La2

Ankit -1nt20is023

Huzaif-1nt20is089

2023-05-22

```
library(readxl)
Book3 <- read_excel("C:/Users/ankit/Downloads/Book3.xlsx")
head(Book3)
```

```
## # A tibble: 6 × 6
##   state    county            candidate       party  votes won
##   <chr>    <chr>             <chr>           <chr>  <dbl> <lgl>
## 1 Delaware Kent County       John Carney     DEM    44352 TRUE
## 2 Delaware Kent County       Julianne Murray REP    39332 FALSE
## 3 Delaware Kent County       Kathy DeMatteis IPD     1115 FALSE
## 4 Delaware Kent County       John Machurek   LIB      616 FALSE
## 5 Delaware New Castle County John Carney     DEM   191678 TRUE
## 6 Delaware New Castle County Julianne Murray REP    82545 FALSE
```

```
head(summary(Book3))
```

```
##     state              county           candidate           party          
##  Length:599         Length:599         Length:599         Length:599        
##  Class :character   Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character  
##                                                                             
##                                                                             
##                                                                             
##      votes             won         
##  Min.   :     4.0   Mode :logical  
##  1st Qu.:   610.5   FALSE:426      
##  Median :  2461.0   TRUE :173      
##  Mean   :  8733.5                  
##  3rd Qu.:  7233.5                  
##  Max.   :203475.0
```

```
head(str(Book3))
```

```
## tibble [599 × 6] (S3: tbl_df/tbl/data.frame)
##  $ state    : chr [1:599] "Delaware" "Delaware" "Delaware" "Delaware" ...
##  $ county   : chr [1:599] "Kent County" "Kent County" "Kent County" "Kent
County" ...
##  $ candidate: chr [1:599] "John Carney" "Julianne Murray" "Kathy
DeMatteis" "John Machurek" ...
##  $ party    : chr [1:599] "DEM" "REP" "IPD" "LIB" ...
```

```
##  $ votes    : num [1:599] 44352 39332 1115 616 191678 ...
##  $ won      : logi [1:599] TRUE FALSE FALSE FALSE TRUE FALSE ...

## NULL

nrow(Book3)

## [1] 599

ncol(Book3)

## [1] 6

subset_Book3 <- Book3[, c("state", "county", "candidate", "party", "votes",
"won")]
print(subset_Book3)

## # A tibble: 599 × 6
##    state    county            candidate       party  votes won
##    <chr>    <chr>             <chr>           <chr>  <dbl> <lgl>
##  1 Delaware Kent County       John Carney     DEM    44352 TRUE
##  2 Delaware Kent County       Julianne Murray REP    39332 FALSE
##  3 Delaware Kent County       Kathy DeMatteis IPD     1115 FALSE
##  4 Delaware Kent County       John Machurek   LIB      616 FALSE
##  5 Delaware New Castle County John Carney     DEM   191678 TRUE
##  6 Delaware New Castle County Julianne Murray REP    82545 FALSE
##  7 Delaware New Castle County Kathy DeMatteis IPD     3785 FALSE
##  8 Delaware New Castle County John Machurek   LIB     2031 FALSE
##  9 Delaware Sussex County     Julianne Murray REP    68435 TRUE
## 10 Delaware Sussex County     John Carney     DEM    56873 FALSE
## # ℹ 589 more rows

won_dataset <- subset(Book3, won == "won")
print(won_dataset)

## # A tibble: 0 × 6
## # ℹ 6 variables: state <chr>, county <chr>, candidate <chr>, party <chr>,
## #   votes <dbl>, won <lgl>

sorted_dataset <- Book3[order(Book3$votes, decreasing = TRUE), ]
print(sorted_dataset)

## # A tibble: 599 × 6
##    state    county            candidate       party  votes won
##    <chr>    <chr>             <chr>           <chr>  <dbl> <lgl>
##  1 Indiana  Marion County     Woody Myers     DEM   203475 TRUE
##  2 Missouri Jackson County    Nicole Galloway DEM   194273 TRUE
##  3 Delaware New Castle County John Carney     DEM   191678 TRUE
##  4 Indiana  Marion County     Eric Holcomb    REP   152405 FALSE
##  5 Missouri Jackson County    Mike Parson     REP   128938 FALSE
##  6 Indiana  Hamilton County   Eric Holcomb    REP   117749 TRUE
##  7 Indiana  Lake County       Woody Myers     DEM   112352 TRUE
```

```
##  8 Indiana  Allen County      Eric Holcomb    REP    98406 TRUE
##  9 Indiana  Lake County       Eric Holcomb    REP    94841 FALSE
## 10 Missouri Greene County     Mike Parson     REP    84582 TRUE
## # ℹ 589 more rows

library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

total_votes <- Book3 %>% group_by(candidate) %>% summarise(total_votes =
sum(votes))
print(total_votes)

## # A tibble: 11 × 2
##    candidate        total_votes
##    <chr>                  <dbl>
##  1 Donald Rainwater      345567
##  2 Eric Holcomb         1706724
##  3 Jerome Bauer            9684
##  4 John Carney           292903
##  5 John Machurek           3270
##  6 Julianne Murray       190312
##  7 Kathy DeMatteis         6150
##  8 Mike Parson          1068545
##  9 Nicole Galloway       610118
## 10 Rik Combs              30002
## 11 Woody Myers           968092

total_votes_party <- Book3%>% group_by(party) %>% summarise(total_votes =
sum(votes))
print(total_votes_party)

## # A tibble: 5 × 2
##   party total_votes
##   <chr>       <dbl>
## 1 DEM       1871113
## 2 GRN          9684
## 3 IPD          6150
## 4 LIB        378839
## 5 REP       2965581

print(head(Book3$percentage_votes <- Book3$votes / sum(Book3$votes) * 100))
```

```
## [1] 0.84780900 0.75184937 0.02131374 0.01177512 3.66401363 1.57788586
```

```
party_percentage_votes <- Book3 %>% group_by(party) %>%
mutate(percentage_votes = votes / sum(votes) * 100)

print(party_percentage_votes)
```

```
## # A tibble: 599 × 7
## # Groups:   party [5]
##    state    county            candidate      party  votes won
percentage_votes
##    <chr>    <chr>             <chr>          <chr>  <dbl> <lgl>
<dbl>
##  1 Delaware Kent County       John Carney    DEM    44352 TRUE
2.37
##  2 Delaware Kent County       Julianne Murr… REP    39332 FALSE
1.33
##  3 Delaware Kent County       Kathy DeMatte… IPD     1115 FALSE
18.1
##  4 Delaware Kent County       John Machurek  LIB      616 FALSE
0.163
##  5 Delaware New Castle County John Carney    DEM   191678 TRUE
10.2
##  6 Delaware New Castle County Julianne Murr… REP    82545 FALSE
2.78
##  7 Delaware New Castle County Kathy DeMatte… IPD     3785 FALSE
61.5
##  8 Delaware New Castle County John Machurek  LIB     2031 FALSE
0.536
##  9 Delaware Sussex County     Julianne Murr… REP    68435 TRUE
2.31
## 10 Delaware Sussex County     John Carney    DEM    56873 FALSE
3.04
## # ℹ 589 more rows
```

```
unique_states <- length(unique(Book3$state))
print(unique_states)
```

```
## [1] 3
```

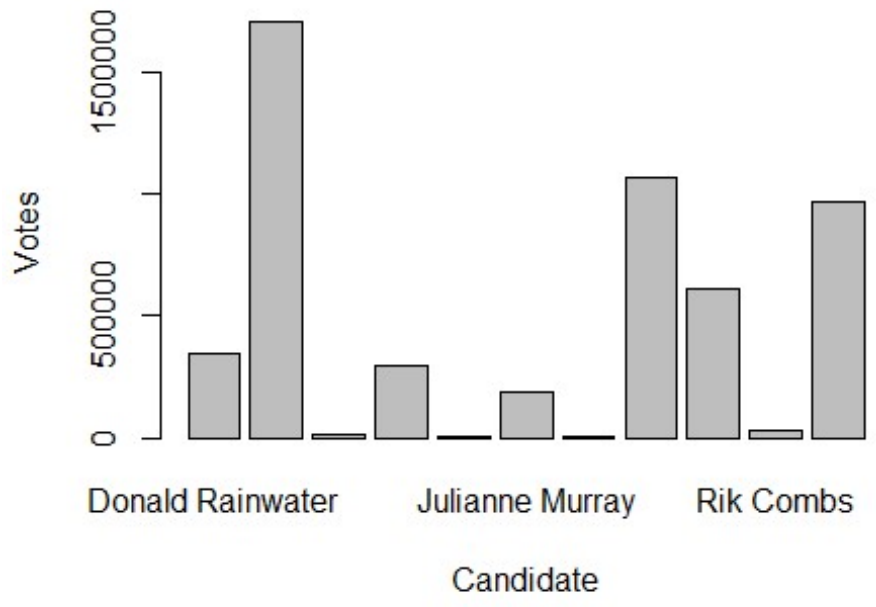
```
unique_counties <- length(unique(Book3$county))

candidate_votes <- sum(Book3$votes[Book3$candidate == "Candidate Name"])
print(candidate_votes)
```

```
## [1] 0
```

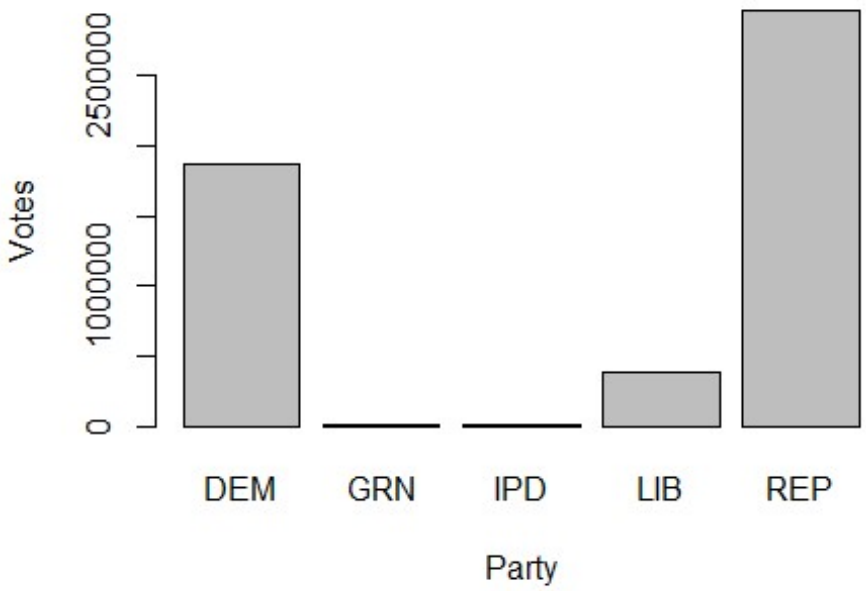
```
party_votes <- sum(Book3$votes[Book3$party == "Party Name"])
print(party_votes)
```

```
## [1] 0
```

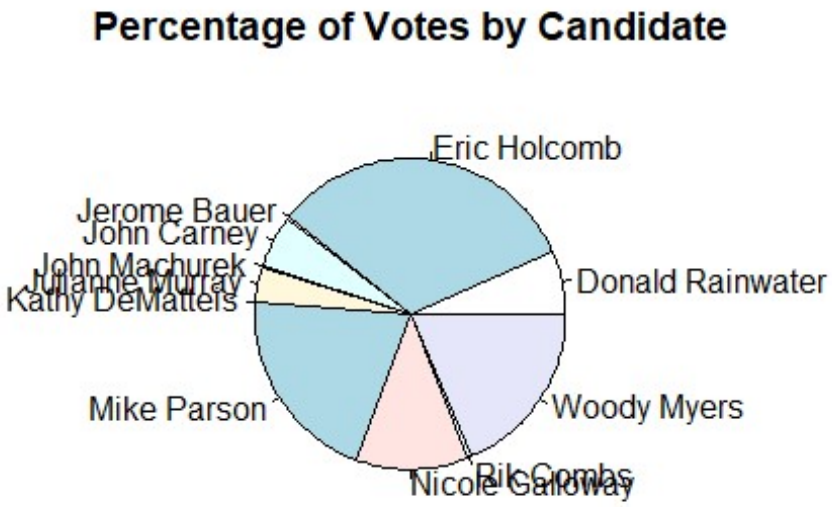
```
mean(Book3$votes)

## [1] 8733.501

median(Book3$votes)

## [1] 2461

max(Book3$votes)

## [1] 203475

min(Book3$votes)

## [1] 4

barplot(total_votes$total_votes, names.arg = total_votes$candidate, xlab =
"Candidate", ylab = "Votes")
```




```
barplot(total_votes_party$total_votes, names.arg = total_votes_party$party,
xlab = "Party", ylab = "Votes")
```




```
pie(total_votes$total_votes, labels = total_votes$candidate, main =
"Percentage of Votes by Candidate")
```



```r
pie(total_votes_party$total_votes, labels = total_votes_party$party, main =
"Percentage of Votes by Party")

avg_votes_by_state <- Book3 %>% group_by(state, candidate) %>%
summarise(avg_votes = mean(votes))

## `summarise()` has grouped output by 'state'. You can override using the
## `.groups` argument.

print(avg_votes_by_state)

## # A tibble: 11 × 3
## # Groups:   state [3]
##    state    candidate        avg_votes
##    <chr>    <chr>                <dbl>
##  1 Delaware John Carney         97634.
##  2 Delaware John Machurek        1090
##  3 Delaware Julianne Murray     63437.
##  4 Delaware Kathy DeMatteis      2050
##  5 Indiana  Donald Rainwater     3756.
##  6 Indiana  Eric Holcomb        18551.
##  7 Indiana  Woody Myers         10523.
##  8 Missouri Jerome Bauer          126.
##  9 Missouri Mike Parson         13699.
## 10 Missouri Nicole Galloway      7822.
## 11 Missouri Rik Combs             385.

avg_votes_by_party_state <- Book3 %>% group_by(state, party) %>%
summarise(avg_votes = mean(votes))

## `summarise()` has grouped output by 'state'. You can override using the
## `.groups` argument.

print(avg_votes_by_party_state)

## # A tibble: 11 × 3
## # Groups:   state [3]
##    state    party avg_votes
##    <chr>    <chr>     <dbl>
##  1 Delaware DEM      97634.
##  2 Delaware IPD       2050
##  3 Delaware LIB       1090
##  4 Delaware REP      63437.
##  5 Indiana  DEM      10523.
##  6 Indiana  LIB       3756.
##  7 Indiana  REP      18551.
##  8 Missouri DEM       7822.
##  9 Missouri GRN        126.
## 10 Missouri LIB        385.
## 11 Missouri REP      13699.
```

```r
percentage_votes_by_party_county <- Book3 %>% group_by(county, party) %>%
mutate(percentage_votes = votes / sum(votes) * 100)
print(percentage_votes_by_party_county)
```

```
## # A tibble: 599 × 7
## # Groups:   county, party [518]
##    state    county            candidate      party  votes won
percentage_votes
##    <chr>    <chr>             <chr>          <chr>  <dbl> <lgl>
<dbl>
##  1 Delaware Kent County       John Carney    DEM    44352 TRUE
100
##  2 Delaware Kent County       Julianne Murr… REP    39332 FALSE
100
##  3 Delaware Kent County       Kathy DeMatte… IPD     1115 FALSE
100
##  4 Delaware Kent County       John Machurek  LIB      616 FALSE
100
##  5 Delaware New Castle County John Carney    DEM   191678 TRUE
100
##  6 Delaware New Castle County Julianne Murr… REP    82545 FALSE
100
##  7 Delaware New Castle County Kathy DeMatte… IPD     3785 FALSE
100
##  8 Delaware New Castle County John Machurek  LIB     2031 FALSE
100
##  9 Delaware Sussex County     Julianne Murr… REP    68435 TRUE
100
## 10 Delaware Sussex County     John Carney    DEM    56873 FALSE
100
## # ℹ 589 more rows
```

```r
total_votes_by_candidate_county <-Book3 %>% group_by(county, candidate) %>%
summarise(total_votes = sum(votes))
```

```
## `summarise()` has grouped output by 'county'. You can override using the
## `.groups` argument.
```

```r
print(total_votes_by_candidate_county)
```

```
## # A tibble: 599 × 3
## # Groups:   county [146]
##    county       candidate        total_votes
##    <chr>        <chr>                  <dbl>
##  1 Adair County Jerome Bauer              44
##  2 Adair County Mike Parson             6597
##  3 Adair County Nicole Galloway         3546
##  4 Adair County Rik Combs                131
##  5 Adams County Donald Rainwater        2570
##  6 Adams County Eric Holcomb            9441
##  7 Adams County Woody Myers             2143
```

```
##  8 Allen County Donald Rainwater       16011
##  9 Allen County Eric Holcomb           98406
## 10 Allen County Woody Myers            53895
## # ℹ 589 more rows
```

```
total_votes_by_candidate_county <-Book3 %>% group_by(county, candidate) %>%
summarise(total_votes = sum(votes))
```

```
## `summarise()` has grouped output by 'county'. You can override using the
## `.groups` argument.
```

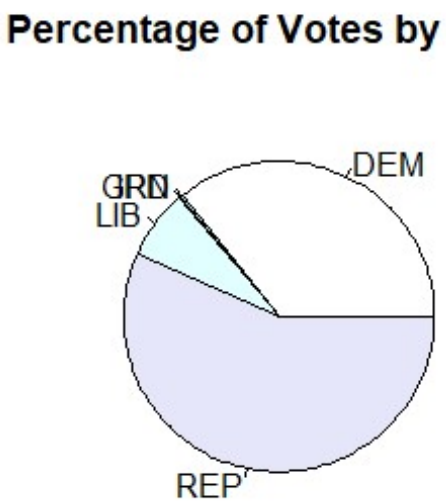
```
print(total_votes_by_candidate_county)
```

```
## # A tibble: 599 × 3
## # Groups:   county [146]
##    county       candidate        total_votes
##    <chr>        <chr>                  <dbl>
##  1 Adair County Jerome Bauer              44
##  2 Adair County Mike Parson             6597
##  3 Adair County Nicole Galloway         3546
##  4 Adair County Rik Combs                131
##  5 Adams County Donald Rainwater        2570
##  6 Adams County Eric Holcomb            9441
##  7 Adams County Woody Myers             2143
##  8 Allen County Donald Rainwater       16011
##  9 Allen County Eric Holcomb           98406
## 10 Allen County Woody Myers            53895
## # ℹ 589 more rows
```

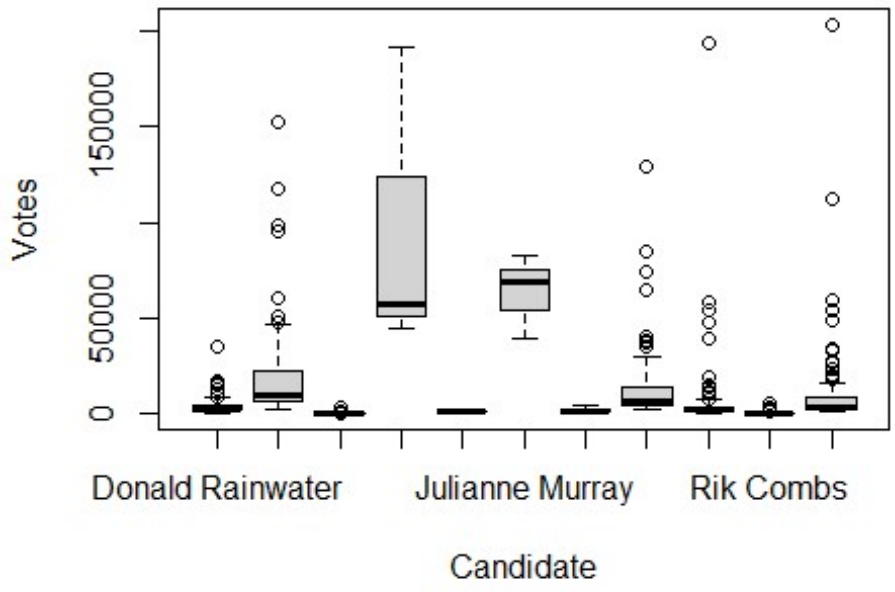
```
 length(unique(Book3$candidate))
```

```
## [1] 11
```

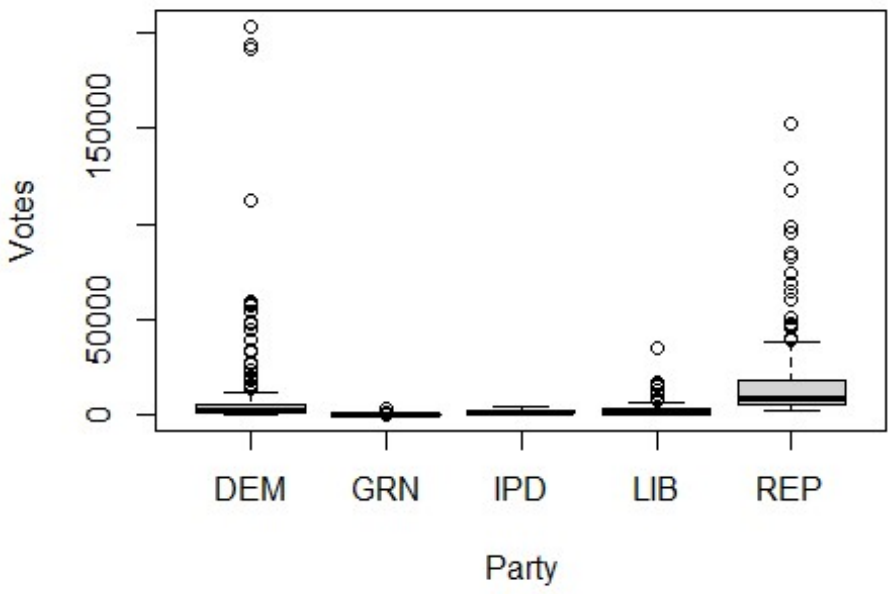
```
library(ggplot2)
```



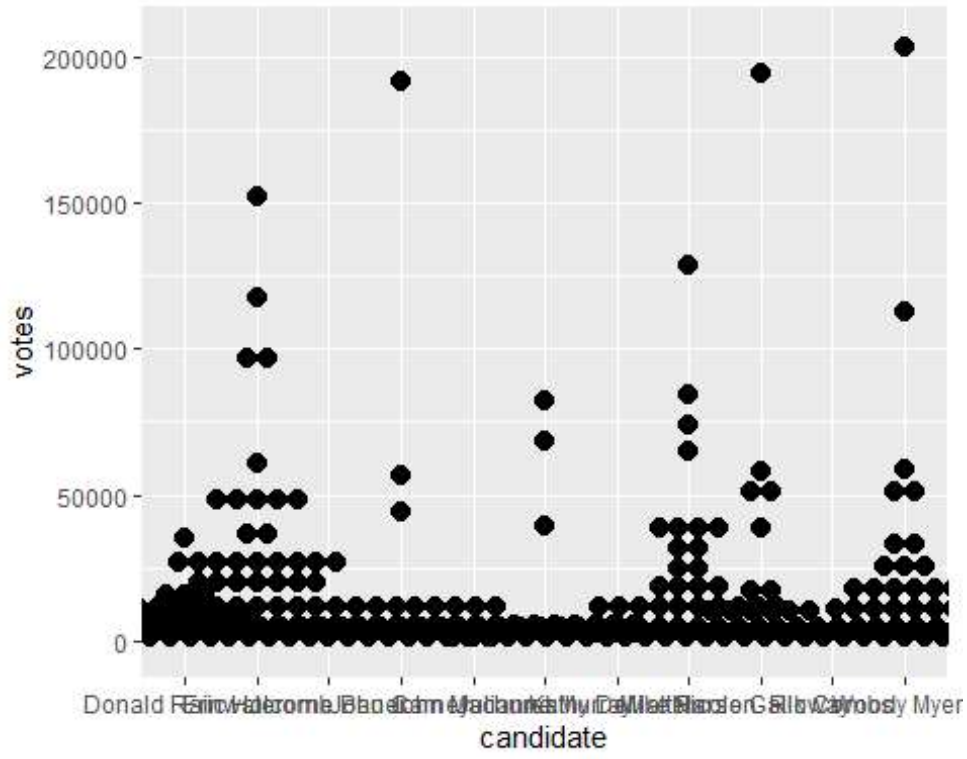

```
boxplot(votes ~ candidate, data = Book3, xlab = "Candidate", ylab = "Votes")
```



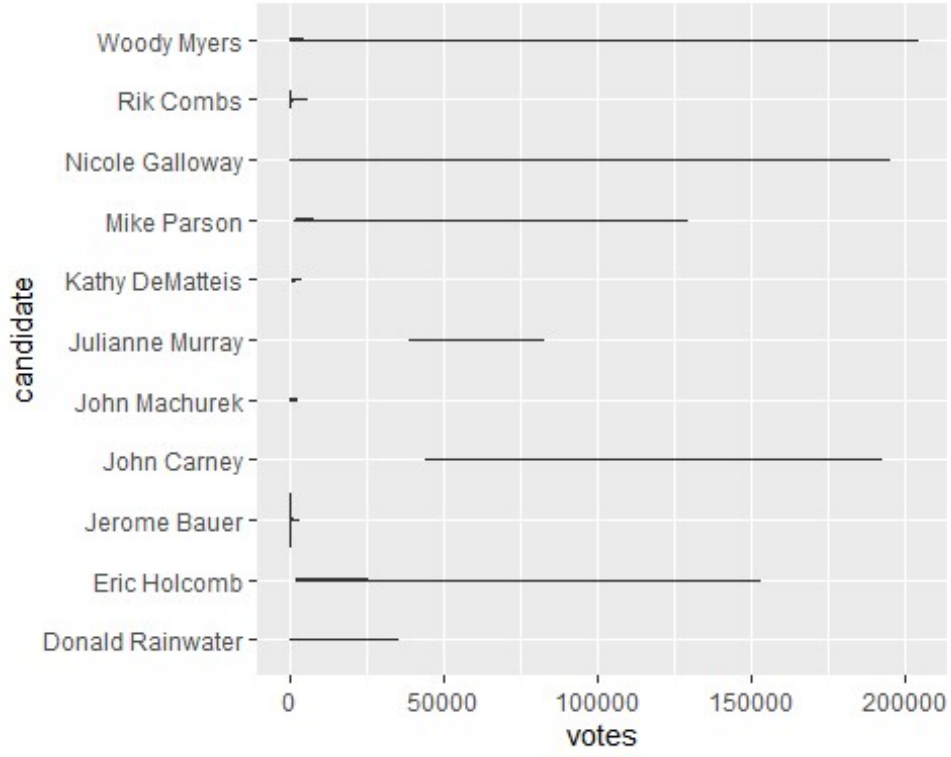

```
boxplot(votes ~ party, data = Book3, xlab = "Party", ylab = "Votes")
```



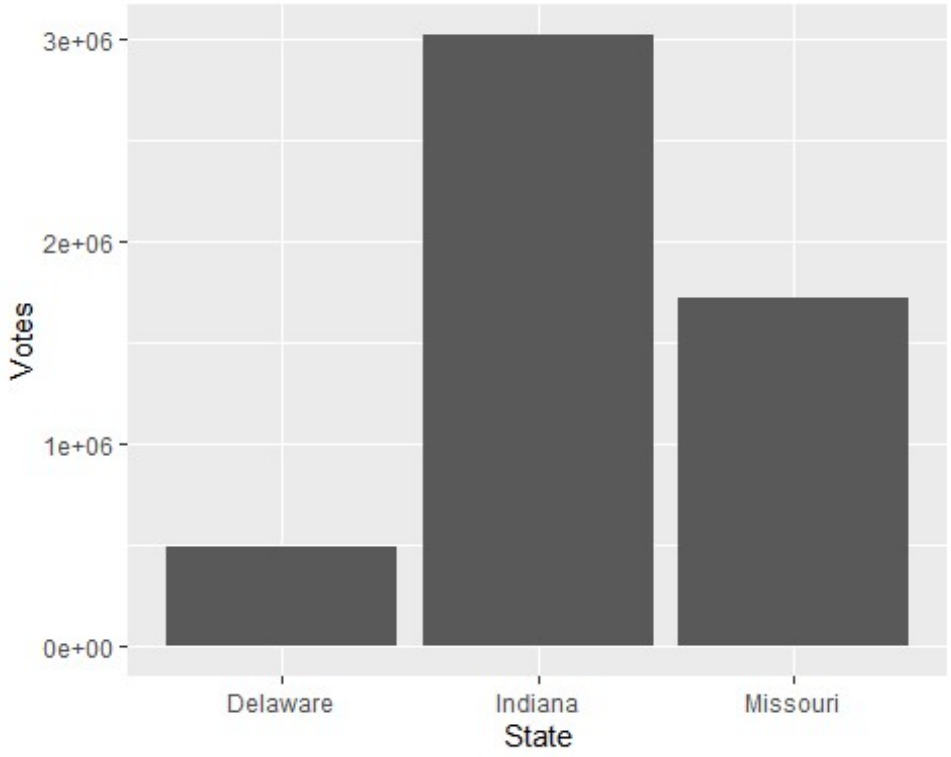

```
IQR(Book3$votes)

## [1] 6623

sd(Book3$votes)

## [1] 21095.05

library(ggplot2)


ggplot(Book3, aes(y = votes, x = candidate)) + geom_dotplot(binaxis = "y",
stackdir = "center")

## Bin width defaults to 1/30 of the range of the data. Pick better value
with
## `binwidth`.
```



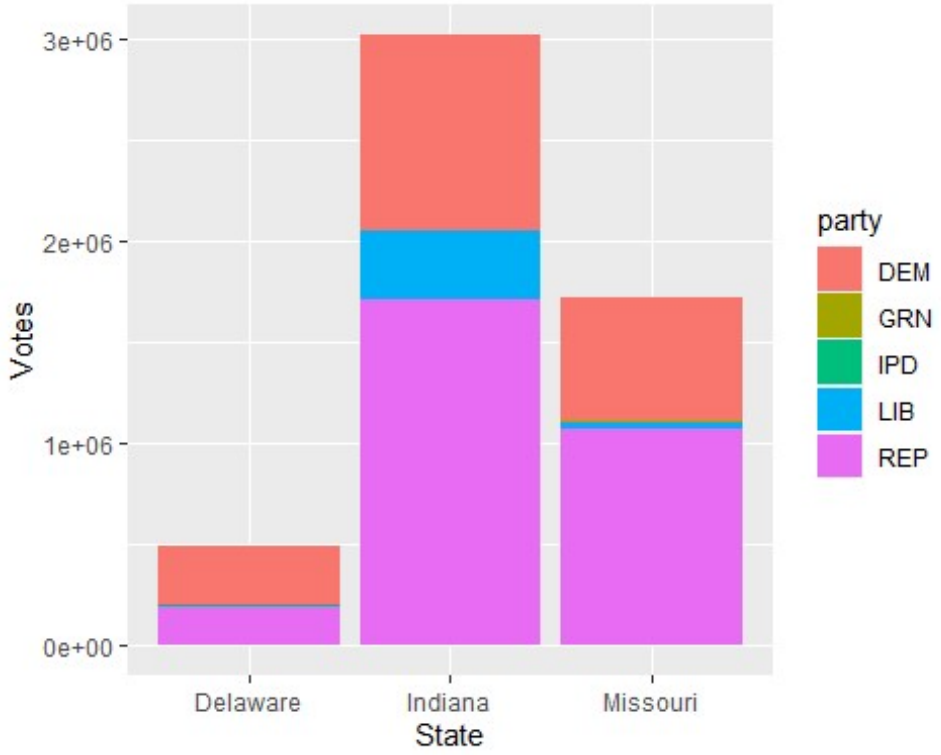

```
ggplot(Book3,aes(x=votes,y=candidate))+geom_violin()
```



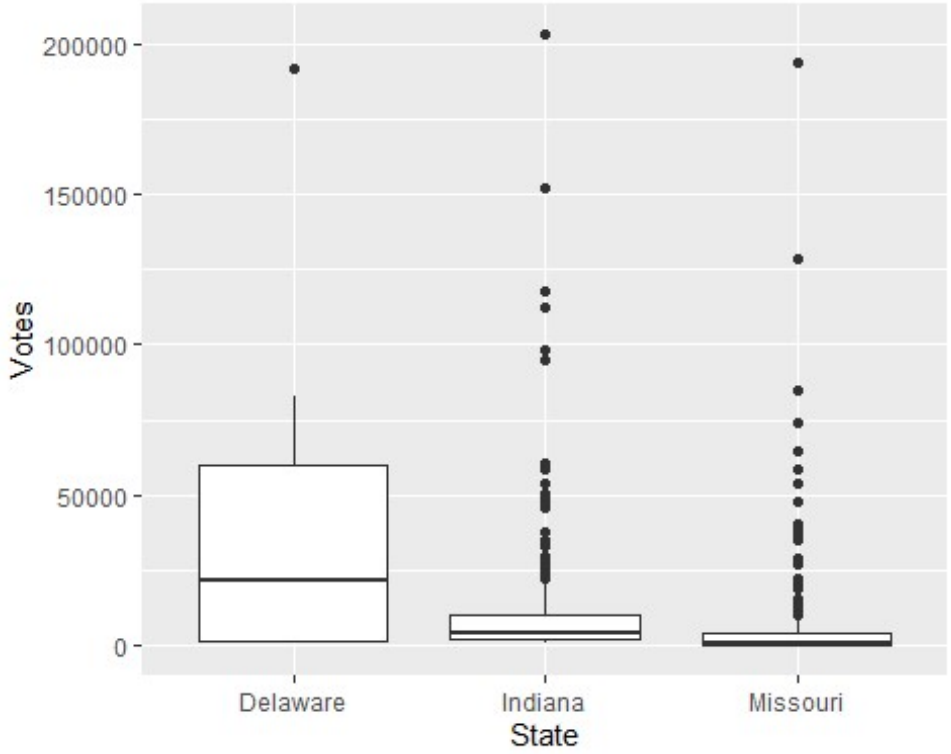
```
ggplot(Book3, aes(x = state, y = votes)) + geom_bar(stat = "identity") +
xlab("State") + ylab("Votes")
```



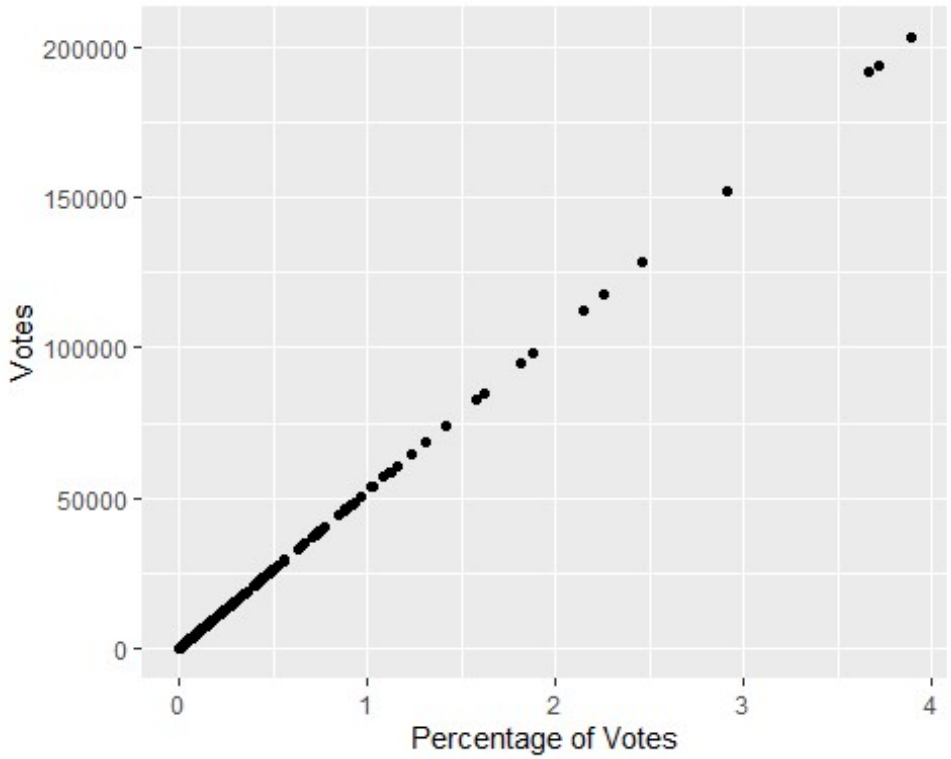

```
ggplot(Book3, aes(x = state, y = votes, fill = party)) + geom_bar(stat =
"identity") + xlab("State") + ylab("Votes") + theme(legend.position =
"right")
```



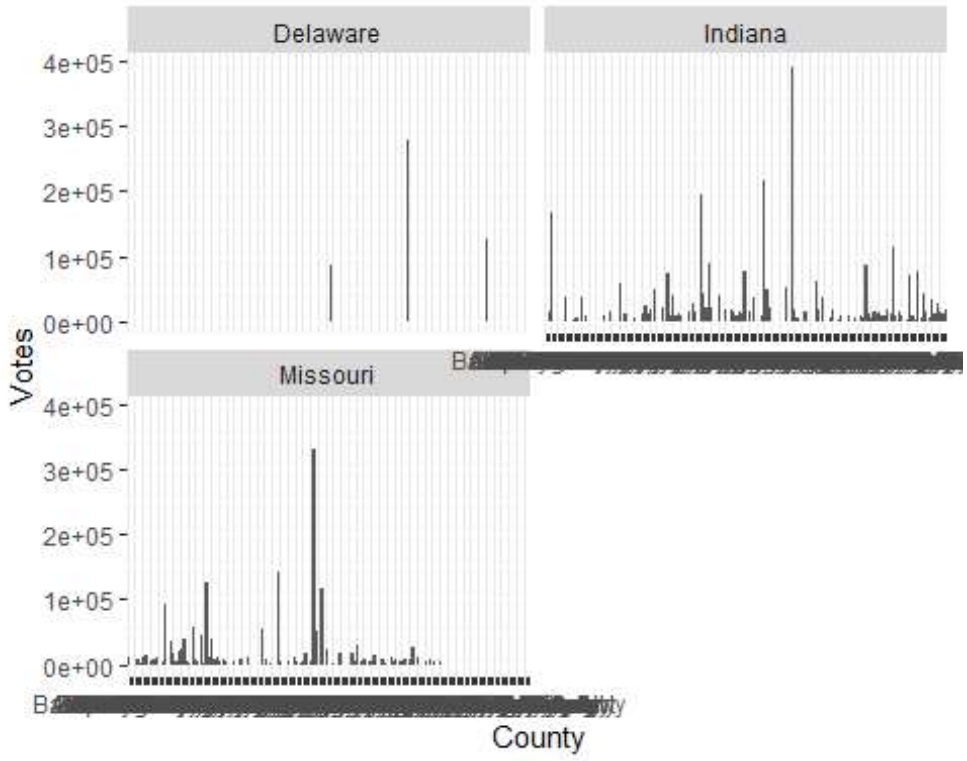

```
ggplot(Book3, aes(x = state, y = votes)) + geom_boxplot() + xlab("State") +
ylab("Votes")
```



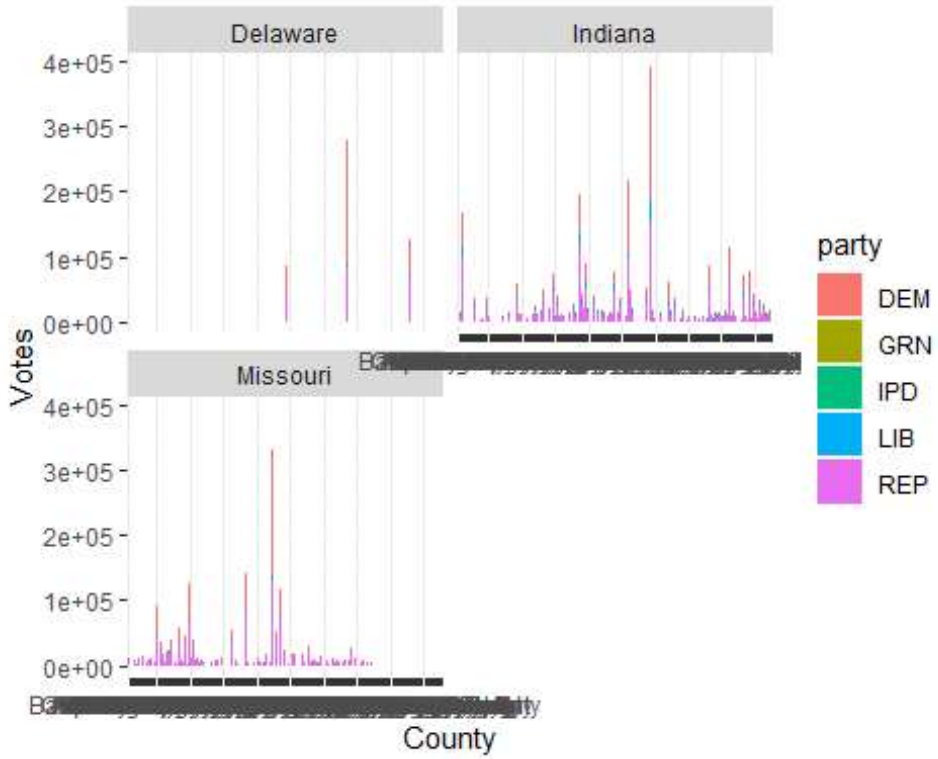
```
ggplot(Book3, aes(x = percentage_votes, y = votes)) + geom_point() +
xlab("Percentage of Votes") + ylab("Votes")
```



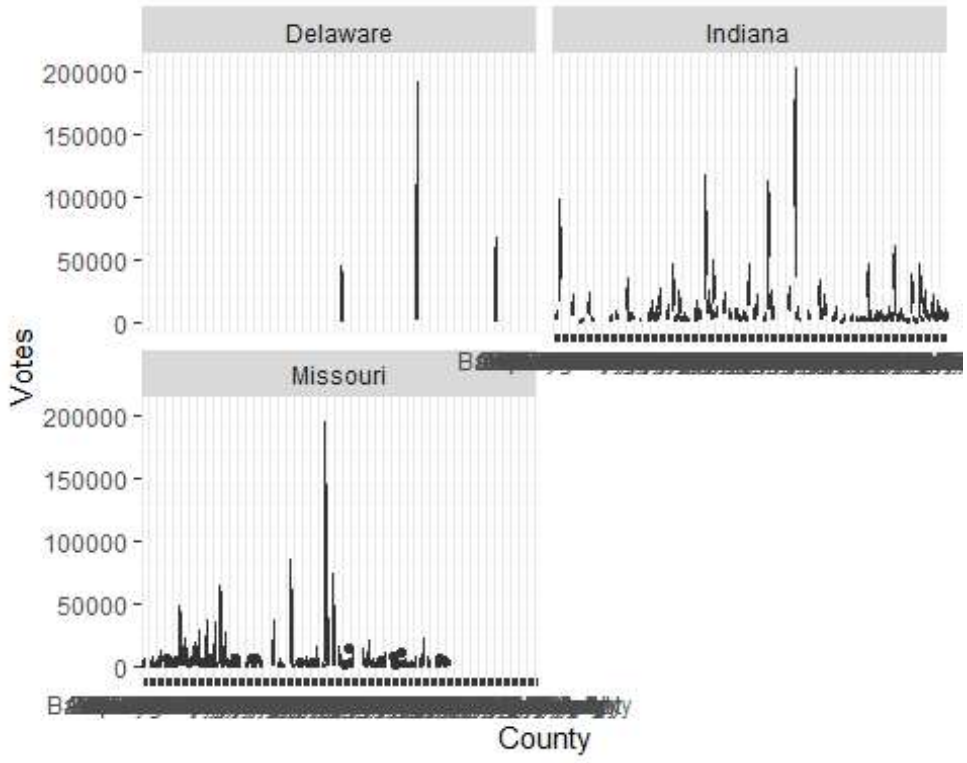

```
ggplot(Book3, aes(x = county, y = votes)) + geom_bar(stat = "identity") +
xlab("County") + ylab("Votes") + facet_wrap(~state, nrow = 2)
```



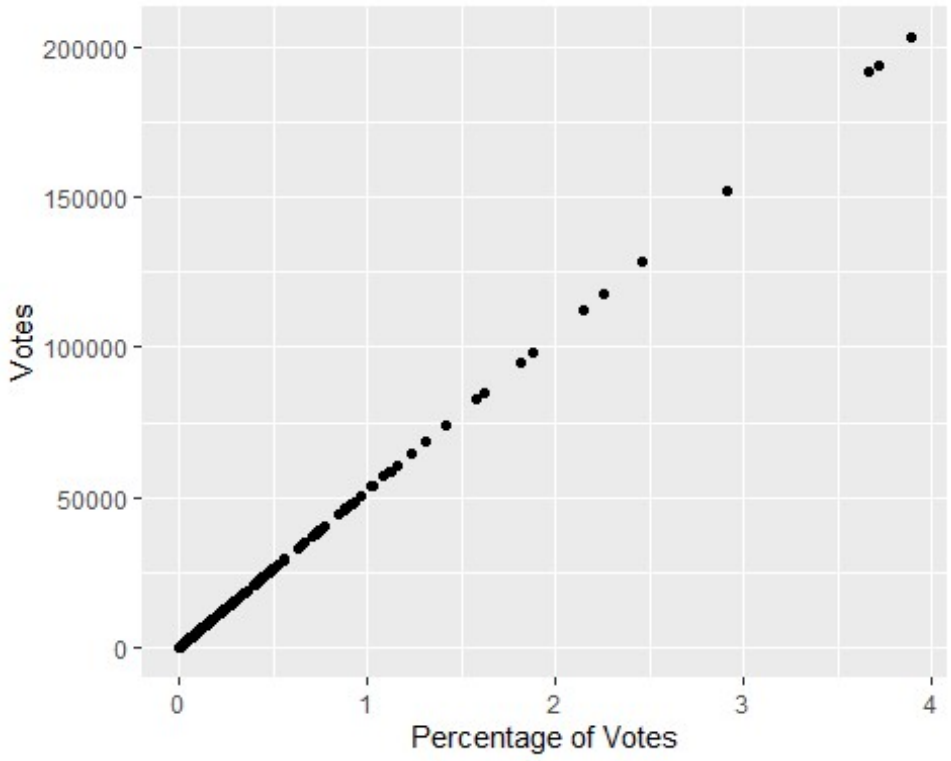

```
ggplot(Book3, aes(x = county, y = votes, fill = party)) + geom_bar(stat =
"identity") + xlab("County") + ylab("Votes") + facet_wrap(~state, nrow = 2) +
theme(legend.position = "right")
```



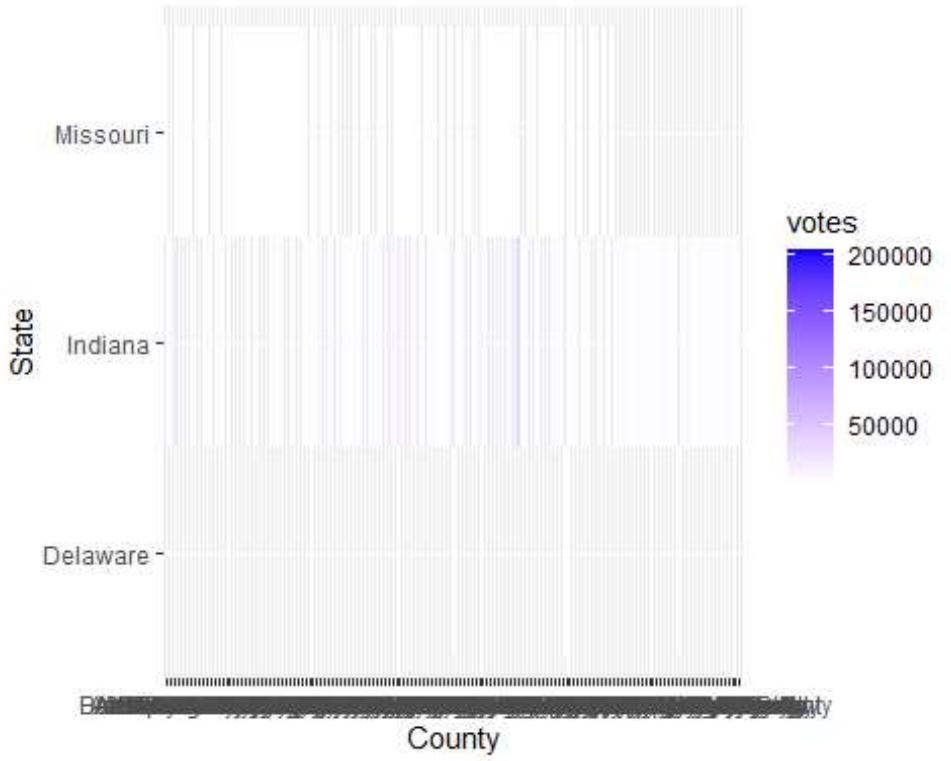
```
ggplot(Book3, aes(x = county, y = votes)) + geom_boxplot() + xlab("County") +
ylab("Votes") + facet_wrap(~state, nrow = 2)
```



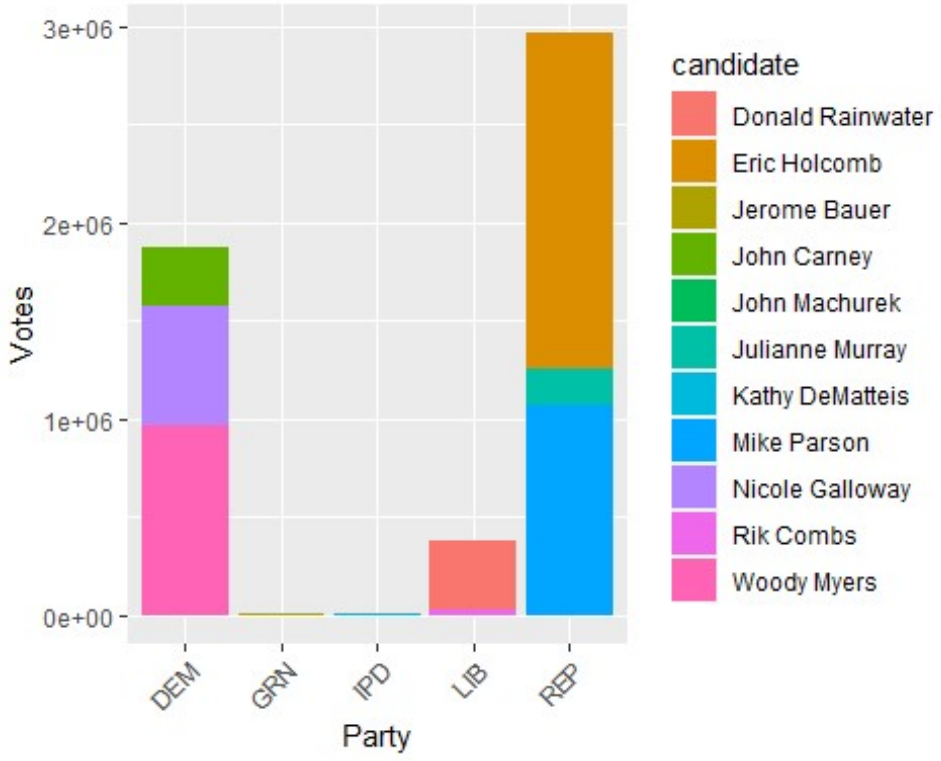

```
ggplot(Book3, aes(x = percentage_votes, y = votes)) + geom_point() +
xlab("Percentage of Votes") + ylab("Votes")
```



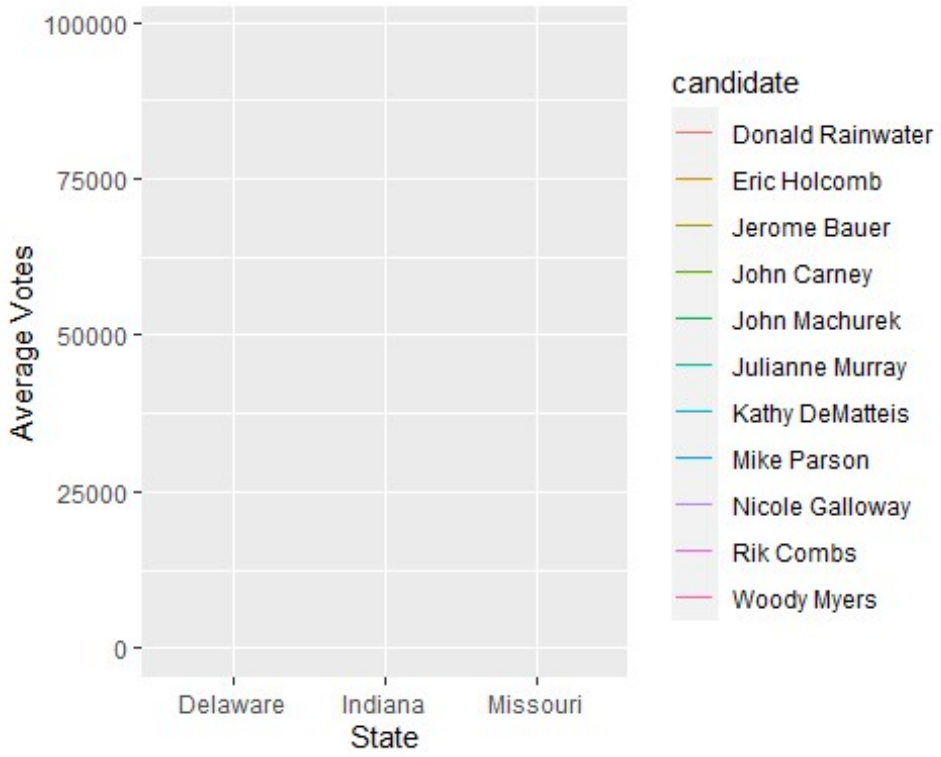

```
ggplot(Book3, aes(x = county, y = state, fill = votes)) + geom_tile() +
xlab("County") + ylab("State") + scale_fill_gradient(low = "white", high =
"blue")
```



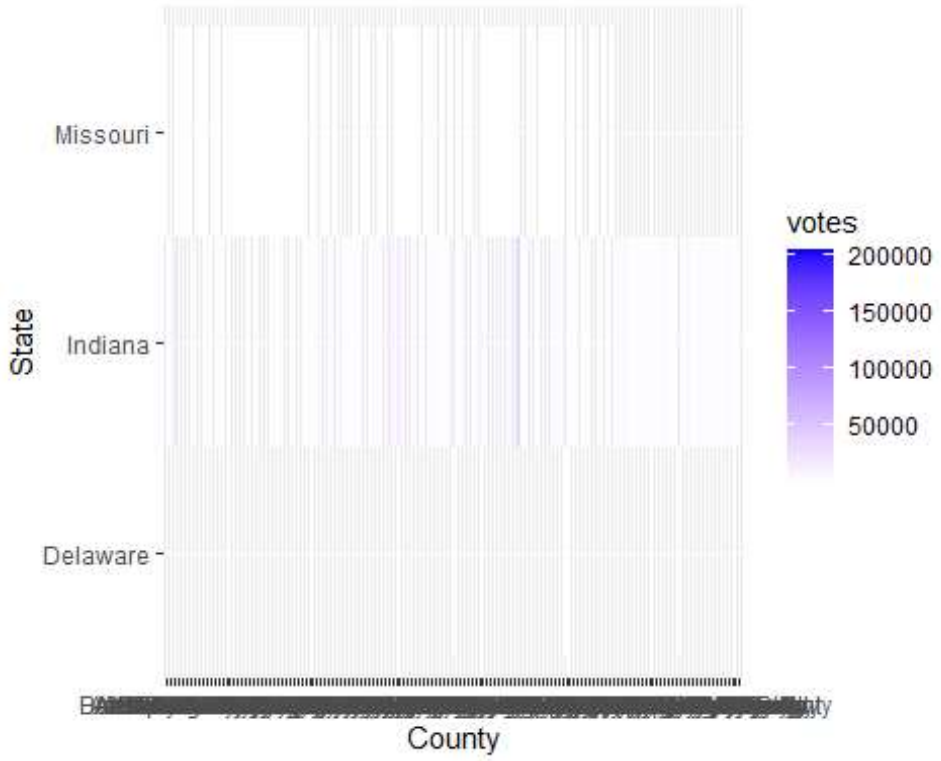
```
ggplot(Book3, aes(x = party, y = votes, fill = candidate)) + geom_bar(stat =
"identity") + xlab("Party") + ylab("Votes") + theme(axis.text.x =
element_text(angle = 45, hjust = 1))
```



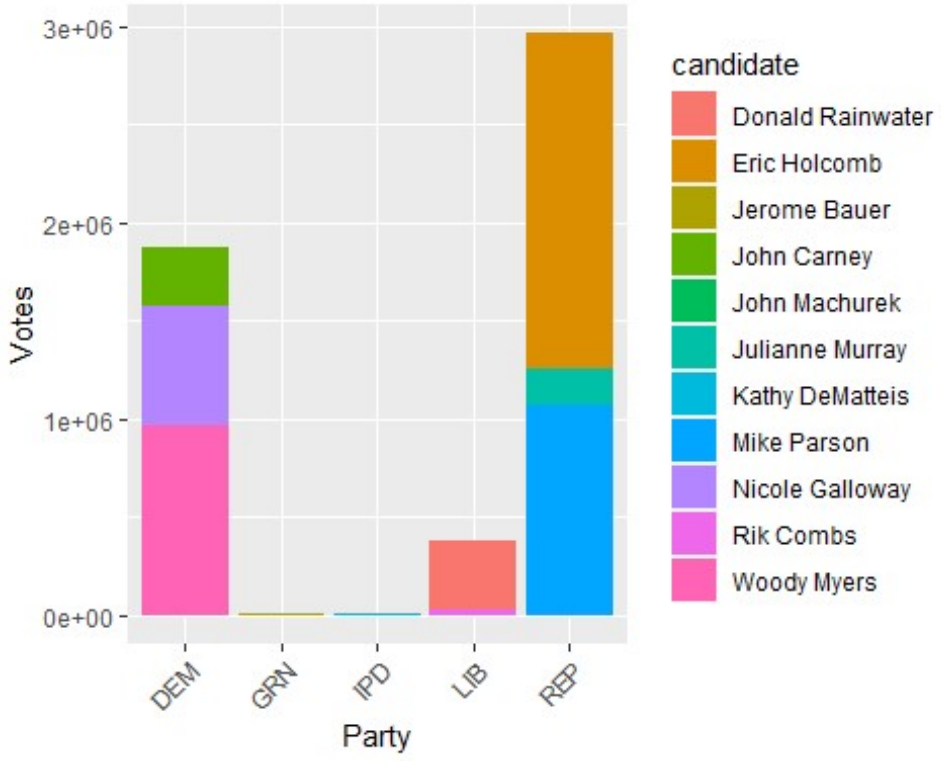

```
 avg_votes_by_state_candidate <- aggregate(votes ~ state + candidate, Book3,
mean)
 ggplot(avg_votes_by_state_candidate, aes(x = state, y = votes, group =
candidate, color = candidate)) + geom_line() + xlab("State") + ylab("Average
Votes")

## `geom_line()`: Each group consists of only one observation.
## ℹ Do you need to adjust the group aesthetic?
```



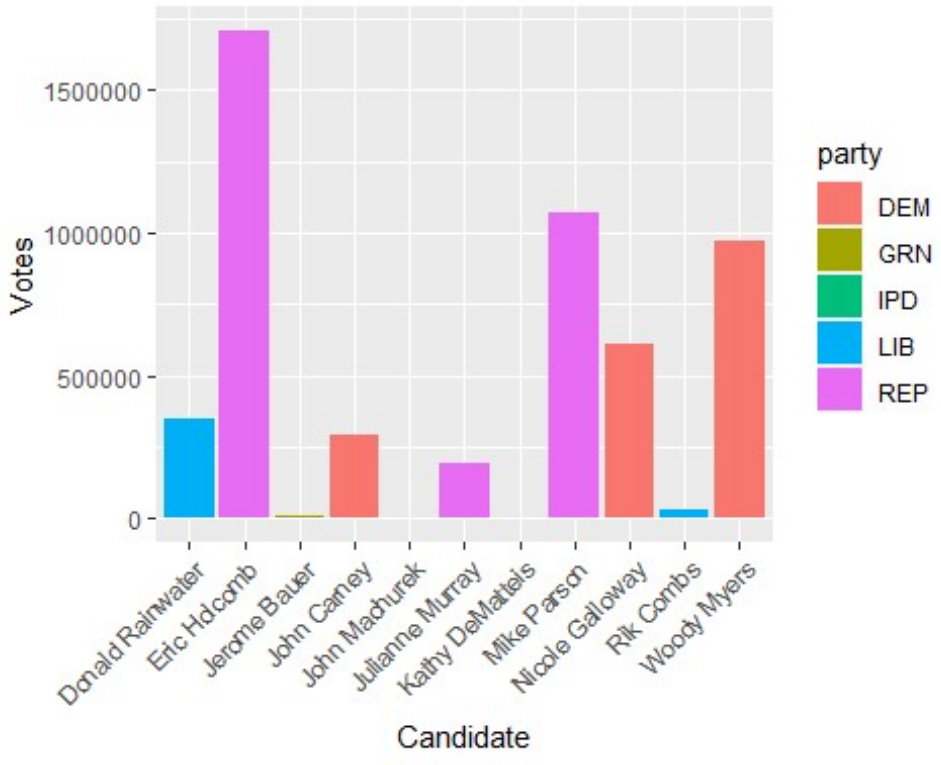

```
ggplot(Book3, aes(x = county, y = state, fill = votes)) + geom_tile() +
xlab("County") + ylab("State") + scale_fill_gradient(low = "white", high =
"blue")
```



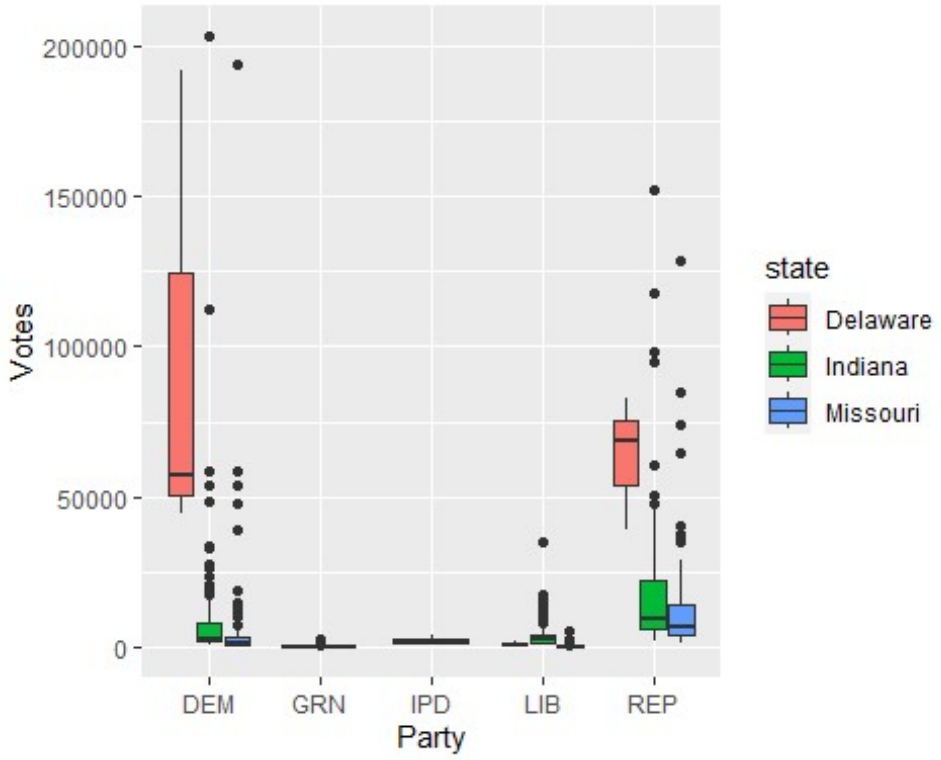
```
ggplot(Book3, aes(x = party, y = votes, fill = candidate)) + geom_bar(stat =
"identity") + xlab("Party") + ylab("Votes") + theme(axis.text.x =
element_text(angle = 45, hjust = 1))
```



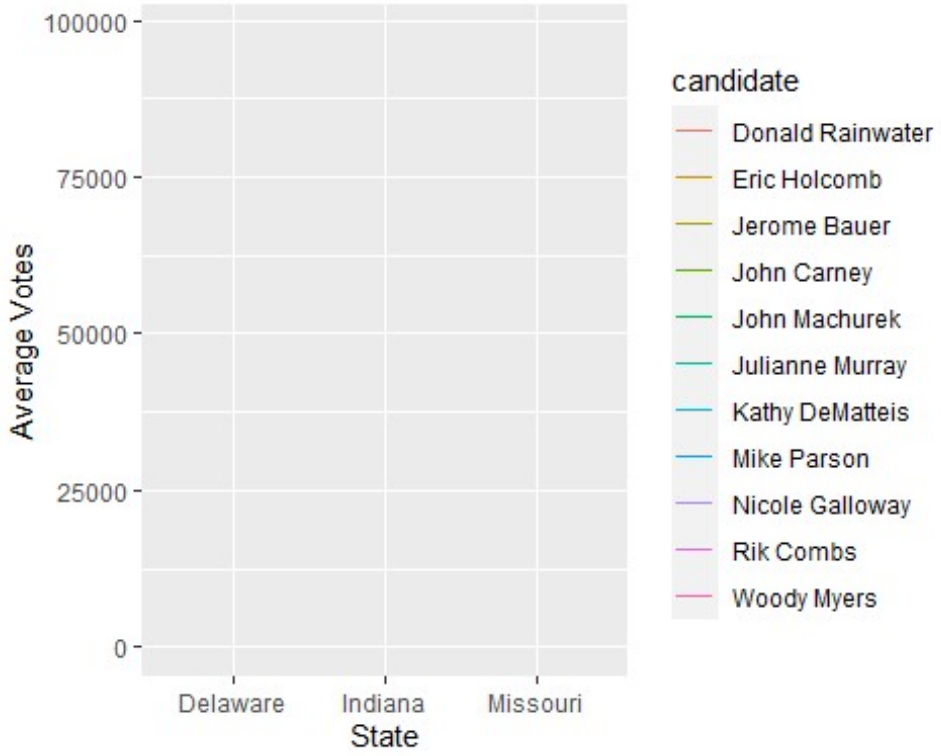

```
ggplot(Book3, aes(x = candidate, y = votes, fill = party)) + geom_bar(stat =
"identity") + xlab("Candidate") + ylab("Votes") + theme(axis.text.x =
element_text(angle = 45, hjust = 1))
```




```
ggplot(Book3, aes(x = party, y = votes, fill = state)) + geom_boxplot() +
xlab("Party") + ylab("Votes")
```



```
 avg_votes_by_state_candidate <- aggregate(votes ~ state + candidate, Book3,
mean)
 ggplot(avg_votes_by_state_candidate, aes(x = state, y = votes, group =
candidate, color = candidate)) + geom_line() + xlab("State") + ylab("Average
Votes")

## `geom_line()`: Each group consists of only one observation.
## ℹ Do you need to adjust the group aesthetic?
```




```
 Book3$cumulative_votes_state <- ave(Book3$votes, Book3$state, FUN = cumsum)
ggplot(Book3, aes(x = state, y = cumulative_votes_state, fill = state)) +
geom_area() + xlab("State") + ylab("Cumulative Votes") +
theme(legend.position = "none")
```


```
 total_votes_by_state_party <- aggregate(votes ~ state + party, Book3, sum)

 print(total_votes_by_state_party)

##       state party   votes
## 1  Delaware   DEM  292903
## 2   Indiana   DEM  968092
## 3  Missouri   DEM  610118
## 4  Missouri   GRN    9684
## 5  Delaware   IPD    6150
## 6  Delaware   LIB    3270
## 7   Indiana   LIB  345567
## 8  Missouri   LIB   30002
## 9  Delaware   REP  190312
## 10  Indiana   REP 1706724
## 11 Missouri   REP 1068545

 Book3$candidate[which.max(Book3$votes)]

## [1] "Woody Myers"

 table(Book3$party)

##
## DEM GRN IPD LIB REP
## 173  77   3 173 173

 aggregate(votes ~ state, Book3, sum)
```

```
##      state   votes
## 1 Delaware  492635
## 2  Indiana 3020383
## 3 Missouri 1718349
```

```
 Book3$votes / sum(Book3$votes) * 100
```

```
##   [1] 8.478090e-01 7.518494e-01 2.131374e-02 1.177512e-02 3.664014e+00
##   [6] 1.577886e+00 7.235203e-02 3.882350e-02 1.308167e+00 1.087154e+00
##  [11] 2.389433e-02 1.190893e-02 1.804691e-01 4.912674e-02 4.096444e-02
##  [16] 1.881076e+00 1.030228e+00 3.060577e-01 4.197564e-01 1.722877e-01
##  [21] 9.681982e-02 5.381003e-02 1.282648e-02 1.173689e-02 6.371184e-02
##  [26] 1.944043e-02 1.783473e-02 4.537437e-01 1.846745e-01 9.737417e-02
##  [31] 9.467889e-02 4.551392e-02 3.102440e-02 1.161647e-01 3.563122e-02
##  [36] 2.999216e-02 1.688469e-01 6.447646e-02 5.619946e-02 6.627140e-01
##  [41] 3.646657e-01 7.034490e-02 1.560586e-01 4.285687e-02 3.402552e-02
##  [46] 1.523694e-01 4.805627e-02 4.599180e-02 5.856978e-02 2.374140e-02
##  [51] 1.057085e-02 1.589068e-01 3.853677e-02 2.926577e-02 3.508261e-01
##  [56] 8.714739e-02 4.555215e-02 1.512224e-01 5.327479e-02 2.985835e-02
##  [61] 2.493039e-01 6.294722e-02 6.036663e-02 5.128487e-01 2.989276e-01
##  [66] 1.047910e-01 2.753200e-01 8.766351e-02 4.968109e-02 8.835740e-01
##  [71] 3.920964e-01 1.469979e-01 1.325275e-01 3.263009e-02 2.702926e-02
##  [76] 4.773513e-01 2.857379e-01 3.190371e-02 9.919014e-02 3.083324e-02
##  [81] 2.198278e-02 1.650238e-01 3.230513e-02 2.865408e-02 1.039117e-01
##  [86] 3.983662e-02 3.064209e-02 2.218732e-01 5.715523e-02 2.930400e-02
##  [91] 3.220382e-01 1.118063e-01 8.267438e-02 1.831070e-01 4.977666e-02
##  [96] 4.799893e-02 2.250827e+00 1.122345e+00 3.272758e-01 4.902543e-01
## [101] 1.648517e-01 1.632078e-01 2.692031e-01 8.628720e-02 3.062297e-02
## [106] 9.690966e-01 4.430773e-01 2.723189e-01 2.387712e-01 8.491088e-02
## [111] 7.890863e-02 4.656336e-01 1.886887e-01 1.207524e-01 2.247787e-01
## [116] 6.055779e-02 5.359976e-02 2.239185e-01 8.005556e-02 6.181941e-02
## [121] 1.983803e-01 5.551130e-02 3.993220e-02 9.991652e-02 3.704577e-02
## [126] 2.370317e-02 1.734919e-01 7.093748e-02 3.345206e-02 1.442644e-01
## [131] 5.115298e-02 3.689284e-02 9.073537e-01 3.370056e-01 2.327690e-01
## [136] 2.142843e-01 5.369533e-02 3.480926e-02 4.402100e-01 1.308453e-01
## [141] 1.105638e-01 1.437483e-01 3.230513e-02 2.609261e-02 2.147660e+00
## [146] 1.812930e+00 1.919001e-01 4.994679e-01 3.466207e-01 8.326695e-02
## [151] 2.555928e-01 7.418711e-02 6.912151e-02 5.626445e-01 2.724718e-01
## [156] 1.551793e-01 3.889519e+00 2.913292e+00 6.685442e-01 2.512728e-01
## [161] 7.693974e-02 5.035013e-02 6.178117e-02 2.249890e-02 1.355286e-02
## [166] 1.760152e-01 5.187937e-02 4.662261e-02 6.314411e-01 4.703360e-01
## [171] 9.337903e-02 2.023754e-01 7.235203e-02 5.375268e-02 4.236178e-01
## [176] 1.564409e-01 1.070848e-01 8.330519e-02 2.194455e-02 2.007124e-02
## [181] 2.389051e-01 6.409415e-02 6.187675e-02 4.224517e-02 1.225301e-02
## [186] 6.403680e-03 1.068746e-01 3.731338e-02 2.324440e-02 1.119593e-01
## [191] 4.084974e-02 3.540184e-02 9.110429e-02 2.154313e-02 2.062558e-02
## [196] 1.055747e-01 4.813273e-02 1.160309e-02 8.026583e-02 2.087409e-02
## [201] 1.624814e-02 8.792348e-01 6.383991e-01 1.363124e-01 1.844260e-01
## [206] 5.300718e-02 1.546441e-02 7.405330e-02 1.947866e-02 1.672603e-02
## [211] 1.792075e-01 8.160391e-02 5.497607e-02 1.414353e-01 3.561211e-02
```

```
## [216] 3.310798e-02 1.819410e-01 4.698581e-02 4.264660e-02 9.460242e-02
## [221] 3.280213e-02 2.322529e-02 1.160232e+00 9.292026e-01 1.029750e-01
## [226] 1.158779e-01 4.386999e-02 3.083324e-02 2.345276e-01 7.835428e-02
## [231] 6.688500e-02 1.390076e-01 4.704315e-02 1.770092e-02 1.286471e-01
## [236] 3.872793e-02 2.727777e-02 2.180501e-01 5.618034e-02 3.324179e-02
## [241] 1.148648e-01 2.962897e-02 2.593968e-02 5.440261e-02 1.571291e-02
## [246] 7.875571e-03 7.259862e-01 5.004237e-01 1.348978e-01 9.880783e-02
## [251] 3.263009e-02 2.293856e-02 4.841947e-02 1.152662e-02 6.174294e-03
## [256] 8.886779e-01 5.207434e-01 6.992436e-02 9.439215e-02 2.926577e-02
## [261] 1.752888e-02 4.954154e-01 2.675974e-01 6.535577e-02 1.741610e-01
## [266] 5.694496e-02 4.627853e-02 5.751843e-02 1.485271e-02 1.284559e-02
## [271] 4.339783e-01 1.734537e-01 3.339471e-02 1.513753e-01 4.603003e-02
## [276] 3.337560e-02 3.233189e-01 1.440159e-01 6.063425e-02 1.791310e-01
## [281] 5.184113e-02 3.760011e-02 1.401928e-01 3.781038e-02 3.490483e-02
## [286] 2.202484e-01 6.099744e-02 5.233814e-02 1.261047e-01 6.778343e-02
## [291] 2.504126e-03 8.410804e-04 1.375358e-01 4.503603e-02 2.676165e-03
## [296] 8.028494e-04 4.149967e-02 1.018854e-02 9.748886e-04 2.867319e-04
## [301] 1.460995e-01 5.090448e-02 4.434787e-03 1.338082e-03 2.361907e-01
## [306] 5.430703e-02 4.625942e-03 1.682161e-03 9.775648e-02 1.559822e-02
## [311] 1.318967e-03 6.308103e-04 1.225301e-01 3.368145e-02 2.561472e-03
## [316] 9.748886e-04 1.532487e-01 4.165259e-02 3.459899e-03 1.070466e-03
## [321] 9.678159e-02 1.477625e-02 1.739507e-03 4.014247e-04 9.186127e-01
## [326] 7.737557e-01 3.270656e-02 8.793113e-03 4.233501e-01 2.528020e-01
## [331] 1.374402e-02 5.791985e-03 2.740584e-01 6.401768e-02 4.912674e-03
## [336] 1.166043e-03 6.887301e-02 1.750976e-02 1.873315e-03 6.308103e-04
## [341] 2.857762e-01 1.072569e-01 8.812228e-03 1.510122e-03 3.600780e-01
## [346] 1.043895e-01 6.231641e-03 1.988008e-03 5.567761e-01 1.963540e-01
## [351] 1.204274e-02 2.523241e-03 7.017286e-02 1.471891e-02 8.410804e-04
## [356] 2.485010e-04 4.610650e-02 7.665300e-03 7.263876e-04 1.911546e-04
## [361] 7.077500e-01 3.587972e-01 1.965070e-02 6.097833e-03 1.103153e-01
## [366] 2.110347e-02 2.370317e-03 5.734639e-04 5.925793e-02 1.739507e-02
## [371] 1.089581e-03 9.557731e-05 6.657342e-01 2.076513e-01 1.536883e-02
## [376] 3.173167e-03 5.098094e-02 1.238682e-02 6.881567e-04 4.014247e-04
## [381] 1.236426e+00 1.112979e+00 4.371706e-02 1.309409e-02 1.437100e-01
## [386] 5.736550e-02 4.071594e-03 1.758623e-03 5.139383e-01 2.241479e-01
## [391] 1.521591e-02 3.154051e-03 1.189746e-01 4.132763e-02 4.473018e-03
## [396] 8.793113e-04 1.620991e-01 4.262748e-02 3.364321e-03 8.793113e-04
## [401] 6.399857e-02 1.276913e-02 1.280736e-03 4.205402e-04 1.270414e-01
## [406] 2.473541e-02 2.179163e-03 7.837340e-04 5.728904e-02 1.496741e-02
## [411] 1.471891e-03 4.396556e-04 7.193149e-02 1.777738e-02 1.777738e-03
## [416] 5.734639e-04 1.102580e-01 2.181074e-02 3.249629e-03 6.499257e-04
## [421] 1.103536e-01 1.999477e-02 2.255625e-03 6.499257e-04 1.506298e-01
## [426] 4.360237e-02 2.255625e-03 5.734639e-04 7.098718e-01 2.859100e-01
## [431] 1.770092e-02 6.709527e-03 1.183629e-01 2.938047e-02 2.140932e-03
## [436] 5.543484e-04 4.853416e-02 1.206186e-02 8.028494e-04 1.529237e-04
## [441] 1.616824e+00 1.023040e+00 5.457465e-02 1.276913e-02 6.761139e-02
## [446] 1.569379e-02 1.013120e-03 2.867319e-04 6.000344e-02 1.150751e-02
## [451] 8.984267e-04 2.293856e-04 1.515474e-01 4.920320e-02 2.962897e-03
## [456] 1.089581e-03 7.716912e-02 1.775826e-02 1.261621e-03 5.925793e-04
## [461] 3.685461e-02 6.900682e-03 6.499257e-04 1.911546e-04 6.738201e-02
```

```
## [466] 2.651315e-02 2.255625e-03 6.881567e-04 2.857188e-01 6.136063e-02
## [471] 5.486138e-03 1.605699e-03 6.243110e-02 2.112259e-02 2.217394e-03
## [476] 8.410804e-04 3.713618e+00 2.464710e+00 1.058997e-01 4.981490e-02
## [481] 7.209206e-01 2.524006e-01 1.827438e-02 5.467022e-03 1.413436e+00
## [486] 7.429416e-01 4.023805e-02 1.492918e-02 2.928680e-01 1.318011e-01
## [491] 1.229124e-02 3.211398e-03 2.897904e-02 5.619946e-03 3.823093e-04
## [496] 7.646185e-05 2.615186e-01 5.251018e-02 4.970020e-03 1.567468e-03
## [501] 2.339350e-01 8.508292e-02 5.237637e-03 1.338082e-03 2.709808e-01
## [506] 6.088275e-02 6.633066e-03 2.007124e-03 6.954205e-02 1.682161e-02
## [511] 5.925793e-04 3.631938e-04 4.016923e-01 1.360256e-01 1.154574e-02
## [516] 3.268744e-03 8.171860e-02 2.500303e-02 1.586583e-03 4.778866e-04
## [521] 1.005091e-01 2.546180e-02 1.701276e-03 5.734639e-04 1.400208e-01
## [526] 2.748804e-02 3.173167e-03 7.263876e-04 1.165279e-01 3.010685e-02
## [531] 1.930662e-03 4.778866e-04 8.338165e-02 2.152401e-02 2.752627e-03
## [536] 4.970020e-04 7.407242e-02 1.557910e-02 1.357198e-03 2.102701e-04
## [541] 1.927221e-01 5.763312e-02 2.752627e-03 7.837340e-04 2.930400e-02
## [546] 3.899554e-03 4.014247e-04 1.338082e-04 1.952262e-01 3.608999e-02
## [551] 3.154051e-03 1.146928e-03 6.730554e-02 2.114170e-02 8.601958e-04
## [556] 4.396556e-04 1.105638e-01 2.364583e-02 2.160047e-03 3.440783e-04
## [561] 6.636889e-02 1.720392e-02 9.748886e-04 2.867319e-04 8.441388e-02
## [566] 2.368406e-02 1.338082e-03 4.587711e-04 1.409192e-01 3.593707e-02
## [571] 2.657049e-03 1.242505e-03 1.020383e-01 3.293594e-02 1.701276e-03
## [576] 4.970020e-04 4.211328e-01 1.088052e-01 8.735766e-03 1.988008e-03
## [581] 1.318967e-01 5.273956e-02 2.790858e-03 7.646185e-04 7.206529e-02
## [586] 1.605699e-02 1.146928e-03 3.440783e-04 1.244799e-01 1.772003e-02
## [591] 1.395429e-03 4.205402e-04 7.575458e-02 1.469979e-02 1.166043e-03
## [596] 4.778866e-04 7.703531e-02 2.848204e-02 1.070466e-03
```

```
 mean(Book3$votes)
```

```
## [1] 8733.501
```

```
 sum(Book3$votes)
```

```
## [1] 5231367
```

```
 Book3[order(Book3$votes, decreasing = TRUE), ]
```

```
## # A tibble: 599 × 8
##    state    county            candidate      party  votes won
percentage_votes
##    <chr>    <chr>             <chr>          <chr>  <dbl> <lgl>
<dbl>
##  1 Indiana  Marion County     Woody Myers    DEM   203475 TRUE
3.89
##  2 Missouri Jackson County    Nicole Gallow… DEM   194273 TRUE
3.71
##  3 Delaware New Castle County John Carney    DEM   191678 TRUE
3.66
##  4 Indiana  Marion County     Eric Holcomb   REP   152405 FALSE
2.91
```

```
##  5 Missouri Jackson County    Mike Parson    REP   128938 FALSE
2.46
##  6 Indiana  Hamilton County   Eric Holcomb   REP   117749 TRUE
2.25
##  7 Indiana  Lake County       Woody Myers    DEM   112352 TRUE
2.15
##  8 Indiana  Allen County      Eric Holcomb   REP    98406 TRUE
1.88
##  9 Indiana  Lake County       Eric Holcomb   REP    94841 FALSE
1.81
## 10 Missouri Greene County     Mike Parson    REP    84582 TRUE
1.62
## # ℹ 589 more rows
## # ℹ 1 more variable: cumulative_votes_state <dbl>
```

```
 votes_threshold <- 10000
 filtered_dataset <- Book3[Book3$votes > votes_threshold, ]
 print(votes_threshold)
```

```
## [1] 10000
```

```
 specific_candidate <- "Candidate Name"
filtered_dataset <- Book3[Book3$candidate == specific_candidate, ]
```