output: pdf_document: default html_document: default

title: “ProblemSet1Skills” author: “Vera Jónsdóttir” “Consulted with Ryan McGinnis” date: “4/8/2021” output: pdf_document: default word_document: default html_document: default

This submission is my work alone and complies with the 30535 integrity policy.

Add your initials to indicate your agreement: VJ

Add your collaborators: Ryan McGinnis

Late coins used this pset: 0. Late coins left: X.

```
library(rmarkdown)
```

```
## Warning: package 'rmarkdown' was built under R version 4.0.5
```

```
library (tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --
```

```
## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.1.0     v dplyr   1.0.5
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
## Warning: package 'tibble' was built under R version 4.0.4
```

```
## Warning: package 'tidyr' was built under R version 4.0.4
```

```
## Warning: package 'readr' was built under R version 4.0.4
```

```
## Warning: package 'purrr' was built under R version 4.0.4
```

```
## Warning: package 'dplyr' was built under R version 4.0.4
```

```
## Warning: package 'stringr' was built under R version 4.0.4
```

```
## Warning: package 'forcats' was built under R version 4.0.4
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library (dslabs)
```

```
## Warning: package 'dslabs' was built under R version 4.0.5
```

```r
library(ggplot2)

tinytex::install_tinytex()

list.of.packages <- c("tidyverse", "dslabs", "rmarkdown", "ggplot2")
```

```r
#Part 2.1

view(polls_us_election_2016)
#2.1.1
nrow(polls_us_election_2016)
```

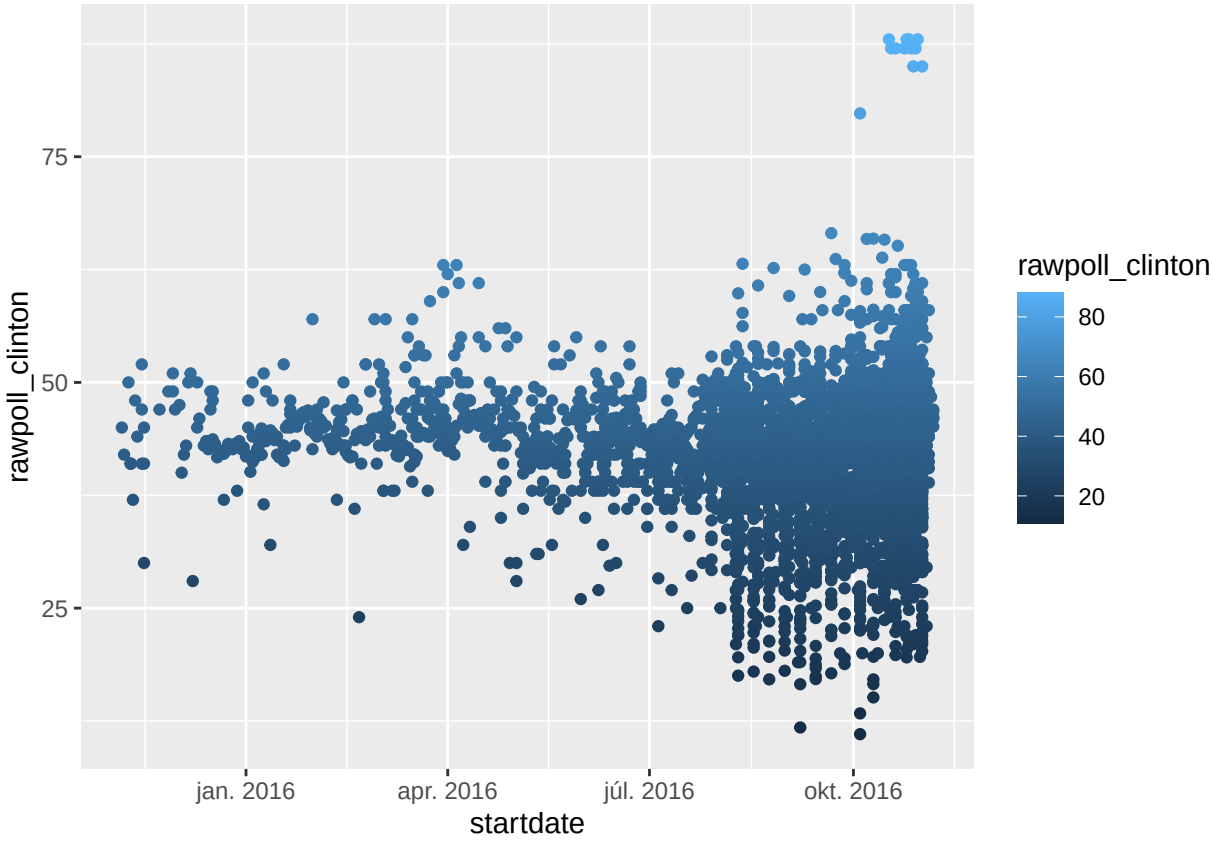
```
## [1] 4208
```

```r
ncol(polls_us_election_2016)
```

```
## [1] 15
```

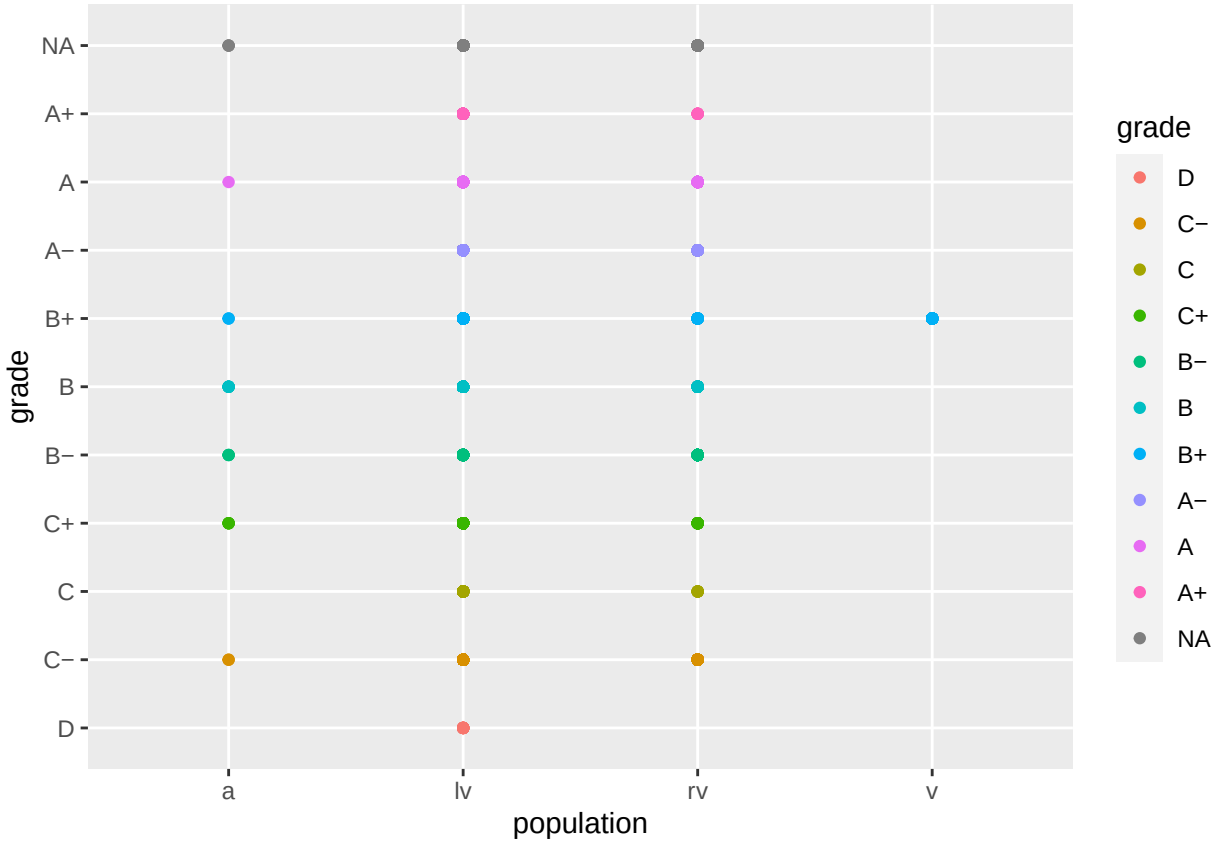
The columns represent the different pollsters that are conducting the survey the US 2016 Elections and the rows are the different indicators that are being tested or looked at for each pollster.

```r
#2.1.2
# Graph 1
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_clinton,
                           color = rawpoll_clinton))
```



3 The variable is the grade assigned by fivethirtyeight to pollster. The variable grade describes the grade assigned by fivethirtyeight to pollster. It's the rating given to pollster based on their accuracy and quality.

```
#4
ggplot(data = polls_us_election_2016)+
  geom_point(mapping = aes(x = population,
                           y = grade,
                           color = grade))
```



It conveys no sort of information on how many values are at each point. We only see that there are values at each point marked on the plot but we do not know the concentration of values at each point which makes this plot quite unuseful.

```
#Part 2.2
#2.2.1
?polls_us_election_2016
```

```
## starting httpd help server ... done
```

```
head(polls_us_election_2016)
```

```
##   state  startdate    enddate
## 1  U.S. 2016-11-03 2016-11-06
## 2  U.S. 2016-11-01 2016-11-07
## 3  U.S. 2016-11-02 2016-11-06
## 4  U.S. 2016-11-04 2016-11-07
## 5  U.S. 2016-11-03 2016-11-06
## 6  U.S. 2016-11-03 2016-11-06
##                                                 pollster grade samplesize
## 1                                ABC News/Washington Post    A+       2220
## 2                                 Google Consumer Surveys     B      26574
## 3                                                   Ipsos    A-       2195
## 4                                                  YouGov     B       3677
## 5                                          Gravis Marketing    B-      16639
## 6 Fox News/Anderson Robbins Research/Shaw & Company Research     A       1295
```

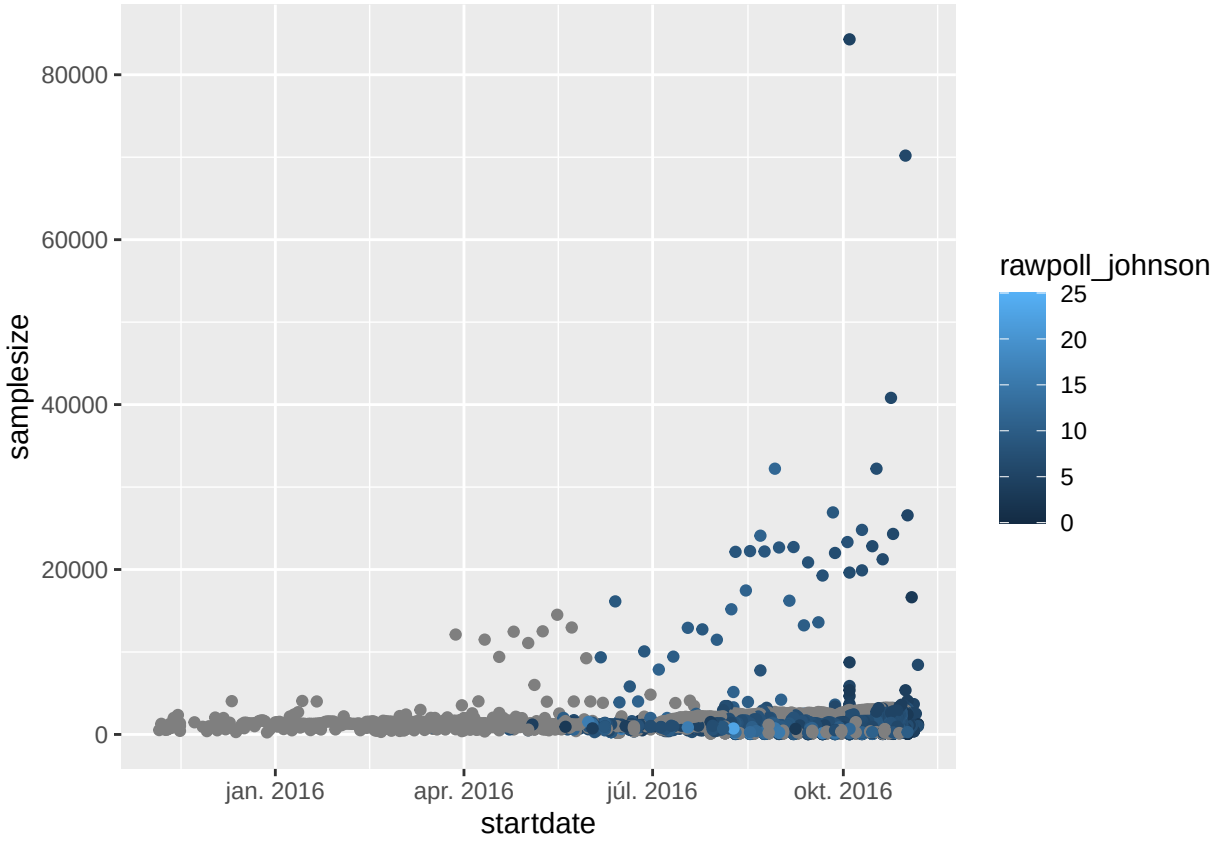
```
##   population rawpoll_clinton rawpoll_trump rawpoll_johnson rawpoll_mcmullin
## 1         lv           47.00         43.00            4.00               NA
## 2         lv           38.03         35.69            5.46               NA
## 3         lv           42.00         39.00            6.00               NA
## 4         lv           45.00         41.00            5.00               NA
## 5         rv           47.00         43.00            3.00               NA
## 6         lv           48.00         44.00            3.00               NA
##   adjpoll_clinton adjpoll_trump adjpoll_johnson adjpoll_mcmullin
## 1        45.20163      41.72430        4.626221               NA
## 2        43.34557      41.21439        5.175792               NA
## 3        42.02638      38.81620        6.844734               NA
## 4        45.65676      40.92004        6.069454               NA
## 5        46.84089      42.33184        3.726098               NA
## 6        49.02208      43.95631        3.057876               NA
```

```
colnames(polls_us_election_2016)
```

```
##  [1] "state"            "startdate"        "enddate"          "pollster"
##  [5] "grade"            "samplesize"       "population"       "rawpoll_clinton"
##  [9] "rawpoll_trump"    "rawpoll_johnson"  "rawpoll_mcmullin" "adjpoll_clinton"
## [13] "adjpoll_trump"    "adjpoll_johnson"  "adjpoll_mcmullin"
```

```
#2.2.2
#Graph 1
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,y = samplesize,color = rawpoll_johnson))
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```



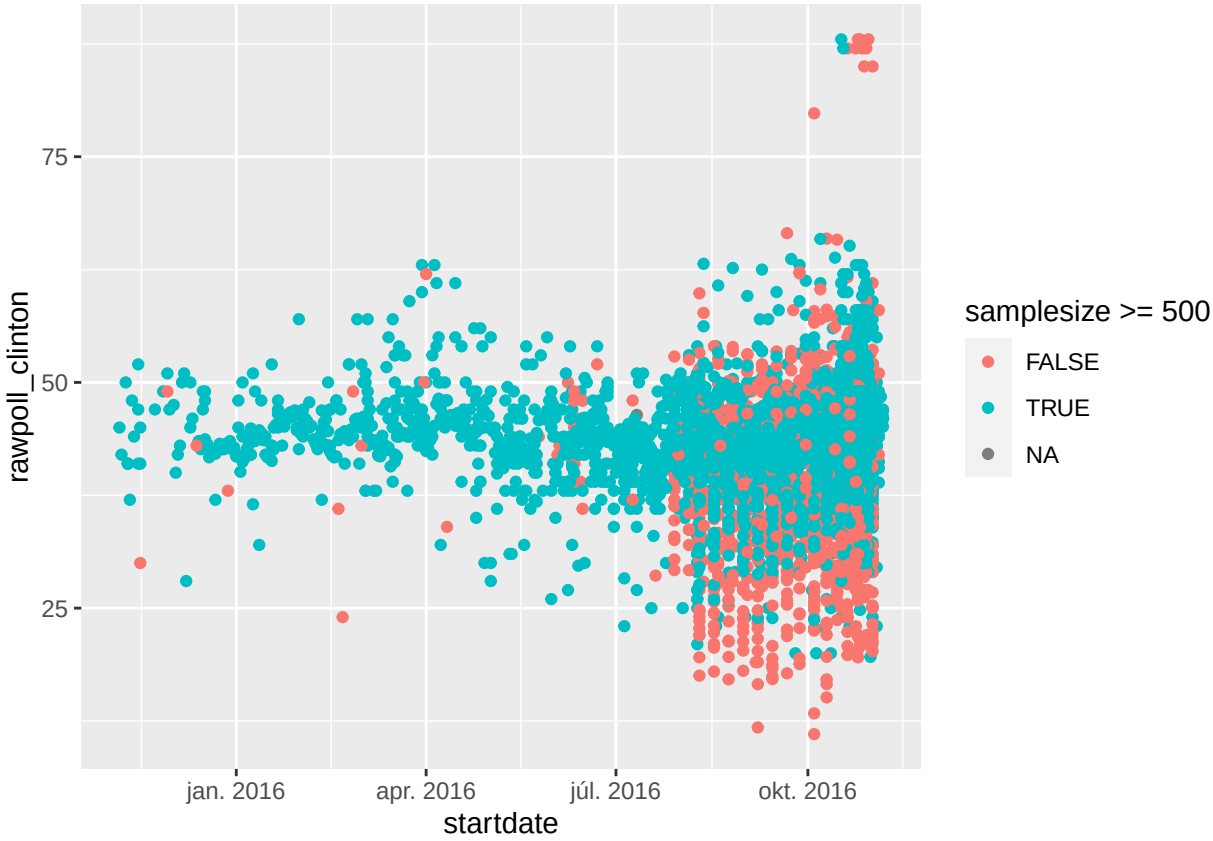

```
#Graph 2
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,y = samplesize,color = as.character(rawpoll_johnson)))
```

```
## Warning: Removed 1 rows containing missing values (geom_point).
```

	●	3.36	●	3.91	●	4.38	●	4.78	●	5.11	●	5.45	●	5.73	●	6.05	●	6.35	●	6.6
5	●	3.37	●	3.96	●	4.4	●	4.8	●	5.12	●	5.46	●	5.74	●	6.06	●	6.36	●	6.61
1	●	3.4	●	3.97	●	4.46	●	4.84	●	5.14	●	5.47	●	5.77	●	6.07	●	6.37	●	6.62
	●	3.5	●	3.98	●	4.48	●	4.85	●	5.15	●	5.48	●	5.78	●	6.08	●	6.38	●	6.65
5	●	3.51	●	3.99	●	4.5	●	4.86	●	5.16	●	5.49	●	5.8	●	6.1	●	6.39	●	6.66
	●	3.53	●	4	●	4.54	●	4.87	●	5.17	●	5.5	●	5.82	●	6.13	●	6.4	●	6.67
7	●	3.58	●	4.03	●	4.55	●	4.88	●	5.19	●	5.52	●	5.83	●	6.15	●	6.42	●	6.68
	●	3.59	●	4.04	●	4.56	●	4.89	●	5.2	●	5.53	●	5.85	●	6.17	●	6.43	●	6.69
	●	3.6	●	4.05	●	4.57	●	4.9	●	5.21	●	5.54	●	5.86	●	6.18	●	6.46	●	6.7
	●	3.64	●	4.07	●	4.59	●	4.92	●	5.24	●	5.57	●	5.87	●	6.19	●	6.47	●	6.71
	●	3.69	●	4.1	●	4.6	●	4.93	●	5.25	●	5.6	●	5.88	●	6.2	●	6.48	●	6.75
	●	3.7	●	4.12	●	4.64	●	4.94	●	5.26	●	5.61	●	5.89	●	6.21	●	6.49	●	6.76
	●	3.71	●	4.14	●	4.65	●	4.95	●	5.27	●	5.62	●	5.9	●	6.22	●	6.5	●	6.77
1	●	3.77	●	4.15	●	4.68	●	4.98	●	5.29	●	5.64	●	5.91	●	6.23	●	6.51	●	6.79
1	●	3.79	●	4.17	●	4.69	●	5	●	5.3	●	5.65	●	5.93	●	6.24	●	6.52	●	6.8
	●	3.8	●	4.2	●	4.7	●	5.03	●	5.31	●	5.66	●	5.97	●	6.3	●	6.54	●	6.82
1	●	3.81	●	4.23	●	4.72	●	5.04	●	5.33	●	5.67	●	5.99	●	6.31	●	6.55	●	6.85
	●	3.86	●	4.27	●	4.73	●	5.07	●	5.36	●	5.69	●	6	●	6.32	●	6.56	●	6.86
1	●	3.89	●	4.3	●	4.76	●	5.09	●	5.4	●	5.7	●	6.01	●	6.33	●	6.57	●	6.87

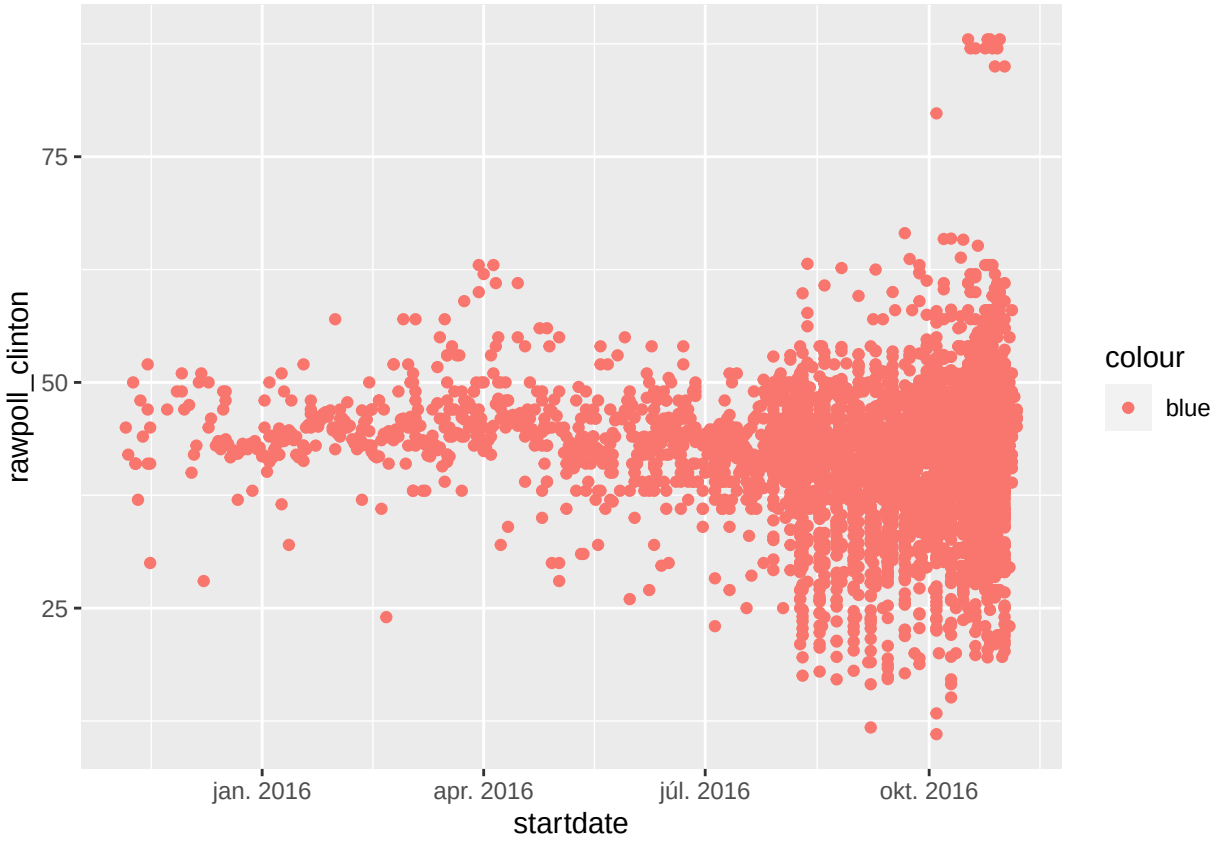
The second graph is making rawpoll_johnson into a character so each individual point on the graph is turned into a color and the color of the number depends on how large it is The smaller the number the greener it becomes and the larger the number the bluer it becomes. The first plot is much more useful as it conveys information more clearly by setting up the scatterplot in a way which is much more clear visually.

```
#2.2.3
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_clinton,
                           color = samplesize >= 500))
```

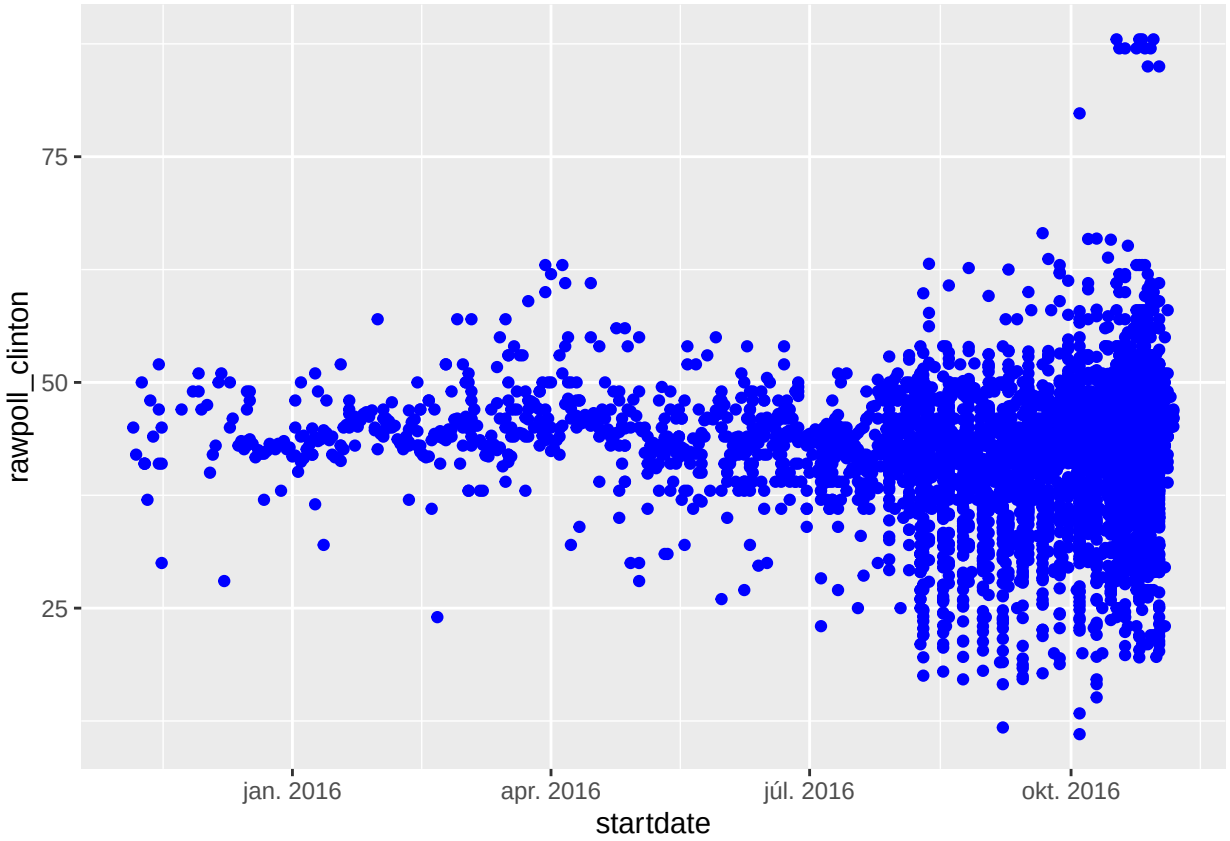


It changes the color of those who are above 500 and those who are below 500

```
#4
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_clinton,
                           color = "blue"))
```

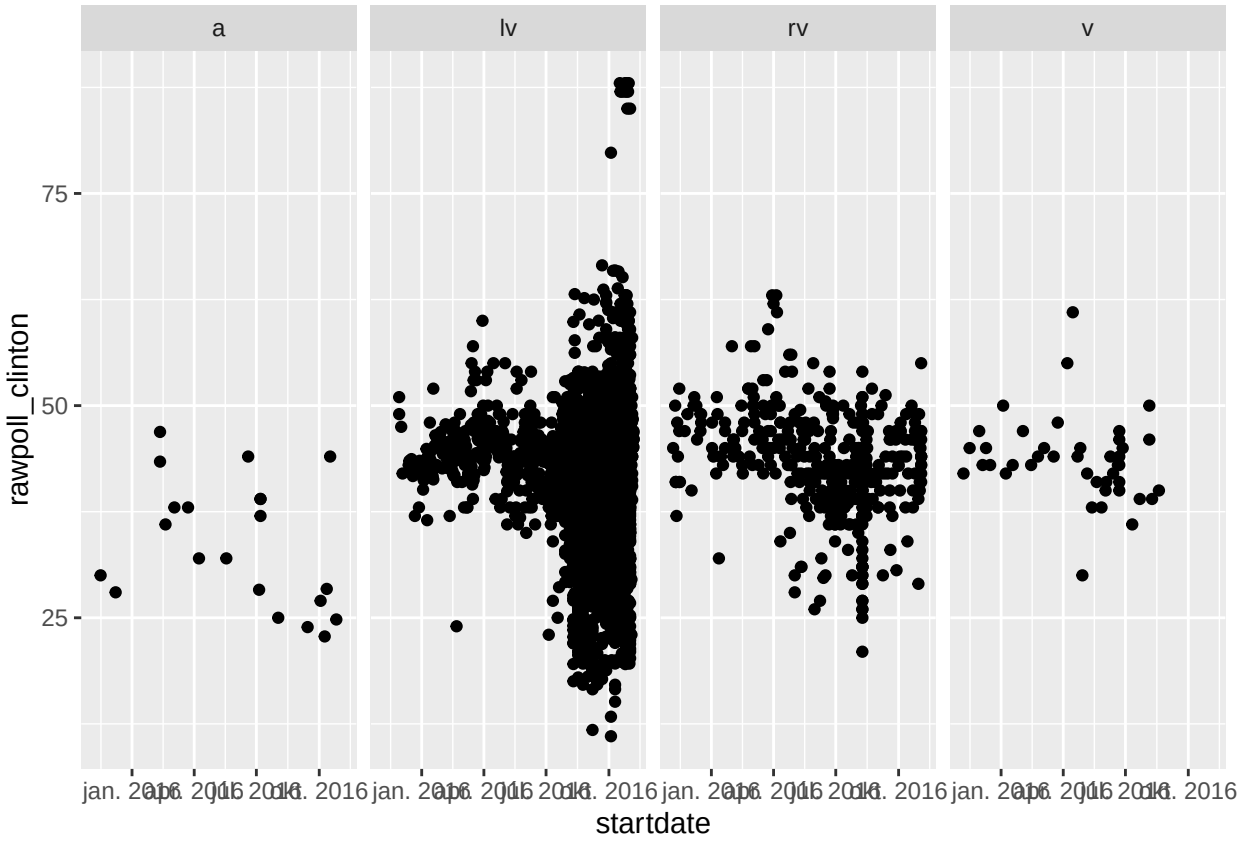



```
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_clinton),
             color = "blue")
```

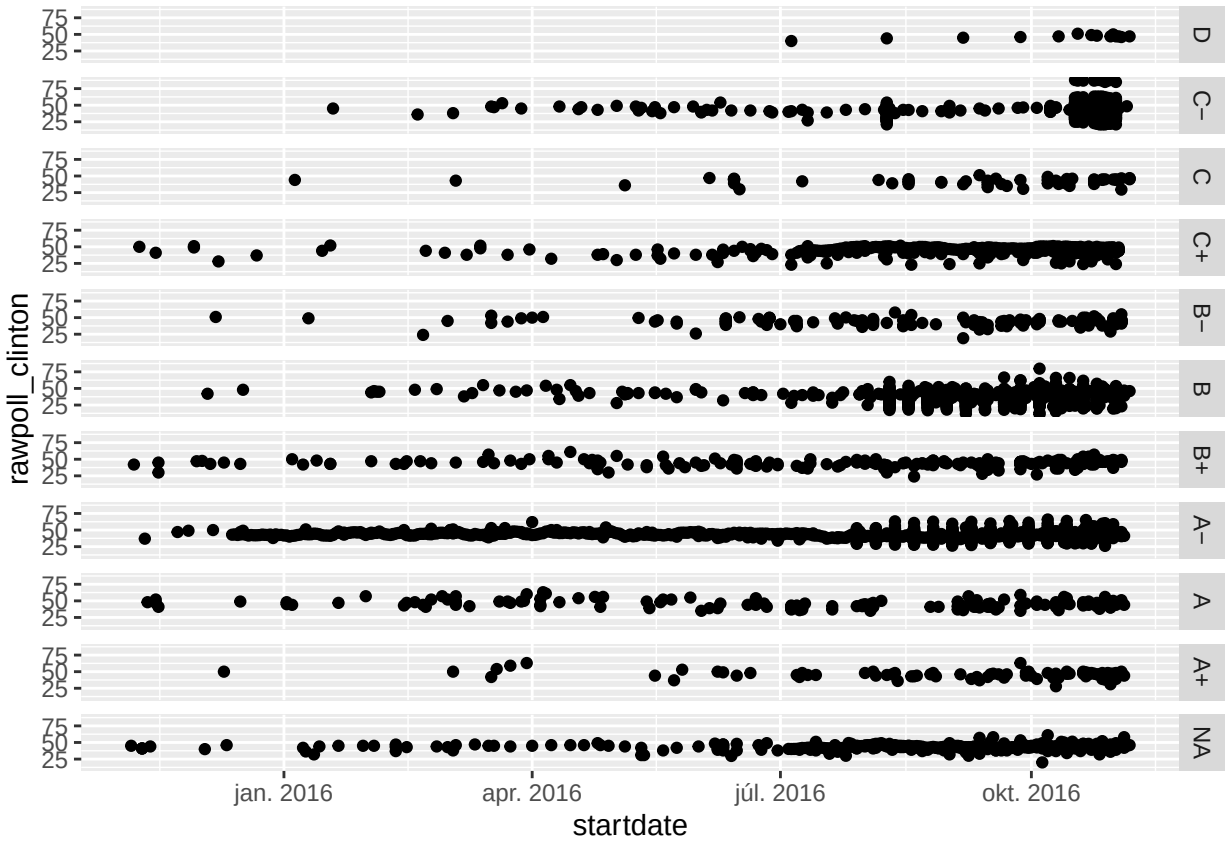


there needs to be a bracket in between color and aes in order for the code to become blue.

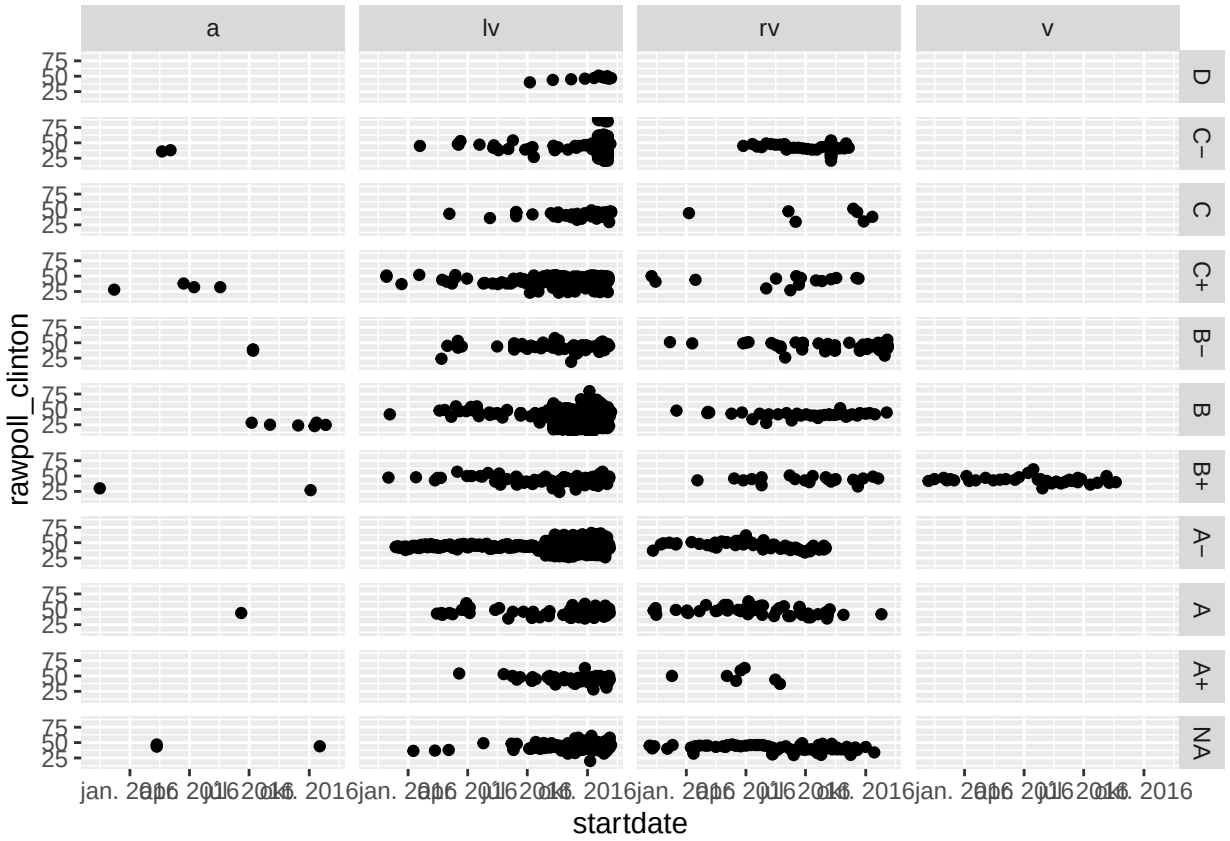
```
#Part 2.3
#2.3.1
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,y = rawpoll_clinton)) +
  facet_grid(cols = vars(population))
```




```
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,y = rawpoll_clinton)) +
  facet_grid(rows = vars(grade))
```



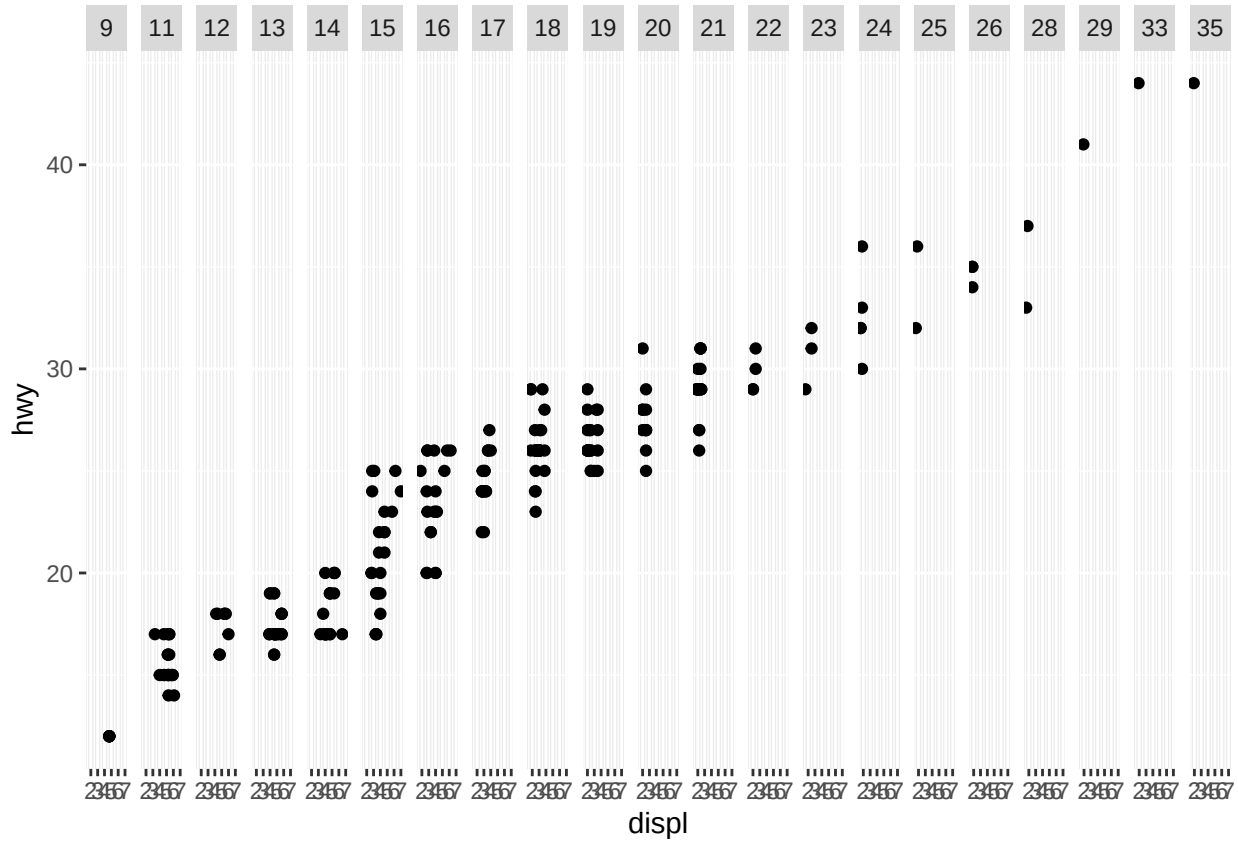

```
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_clinton)) +
  facet_grid(rows = vars(grade), cols = vars(population))
```



The facet clearly divides different classifications down into a matrix. The rows are classified by changes in the grade and the columns are classified by the changes in population. Here a = all adults, lv = likely voters, rv = registered voters, and v = voters

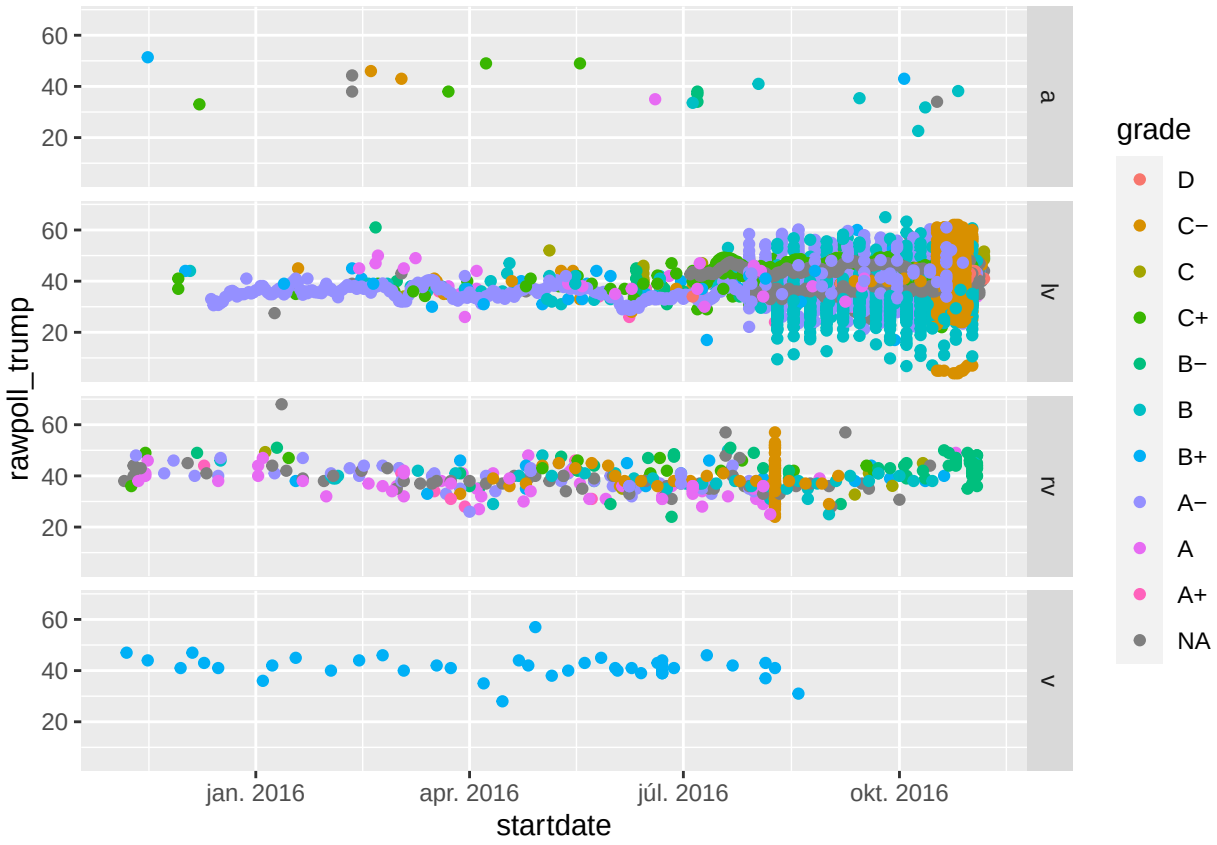
```
#2.3.2

ggplot(mpg, aes(x = displ, y = hwy)) +
  geom_point() +
  facet_grid(. ~ cty)
```

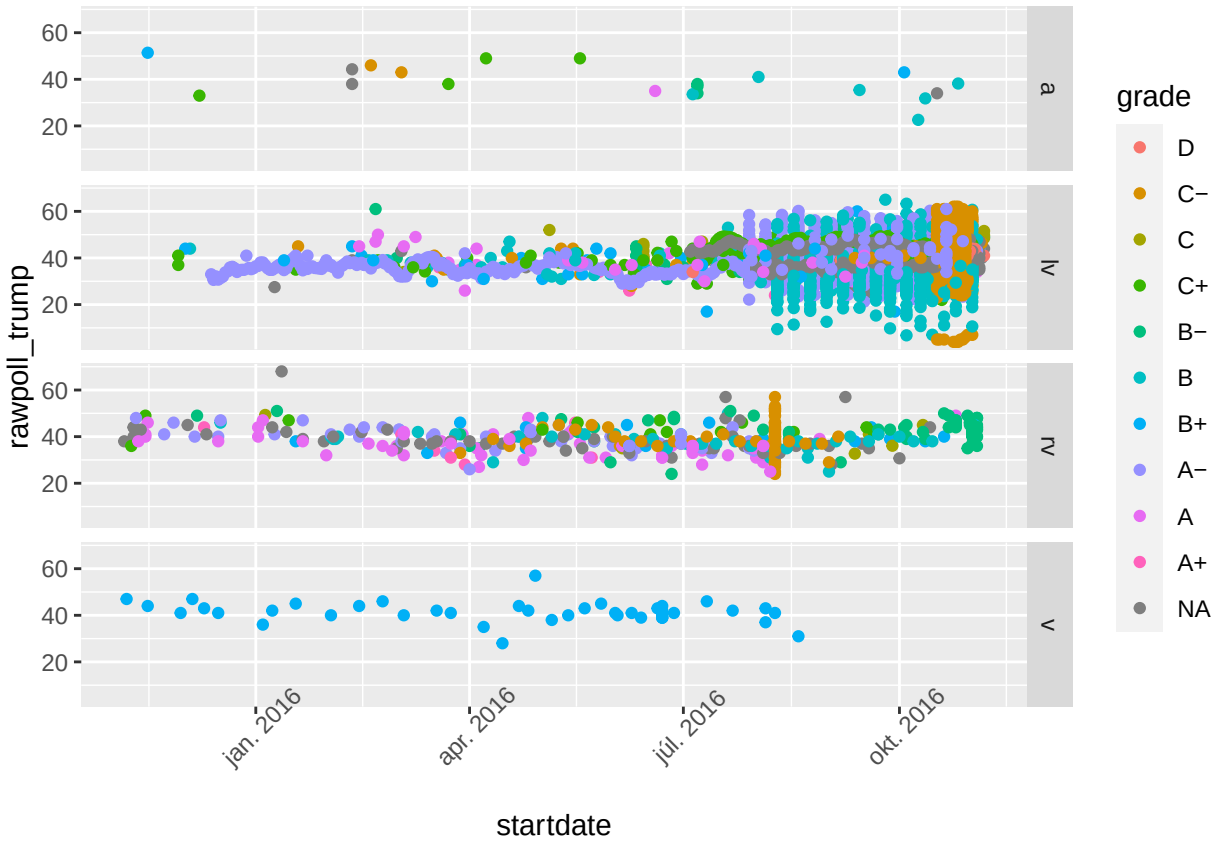


The continuous variable will be converted to a categorical variable, and the plot contains a facet for each distinct value. source: https://jrnold.github.io/r4ds-exercise-solutions/data-visualisation.html#facets

```
#2.3.3
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_trump,
                           color = grade))+
  facet_grid(rows=vars(population))
```




```
#2.3.4
ggplot(data = polls_us_election_2016) +
                theme(axis.text.x = element_text(angle = 45))+
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_trump,
                           color = grade))+
  facet_grid(rows=vars(population))
```

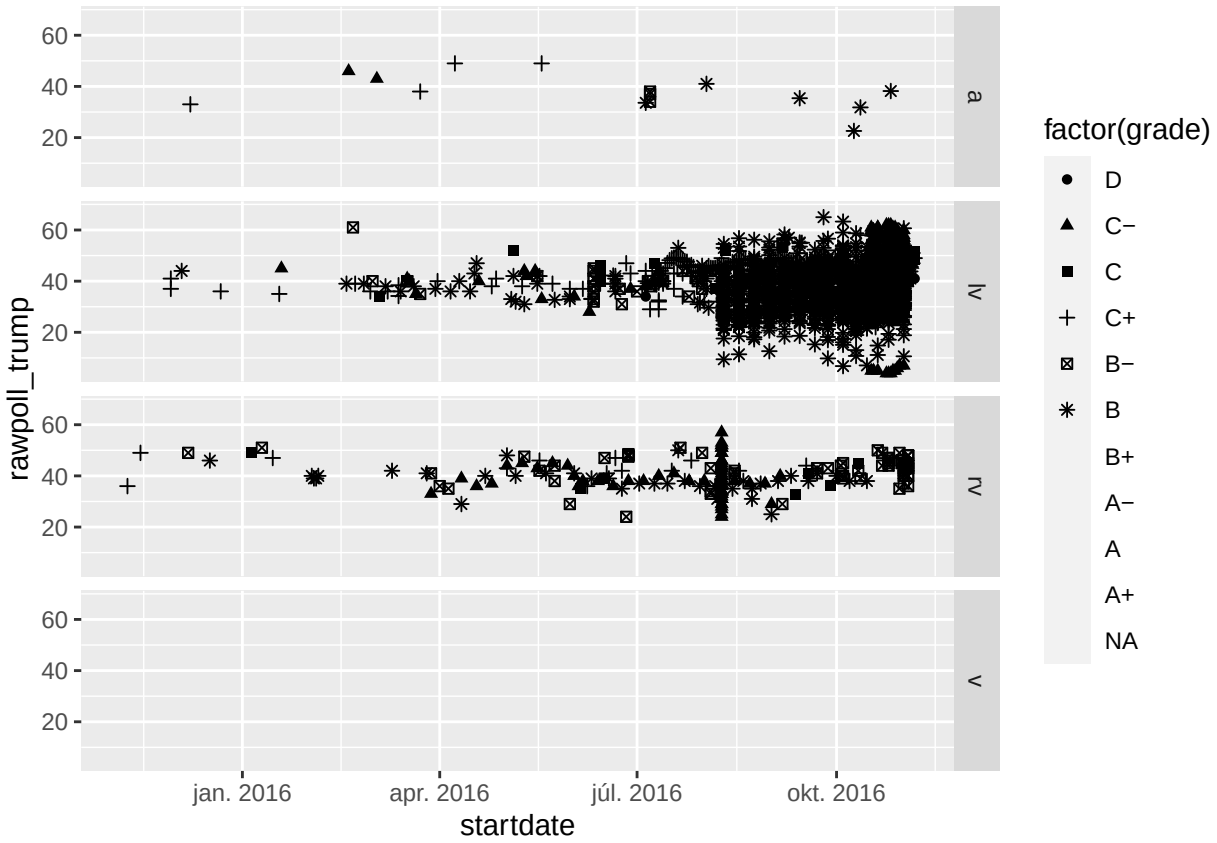



```
#2.3.5

ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_trump,
                           shape = factor(grade)))+
  facet_grid(rows=vars(population))
```

```
## Warning: The shape palette can deal with a maximum of 6 discrete values because
## more than 6 becomes difficult to discriminate; you have 10. Consider
## specifying shapes manually if you must have them.
```

```
## Warning: Removed 1961 rows containing missing values (geom_point).
```

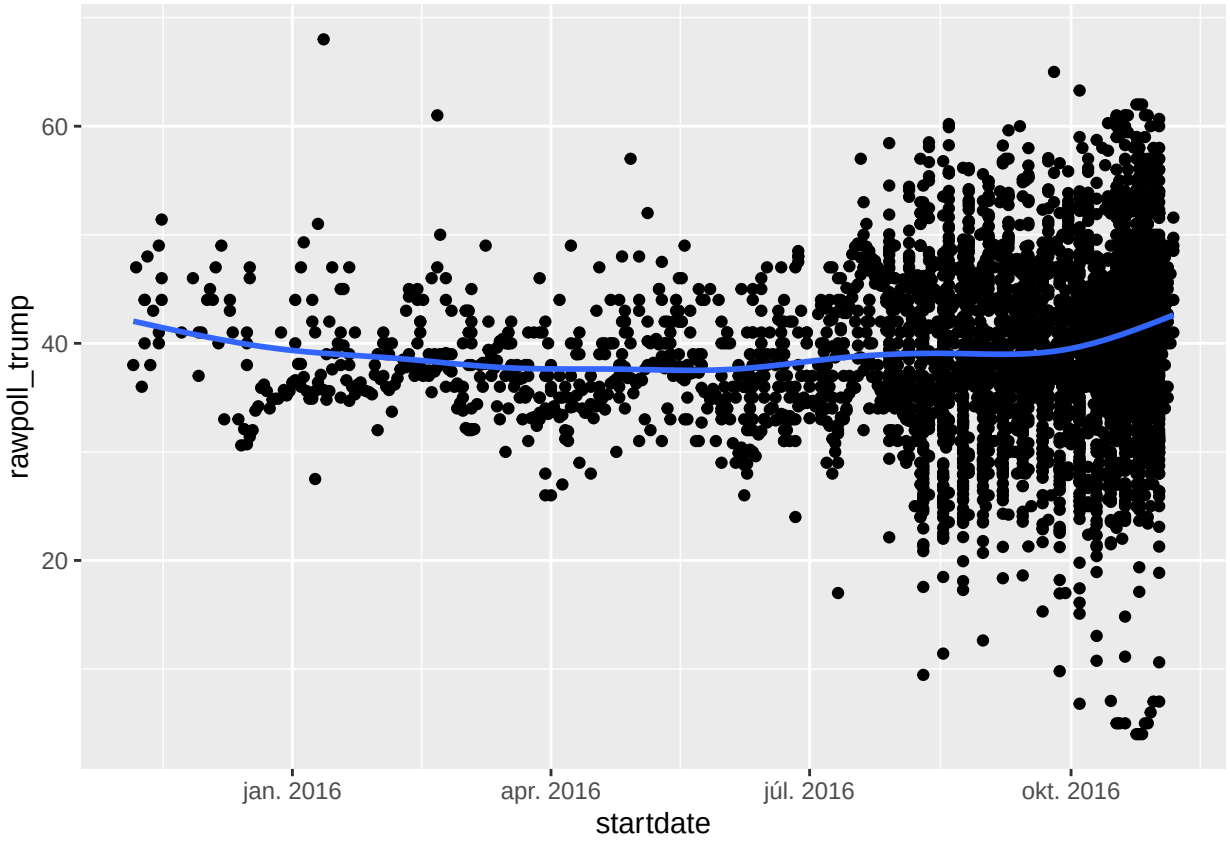


The shape palette can deal with a maximum of 6 discrete values because more than 6 becomes difficult to discriminate; here we have 10. It is possible to specify shapes manually if they are very necessary.

Part 2.4 2.4.1 We use geom_line to draw a line chart. We use geom_poxplot in order to create a boxplot. We use geom_histogram to create a histogram with geom. We use geom_are in order to create an area chart using geom. Source:https://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.pdf

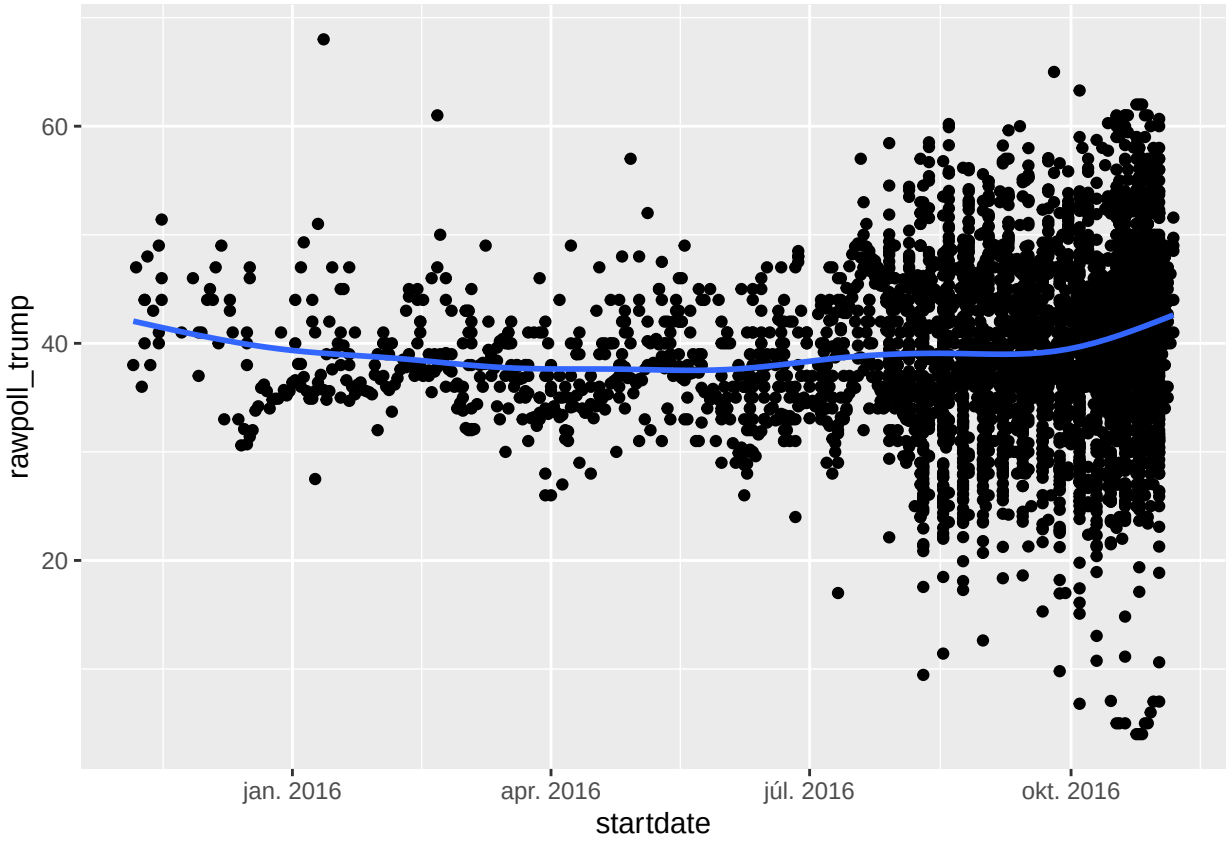
```
#2.4.2
ggplot(data = polls_us_election_2016,mapping = aes(x = startdate,y = rawpoll_trump))+
  geom_point() +
  geom_smooth(se=FALSE)
```

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




```
#2.4.2
ggplot() +
  geom_point(data = polls_us_election_2016,mapping = aes(x = startdate,y = rawpoll_trump)) +
  geom_smooth(data = polls_us_election_2016,mapping = aes(x = startdate,y = rawpoll_trump),
              se=FALSE)
```

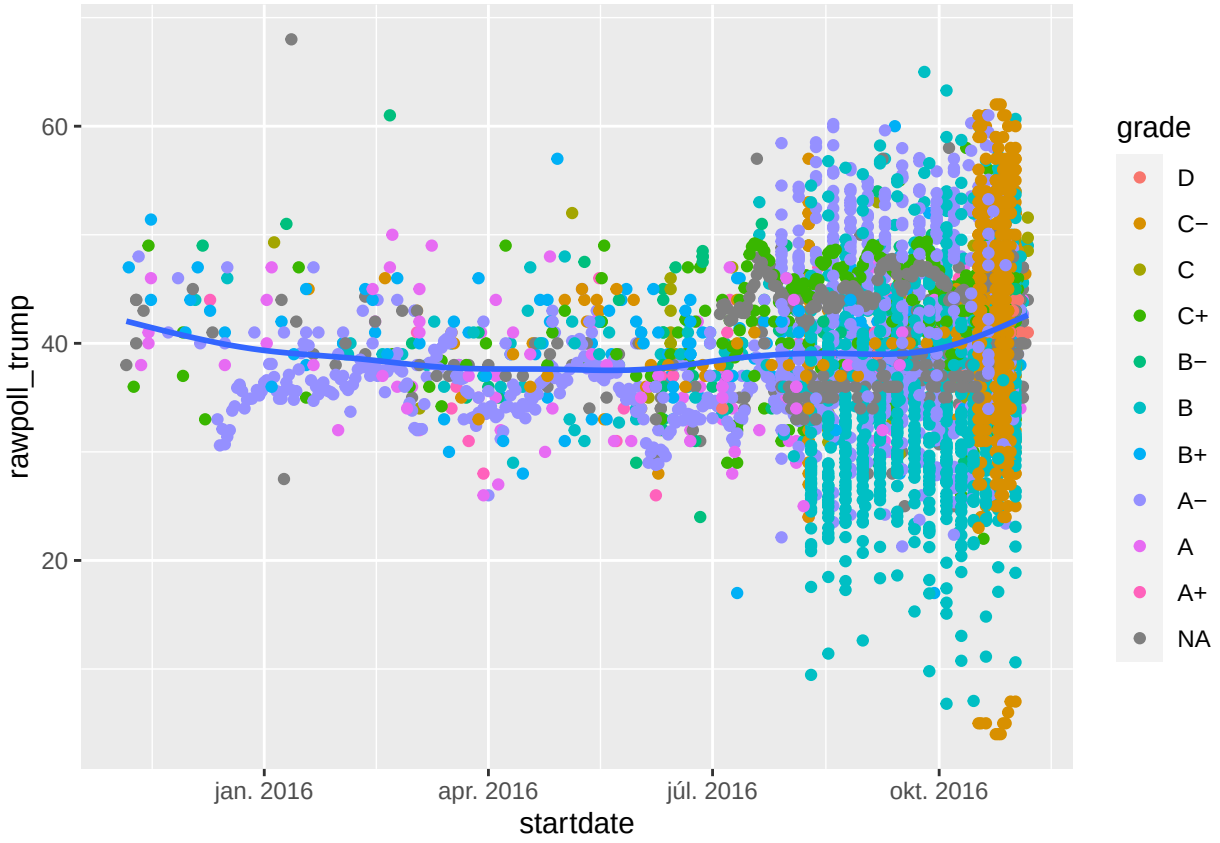
```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



These graphs are identical. It is the same command however it is just phrased slightly differently.

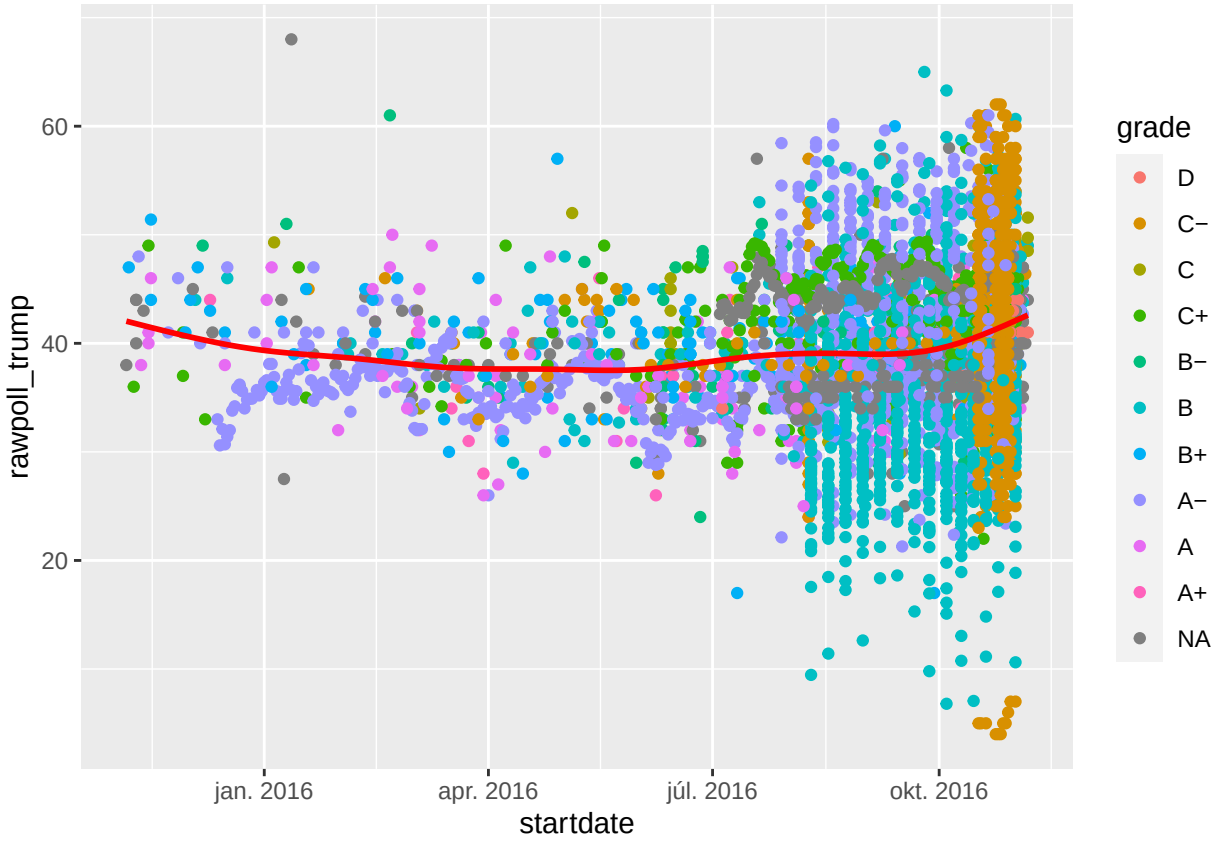
```
#2.4.3
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_trump,
                           color = grade))+
  geom_smooth(data = polls_us_election_2016,mapping = aes(x = startdate,y = rawpoll_trump),
              se=FALSE)
```

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



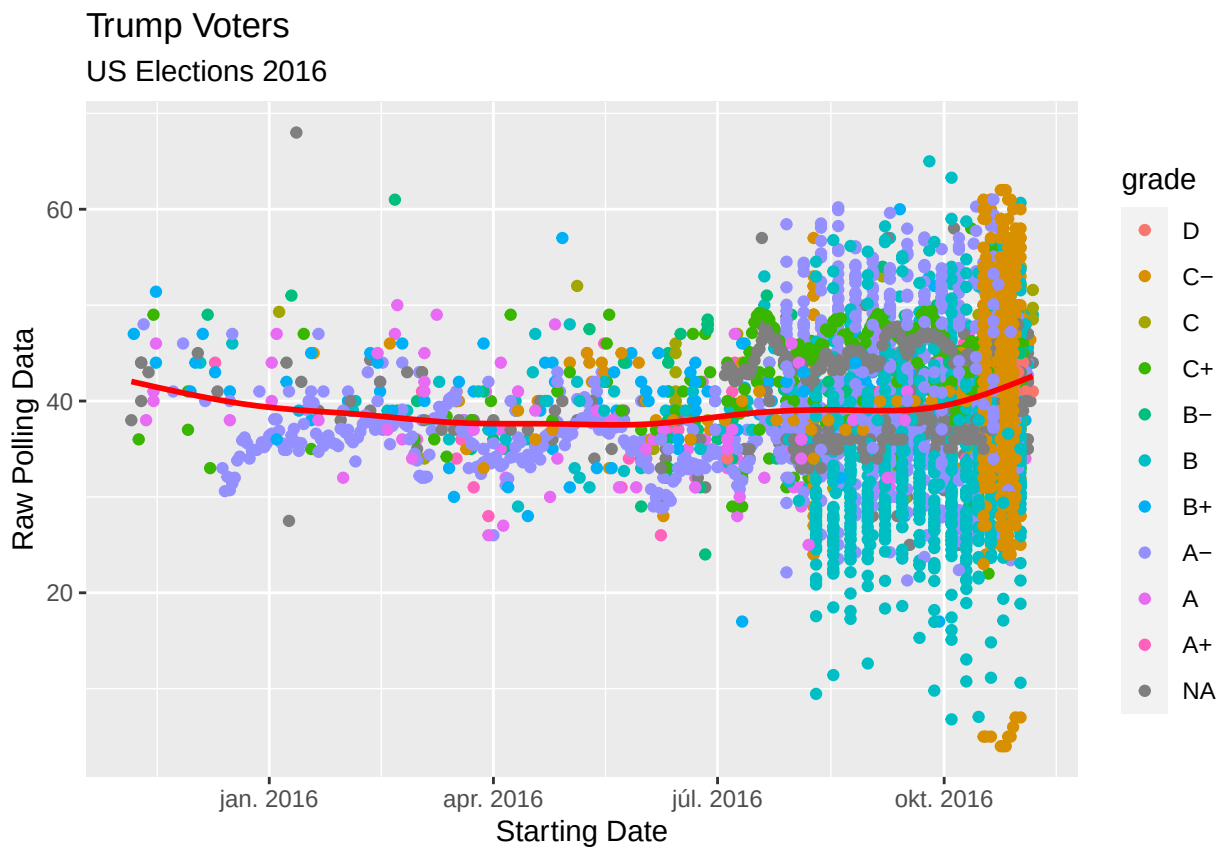

```
#2.4.4
#make line red
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_trump,
                           color = grade))+
  geom_smooth(data = polls_us_election_2016,mapping = aes(x = startdate,y = rawpoll_trump), color = "re
```

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



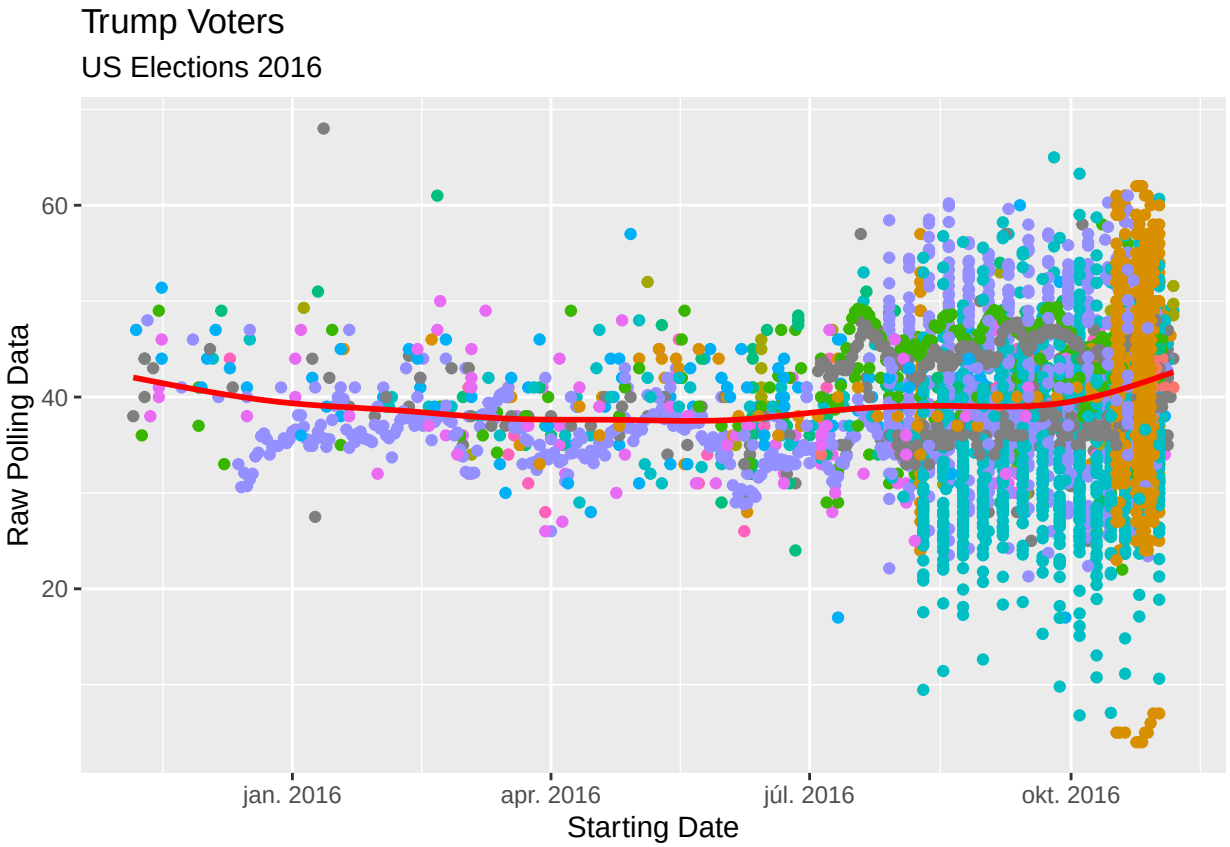

```
#2.4.4
#make the x and y axes labels more informative using +labs() and use and informative title
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_trump,
                           color = grade))+
  geom_smooth(data = polls_us_election_2016,mapping = aes(x = startdate,y = rawpoll_trump), color = "re
  labs(title = "Trump Voters", subtitle = "US Elections 2016", y = "Raw Polling Data", x = "Starting Da
```

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




```
#Remove legend
ggplot(data = polls_us_election_2016) +
  geom_point(mapping = aes(x = startdate,
                           y = rawpoll_trump,
                           color = grade))+
  geom_smooth(data = polls_us_election_2016,mapping = aes(x = startdate,y = rawpoll_trump), color = "red
  labs(title = "Trump Voters", subtitle = "US Elections 2016", y = "Raw Polling Data", x = "Starting Da
  theme(legend.position = "none")
```

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

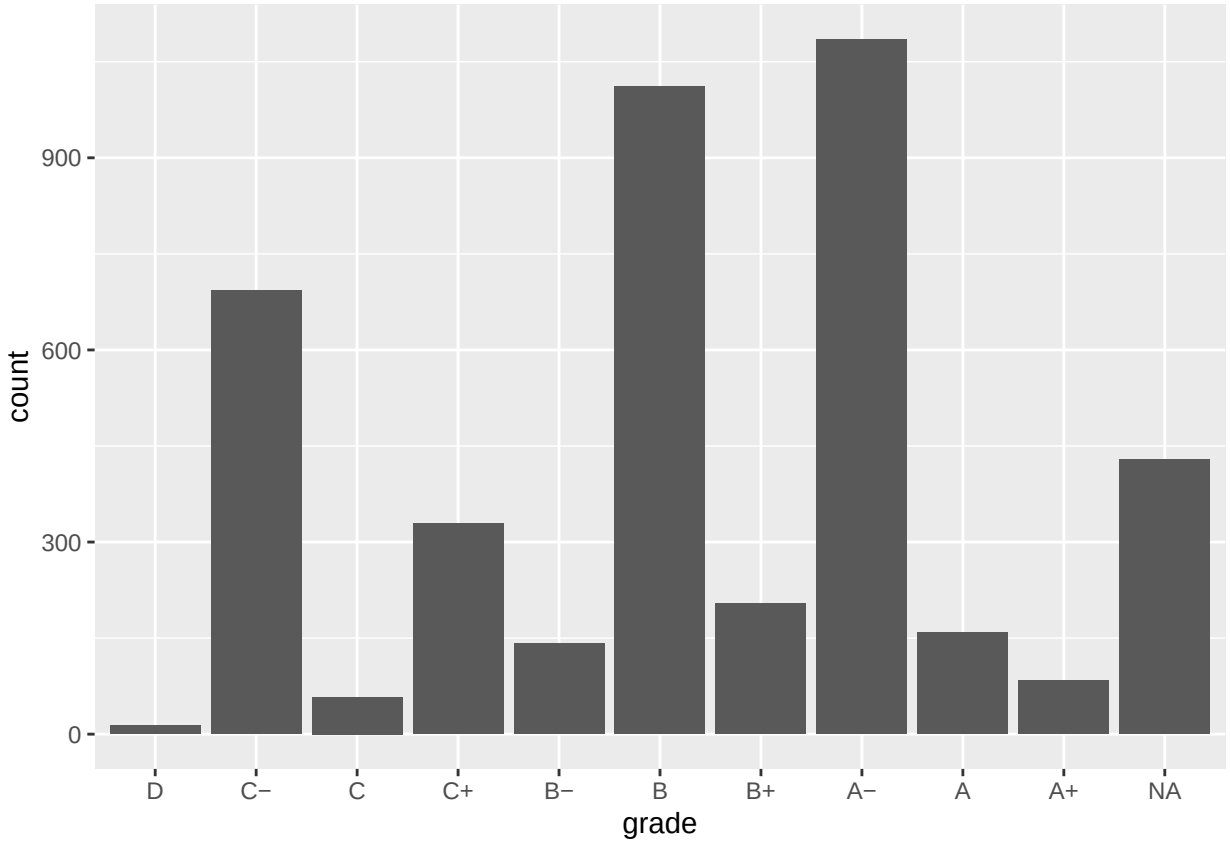


Taking the grade out of the grade/legend made the graph less informing as now we have no idea what the quality of the pollsters are. Adding a more informative title and information to the x and y lab was definetly an improvement. The changing of the color of the however was not extreemly necessary however it does not make the graph any worse off.

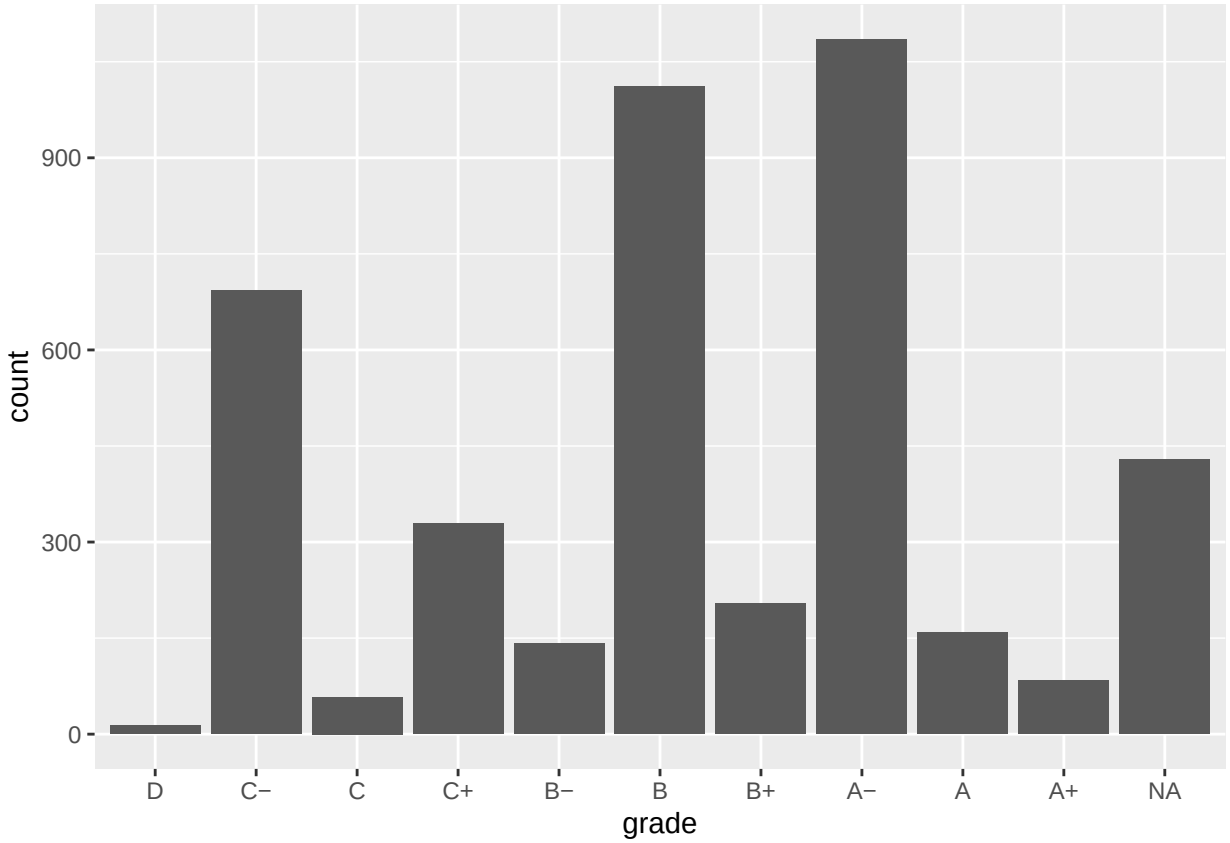
```
#2.4.1
```

2.4.1.1 In geom it is possible to choose between two bar charts: geom_bar or geom_col. They slighlty different.In order to make the heights of the bars represent the values in the data it is more beneficial to use geom_col. However geom_bar will make the height of the bar proportional to the number of cases in each group or the sum of weights if aesthetic is supplied. If you want the heights of the bars to represent values in the data, use geom_col()

```
#2.4.1.2
ggplot(data=polls_us_election_2016, aes(x=grade)) + geom_bar()
```



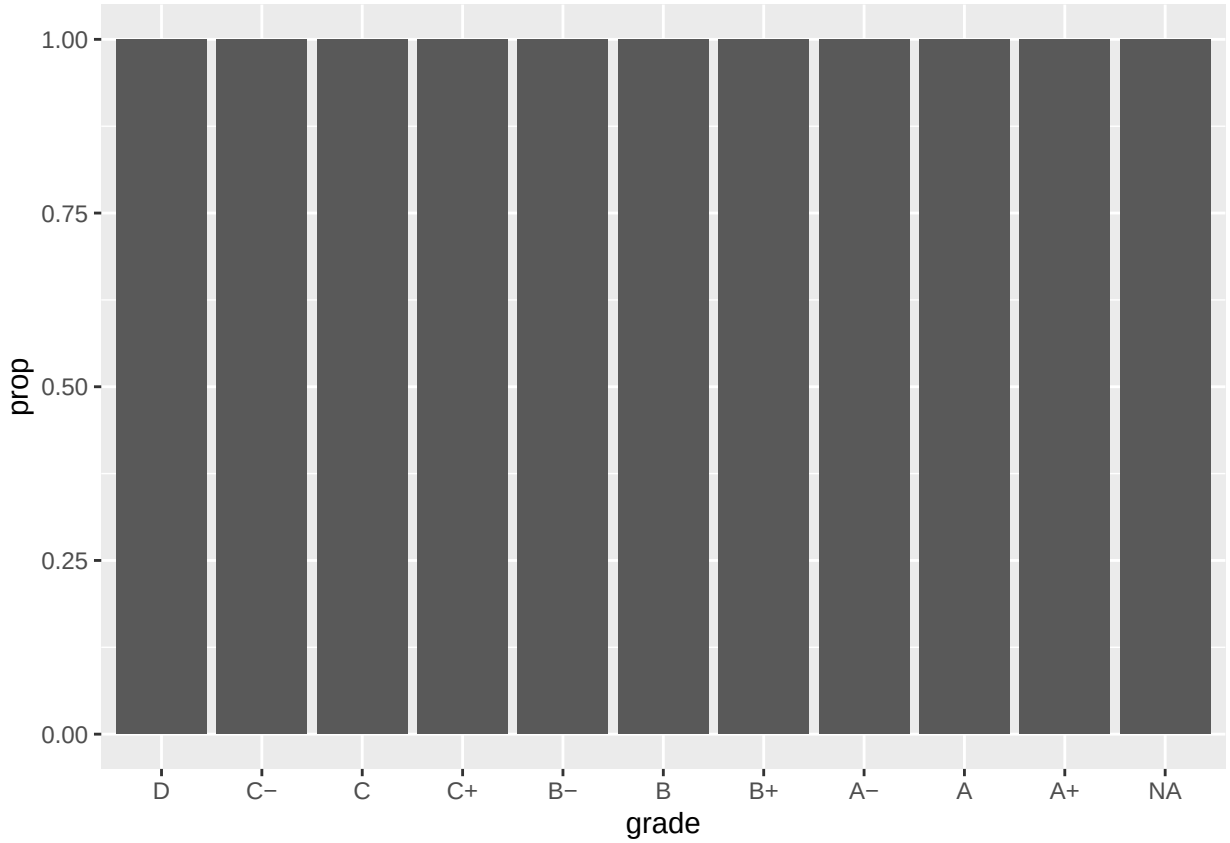

```
#2.4.1.2
ggplot(data=polls_us_election_2016, aes(x=grade)) + stat_count()
```




```
#2.4.1.3
```

stat_smooth computes y which is predicted value, ymin which is lower pointwise confidence interval around the mean, ymax which is upper pointwise confidence interval around the mean, se which is the standard error. The x and y data are the parameters that control its behavior.

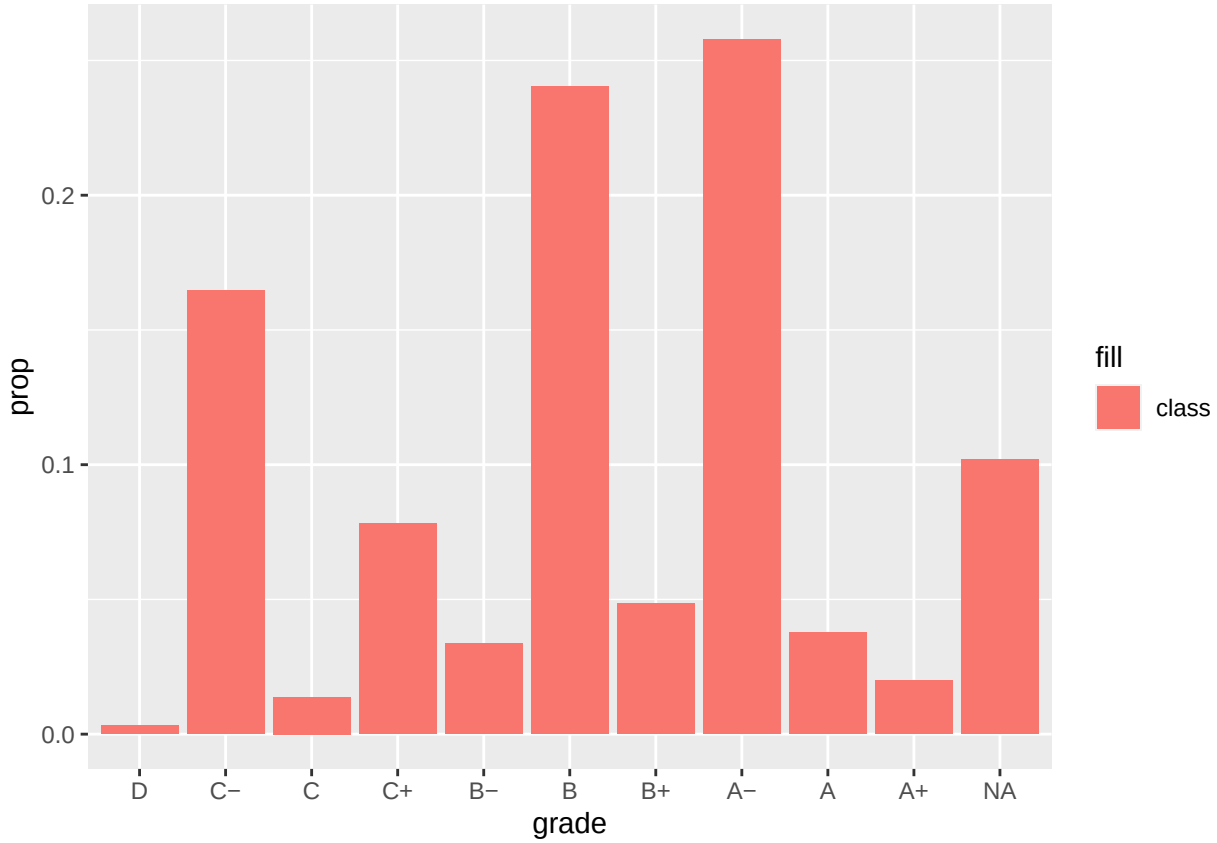
```
#2.4.1.4
ggplot(data = polls_us_election_2016) +
  geom_bar(mapping = aes(x = grade, y = ..prop..))
```



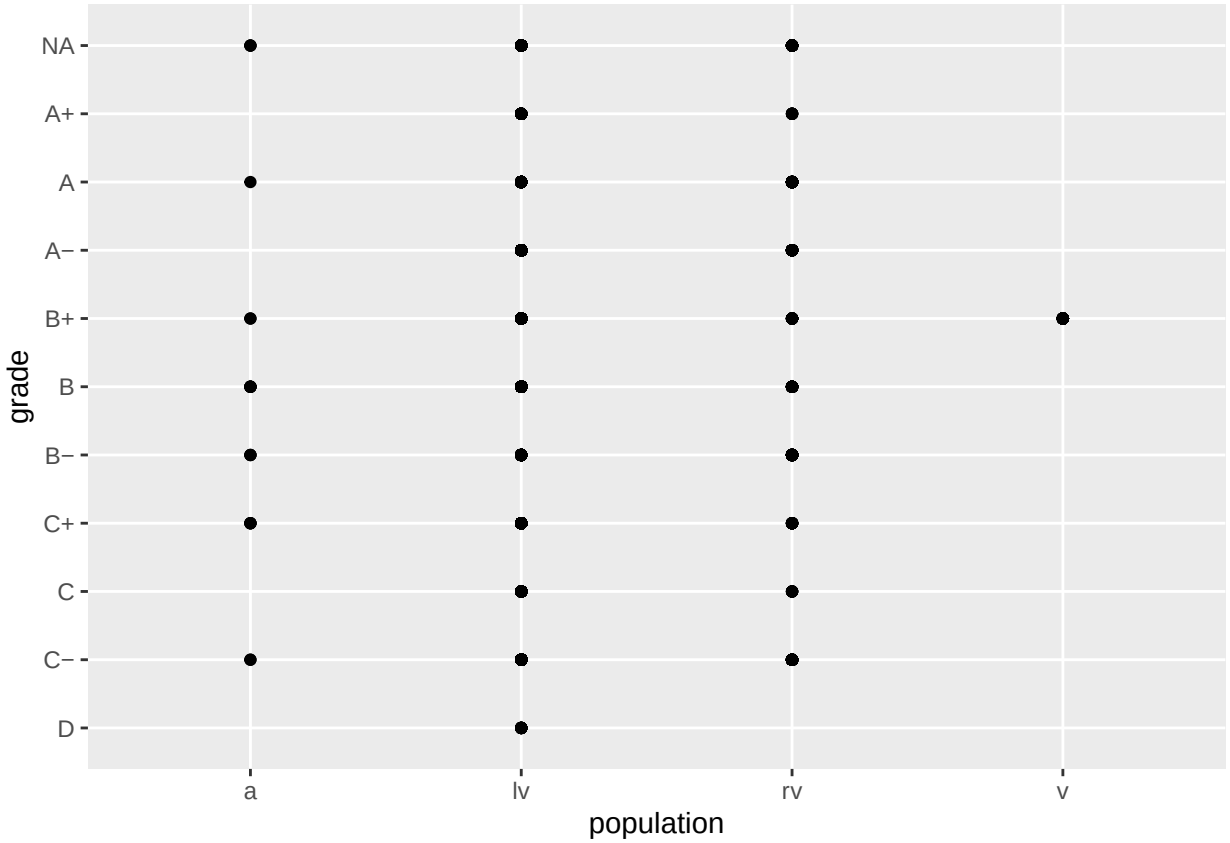
the problem is with the y axis which is labelled as "..prop.." Which signifies proportion. you should rather use stat and count as given an example in the graph below.

```
ggplot(data = polls_us_election_2016) +
  geom_bar(mapping = aes(x = grade, y = ..prop..,fill='class',stat='count',group=factor(1)))
```

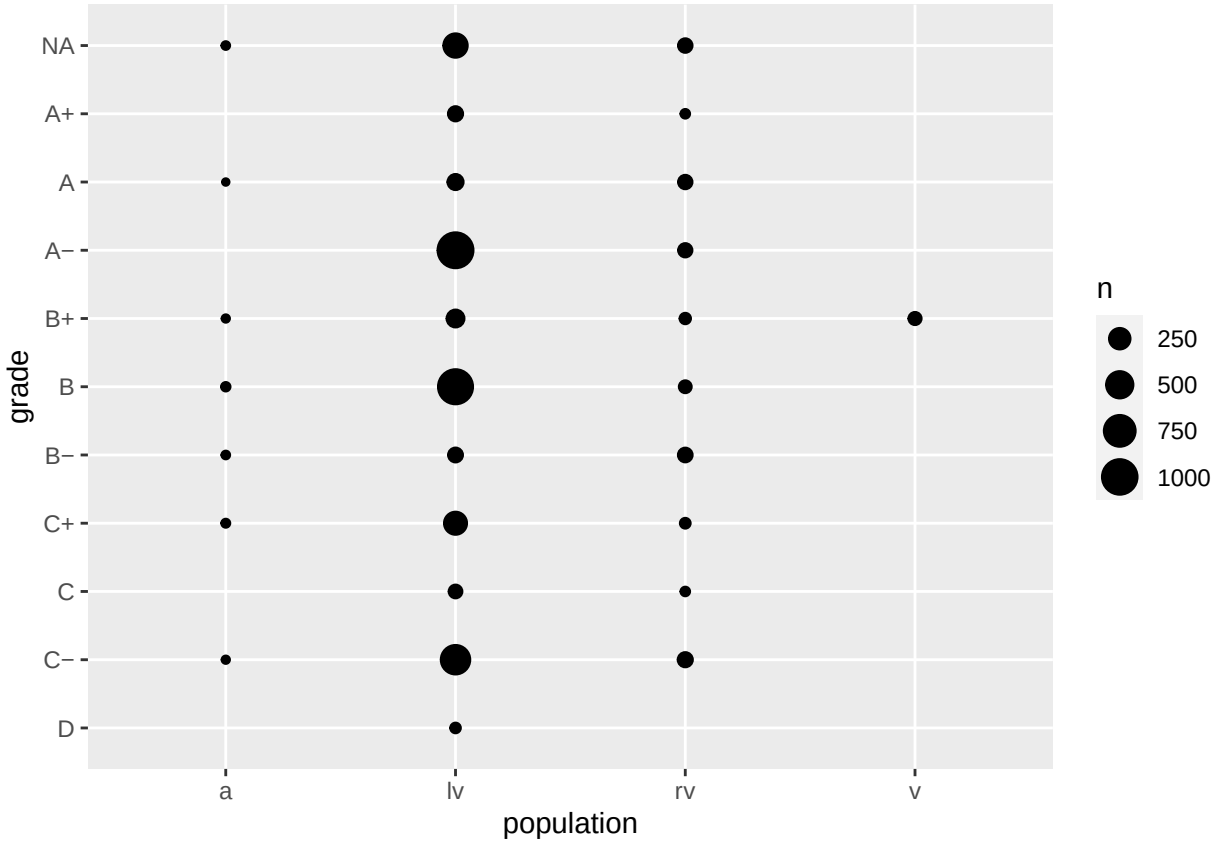
```
## Warning: Ignoring unknown aesthetics: stat
```




```
#2.5.1
#plot 1
ggplot(data = polls_us_election_2016,mapping = aes(x = population, y = grade)) +geom_point()
```




```
#plot 2
ggplot(data = polls_us_election_2016,mapping = aes(x = population, y = grade)) +geom_count()
```

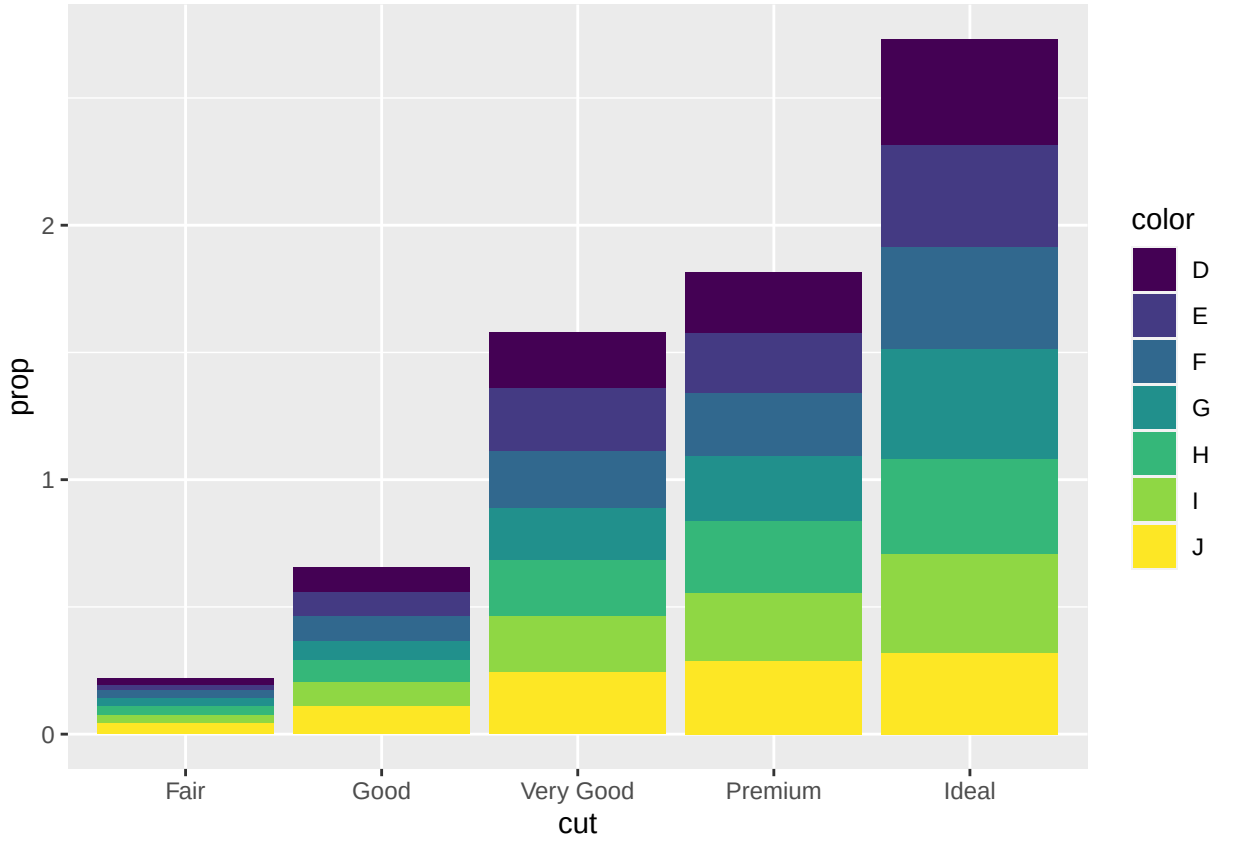


The problem with this first plot is only giving us particular points in the data but not really accurately showing us the distribution in the graph. The second graph is much more concise as it shows us how many values are on each point.We can therefore more accurately see the distribution of the data in the second version.

2.5.2 Geom_jitter spreads out points that would otherwise usually overlap under geom_count so that you can see them better. geom_jitter adds a small variation to the location of each point and is more useful for smaller datasets.

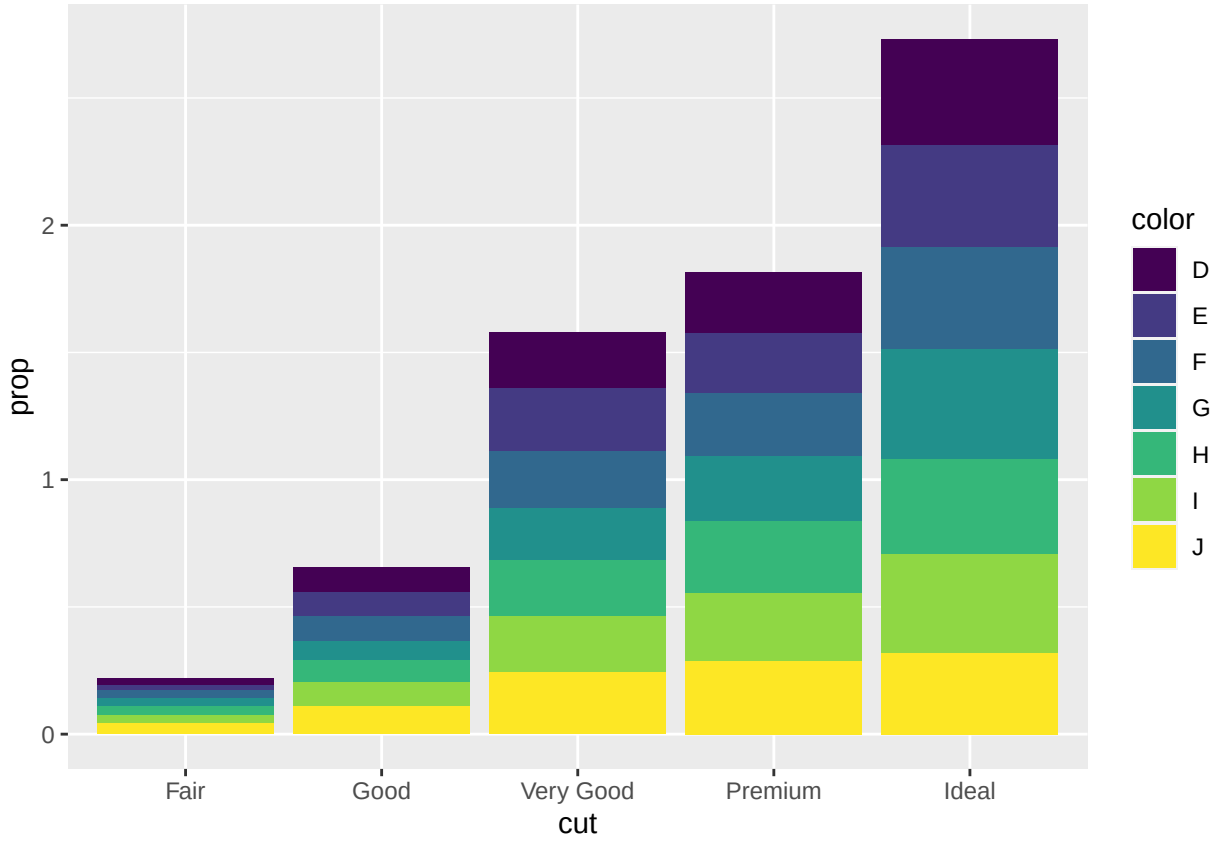
```
#2.5.3
ggplot(data = diamonds)+
geom_bar(mapping = aes(x = cut, fill = color, y = ..prop.., group=color), postion = "dodge")
```

```
## Warning: Ignoring unknown parameters: postion
```




```
ggplot(data = diamonds)+
geom_bar(mapping = aes(x = cut, fill = color, y = ..prop.., group=color), postion = "stack")
```

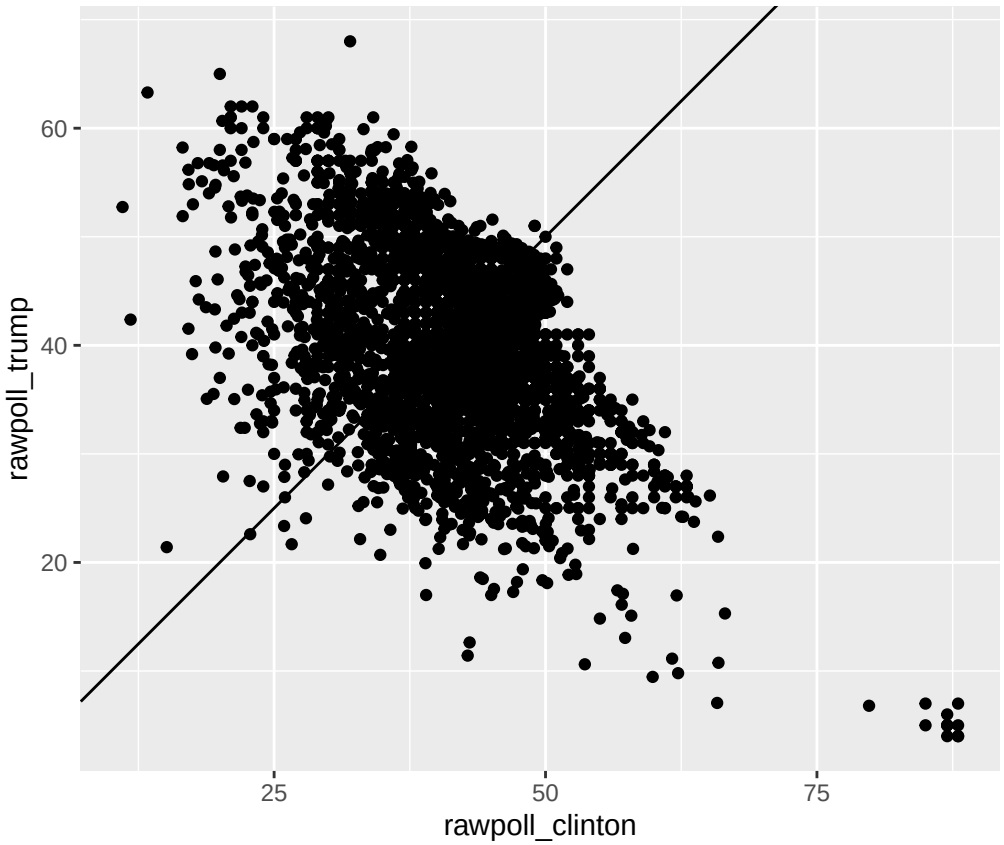
```
## Warning: Ignoring unknown parameters: postion
```




```
You need to run stack instead of dodge

2.6.1
Flip cartesian will change horizontal to vertical and vice versa. This is mostly useful when convering g


'''r
#2.6.2
ggplot(data = polls_us_election_2016,mapping = aes(x = rawpoll_clinton,y = rawpoll_trump))+geom_point()
```



geom_abline() draws a line through the plot. coord_fixed() fixes the ration between x and y so they are proportionally correct. This plot is explaing the us elections and the raw data from the polls of Hillary Clinton and Donald Trump.