Team 5
Yifan Feng: 2671027
Yunxiang Li: 2674844

1a)

```
> (rel_by_region <- gss_sm %>%group_by(bigregion, religion)
%>%summarize(N = n()))
```

```
# A tibble: 24 x 3
# Groups:   bigregion [4]
  bigregion religion       N
  <fct>     <fct>      <int>
 1 Northeast Protestant   158
 2 Northeast Catholic     162
 3 Northeast Jewish        27
 4 Northeast None         112
 5 Northeast Other         28
 6 Northeast NA             1
 7 Midwest   Protestant   325
 8 Midwest   Catholic     172
 9 Midwest   Jewish         3
10 Midwest   None         157
# … with 14 more rows
```

1b)

```
> (rel_by_region <- gss_sm %>%group_by(bigregion, religion))
```

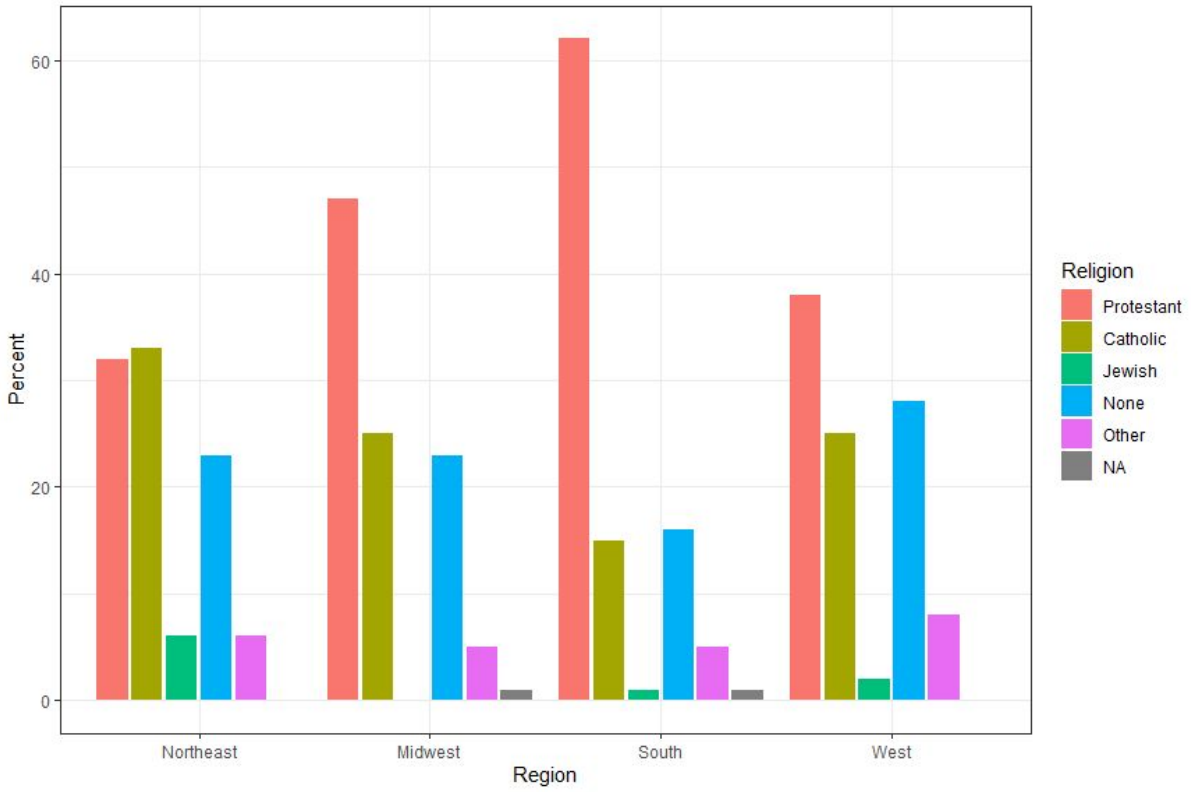
```
# A tibble: 2,867 x 32
# Groups:   bigregion, religion [24]
    year    id ballot   age childs  sibs degree   race  sex
   <dbl> <dbl>  <dbl> <dbl>  <dbl> <dbl> <fct>    <fct> <fct>
 1  2016     1      1    47      3     2 Bachelor White Male
 2  2016     2      2    61      0     3 High Sc… White Male
 3  2016     3      3    72      2     3 Bachelor White Male
 4  2016     4      1    43      4     3 High Sc… White Fema…
 5  2016     5      3    55      2     2 Graduate White Fema…
 6  2016     6      2    53      2     2 Junior … White Fema…
 7  2016     7      1    50      2     2 High Sc… White Male
 8  2016     8      3    23      3     6 High Sc… Other Fema…
 9  2016     9      1    45      3     5 High Sc… Black Male
10  2016    10      3    71      4     1 Junior … White Male
# … with 2,857 more rows, and 23 more variables:
#   region <fct>, income16 <fct>, relig <fct>,
```

```
#   marital <fct>, padeg <fct>, madeg <fct>, partyid <fct>,
#   polviews <fct>, happy <fct>, partners <fct>,
#   grass <fct>, zodiac <fct>, pres12 <dbl>, wtssall <dbl>,
#   income_rc <fct>, agegrp <fct>, ageq <fct>,
#   siblings <fct>, kids <fct>, religion <fct>,
#   bigregion <fct>, partners_rc <fct>, obama <dbl>
```

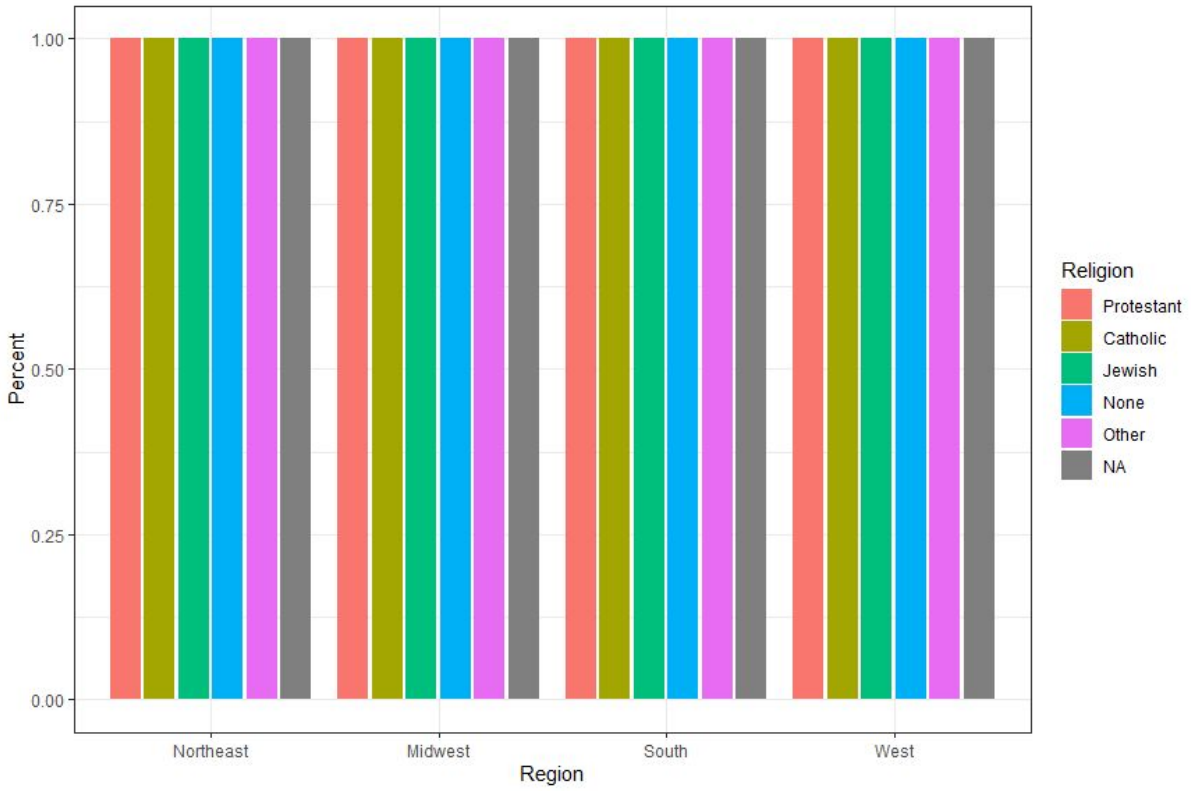
It returns a huge table with 2867 rows and 32 columns, which is grouped by bigregion and religion. There are 32 variables added to the table in total. N has been removed.

1c)

The graph by geom_col:




```
> ggplot(rel_by_region, aes(x = bigregion, fill = religion)) + geom_bar(position =
"dodge2") + labs(x = "Region",y = "Percent", fill = "Religion") + theme(legend.position =
"top") + theme_bw()
```

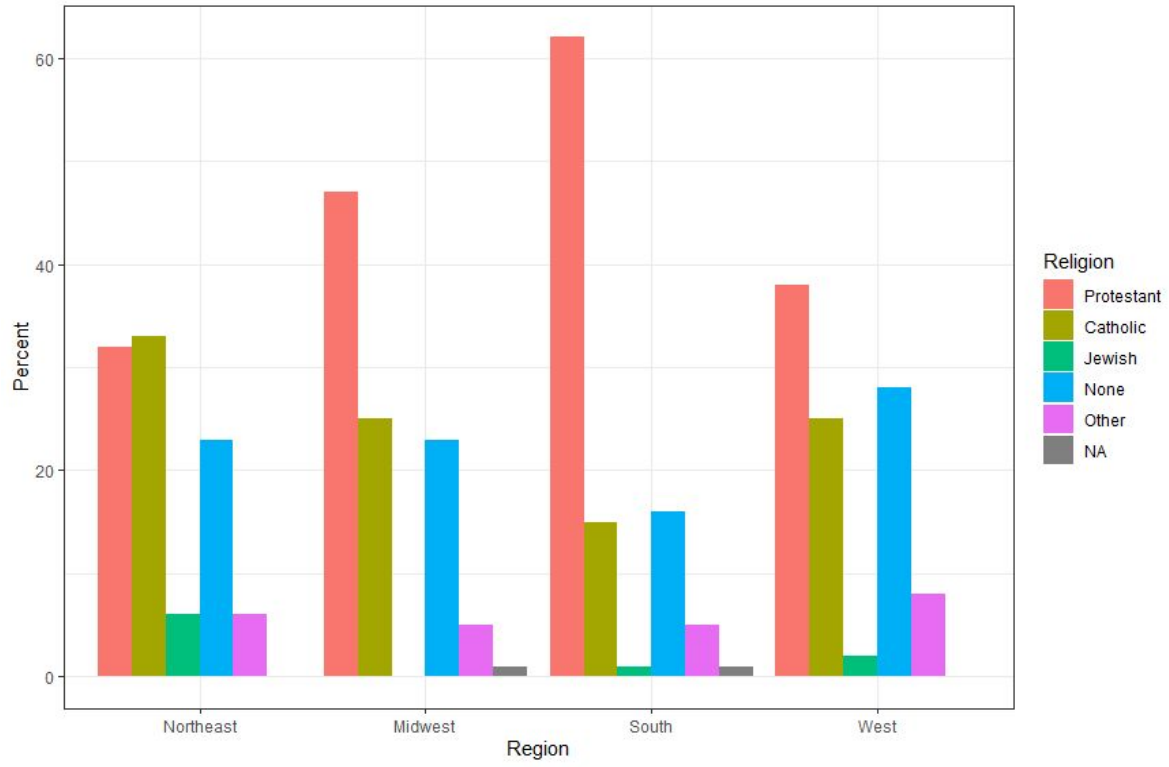


#The **geom_bar( )** doesn't take y aesthetic, so there are no continuous value y showed in the plot.

1d)

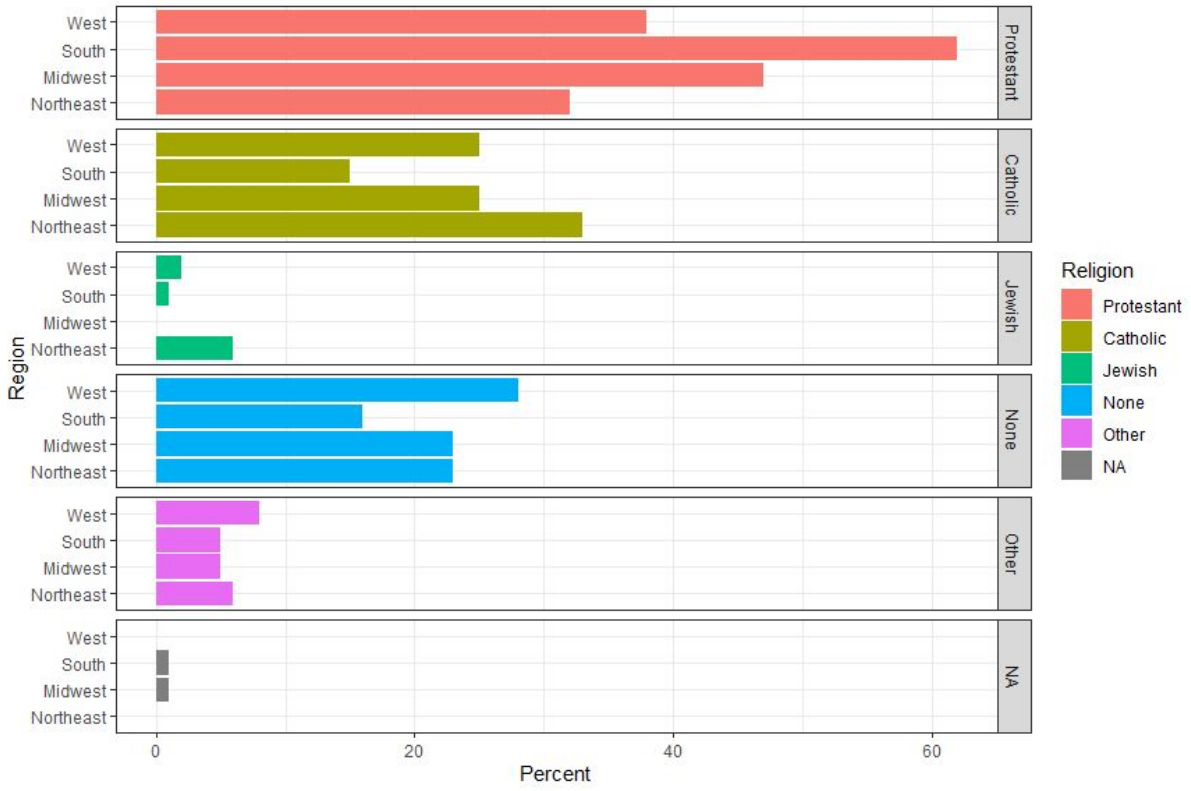
```
> ggplot(rel_by_region, aes(x = bigregion, y = pct, fill = religion)) + geom_col(position = "dodge") + labs(x = "Region",y = "Percent", fill = "Religion") + theme(legend.position = "top") + theme_bw()
```

#The white space (padding) between bars in each religion has disappeared. In dodge2, default argument padding is added (default = 0.1), which can make the bars more distinguishable from one another.



1e)

```
>(rel_by_region <- gss_sm %>% group_by(bigregion, religion) %>% summarize(N = n()) %>% mutate(freq = N / sum(N),  pct = round((freq*100), 0)))

> ggplot(rel_by_region, aes(x = bigregion, y = pct, fill = religion)) + geom_col(position = "dodge2") + labs(x = "Region",y = "Percent", fill = "Religion") + theme(legend.position = "top") + theme_bw() + coord_flip() + facet_grid(religion~.)
```

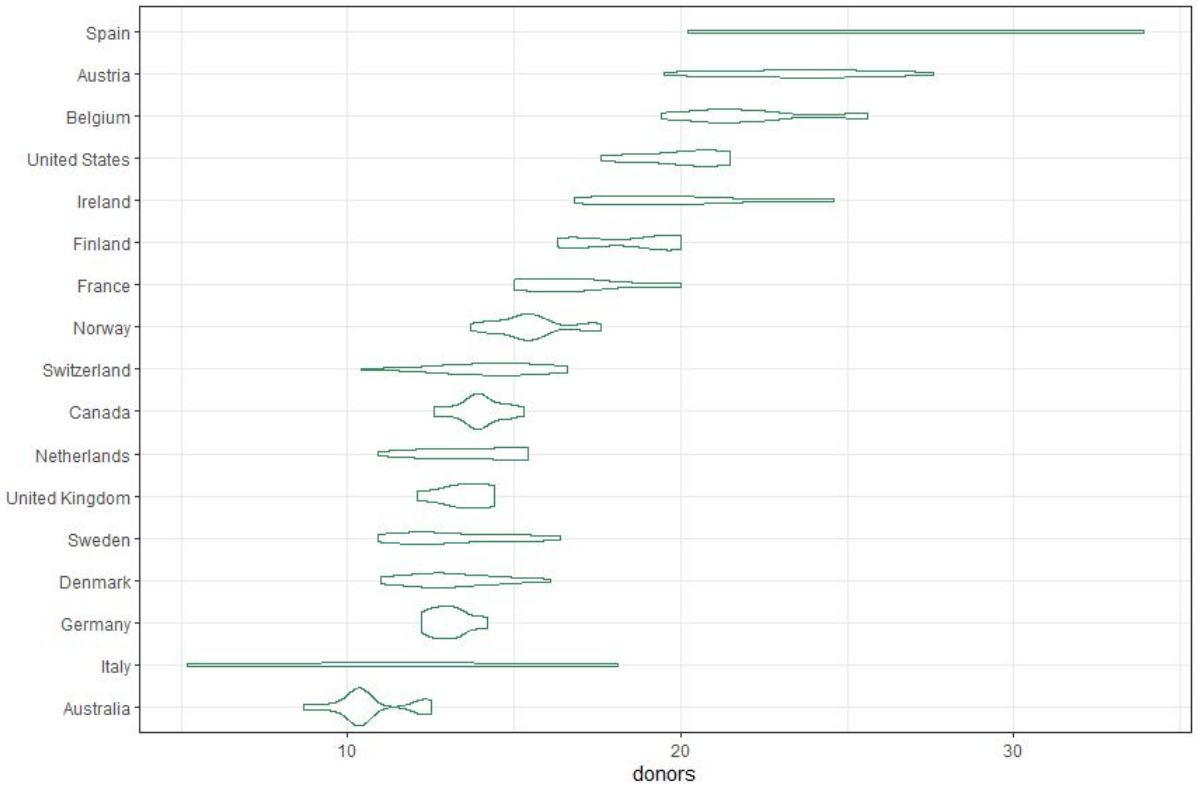


#**coord_flip( )** transforms a graph's x-axis and y-axis, meaning that horizontal becomes vertical and vice versa. This is useful only when it comes to a situation we want to display y condicional on x (stress on x).
If we only apply facet_grid(.~religion) without the coord_flip(), the region names on the x-axis will overlap.

#**facet_grid( )** can split the data by variables vary on the horizontal or vertical direction. Divided by the religion in the vertical direction, the graph shows the popularity of each religion in relation to different areas. It becomes much easier to compare the differences between the region.

2a)

```
> ggplot(data = organdata, mapping = aes(x = reorder(country, donors, na.rm=TRUE), y = donors)) + geom_violin(colour = "seagreen") + labs(x=NULL) + coord_flip() + theme_bw()
```

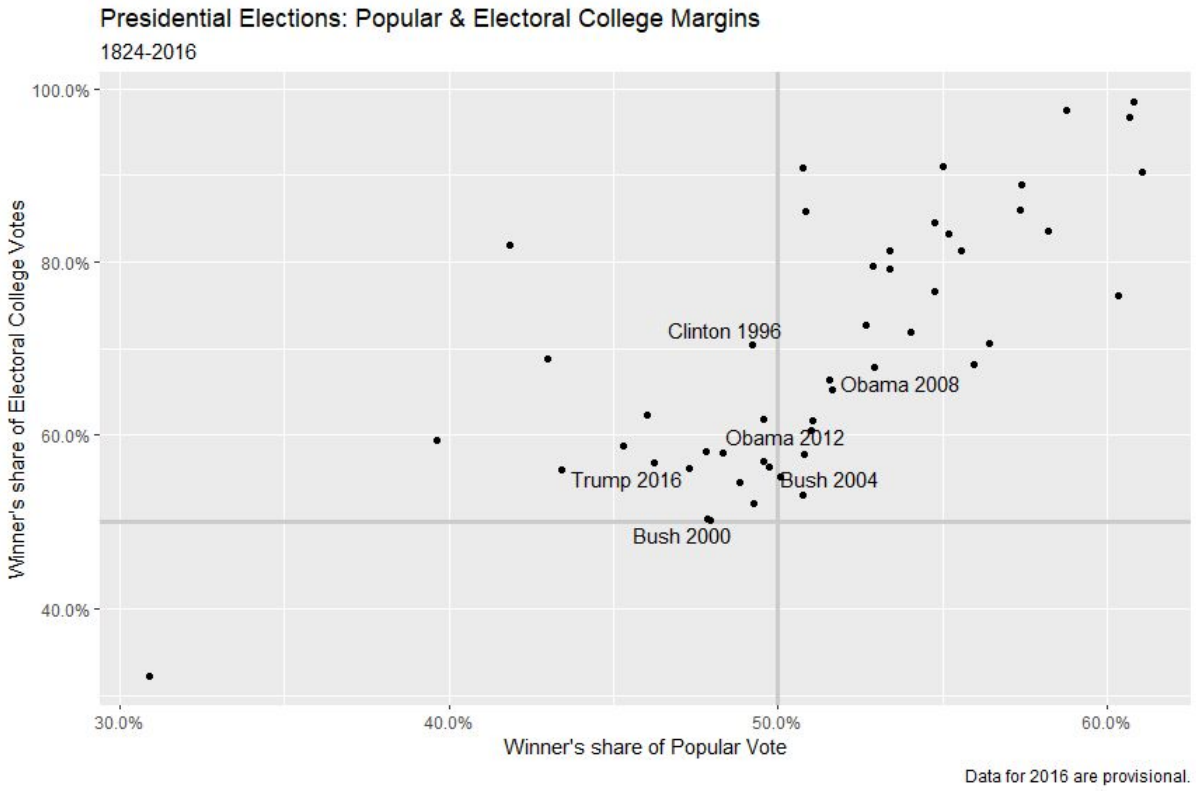


2b)

#The shape of one plot has been changed from box to violin. **Geom_boxplot ( )** can only show the distribution of data regarding five limited summary statistics (i.e. min, max, median, Q1 and Q2) but Violin plot can display the variation of data.

2c)

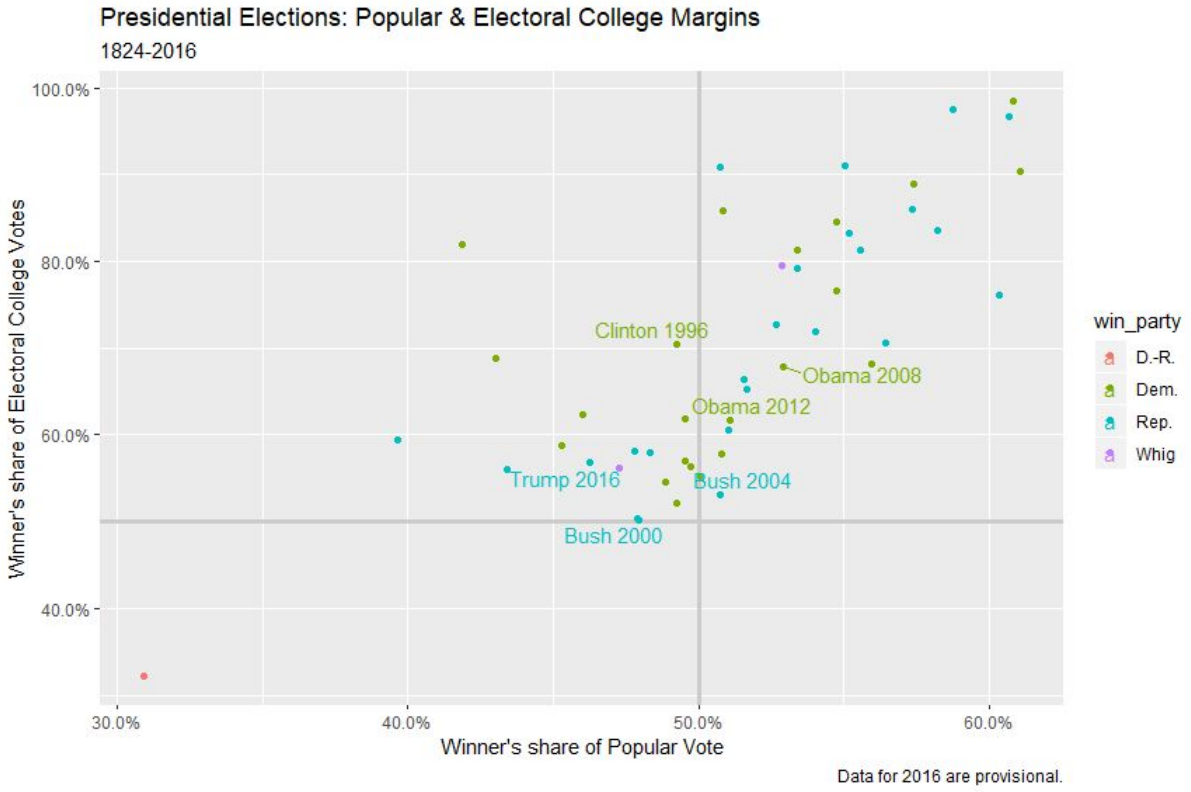
```
> p_title <- "Presidential Elections: Popular & Electoral College Margins"
p_subtitle <- "1824-2016"
p_caption <- "Data for 2016 are provisional."

x_label <- "Winner's share of Popular Vote"
y_label <- "Winner's share of Electoral College Votes"
p <- ggplot(elections_historic, aes(x=popular_pct, y=ec_pct, label = winner_label))
p + geom_hline(yintercept = 0.5, size = 1.4, color = "gray80") + geom_vline(xintercept =
0.5, size = 1.4, color = "gray80") + geom_point() + geom_text_repel(data =
subset(elections_historic, year > 1992)) + scale_x_continuous(labels = scales::percent) +
scale_y_continuous(labels = scales::percent) + labs(x=x_label, y=y_label, title = p_title,
subtitle = p_subtitle, caption = p_caption)
```



2d)

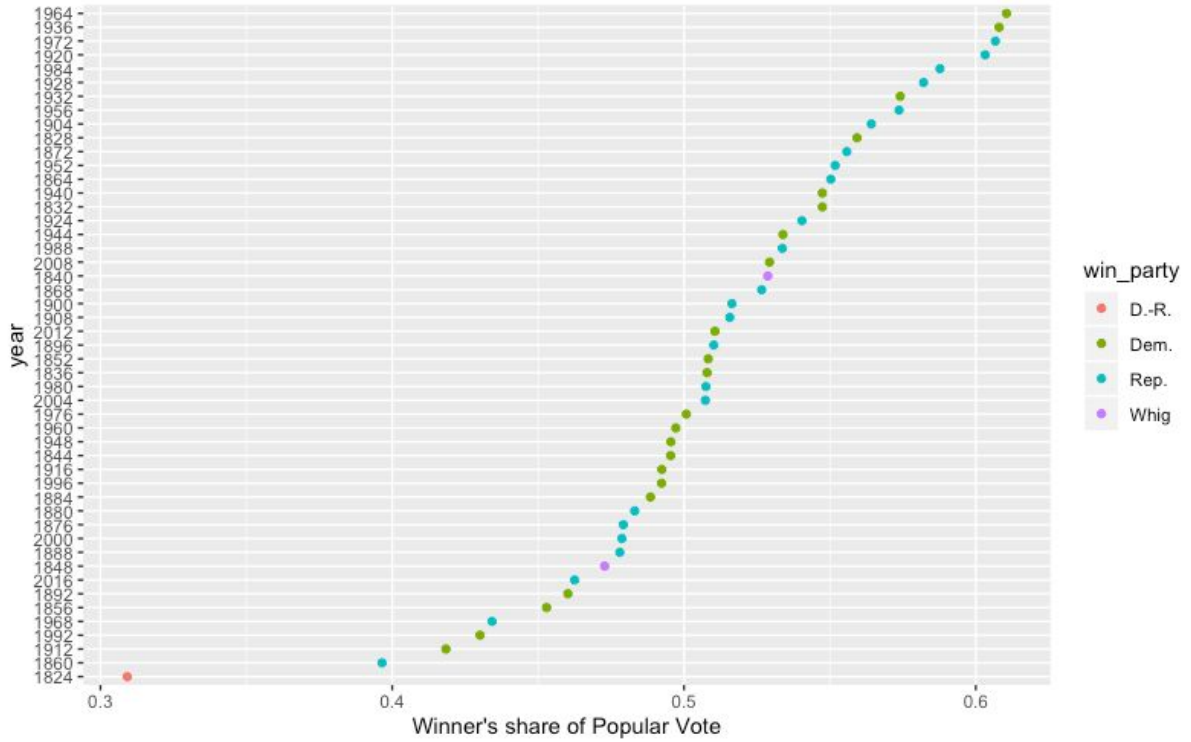
```
> x_label <- "Winner's share of Popular Vote"
> y_label <- "Winner's share of Electoral College Votes"
> p <- ggplot(elections_historic, aes(x=popular_pct, y=ec_pct, label = winner_label), color
=win_party)
> p + geom_hline(yintercept = 0.5, size = 1.4, color = "gray80") + geom_vline(xintercept =
0.5, size = 1.4, color = "gray80") + geom_point() + geom_text_repel(data =
subset(elections_historic, year > 1992)) + scale_x_continuous(labels = scales::percent) +
scale_y_continuous(labels = scales::percent) + labs(x=x_label, y=y_label, title = p_title,
subtitle = p_subtitle, caption = p_caption)
> p_title <- "Presidential Elections: Popular & Electoral College Margins"
> p_subtitle <- "1824-2016"
> p_caption <- "Data for 2016 are provisional."
>
> x_label <- "Winner's share of Popular Vote"
> y_label <- "Winner's share of Electoral College Votes"
> p <- ggplot(elections_historic, aes(x=popular_pct, y=ec_pct, label = winner_label, color
=win_party))
> p + geom_hline(yintercept = 0.5, size = 1.4, color = "gray80") + geom_vline(xintercept =
0.5, size = 1.4, color = "gray80") + geom_point() + geom_text_repel(data =
subset(elections_historic, year > 1992)) + scale_x_continuous(labels = scales::percent) +
```

```
scale_y_continuous(labels = scales::percent) + labs(x=x_label, y=y_label, title = p_title, subtitle = p_subtitle, caption = p_caption)
```



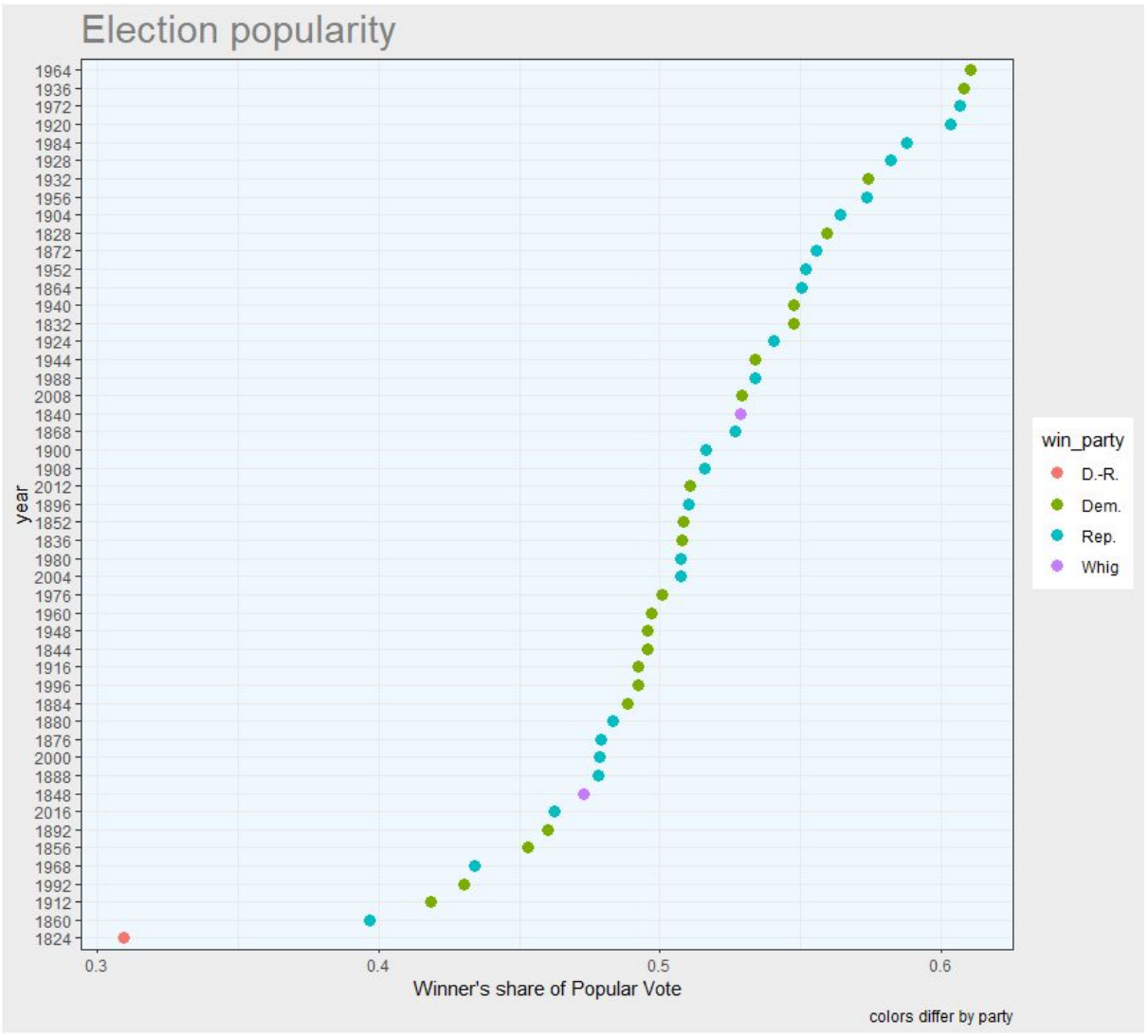
2e)

```
>ggplot(data = elections_historic, mapping = aes(x = popular_pct , y = reorder(year, popular_pct, na.rm=TRUE), color=win_party)) + geom_point() + labs(x = "Winner's share of Popular Vote", y = "year")
```



2f)

```
#We add grey and light blue background to make the graph more readable. Besides, we create headline for better understanding of the value of the data.

> ggplot(data = elections_historic, mapping = aes(x = popular_pct , y = reorder(year, popular_pct, na.rm=TRUE), color = win_party)) + geom_point(size = 3) + labs(x = "Winner's share of Popular Vote", y = "year", caption = "colors differ by party", title = "Election popularity") + theme(panel.background = element_rect(fill = "aliceblue"),plot.background = element_rect(fill = "gray93"),plot.title = element_text(size = rel(2),color = "gray50"))
```


Election popularity
year
1964
1936
1972
1920
1984
1928
1932
1956
1904
1828
1872
1952
1864
1940
1832
1924
1944
1988
2008
1840
1868
1900
1908
2012
1896
1852
1836
1980
2004
1976
1960
1948
1844
1916
1996
1884
1880
1876
2000
1888
1848
2016
1892
1856
1968
1992
1912
1860
1824
0.3
0.4
0.5
0.6
Winner's share of Popular Vote
colors differ by party
win_party
D.-R.
Dem.
Rep.
Whig