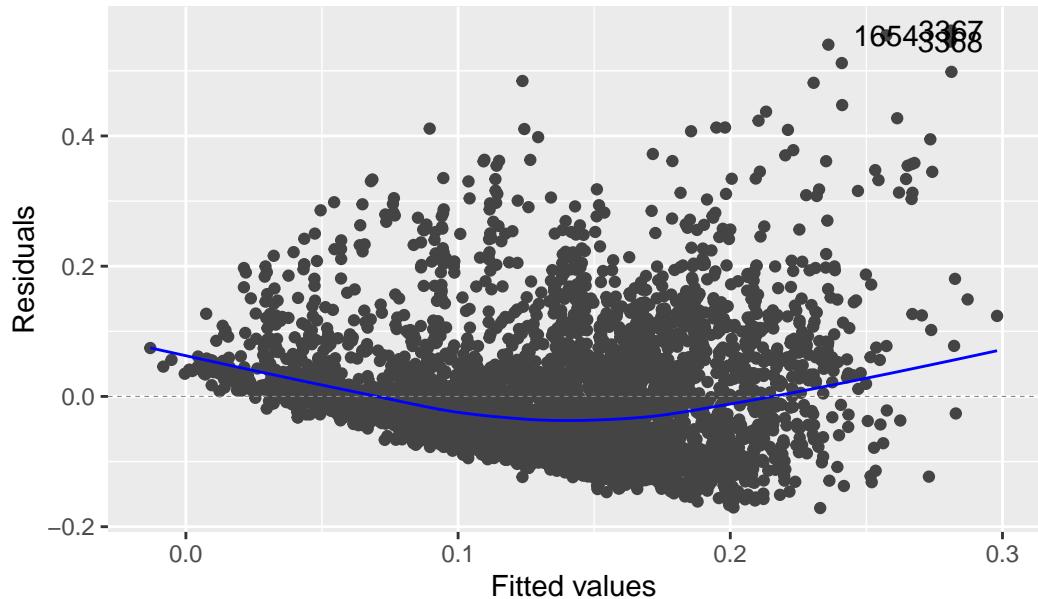
```
f1 <- reg_vs ~ psla + elect_system + v2x_polyarchy + v2pariglef +
v2papariah + v2paelcont
```

Next is fitting the model and looking at the assumption

```
m1 <- lm(f1)

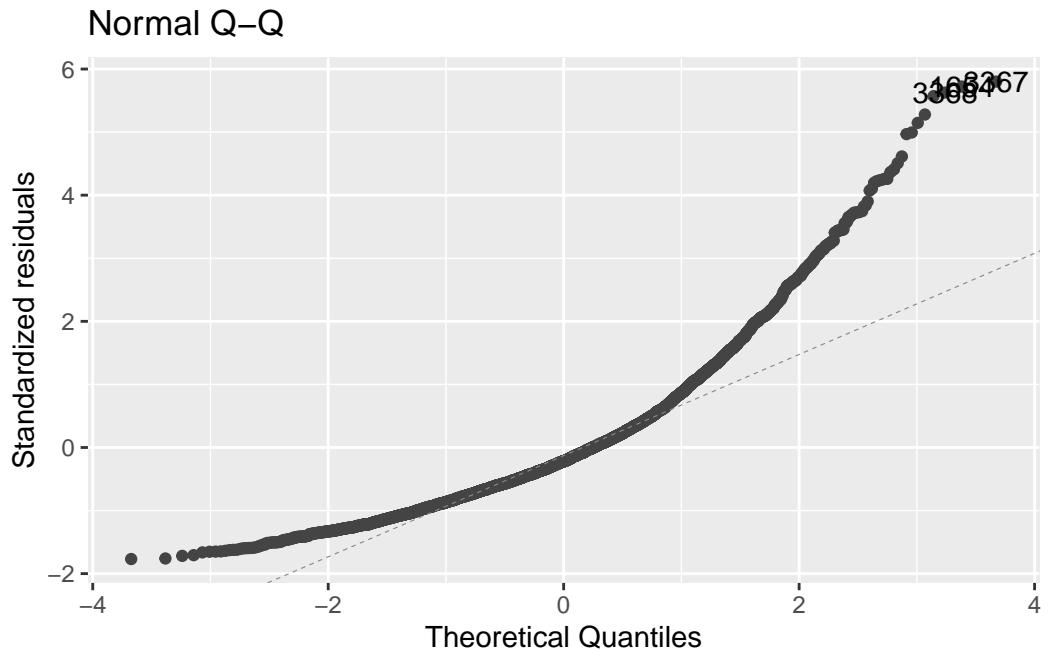
autoplot(m1, which = 1, nrow = 1, ncol = 1)
```

## Residuals vs Fitted

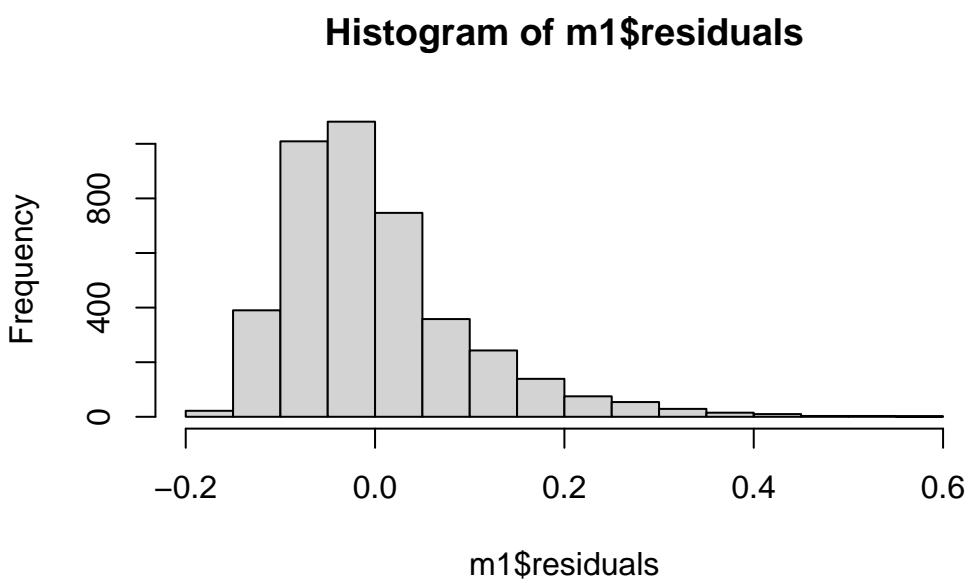


```
#### Linearity looks surprisingly good here.
```

```
autoplot(m1, which = 2, nrow = 1, ncol = 1)
```

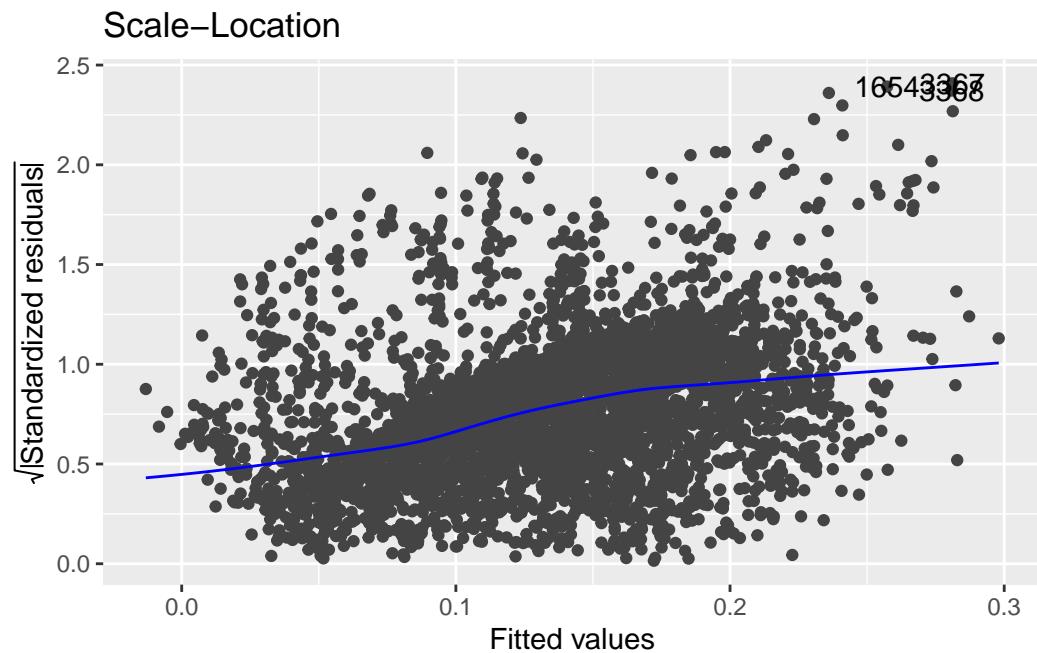


```
hist(m1$residuals)
```



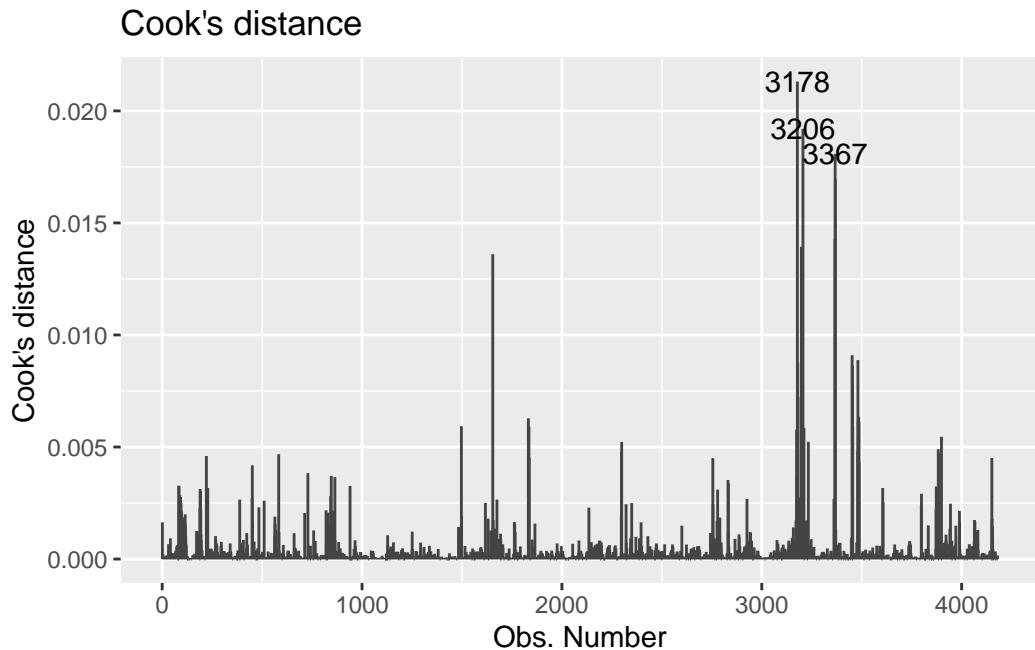
```
#### Errors are not normal!
```

```
autoplot(m1, which = 3, nrow = 1, ncol = 1)
```



```
#### There might be some heterogeneity, but its pretty good here.
```

```
autoplot(m1, which = 4, nrow = 1, ncol = 1)
```



```
#### Cooks Distance looks pretty good
```

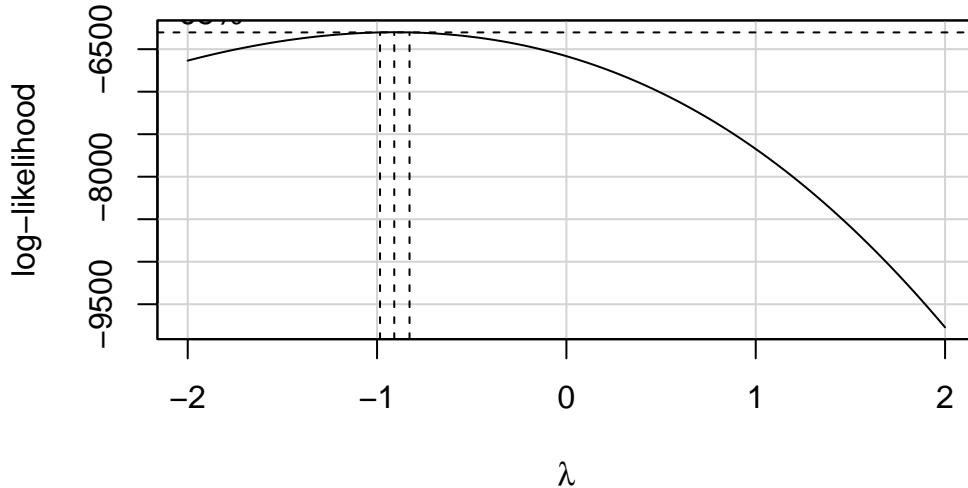
We needed a Box Cox transformation. This code gave us an optimal transformation.

```
f1
```

```
reg_vs ~ psla + elect_system + v2x_polyarchy + v2pariglef + v2papariah +
v2paelcont
```

```
bc_results <- boxCox((reg_vs+.1) ~ psla + elect_system + v2x_polyarchy + v2pariglef + v2pa
v2paelcont)
```

## Profile Log-likelihood



```
#### It looks like we should mutate the DV and make it reg_vs^-1
```

```
dat_new <- dat %>%
  mutate(reg_vs = (reg_vs + .1)^(-1))
```

Recheck assumptions to see if this helped

```
attach(dat_new)
```

The following objects are masked from dat (pos = 3):

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,
v2xpa_antiplural
```

The following objects are masked from dat\_center:

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,
```

```
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,  
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,  
v2xpa_antiplural
```

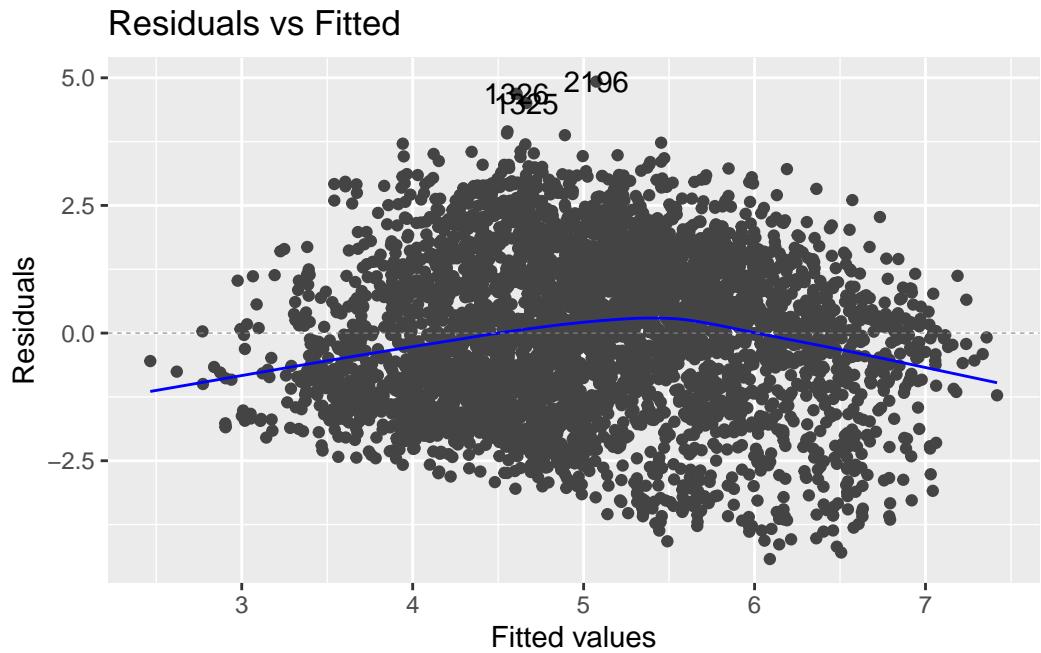
The following objects are masked from dat\_new (pos = 5):

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,  
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,  
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,  
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,  
v2xpa_antiplural
```

The following objects are masked from dat (pos = 6):

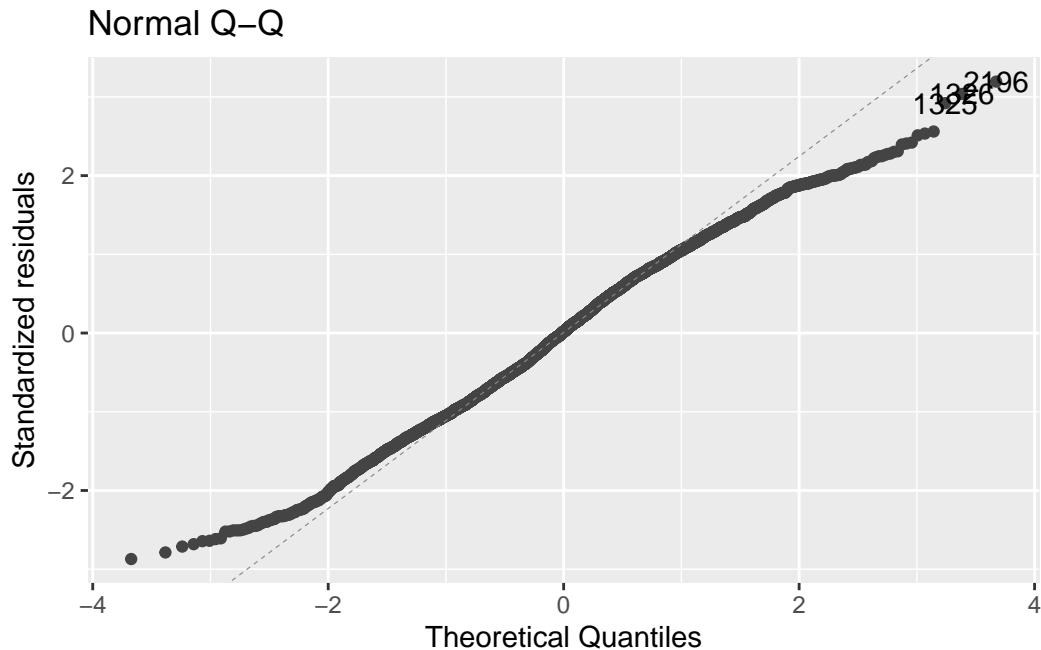
```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,  
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,  
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,  
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,  
v2xpa_antiplural
```

```
m2 <- lm(f1)  
  
autoplot(m2, which = 1, nrow = 1, ncol = 1)
```

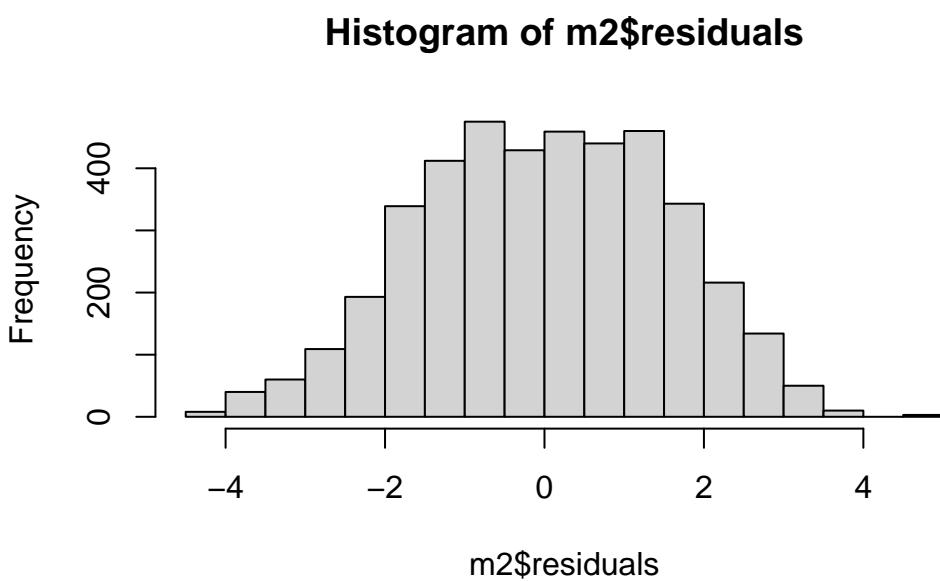


```
#### Linearity looks surprisingly good here.
```

```
autoplot(m2, which = 2, nrow = 1, ncol = 1)
```

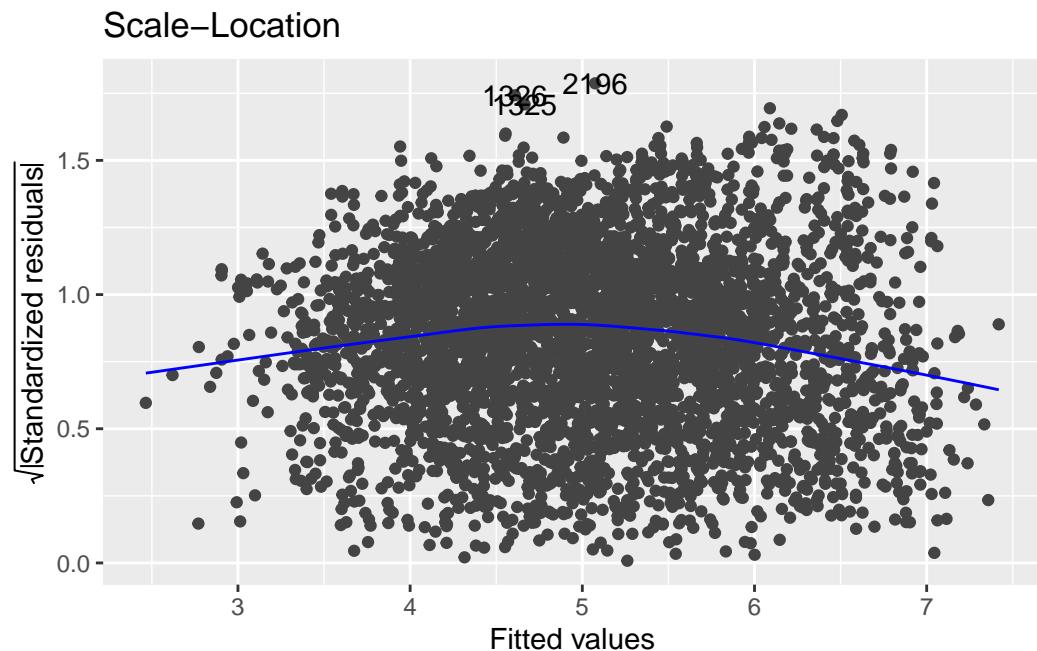


```
hist(m2$residuals)
```



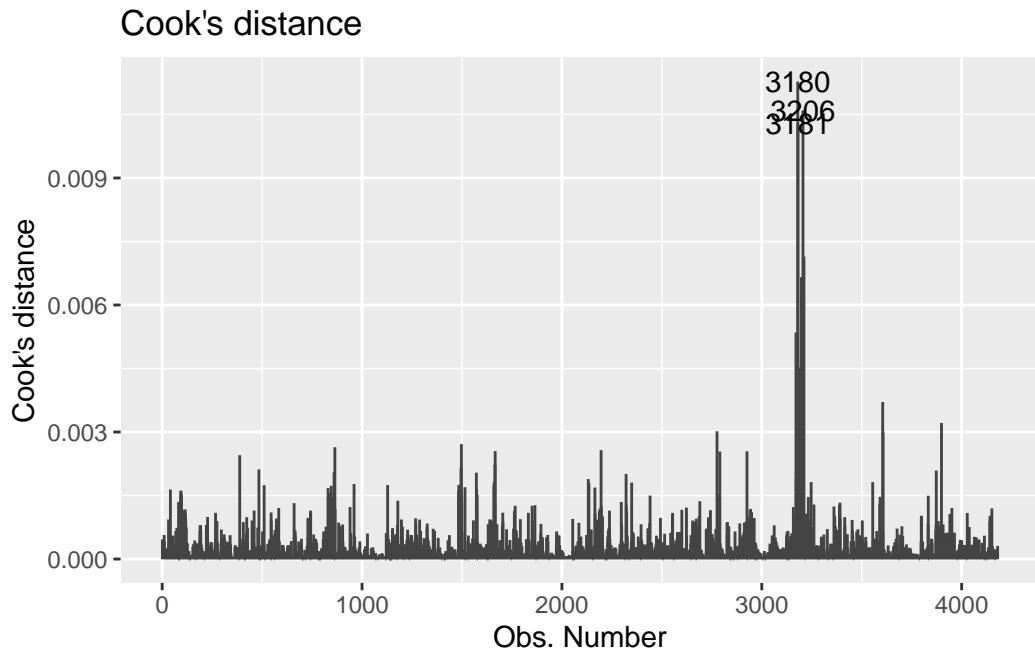
```
#### Errors are more normal!
```

```
autoplot(m2, which = 3, nrow = 1, ncol = 1)
```



```
#### There might be some heterogeneity, but its pretty good here.
```

```
autoplot(m2, which = 4, nrow = 1, ncol = 1)
```



```
#### Cooks Distance looks pretty good
```

To ensure that we have the best model here, lets rerun this with the stepwise selection

```
base_model <- lm(reg_vs ~ 1)
full_model <- lm(reg_vs ~ elect_system + v2pariglef + psla + v2papariah + exe_elect + v2pa

forward <- step(base_model,
                  direction = "forward",
                  k = 2,
                  scope = list(lower = base_model, upper = full_model))
```

```
Start: AIC=4762.29
reg_vs ~ 1

          Df Sum of Sq    RSS    AIC
+ psla      1   2667.60 10387 3808.8
+ v2paelcont 2    433.67 12621 4625.1
+ elect_system 3    181.05 12874 4709.9
+ v2pariglef  1     52.79 13002 4747.3
+ v2x_polyarchy 1    11.13 13043 4760.7
<none>                 13054 4762.3
```

```

+ exe_elect      1      3.77 13051 4763.1
+ v2xpa_antiplural 1      0.62 13054 4764.1
+ v2papariah    1      0.54 13054 4764.1

```

Step: AIC=3808.79

reg\_vs ~ psla

	Df	Sum of Sq	RSS	AIC
+ elect_system	3	233.272	10154	3719.8
+ v2pariglef	1	168.310	10219	3742.5
+ v2x_polyarchy	1	23.334	10364	3801.4
+ exe_elect	1	19.974	10367	3802.7
+ v2paelcont	2	17.099	10370	3805.9
+ v2xpa_antiplural	1	6.269	10381	3808.3
+ v2papariah	1	5.295	10382	3808.7
<none>			10387	3808.8

Step: AIC=3719.84

reg\_vs ~ psla + elect\_system

	Df	Sum of Sq	RSS	AIC
+ v2pariglef	1	155.227	9998.4	3657.4
+ v2x_polyarchy	1	34.550	10119.1	3707.6
+ v2papariah	1	9.198	10144.5	3718.1
+ v2paelcont	2	10.646	10143.0	3719.5
+ v2xpa_antiplural	1	5.253	10148.4	3719.7
<none>			10153.7	3719.8
+ exe_elect	1	0.465	10153.2	3721.7

Step: AIC=3657.45

reg\_vs ~ psla + elect\_system + v2pariglef

	Df	Sum of Sq	RSS	AIC
+ v2x_polyarchy	1	33.110	9965.3	3645.6
+ v2papariah	1	18.972	9979.5	3651.5
+ v2paelcont	2	12.374	9986.1	3656.3
<none>			9998.4	3657.4
+ v2xpa_antiplural	1	0.957	9997.5	3659.0
+ exe_elect	1	0.017	9998.4	3659.4

Step: AIC=3645.58

reg\_vs ~ psla + elect\_system + v2pariglef + v2x\_polyarchy

```

          Df Sum of Sq   RSS   AIC
+ v2xpa_antiplural  1   12.2209 9953.1 3642.5
+ v2paelcont        2   16.0779 9949.2 3642.8
+ v2papariah        1    9.3837 9955.9 3643.6
<none>                  9965.3 3645.6
+ exe_elect         1   0.9103 9964.4 3647.2

Step: AIC=3642.45
reg_vs ~ psla + elect_system + v2pariglelf + v2x_polyarchy + v2xpa_antiplural

          Df Sum of Sq   RSS   AIC
+ v2papariah  1   15.865 9937.2 3637.8
+ v2paelcont  2   17.252 9935.8 3639.2
<none>                  9953.1 3642.5
+ exe_elect   1   0.422 9952.7 3644.3

Step: AIC=3637.79
reg_vs ~ psla + elect_system + v2pariglelf + v2x_polyarchy + v2xpa_antiplural +
v2papariah

          Df Sum of Sq   RSS   AIC
+ v2paelcont  2   17.6362 9919.6 3634.4
<none>                  9937.2 3637.8
+ exe_elect   1   0.5496 9936.7 3639.6

Step: AIC=3634.36
reg_vs ~ psla + elect_system + v2pariglelf + v2x_polyarchy + v2xpa_antiplural +
v2papariah + v2paelcont

          Df Sum of Sq   RSS   AIC
<none>                  9919.6 3634.4
+ exe_elect  1   0.3452 9919.3 3636.2

back <- step(full_model,
              direction = "backward",
              k = 2,
              scope = list(lower = base_model, upper = full_model))

Start: AIC=3636.22
reg_vs ~ elect_system + v2pariglelf + psla + v2papariah + exe_elect +
v2paelcont + v2xpa_antiplural + v2x_polyarchy

```

	Df	Sum of Sq	RSS	AIC
- exe_elect	1	0.35	9919.6	3634.4
<none>			9919.3	3636.2
- v2paelcont	2	17.43	9936.7	3639.6
- v2papariah	1	16.35	9935.6	3641.1
- v2xpa_antiplural	1	19.66	9938.9	3642.5
- v2x_polyarchy	1	47.08	9966.3	3654.0
- v2pariglef	1	176.83	10096.1	3708.1
- elect_system	3	223.39	10142.6	3723.3
- psla	1	2464.31	12383.6	4561.7

Step: AIC=3634.36

```
reg_vs ~ elect_system + v2pariglef + psla + v2papariah + v2paelcont +
v2xpa_antiplural + v2x_polyarchy
```

	Df	Sum of Sq	RSS	AIC
<none>			9919.6	3634.4
- v2paelcont	2	17.64	9937.2	3637.8
- v2papariah	1	16.25	9935.8	3639.2
- v2xpa_antiplural	1	20.22	9939.8	3640.9
- v2x_polyarchy	1	46.73	9966.3	3652.0
- v2pariglef	1	176.51	10096.1	3706.1
- elect_system	3	234.15	10153.8	3725.9
- psla	1	2481.10	12400.7	4565.5

```
both <- step(base_model,
              direction = "both",
              k = 2,
              scope = list(lower = base_model, upper = full_model))
```

Start: AIC=4762.29

```
reg_vs ~ 1
```

	Df	Sum of Sq	RSS	AIC
+ psla	1	2667.60	10387	3808.8
+ v2paelcont	2	433.67	12621	4625.1
+ elect_system	3	181.05	12874	4709.9
+ v2pariglef	1	52.79	13002	4747.3
+ v2x_polyarchy	1	11.13	13043	4760.7
<none>		13054	4762.3	

+ exe_elect	1	3.77	13051	4763.1
+ v2xpa_antiplural	1	0.62	13054	4764.1
+ v2papariah	1	0.54	13054	4764.1

Step: AIC=3808.79

reg\_vs ~ psla

	Df	Sum of Sq	RSS	AIC
+ elect_system	3	233.27	10154	3719.8
+ v2pariglef	1	168.31	10219	3742.5
+ v2x_polyarchy	1	23.33	10364	3801.4
+ exe_elect	1	19.97	10367	3802.7
+ v2paelcont	2	17.10	10370	3805.9
+ v2xpa_antiplural	1	6.27	10381	3808.3
+ v2papariah	1	5.30	10382	3808.7
<none>			10387	3808.8
- psla	1	2667.60	13054	4762.3

Step: AIC=3719.84

reg\_vs ~ psla + elect\_system

	Df	Sum of Sq	RSS	AIC
+ v2pariglef	1	155.23	9998.4	3657.4
+ v2x_polyarchy	1	34.55	10119.1	3707.6
+ v2papariah	1	9.20	10144.5	3718.1
+ v2paelcont	2	10.65	10143.0	3719.5
+ v2xpa_antiplural	1	5.25	10148.4	3719.7
<none>			10153.7	3719.8
+ exe_elect	1	0.46	10153.2	3721.7
- elect_system	3	233.27	10386.9	3808.8
- psla	1	2719.83	12873.5	4709.9

Step: AIC=3657.45

reg\_vs ~ psla + elect\_system + v2pariglef

	Df	Sum of Sq	RSS	AIC
+ v2x_polyarchy	1	33.11	9965.3	3645.6
+ v2papariah	1	18.97	9979.5	3651.5
+ v2paelcont	2	12.37	9986.1	3656.3
<none>			9998.4	3657.4
+ v2xpa_antiplural	1	0.96	9997.5	3659.0
+ exe_elect	1	0.02	9998.4	3659.4
- v2pariglef	1	155.23	10153.7	3719.8

- elect_system	3	220.19	10218.6	3742.5
- psla	1	2832.64	12831.1	4698.1

Step: AIC=3645.58

reg\_vs ~ psla + elect\_system + v2pariglelf + v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ v2xpa_antiplural	1	12.22	9953.1	3642.5
+ v2paelcont	2	16.08	9949.2	3642.8
+ v2papariah	1	9.38	9955.9	3643.6
<none>			9965.3	3645.6
+ exe_elect	1	0.91	9964.4	3647.2
- v2x_polyarchy	1	33.11	9998.4	3657.4
- v2pariglelf	1	153.79	10119.1	3707.6
- elect_system	3	231.59	10196.9	3735.6
- psla	1	2852.78	12818.1	4695.9

Step: AIC=3642.45

reg\_vs ~ psla + elect\_system + v2pariglelf + v2x\_polyarchy + v2xpa\_antiplural

	Df	Sum of Sq	RSS	AIC
+ v2papariah	1	15.86	9937.2	3637.8
+ v2paelcont	2	17.25	9935.8	3639.2
<none>			9953.1	3642.5
+ exe_elect	1	0.42	9952.7	3644.3
- v2xpa_antiplural	1	12.22	9965.3	3645.6
- v2x_polyarchy	1	44.37	9997.5	3659.0
- v2pariglelf	1	162.52	10115.6	3708.2
- elect_system	3	237.38	10190.5	3735.0
- psla	1	2864.82	12817.9	4697.8

Step: AIC=3637.79

reg\_vs ~ psla + elect\_system + v2pariglelf + v2x\_polyarchy + v2xpa\_antiplural + v2papariah

	Df	Sum of Sq	RSS	AIC
+ v2paelcont	2	17.64	9919.6	3634.4
<none>			9937.2	3637.8
+ exe_elect	1	0.55	9936.7	3639.6
- v2papariah	1	15.86	9953.1	3642.5
- v2xpa_antiplural	1	18.70	9955.9	3643.6
- v2x_polyarchy	1	42.00	9979.2	3653.4
- v2pariglelf	1	173.83	10111.1	3708.3

```

- elect_system      3    242.90 10180.1 3732.7
- psla             1    2879.81 12817.0 4699.5

Step: AIC=3634.36
reg_vs ~ psla + elect_system + v2pariglef + v2x_polyarchy + v2xpa_antiplural +
v2papariah + v2paelcont

```

	Df	Sum of Sq	RSS	AIC
<none>			9919.6	3634.4
+ exe_elect	1	0.35	9919.3	3636.2
- v2paelcont	2	17.64	9937.2	3637.8
- v2papariah	1	16.25	9935.8	3639.2
- v2xpa_antiplural	1	20.22	9939.8	3640.9
- v2x_polyarchy	1	46.73	9966.3	3652.0
- v2pariglef	1	176.51	10096.1	3706.1
- elect_system	3	234.15	10153.8	3725.9
- psla	1	2481.10	12400.7	4565.5

forward

Call:

```
lm(formula = reg_vs ~ psla + elect_system + v2pariglef + v2x_polyarchy +
v2xpa_antiplural + v2papariah + v2paelcont)
```

Coefficients:

	(Intercept)	psla	elect_system1	elect_system2
	6.87189	-4.27579	-0.39810	-0.58638
elect_system3	-1.10528	v2pariglef	v2x_polyarchy	v2xpa_antiplural
		-0.14492	0.65962	0.32292
v2papariah	0.04857	v2paelcont1	v2paelcont2	
		0.14113	0.22338	

back

Call:

```
lm(formula = reg_vs ~ elect_system + v2pariglef + psla + v2papariah +
v2paelcont + v2xpa_antiplural + v2x_polyarchy)
```

Coefficients:

	elect_system1	elect_system2	elect_system3
(Intercept)	6.87189	-0.39810	-0.58638
v2pariglef		psla	v2papariah
	-0.14492	-4.27579	0.04857
v2paelcont2	v2xpa_antiplural	v2x_polyarchy	v2paelcont1
	0.22338	0.32292	0.14113

both

Call:

```
lm(formula = reg_vs ~ psla + elect_system + v2pariglef + v2x_polyarchy +
    v2xpa_antiplural + v2papariah + v2paelcont)
```

Coefficients:

	psla	elect_system1	elect_system2
(Intercept)	6.87189	-4.27579	-0.39810
elect_system3	v2pariglef	v2x_polyarchy	v2xpa_antiplural
	-1.10528	-0.14492	0.65962
v2papariah	v2paelcont1	v2paelcont2	0.32292
	0.04857	0.14113	0.22338

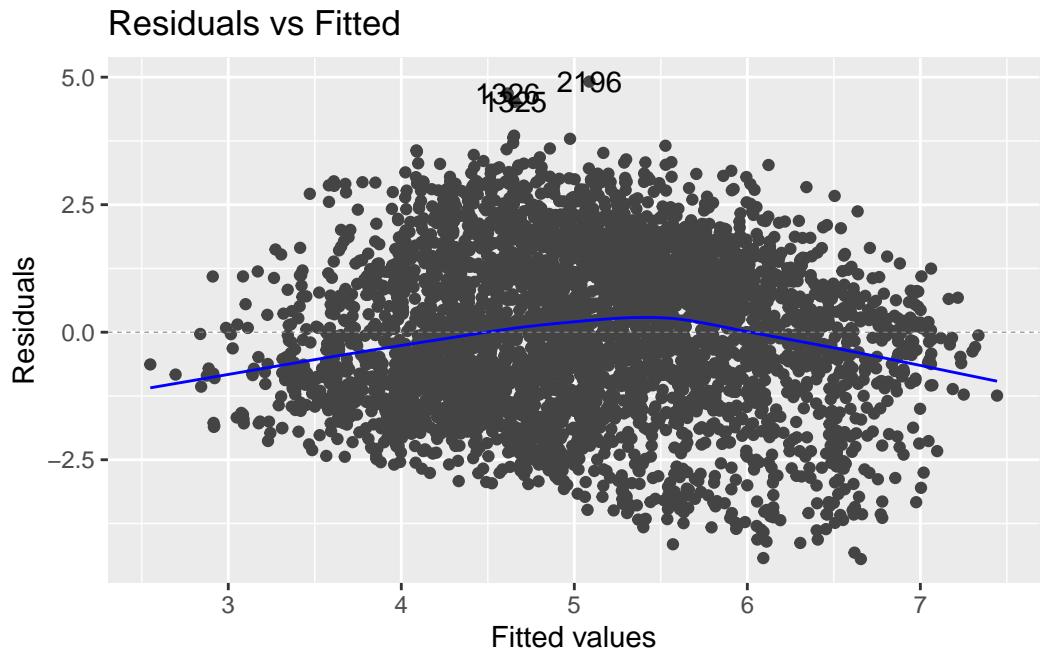
#### All of the selection methods choose this model, so we will go with that:

```
f2 <- reg_vs ~ psla + elect_system + v2pariglef + v2x_polyarchy +
    v2xpa_antiplural + v2papariah + v2paelcont
```

Recheck assumptions to see if the model is up to snuff.

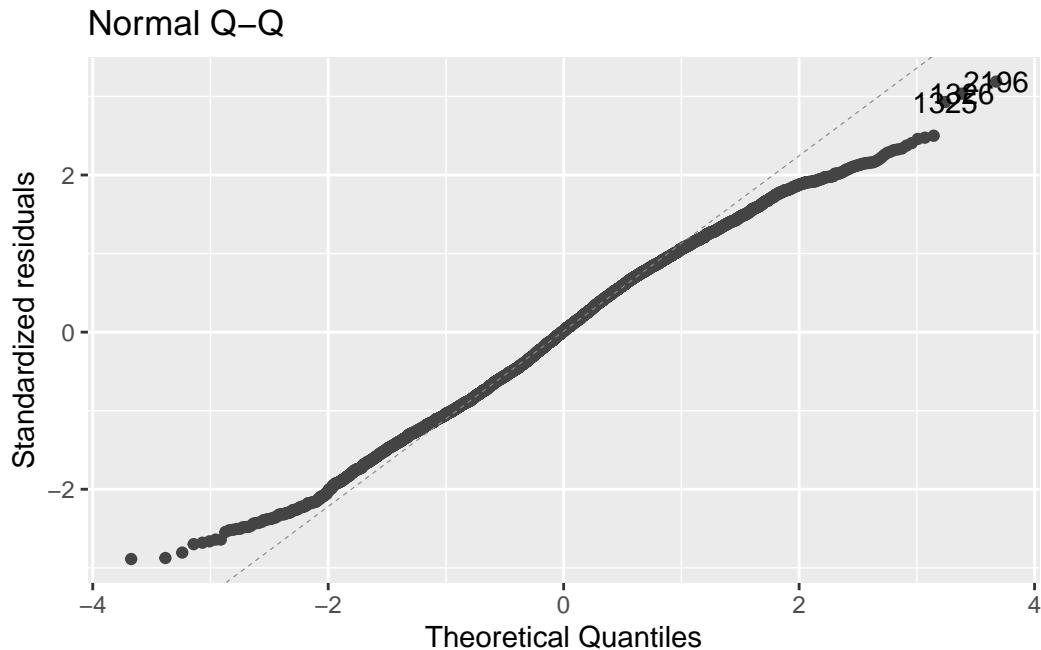
```
m3 <- lm(f2)

autoplot(m3, which = 1, nrow = 1, ncol = 1)
```



```
#### Linearity looks good here.
```

```
autoplot(m3, which = 2, nrow = 1, ncol = 1)
```



```
shapiro.test(m3$residuals)
```

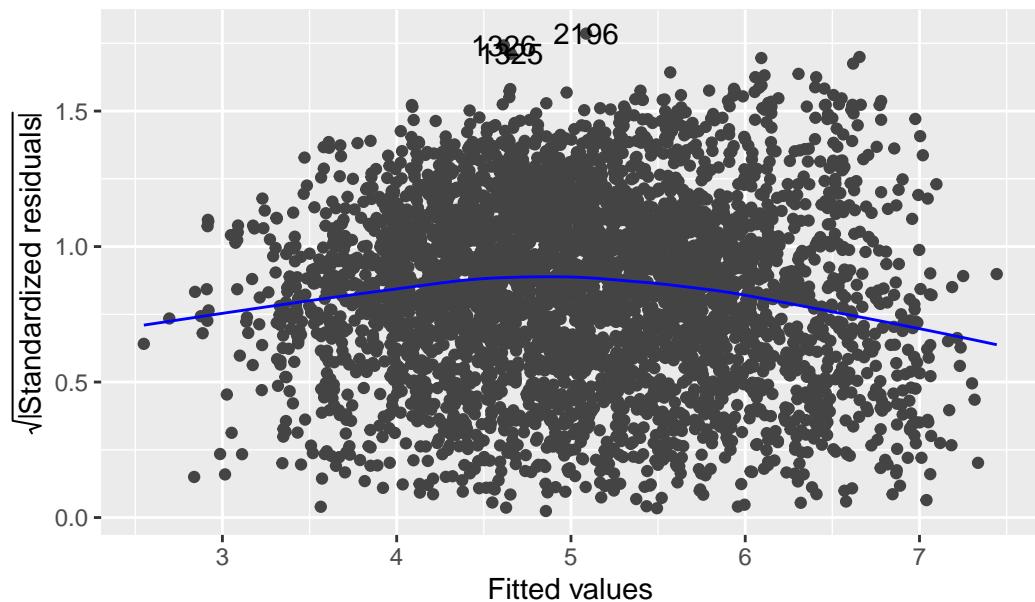
Shapiro-Wilk normality test

```
data: m3$residuals
W = 0.99383, p-value = 2.093e-12
```

#### Errors are more normal than the other models. Best we are probaly going to get.

```
autoplot(m3, which = 3, nrow = 1, ncol = 1)
```

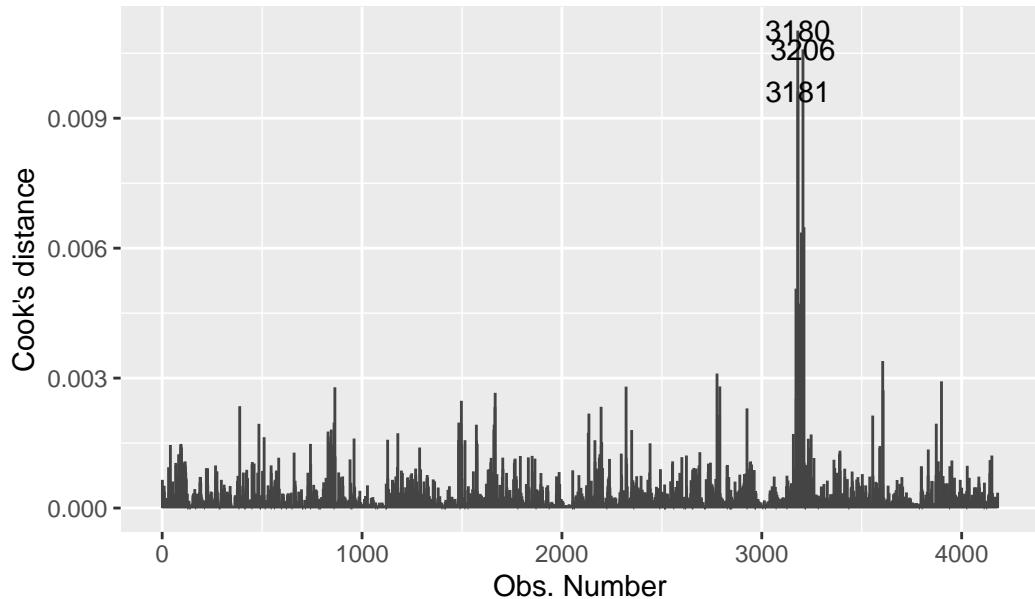
## Scale–Location



```
#### There might be some heterogeneity, but its pretty good here.
```

```
autoplot(m3, which = 4, nrow = 1, ncol = 1)
```

## Cook's distance



```
##### no outliers
```

```
car::vif(m3)
```

	GVIF	Df	GVIF^(1/(2*Df))
psla	1.208220	1	1.099191
elect_system	1.097085	3	1.015563
v2pariglef	1.055315	1	1.027285
v2x_polyarchy	1.812839	1	1.346417
v2xpa_antiplural	1.862623	1	1.364779
v2papariah	1.150434	1	1.072583
v2paelcont	1.174909	2	1.041121

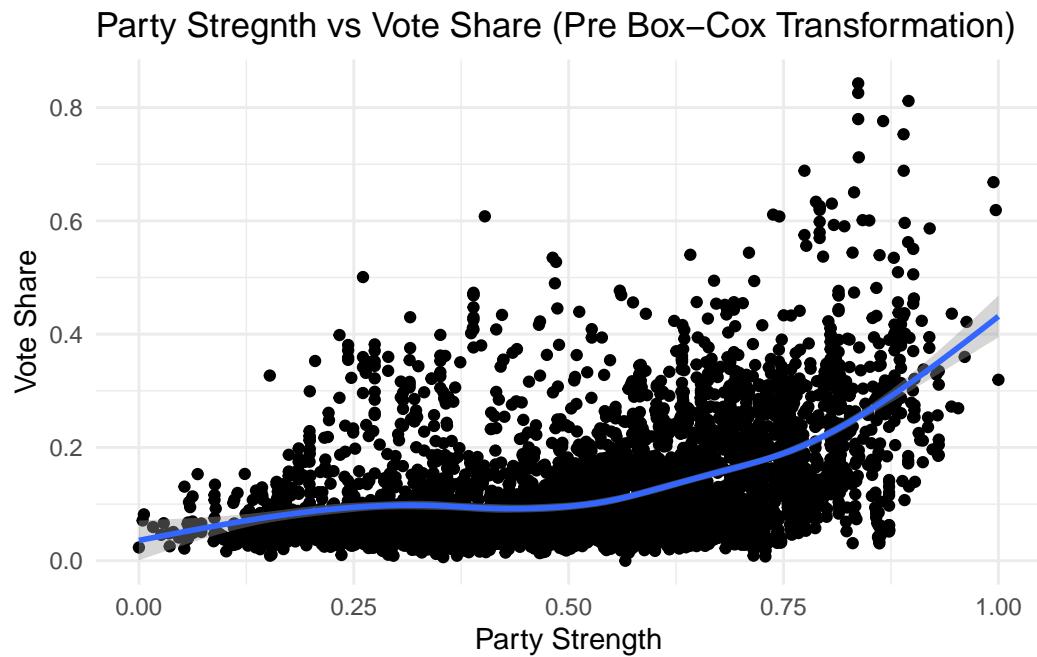
```
##### No multicollinearity!
```

Now for exploratory data analysis

```
ggplot(data = dat, aes(x = psla, y = reg_vs)) +  
  geom_point() +  
  geom_smooth() +  
  labs(title = "Party Strength vs Vote Share (Pre Box-Cox Transformation)", x = "Party Str
```

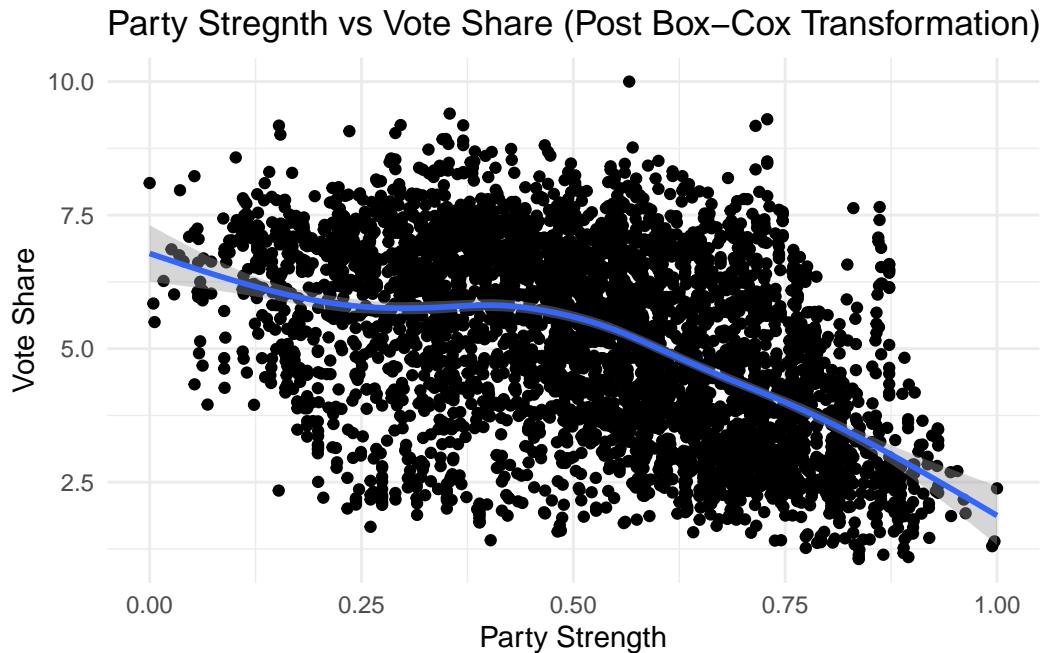
```
theme_minimal()
```

```
`geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```



```
ggplot(data = dat_new, aes(x = psla, y = reg_vs)) +  
  geom_point() +  
  geom_smooth() +  
  labs(title = "Party Strength vs Vote Share (Post Box–Cox Transformation)", x = "Party St  
theme_minimal()
```

```
`geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

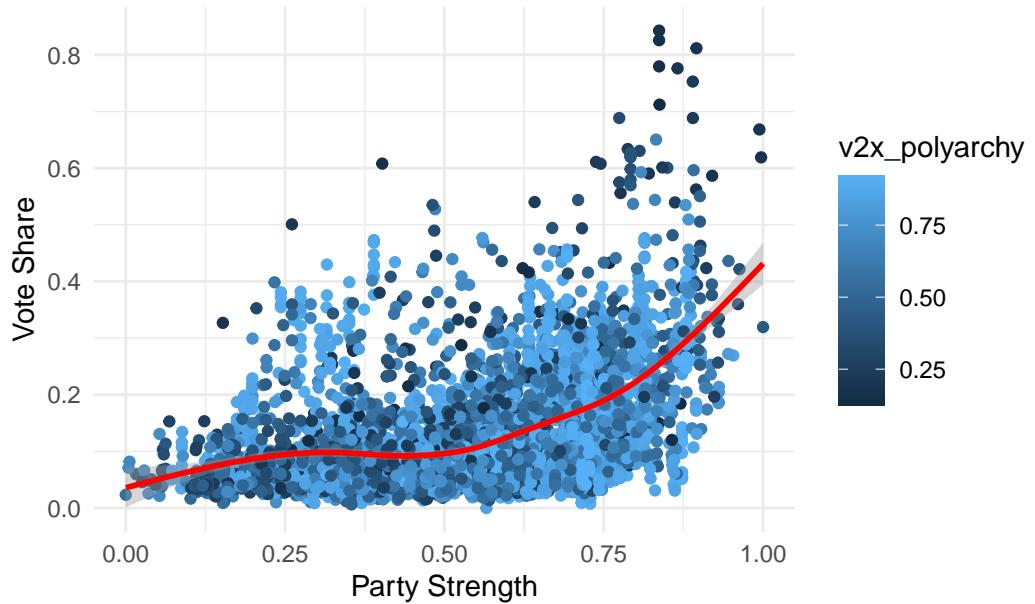


```
#### for Question 2
```

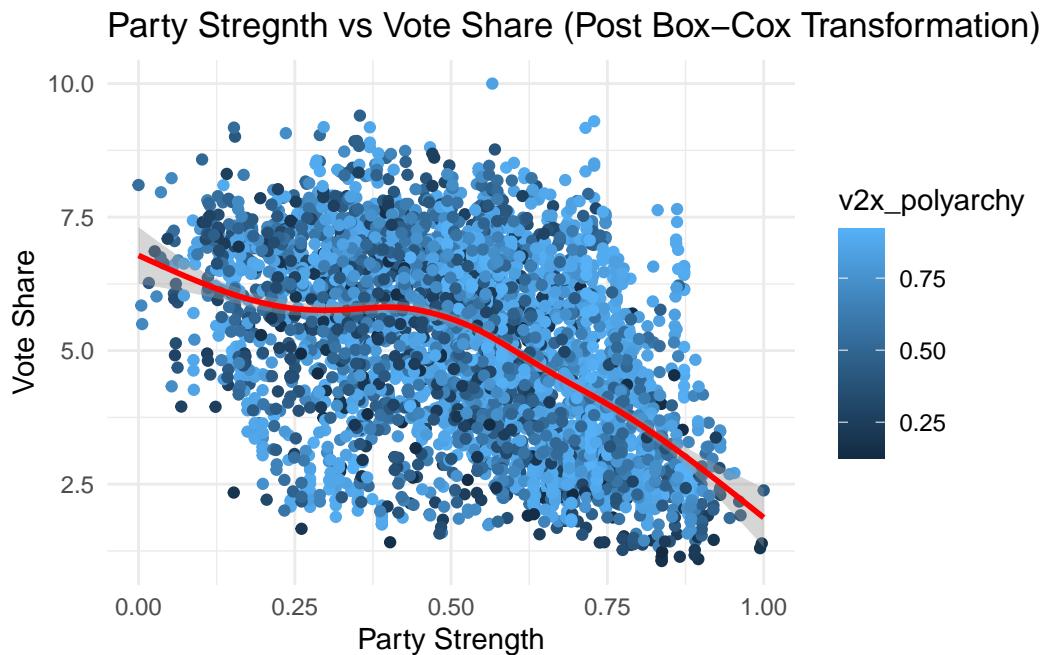
```
ggplot(data = dat, aes(x = psla, y = reg_vs)) +
  geom_point(aes(color = v2x_polyarchy)) +
  geom_smooth(color = "red") +
  labs(title = "Party Strength vs Vote Share (Pre Box–Cox Transformation)", x = "Party Strength", y = "Vote Share")
  theme_minimal()
```

```
`geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

Party Strength vs Vote Share (Pre Box–Cox Transformation)

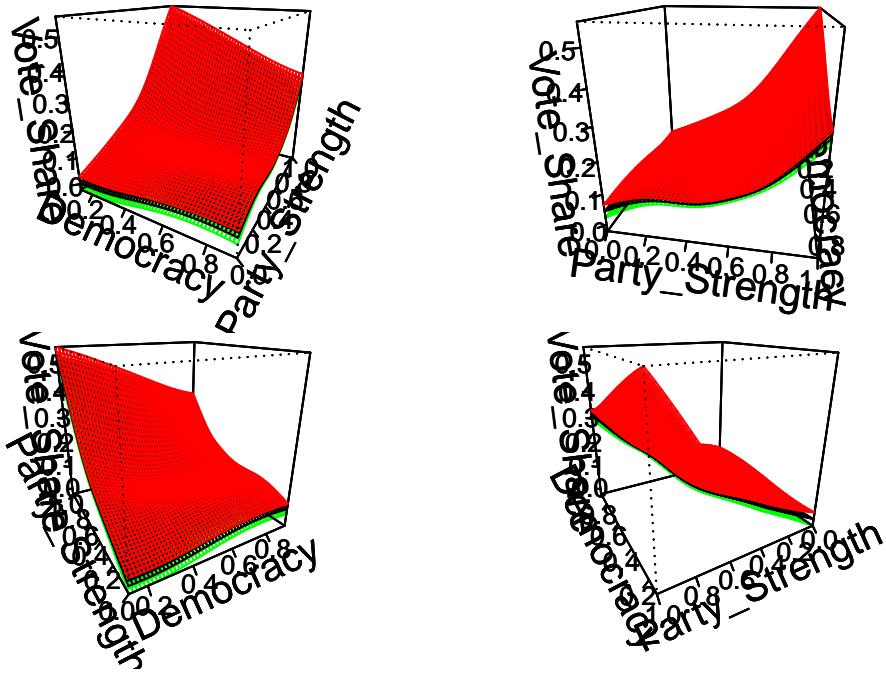


```
ggplot(data = dat_new, aes(x = psla, y = reg_vs)) +  
  geom_point(aes(color = v2x_polyarchy)) +  
  geom_smooth(color = "red") +  
  labs(title = "Party Strength vs Vote Share (Post Box-Cox Transformation)", x = "Party St  
theme_minimal()  
  
`geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```



```
dat_interflex <- dat %>%
  rename(Democracy = v2x_polyarchy, Vote_Share = reg_vs, Party_Strength = psla)

interflex(estimator = "gam", Y = "Vote_Share", X = "Party_Strength", D = "Democracy", data =
```



Family: gaussian

Link function: identity

Formula:

$\text{Vote\_Share} \sim s(\text{Democracy}, \text{Party\_Strength}, k = 10)$

Estimated degrees of freedom:

8.83 total = 9.83

GCV score: 0.008689635

Regression models

```
summary(m3) ##### OLS
```

Call:

```
lm(formula = f2)
```

Residuals:

```

      Min       1Q   Median     3Q    Max
-4.4470 -1.1404  0.0217  1.1804  4.9130

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 6.87189  0.14306 48.036 < 2e-16 ***
psla        -4.27579  0.13241 -32.292 < 2e-16 ***
elect_system1 -0.39810  0.07956 -5.003 5.86e-07 ***
elect_system2 -0.58638  0.07041 -8.328 < 2e-16 ***
elect_system3 -1.10528  0.27151 -4.071 4.77e-05 ***
v2pariglef   -0.14492  0.01683 -8.613 < 2e-16 ***
v2x_polyarchy 0.65962  0.14883  4.432 9.58e-06 ***
v2xpa_antiplural 0.32292  0.11077  2.915 0.00357 **
v2papariah   0.04857  0.01859  2.613 0.00900 **
v2paelcont1  0.14113  0.07382  1.912 0.05596 .
v2paelcont2  0.22338  0.10421  2.144 0.03212 *
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

```

Residual standard error: 1.543 on 4169 degrees of freedom
Multiple R-squared: 0.2401, Adjusted R-squared: 0.2383
F-statistic: 131.8 on 10 and 4169 DF, p-value: < 2.2e-16

```

```

##### Good F stat and r squared value is about what is industry standard for poli sci. Most

##### partial F test here

##### How important is the effect of party strength

m3 <- lm(reg_vs ~ psla + elect_system + v2pariglef + v2x_polyarchy + v2xpa_antiplural + v2papariah + v2paelcont)

m4 <- lm(reg_vs ~ elect_system + v2pariglef + v2x_polyarchy + v2xpa_antiplural + v2papariah + v2paelcont)

anova(m4, m3) ##### Party strength is really important

```

#### Analysis of Variance Table

```

Model 1: reg_vs ~ elect_system + v2pariglef + v2x_polyarchy + v2xpa_antiplural +
          v2papariah + v2paelcont
Model 2: reg_vs ~ psla + elect_system + v2pariglef + v2x_polyarchy + v2xpa_antiplural +
          v2papariah + v2paelcont

```

```

Res.Df      RSS Df Sum of Sq      F    Pr(>F)
1     4170 12400.7
2     4169  9919.6  1     2481.1 1042.8 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

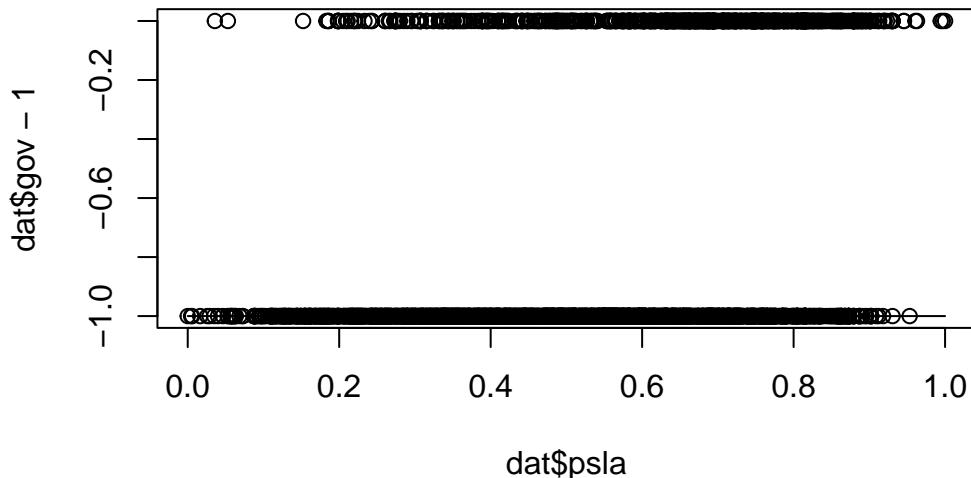
```

Logit model. Mostly a robustness check, so we did not go too crazy here.

```

robust <- glm(gov ~ psla + elect_system + v2pariglef + v2x_polyarchy + v2xpa_antiplural +
scatter.smooth(x = dat$psla, y = dat$gov - 1)

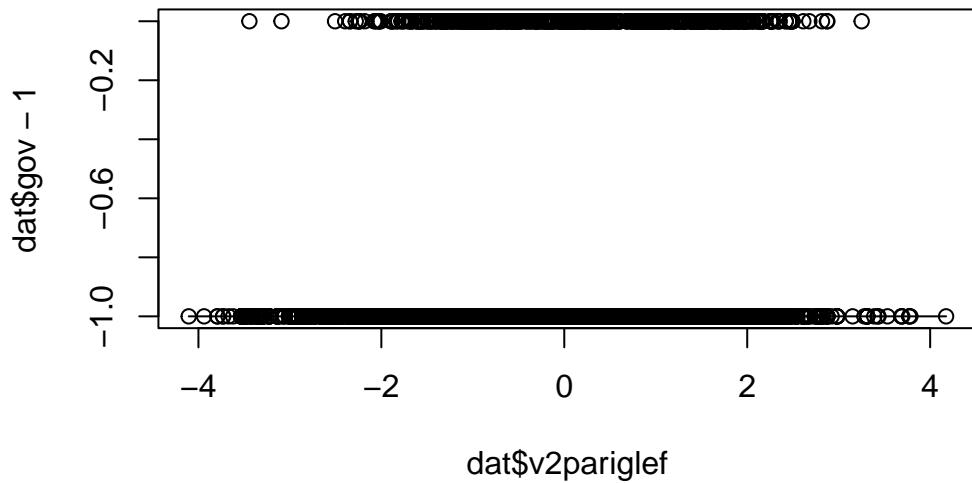
```



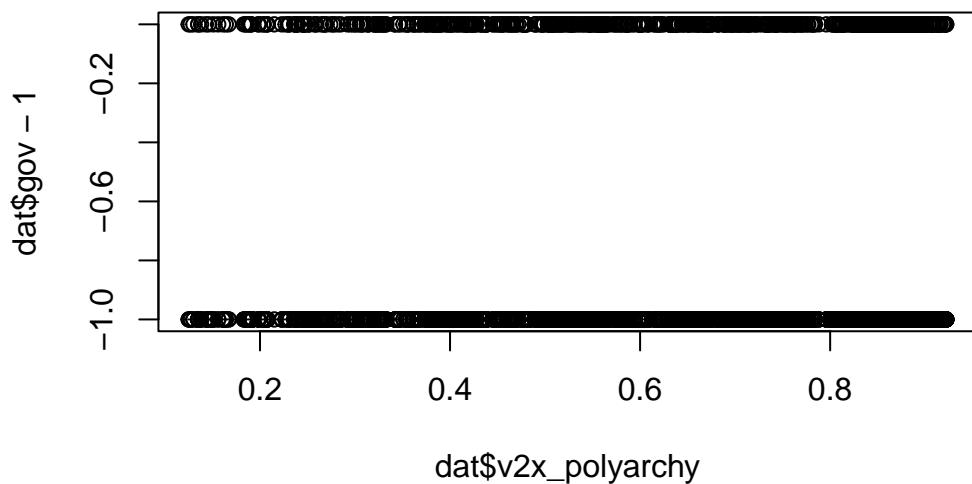
```

scatter.smooth(x = dat$v2pariglef, y = dat$gov - 1)

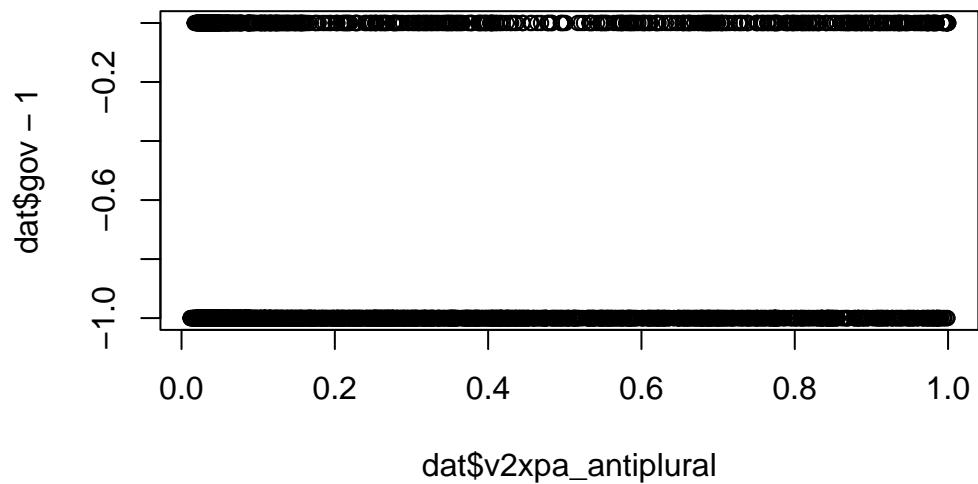
```



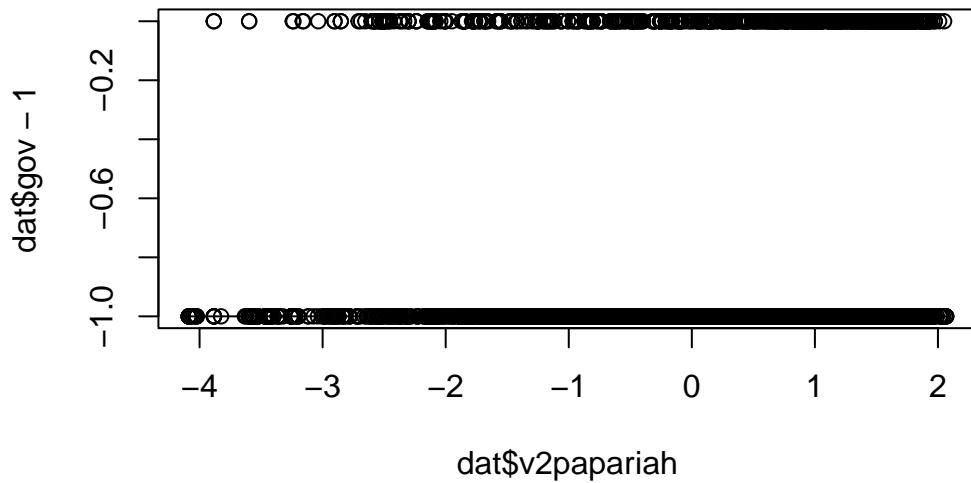
```
scatter.smooth(x = dat$v2x_polyarchy, y = dat$gov - 1)
```



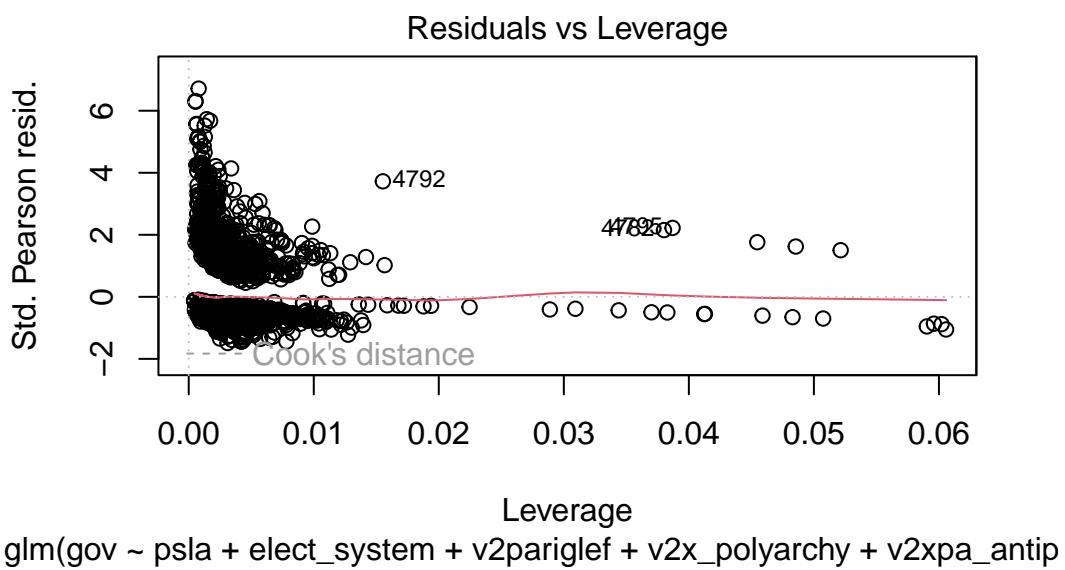
```
scatter.smooth(x = dat$v2xpa_antiplural, y = dat$gov - 1)
```



```
scatter.smooth(x = dat$v2papariah, y = dat$gov - 1)
```



```
plot(robust, which = 5, cook.levels = .5)
```



```
vif(robust)
```

	GVIF	Df	GVIF^(1/(2*Df))
psla	1.211162	1	1.100528
elect_system	1.100599	3	1.016104
v2pariglef	1.052634	1	1.025980
v2x_polyarchy	2.011111	1	1.418136
v2xpa_antiplural	2.066863	1	1.437659
v2papariah	1.191706	1	1.091653
v2paelcont	1.157573	2	1.037259

```
summary(robust)
```

Call:

```
glm(formula = gov ~ psla + elect_system + v2pariglef + v2x_polyarchy +
    v2xpa_antiplural + v2papariah + v2paelcont, family = binomial(link = "logit"),
    data = dat_new)
```

Coefficients:

	Estimate	Std. Error	z value	Pr(> z )
(Intercept)	-4.30413	0.28253	-15.234	< 2e-16 ***
psla	5.07036	0.27394	18.509	< 2e-16 ***
elect_system1	-0.14543	0.13275	-1.096	0.273263
elect_system2	0.15582	0.12720	1.225	0.220562
elect_system3	0.37756	0.48677	0.776	0.437966
v2pariglef	0.20123	0.02996	6.716	1.87e-11 ***
v2x_polyarchy	-0.53430	0.27120	-1.970	0.048829 *
v2xpa_antiplural	0.77461	0.19129	4.049	5.13e-05 ***
v2papariah	0.12071	0.03429	3.520	0.000431 ***
v2paelcont1	0.20006	0.14613	1.369	0.170983
v2paelcont2	-0.23430	0.19653	-1.192	0.233195
---				

```
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for binomial family taken to be 1)

```
Null deviance: 4175.0 on 4179 degrees of freedom
Residual deviance: 3635.6 on 4169 degrees of freedom
AIC: 3657.6
```

Number of Fisher Scoring iterations: 5

Model selection and assumption check for interaction model

```
attach(dat_center)
```

The following objects are masked from dat\_new (pos = 3):

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,  
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,  
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,  
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,  
v2xpa_antiplural
```

The following objects are masked from dat (pos = 4):

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,  
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,  
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,  
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,  
v2xpa_antiplural
```

The following objects are masked from dat\_center (pos = 5):

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,  
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,  
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,  
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,  
v2xpa_antiplural
```

The following objects are masked from dat\_new (pos = 6):

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendla,  
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,  
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,  
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,  
v2xpa_antiplural
```

The following objects are masked from dat (pos = 7):

```
COW, COWcode.x, elect_system, exe_elect, gov, leg_elect, piendl,
psla, reg_vs, v2elcomvot, v2elparallel, v2eltrnout, v2eltype_0,
v2eltype_6, v2elvaptrn, v2paelcont, v2pagovsup, v2papariah,
v2pariglef, v2paseatshare, v2pavote, v2x_polyarchy,
v2xpa_antiplural
```

```
base_model <- lm(reg_vs ~ psla*v2x_polyarchy)
full_model <- lm(reg_vs ~ elect_system + v2pariglef + v2papariah + exe_elect + v2paelcont

forward <- step(base_model,
                  direction = "forward",
                  k = 2,
                  scope = list(lower = base_model, upper = full_model))
```

Start: AIC=3785.96

reg\_vs ~ psla \* v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ elect_system	3	231.152	10089	3697.3
+ v2pariglef	1	162.537	10158	3721.6
+ v2paelcont	2	20.017	10300	3781.8
+ exe_elect	1	10.017	10310	3783.9
<none>		10320	3786.0	
+ v2xpa_antiplural	1	3.646	10317	3786.5
+ v2papariah	1	0.577	10320	3787.7

Step: AIC=3697.28

reg\_vs ~ psla + v2x\_polyarchy + elect\_system + psla:v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ v2pariglef	1	151.032	9938.3	3636.2
+ v2paelcont	2	13.069	10076.3	3695.9
+ v2xpa_antiplural	1	7.819	10081.5	3696.0
<none>		10089.3	3697.3	
+ v2papariah	1	1.869	10087.5	3698.5
+ exe_elect	1	0.571	10088.8	3699.0

Step: AIC=3636.23

reg\_vs ~ psla + v2x\_polyarchy + elect\_system + v2pariglef + psla:v2x\_polyarchy

```

              Df Sum of Sq    RSS    AIC
+ v2xpa_antiplural  1   19.3215 9919.0 3630.1
+ v2paelcont        2   14.8828 9923.4 3634.0
+ v2papariah        1    7.5694 9930.7 3635.0
<none>                  9938.3 3636.2
+ exe_elect         1   1.6232 9936.7 3637.5

Step:  AIC=3630.1
reg_vs ~ psla + v2x_polyarchy + elect_system + v2pariglef + v2xpa_antiplural +
       psla:v2x_polyarchy

              Df Sum of Sq    RSS    AIC
+ v2papariah  1   14.7577 9904.2 3625.9
+ v2paelcont  2   16.1174 9902.9 3627.3
<none>                  9919.0 3630.1
+ exe_elect   1   0.8634 9918.1 3631.7

Step:  AIC=3625.87
reg_vs ~ psla + v2x_polyarchy + elect_system + v2pariglef + v2xpa_antiplural +
       v2papariah + psla:v2x_polyarchy

              Df Sum of Sq    RSS    AIC
+ v2paelcont  2   16.4699 9887.7 3622.9
<none>                  9904.2 3625.9
+ exe_elect   1   1.0269 9903.2 3627.4

Step:  AIC=3622.92
reg_vs ~ psla + v2x_polyarchy + elect_system + v2pariglef + v2xpa_antiplural +
       v2papariah + v2paelcont + psla:v2x_polyarchy

              Df Sum of Sq    RSS    AIC
<none>                  9887.7 3622.9
+ exe_elect  1   0.74024 9887.0 3624.6

back <- step(full_model,
               direction = "backward",
               k = 2,
               scope = list(lower = base_model, upper = full_model))

```

Start: AIC=3624.6

```
reg_vs ~ elect_system + v2pariglef + v2papariah + exe_elect +
v2paelcont + v2xpa_antiplural + psla * v2x_polyarchy
```

	Df	Sum of Sq	RSS	AIC
- exe_elect	1	0.740	9887.7	3622.9
<none>			9887.0	3624.6
- v2paelcont	2	16.183	9903.2	3627.4
- v2papariah	1	15.246	9902.2	3629.0
- v2xpa_antiplural	1	27.219	9914.2	3634.1
- v2pariglef	1	176.493	10063.5	3696.6
- elect_system	3	215.908	10102.9	3708.9

Step: AIC=3622.92

```
reg_vs ~ elect_system + v2pariglef + v2papariah + v2paelcont +
v2xpa_antiplural + psla + v2x_polyarchy + psla:v2x_polyarchy
```

	Df	Sum of Sq	RSS	AIC
<none>			9887.7	3622.9
- v2paelcont	2	16.470	9904.2	3625.9
- v2papariah	1	15.110	9902.9	3627.3
- v2xpa_antiplural	1	28.036	9915.8	3632.8
- v2pariglef	1	175.924	10063.7	3694.6
- elect_system	3	223.792	10111.5	3710.5

```
both <- step(base_model,
               direction = "both",
               k = 2,
               scope = list(lower = base_model, upper = full_model))
```

Start: AIC=3785.96

```
reg_vs ~ psla * v2x_polyarchy
```

	Df	Sum of Sq	RSS	AIC
+ elect_system	3	231.152	10089	3697.3
+ v2pariglef	1	162.537	10158	3721.6
+ v2paelcont	2	20.017	10300	3781.8
+ exe_elect	1	10.017	10310	3783.9
<none>			10320	3786.0
+ v2xpa_antiplural	1	3.646	10317	3786.5
+ v2papariah	1	0.577	10320	3787.7

Step: AIC=3697.28

reg\_vs ~ psla + v2x\_polyarchy + elect\_system + psla:v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ v2pariglef	1	151.032	9938.3	3636.2
+ v2paelcont	2	13.069	10076.3	3695.9
+ v2xpa_antiplural	1	7.819	10081.5	3696.0
<none>			10089.3	3697.3
+ v2papariah	1	1.869	10087.5	3698.5
+ exe_elect	1	0.571	10088.8	3699.0
- elect_system	3	231.152	10320.5	3786.0

Step: AIC=3636.23

reg\_vs ~ psla + v2x\_polyarchy + elect\_system + v2pariglef + psla:v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ v2xpa_antiplural	1	19.321	9919.0	3630.1
+ v2paelcont	2	14.883	9923.4	3634.0
+ v2papariah	1	7.569	9930.7	3635.0
<none>			9938.3	3636.2
+ exe_elect	1	1.623	9936.7	3637.5
- v2pariglef	1	151.032	10089.3	3697.3
- elect_system	3	219.648	10157.9	3721.6

Step: AIC=3630.1

reg\_vs ~ psla + v2x\_polyarchy + elect\_system + v2pariglef + v2xpa\_antiplural + psla:v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ v2papariah	1	14.758	9904.2	3625.9
+ v2paelcont	2	16.117	9902.9	3627.3
<none>			9919.0	3630.1
+ exe_elect	1	0.863	9918.1	3631.7
- v2xpa_antiplural	1	19.321	9938.3	3636.2
- v2pariglef	1	162.535	10081.5	3696.0
- elect_system	3	225.853	10144.8	3718.2

Step: AIC=3625.87

reg\_vs ~ psla + v2x\_polyarchy + elect\_system + v2pariglef + v2xpa\_antiplural + v2papariah + psla:v2x\_polyarchy

	Df	Sum of Sq	RSS	AIC
+ v2paelcont	2	16.470	9887.7	3622.9

```

<none>                                9904.2 3625.9
+ exe_elect      1     1.027  9903.2 3627.4
- v2papariah    1    14.758  9919.0 3630.1
- v2xpa_antiplural 1   26.510  9930.7 3635.0
- v2pariglef    1   173.321 10077.5 3696.4
- elect_system   3   231.168 10135.4 3716.3

```

Step: AIC=3622.92

```

reg_vs ~ psla + v2x_polyarchy + elect_system + v2pariglef + v2xpa_antiplural +
v2papariah + v2paelcont + psla:v2x_polyarchy

```

	Df	Sum of Sq	RSS	AIC
<none>			9887.7	3622.9
+ exe_elect	1	0.740	9887.0	3624.6
- v2paelcont	2	16.470	9904.2	3625.9
- v2papariah	1	15.110	9902.9	3627.3
- v2xpa_antiplural	1	28.036	9915.8	3632.8
- v2pariglef	1	175.924	10063.7	3694.6
- elect_system	3	223.792	10111.5	3710.5

forward

Call:

```

lm(formula = reg_vs ~ psla + v2x_polyarchy + elect_system + v2pariglef +
v2xpa_antiplural + v2papariah + v2paelcont + psla:v2x_polyarchy)

```

Coefficients:

(Intercept)	psla	v2x_polyarchy	elect_system1
5.1401	-1.6846	0.3223	-0.3820
elect_system2	elect_system3	v2pariglef	v2xpa_antiplural
-0.5723	-1.1252	-0.4216	0.2261
v2papariah	v2paelcont1	v2paelcont2	psla:v2x_polyarchy
0.1290	0.1289	0.2245	0.3526

back

Call:

```

lm(formula = reg_vs ~ elect_system + v2pariglef + v2papariah +

```

```
v2paelcont + v2xpa_antiplural + psla + v2x_polyarchy + psla:v2x_polyarchy)
```

Coefficients:

	elect_system1	elect_system2	elect_system3
(Intercept)	5.1401	-0.3820	-0.5723
v2pariglef	v2papariah	v2paelcont1	v2paelcont2
-0.4216	0.1290	0.1289	0.2245
v2xpa_antiplural	psla	v2x_polyarchy	psla:v2x_polyarchy
0.2261	-1.6846	0.3223	0.3526

both

Call:

```
lm(formula = reg_vs ~ psla + v2x_polyarchy + elect_system + v2pariglef +
v2xpa_antiplural + v2papariah + v2paelcont + psla:v2x_polyarchy)
```

Coefficients:

	psla	v2x_polyarchy	elect_system1
(Intercept)	5.1401	-1.6846	-0.3820
elect_system2	elect_system3	v2pariglef	v2xpa_antiplural
-0.5723	-1.1252	-0.4216	0.2261
v2papariah	v2paelcont1	v2paelcont2	psla:v2x_polyarchy
0.1290	0.1289	0.2245	0.3526

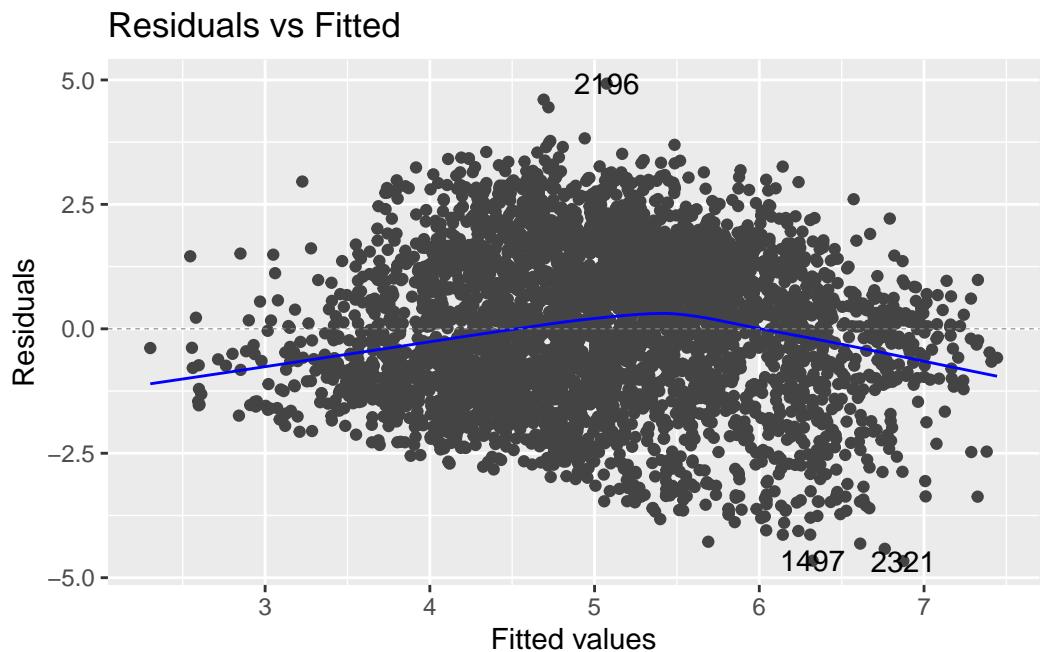
#### all the models give the same formula here, so we are going to go with:

```
f3 <- reg_vs ~ psla + v2x_polyarchy + elect_system + v2pariglef +
v2xpa_antiplural + v2papariah + v2paelcont + psla:v2x_polyarchy
```

#### check assumptions

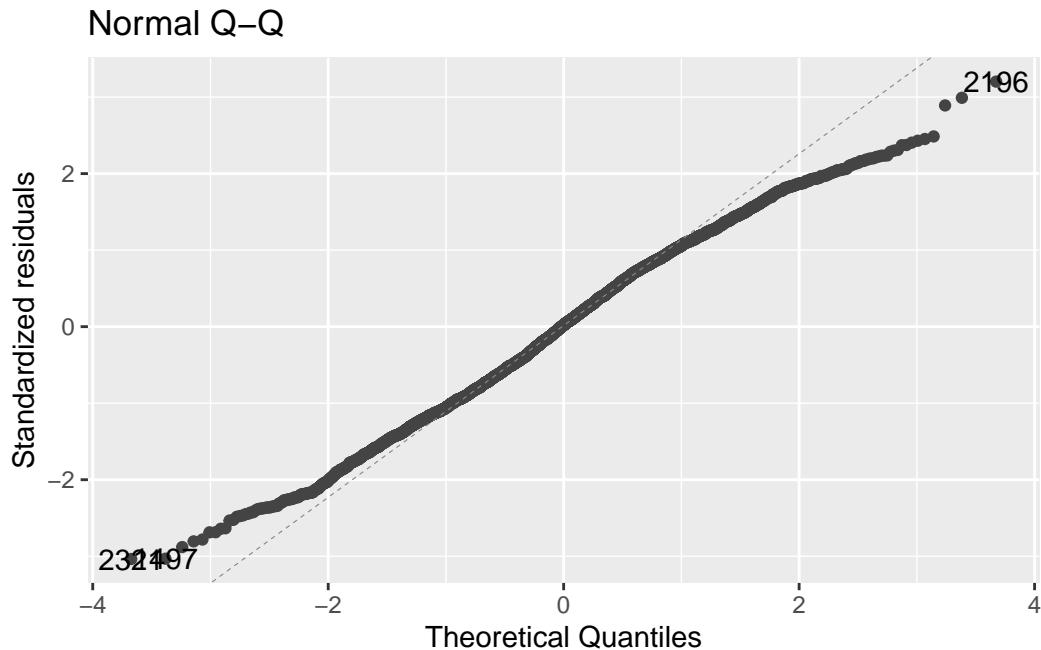
```
inter1 <- lm(f3)
```

```
autoplot(inter1, which = 1, nrow = 1, ncol = 1)
```

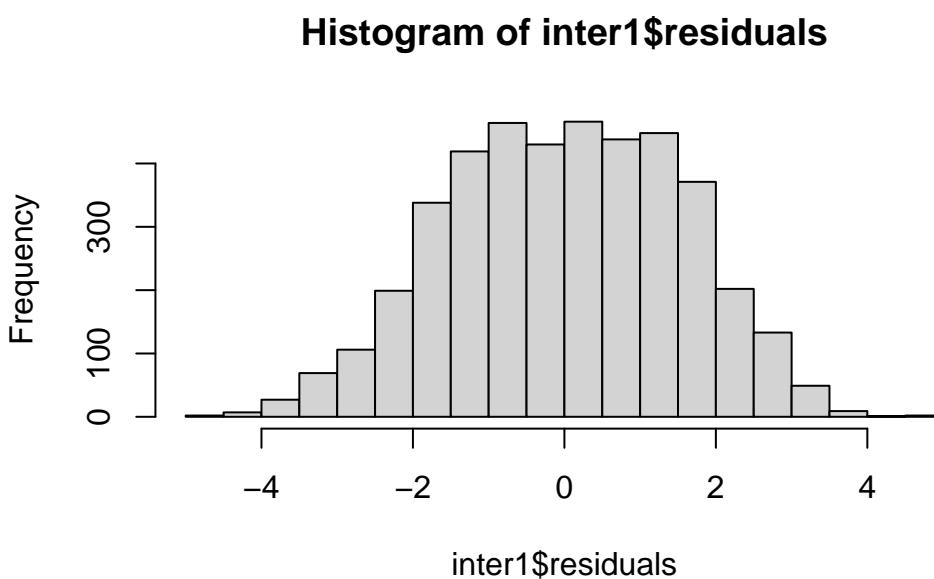


```
#### linearity looks good
```

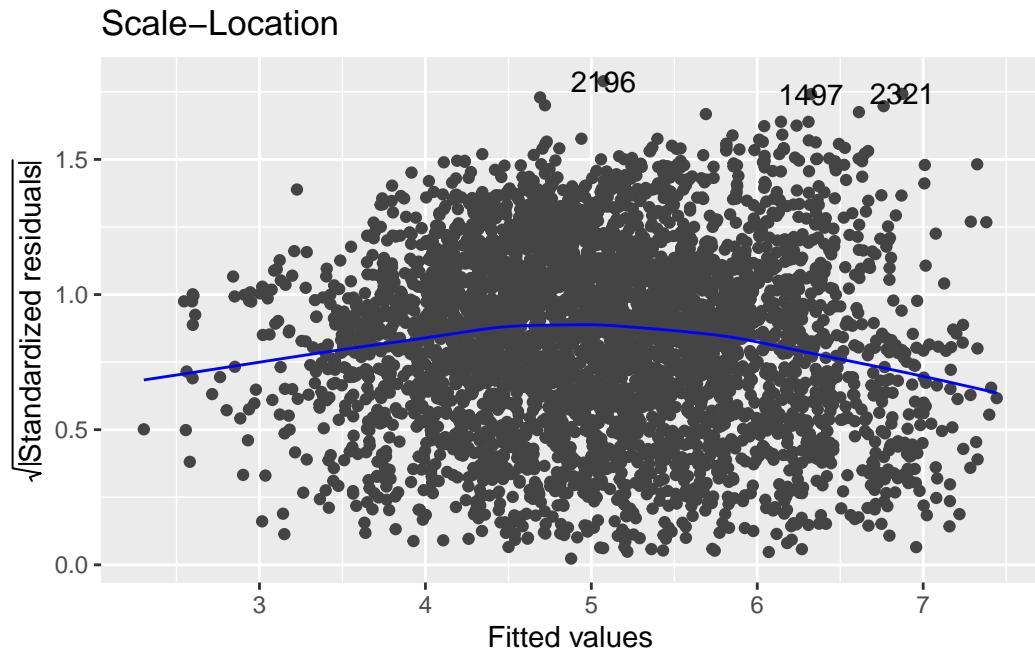
```
autoplot(inter1, which = 2, nrow = 1, ncol = 1)
```



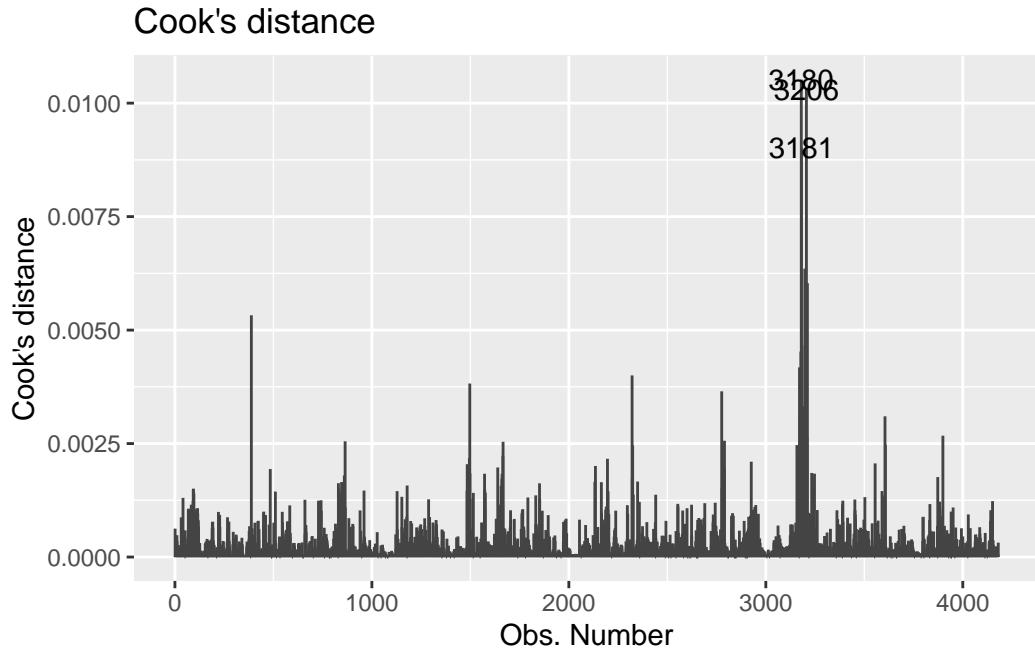
```
hist(inter1$residuals)
```



```
#### looks pretty normal  
autoplot(inter1, which = 3, nrow = 1, ncol = 1)
```



```
#### constant variance looks good  
autoplot(inter1, which = 4, nrow = 1, ncol = 1)
```



```
#### no serious outliers.
```

```
car::vif(inter1, type = "predictor")
```

GVIFs computed for predictors

	GVIF	Df	GVIF^(1/(2*Df))	Interacts With
psla	2.187939	3	1.139390	v2x_polyarchy
v2x_polyarchy	2.187939	3	1.139390	psla
elect_system	1.103624	3	1.016569	--
v2pariglef	1.055331	1	1.027293	--
v2xpa_antiplural	1.905785	1	1.380502	--
v2papariah	1.151168	1	1.072925	--
v2paelcont	1.177381	2	1.041668	--

#### Other Predictors

psla	elect_system, v2pariglef, v2xpa_antiplural, v2papariah, v2paelcont
v2x_polyarchy	elect_system, v2pariglef, v2xpa_antiplural, v2papariah, v2paelcont
elect_system	psla, v2x_polyarchy, v2pariglef, v2xpa_antiplural, v2papariah, v2paelcont
v2pariglef	psla, v2x_polyarchy, elect_system, v2xpa_antiplural, v2papariah, v2paelcont
v2xpa_antiplural	psla, v2x_polyarchy, elect_system, v2pariglef, v2papariah, v2paelcont
v2papariah	psla, v2x_polyarchy, elect_system, v2pariglef, v2xpa_antiplural, v2paelcont
v2paelcont	psla, v2x_polyarchy, elect_system, v2pariglef, v2xpa_antiplural, v2papariah

```
#### no multicollinearity
```

```
#### meets assumptions
```

Look at results

```
summary(inter1)
```

```
Call:  
lm(formula = f3)
```

Residuals:

Min	1Q	Median	3Q	Max
-4.6653	-1.1422	0.0331	1.1859	4.9262

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	5.14008	0.03123	164.568	< 2e-16 ***
psla	-1.68465	0.05244	-32.126	< 2e-16 ***
v2x_polyarchy	0.32225	0.06497	4.960	7.33e-07 ***
elect_system1	-0.38201	0.07957	-4.801	1.63e-06 ***
elect_system2	-0.57230	0.07041	-8.128	5.71e-16 ***
elect_system3	-1.12520	0.27116	-4.150	3.40e-05 ***
v2pariglef	-0.42155	0.04895	-8.611	< 2e-16 ***
v2xpa_antiplural	0.22615	0.06578	3.438	0.000592 ***
v2papariah	0.12903	0.05113	2.524	0.011647 *
v2paelcont1	0.12894	0.07378	1.748	0.080605 .
v2paelcont2	0.22454	0.10405	2.158	0.030983 *
psla:v2x_polyarchy	0.35257	0.09622	3.664	0.000251 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.54 on 4168 degrees of freedom

Multiple R-squared: 0.2426, Adjusted R-squared: 0.2406

F-statistic: 121.4 on 11 and 4168 DF, p-value: < 2.2e-16

```
#### F stat is not too much larger, and the adjusted r-squared is not that much better, so
```

```
anova(m3, inter1)
```

### Analysis of Variance Table

```
Model 1: reg_vs ~ psla + elect_system + v2pariglef + v2x_polyarchy + v2xpa_antiplural +
          v2papariah + v2paelcont
Model 2: reg_vs ~ psla + v2x_polyarchy + elect_system + v2pariglef + v2xpa_antiplural +
          v2papariah + v2paelcont + psla:v2x_polyarchy
Res.Df      RSS Df Sum of Sq    F   Pr(>F)
1     4169  9919.6
2     4168  9887.7  1     31.855 13.428 0.000251 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

#### yeah, looks like it is.