# Data 608 Story 1

Keith DeNivo

Given the Dataset on Infrastructure Investment and Jobs Act funding by State

we must address the following questions

### Questions

1. Is the allocation equitable based on the population of each of the States and Territories, or is bias apparent?
2. Does the allocation favor the political interests of the Biden administration?

### Story start

The main issue is whether the money allocated through the Infrastructure Investment and Jobs Act Funding Act was distributed fairly among the states and was there any bias? This may open up an understanding of how government operates under the political parties and whether they are to be trusted or challenged. If they need to be challenged then democracy must take precedence to elect the officials that will provide the fairness that Americans' deserve.

### Libraries Loaded

```
library(readxl)
library(httr)
```

```
Warning: package 'httr' was built under R version 4.4.3
```

```
library(devtools)
```

```
Loading required package: usethis
```

```
library(RCurl)
library(plyr)
library(dplyr)
```

```

Attaching package: 'dplyr'

The following objects are masked from 'package:plyr':

    arrange, count, desc, failwith, id, mutate, rename, summarise,
    summarize

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union
```

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
v forcats   1.0.0     v readr     2.1.5
v ggplot2   3.5.1     v stringr   1.5.1
v lubridate 1.9.3     v tibble    3.2.1
v purrr     1.0.2     v tidyr     1.3.1

-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::arrange()   masks plyr::arrange()
x purrr::compact()   masks plyr::compact()
x tidyr::complete()  masks RCurl::complete()
x dplyr::count()     masks plyr::count()
x dplyr::desc()      masks plyr::desc()
x dplyr::failwith()  masks plyr::failwith()
x dplyr::filter()    masks stats::filter()
x dplyr::id()        masks plyr::id()
x dplyr::lag()       masks stats::lag()
```

```
x dplyr::mutate()    masks plyr::mutate()
x dplyr::rename()    masks plyr::rename()
x dplyr::summarise() masks plyr::summarise()
x dplyr::summarize() masks plyr::summarize()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becor
```

```
library(DescTools)
library(ggpubr)
```

```
Attaching package: 'ggpubr'

The following object is masked from 'package:plyr':

    mutate
```

```
library(openintro)
```

```
Loading required package: airports
Loading required package: cherryblossom
Loading required package: usdata

Attaching package: 'openintro'

The following object is masked from 'package:DescTools':

    cards
```

```
library(readr)
library(rvest)
```

```
Warning: package 'rvest' was built under R version 4.4.3

Attaching package: 'rvest'

The following object is masked from 'package:readr':

    guess_encoding
```

```
library(fpp3)
```

```
Registered S3 method overwritten by 'tsibble':
  method               from
  as_tibble.grouped_df dplyr
-- Attaching packages --------------------------------------- fpp3 1.0.1 --
v tsibble     1.1.5     v feasts      0.4.1
v tsibbledata 0.4.1     v fable       0.4.0

Warning: package 'feasts' was built under R version 4.4.3

-- Conflicts ------------------------------------------------ fpp3_conflicts --
x dplyr::arrange()     masks plyr::arrange()
x purrr::compact()     masks plyr::compact()
x tidyr::complete()    masks RCurl::complete()
x dplyr::count()       masks plyr::count()
x lubridate::date()    masks base::date()
x dplyr::desc()        masks plyr::desc()
x dplyr::failwith()    masks plyr::failwith()
x dplyr::filter()      masks stats::filter()
x dplyr::id()          masks plyr::id()
x tsibble::intersect() masks base::intersect()
x tsibble::interval()  masks lubridate::interval()
x dplyr::lag()         masks stats::lag()
x fabletools::MAE()    masks DescTools::MAE()
x fabletools::MAPE()   masks DescTools::MAPE()
x fabletools::MSE()    masks DescTools::MSE()
x ggpubr::mutate()     masks dplyr::mutate(), plyr::mutate()
x dplyr::rename()      masks plyr::rename()
x fabletools::RMSE()   masks DescTools::RMSE()
x tsibble::setdiff()   masks base::setdiff()
x dplyr::summarise()   masks plyr::summarise()
x dplyr::summarize()   masks plyr::summarize()
x tsibble::union()     masks base::union()
```

```
library(ggplot2)
library(tsibble)
library(feasts)
library(openxlsx)
library(latex2exp)
```

```
Warning: package 'latex2exp' was built under R version 4.4.3
```

```
library(seasonal)
```

```
Warning: package 'seasonal' was built under R version 4.4.3


Attaching package: 'seasonal'

The following object is masked from 'package:tibble':

    view
```

```
library(seasonalview)
```

```
Attaching package: 'seasonalview'

The following object is masked from 'package:seasonal':

    view

The following object is masked from 'package:tibble':

    view
```

```
library(fable)
library(rio)
```

```
Warning: package 'rio' was built under R version 4.4.3
```

```
library(urca)
```

```
Warning: package 'urca' was built under R version 4.4.3
```

```
library(zoo)
```

```
Attaching package: 'zoo'

The following object is masked from 'package:tsibble':

    index

The following objects are masked from 'package:base':

    as.Date, as.Date.numeric
```

```
library(lubridate)
library(writexl)
```

```
Warning: package 'writexl' was built under R version 4.4.3
```

```
library(janitor)
```

```
Warning: package 'janitor' was built under R version 4.4.3


Attaching package: 'janitor'

The following objects are masked from 'package:stats':

    chisq.test, fisher.test
```

In order to answer the first question population data was downloaded. It will be used to calculate the revenue spent from the Act on each individual for each state. In order to address the second question there was a comparison of the Biden supporting states and the Trump supporting states. The election results of each state was gathered to determine if the state supported Trump or Biden.

### Sources of the Data

#### Population of each state:

https://www.census.gov/data/tables/time-series/demo/popest/2020s-state-total.html

US Census Bureau. (n.d.). *State Population Totals and Components of Change: 2020-2024*. Census.gov. https://www.census.gov/data/tables/time-series/demo/popest/2020s-state-total.html

**General election results:**

https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/42MVDX

MIT Election Data and Science Lab, 2017, "U.S. President 1976–2020", https://doi.org/10.7910/DVN/42MVDX, Harvard Dataverse, V8, UNF:6:F0opd1IRbeYI9QyVfzglUw== [fileUNF]

## Data imported into R

### Infrastructure and Jobs Act Dataset

```
#stored the data sets on Github for download/read
c <- "https://github.com/division-zero/Data-608/raw/refs/heads/main/Story%201/IIJA%20FUNDING%
#read in the raw file
temp <- tempfile(fileext = ".xlsx")
#download the file
GET(c, write_disk(temp))
```

```
Response [https://raw.githubusercontent.com/division-zero/Data-608/refs/heads/main/Story%201/
  Date: 2025-03-28 01:42
  Status: 200
  Content-Type: application/octet-stream
  Size: 11.4 kB
<ON DISK>  C:\Users\keith\AppData\Local\Temp\RtmpGea0bX\file15f68da554fb.xlsx
```

```
# Load the file into a dataframe
funding_df <- read_excel(temp)

# temp file removed
unlink(temp)

names(funding_df)[1] <- "state"

#fix typo for delaware
funding_df[9,1] <- 'DELAWARE'

head(funding_df)
```

```
# A tibble: 6 x 2
  state          `Total (Billions)`
  <chr>                       <dbl>
1 ALABAMA                     3
2 ALASKA                      3.7
3 AMERICAN SAMOA              0.0686
4 ARIZONA                     3.5
5 ARKANSAS                    2.8
6 CALIFORNIA                 18.4
```

**Population of each state.**

Data was imported and cleaned.

```
c1 <- "https://github.com/division-zero/Data-608/raw/refs/heads/main/Story%201/NST-EST2024-P
#read in the raw file
temp <- tempfile(fileext = ".xlsx")
#download the file
GET(c1, write_disk(temp))
```

```
Response [https://raw.githubusercontent.com/division-zero/Data-608/refs/heads/main/Story%201/
  Date: 2025-03-28 01:42
  Status: 200
  Content-Type: application/octet-stream
  Size: 15.5 kB
<ON DISK>  C:\Users\keith\AppData\Local\Temp\RtmpGea0bX\file15f684c23796b.xlsx
```

```
# Load the file into a dataframe
statepop_df <- read_excel(temp)
```

```
New names:
* `` -> `...2`
* `` -> `...3`
* `` -> `...4`
* `` -> `...5`
* `` -> `...6`
* `` -> `...7`
```

```r
# temp file removed
unlink(temp)

head(statepop_df)
```

```
# A tibble: 6 x 7
  table with row headers in column~1 ...2  ...3     ...4    ...5    ...6    ...7
  <chr>                              <chr> <chr>   <dbl>   <dbl>   <dbl>   <dbl>
1 Annual Estimates of the Resident ~ <NA>  <NA>  NA      NA      NA      NA
2 Geographic Area                    Apri~ Popu~ NA      NA      NA      NA
3 <NA>                               <NA>  2020   2.02e3  2.02e3  2.02e3  2.02e3
4 United States                      3315~ 3315~  3.32e8  3.34e8  3.37e8  3.40e8
5 Northeast                          5761~ 5743~  5.73e7  5.72e7  5.74e7  5.78e7
6 Midwest                            6899~ 6898~  6.89e7  6.89e7  6.92e7  6.96e7
# i abbreviated name:
#   1: `table with row headers in column A and column headers in rows 3 through 4. (leading c
```

```r
statepop_df <- statepop_df |>
  row_to_names(row_number = 3)
```

```
Warning: Row 3 does not provide unique names. Consider running clean_names()
after row_to_names().
```

```r
names(statepop_df)[1] <- "state"
```

```
Warning: The `value` argument of `names<-()` can't be empty as of tibble 3.0.0.
```

```r
statepop_df <- statepop_df[6:56, ] |> select( 'state', '2020')
names(statepop_df)[2] <- "population"

#remove the '.' in front of the state names
statepop_df <- statepop_df %>%
  mutate(state = gsub("^\\.", "", state))

  statepop_df <- statepop_df |> mutate(`state` = toupper(`state`))

head(statepop_df)
```

```
# A tibble: 6 x 2
  state      population
  <chr>      <chr>
1 ALABAMA    5033094
2 ALASKA     733017
3 ARIZONA    7187135
4 ARKANSAS   3014546
5 CALIFORNIA 39521958
6 COLORADO   5787129
```

Population of each state put into dataframe

### Election Results

Imported the election results into R cleaned and stored into a data frame

```
c2 <- getURL("https://raw.githubusercontent.com/division-zero/Data-608/refs/heads/main/Story%
#read in the raw file
genelection_df <- data.frame(read.csv(text = c2 ))
#put the csv into a dataframe
head(genelection_df)
```

```
  year   state state_po state_fips state_cen state_ic       office
1 1976 ALABAMA       AL          1        63       41 US PRESIDENT
2 1976 ALABAMA       AL          1        63       41 US PRESIDENT
3 1976 ALABAMA       AL          1        63       41 US PRESIDENT
4 1976 ALABAMA       AL          1        63       41 US PRESIDENT
5 1976 ALABAMA       AL          1        63       41 US PRESIDENT
6 1976 ALABAMA       AL          1        63       41 US PRESIDENT
                candidate             party_detailed writein candidatevotes
1           CARTER, JIMMY                   DEMOCRAT   FALSE         659170
2            FORD, GERALD                 REPUBLICAN   FALSE         504070
3         MADDOX, LESTER AMERICAN INDEPENDENT PARTY   FALSE           9198
4 BUBAR, BENJAMIN ""BEN""                PROHIBITION   FALSE           6669
5              HALL, GUS        COMMUNIST PARTY USE   FALSE           1954
6        MACBRIDE, ROGER                LIBERTARIAN   FALSE           1481
  totalvotes  version notes party_simplified
1    1182850 20210113    NA         DEMOCRAT
2    1182850 20210113    NA       REPUBLICAN
3    1182850 20210113    NA            OTHER
4    1182850 20210113    NA            OTHER
```

```
5    1182850 20210113    NA            OTHER
6    1182850 20210113    NA      LIBERTARIAN
```

```
#extract Biden and Trump election data for year 2020
election2020_df <- genelection_df |>  filter(year == 2020)
head(election2020_df)
```

```
  year   state state_po state_fips state_cen state_ic       office
1 2020 ALABAMA       AL          1        63       41 US PRESIDENT
2 2020 ALABAMA       AL          1        63       41 US PRESIDENT
3 2020 ALABAMA       AL          1        63       41 US PRESIDENT
4 2020 ALABAMA       AL          1        63       41 US PRESIDENT
5 2020  ALASKA       AK          2        94       81 US PRESIDENT
6 2020  ALASKA       AK          2        94       81 US PRESIDENT
            candidate party_detailed writein candidatevotes totalvotes  version
1 BIDEN, JOSEPH R. JR       DEMOCRAT   FALSE         849624    2323282 20210113
2    TRUMP, DONALD J.     REPUBLICAN   FALSE        1441170    2323282 20210113
3       JORGENSEN, JO    LIBERTARIAN   FALSE          25176    2323282 20210113
4                                      TRUE           7312    2323282 20210113
5 BIDEN, JOSEPH R. JR       DEMOCRAT   FALSE         153778     359530 20210113
6    TRUMP, DONALD J.     REPUBLICAN   FALSE         189951     359530 20210113
  notes party_simplified
1    NA         DEMOCRAT
2    NA       REPUBLICAN
3    NA      LIBERTARIAN
4    NA            OTHER
5    NA         DEMOCRAT
6    NA       REPUBLICAN
```

```
#remove blanks in candidate column
election2020_df <- election2020_df |>
  filter(candidate != "")

#if the string under candidate contains "BIDEN" relabel the candidate name to BIDEN. If the C
election_df <- election2020_df |>
  mutate(candidate = ifelse(grepl("BIDEN", candidate), "BIDEN", candidate),
         candidate = ifelse(grepl("TRUMP", candidate), "TRUMP", candidate))


#remove columns that are not of interest
 election2020_df1 <-election_df  |> select('state', 'candidate', 'candidatevotes')
```

```
 #select the rows that have Biden or Trump as the candidate
 election2020_df2 <- election2020_df1 |> filter( candidate %in% c("BIDEN" , "TRUMP"))

 #take the votes for each candidate and put them under the candidate name.  the candidate nar
 electionBT_df <- election2020_df2 |>
   pivot_wider(names_from = candidate, values_from = candidatevotes)

 #if the votes for one candidate are greater than the other, then declare that candidate the
 electionBT_df <- electionBT_df |>  mutate(winner = ifelse(TRUMP > BIDEN, "TRUMP", "BIDEN"))
head(electionBT_df)
```

```
# A tibble: 6 x 4
  state         BIDEN   TRUMP winner
  <chr>         <int>   <int> <chr>
1 ALABAMA      849624 1441170 TRUMP
2 ALASKA       153778  189951 TRUMP
3 ARIZONA     1672143 1661686 BIDEN
4 ARKANSAS     423932  760647 TRUMP
5 CALIFORNIA 11110250 6006429 BIDEN
6 COLORADO    1804352 1364607 BIDEN
```

### Merging the data sets.

The data frames were merged by state. The revenue spent from the act for each state was divided by the population of the respective state.

```
merged_df <- funding_df |>
  full_join(statepop_df, by = "state") |>
  full_join(electionBT_df, by = "state")
names(merged_df)[2] <- "Total_Billions"

merged_df <- merged_df |>
  mutate(per_capita = as.numeric(Total_Billions)*1000000000 / as.numeric(population))

head(merged_df)
```

```
# A tibble: 6 x 7
  state          Total_Billions population    BIDEN   TRUMP winner per_capita
  <chr>                   <dbl> <chr>         <int>   <int> <chr>       <dbl>
1 ALABAMA                     3 5033094      849624 1441170 TRUMP        596.
```

```
2 ALASKA                3.7    733017       153778  189951 TRUMP       5048.
3 AMERICAN SAMOA        0.0686 <NA>             NA      NA <NA>          NA
4 ARIZONA               3.5    7187135     1672143 1661686 BIDEN        487.
5 ARKANSAS              2.8    3014546      423932  760647 TRUMP        929.
6 CALIFORNIA           18.4    39521958   11110250 6006429 BIDEN        466.
```

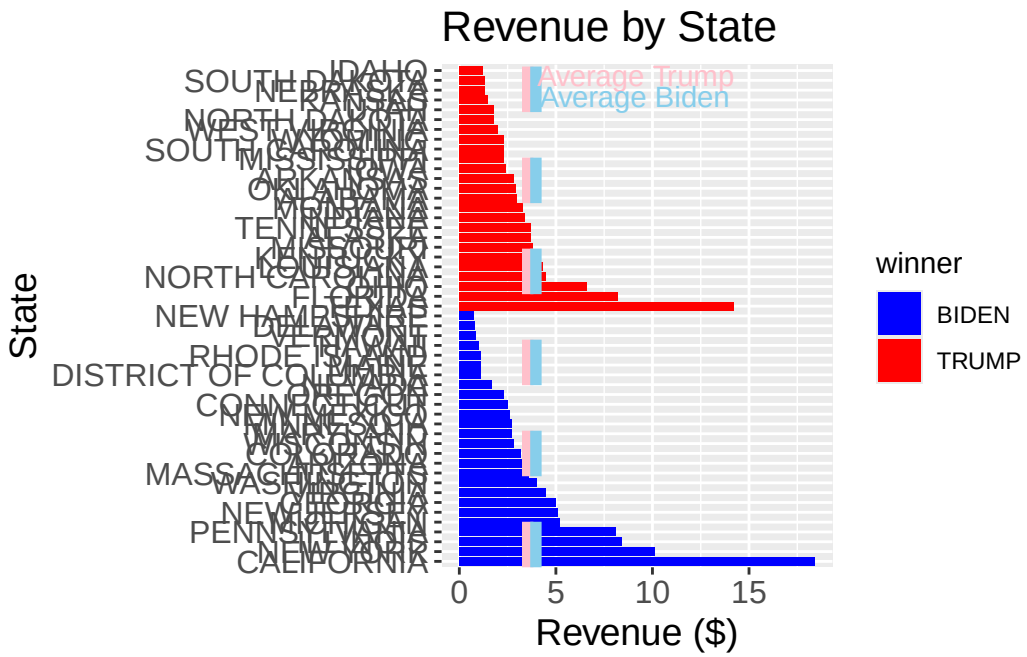
### Visualize the Data

At first we can look at how much money was spent on each state.

```
merged_df <- merged_df |>
  filter(!is.na(per_capita))

averageT_revenue <- merged_df |>
  filter(winner == 'TRUMP') |>
  summarize(mean_revenue = mean(Total_Billions, na.rm = TRUE)) |>
  pull(mean_revenue)

averageB_revenue <- merged_df |>
  filter(winner == 'BIDEN') |>
  summarize(mean_revenue = mean(Total_Billions, na.rm = TRUE)) |>
  pull(mean_revenue)

merged_df <- merged_df |>
  arrange(winner, desc(Total_Billions)) |>
  mutate(state = factor(state, levels = unique(state)))

merged_df |> ggplot( aes(x = state, y = Total_Billions, fill = winner)) +
  geom_col( ) +
  labs(title = "Revenue by State") +
  scale_fill_manual(values = c("BIDEN" = "blue", "TRUMP" = "red")) +
geom_hline(yintercept = averageT_revenue, linetype = "dashed", color = "pink", size = 2)  +
geom_hline(yintercept = averageB_revenue, linetype = "dashed", color = "skyblue", size = 2) +
  annotate("text", x = Inf, y = averageT_revenue, label = "Average Trump", hjust = -0.05, vju
  annotate("text", x = Inf, y = averageB_revenue, label = "Average Biden", hjust = -0.02, vju
   labs(title = "Revenue by State",
       x = "State",
       y = "Revenue ($)") +
  coord_flip() +
  theme(axis.text.y = element_text(size = 12),
    axis.title.y = element_text(size = 14),
    plot.title = element_text(size = 16),
```

```
    axis.text.x = element_text(size = 12),
    axis.title.x = element_text(size = 14))
```

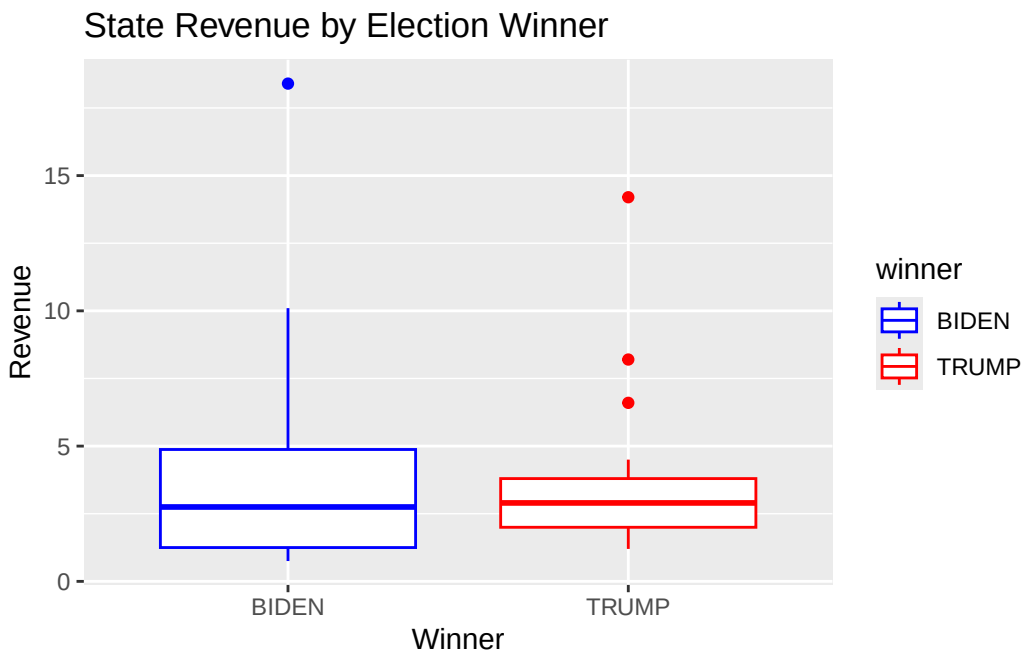
```
Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
i Please use `linewidth` instead.
```




Money was not spent equally on each state. It has a distribution. some states Received much more money than others. California, Florida, New York, and Ohio were among the states that received the highest amounts. New Hampshire, Delaware, Vermont, and Idaho were among the smallest recipients.

### Looking at the revenue distribution via box and whisker plot

```
ggplot(merged_df, aes(x = winner, y = Total_Billions, color = winner)) +
  scale_color_manual(values = c("BIDEN" = "blue", "TRUMP" = "red"))+
  geom_boxplot() +
  labs(title = "State Revenue by Election Winner",
       x = "Winner",
       y = "Revenue")
```



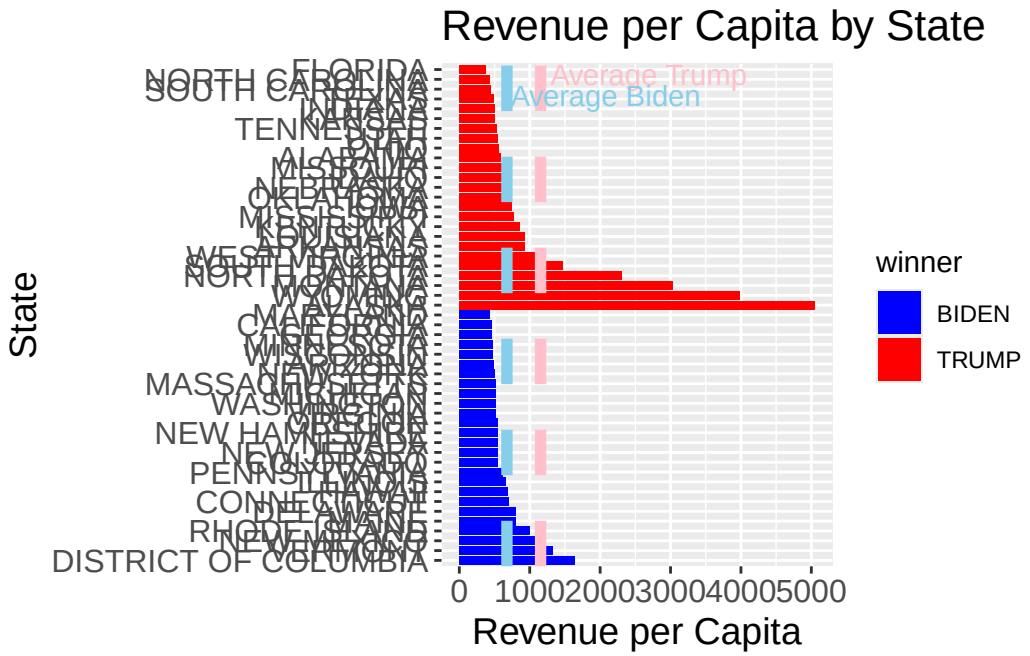

Though money was not spent equally on each state. The red states based on mean and median did not receive significantly more money than blue states.

### Infrastructure and Jobs Act Per Capita.

Ideally the money spent per person or capita should be completely evenly distributed. we can visualize the distribution for red and blue states.

```
merged_df <- merged_df |>
  filter(!is.na(per_capita))

averageT_per_capita <- merged_df |>
  filter(winner == 'TRUMP') |>
  summarize(mean_per_capita = mean(per_capita, na.rm = TRUE)) |>
  pull(mean_per_capita)

averageB_per_capita <- merged_df |>
  filter(winner == 'BIDEN') |>
  summarize(mean_per_capita = mean(per_capita, na.rm = TRUE)) |>
  pull(mean_per_capita)

merged_df <- merged_df |>
```

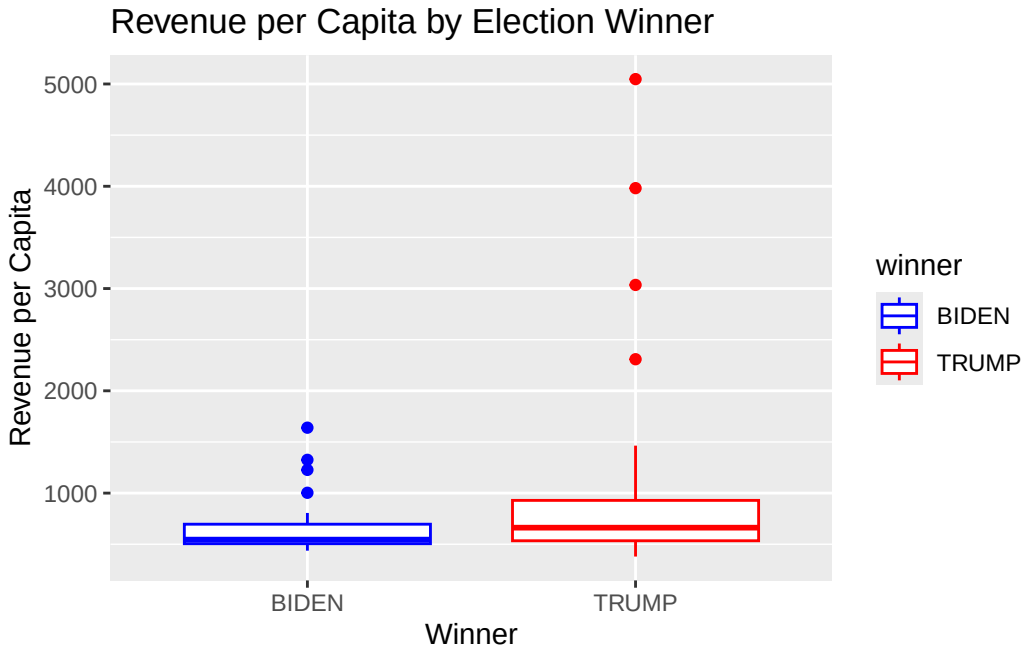
```
  arrange(winner, desc(per_capita)) |>
  mutate(state = factor(state, levels = unique(state)))

merged_df |> ggplot( aes(x = state, y = per_capita, fill = winner)) +
  geom_col( ) +
  labs(title = "Revenue per Capita by State") +
  scale_fill_manual(values = c("BIDEN" = "blue", "TRUMP" = "red")) +
geom_hline(yintercept = averageT_per_capita, linetype = "dashed", color = "pink", size = 2)
geom_hline(yintercept = averageB_per_capita, linetype = "dashed", color = "skyblue", size = 
  annotate("text", x = Inf, y = averageT_per_capita, label = "Average Trump", hjust = -0.05,
  annotate("text", x = Inf, y = averageB_per_capita, label = "Average Biden", hjust = -0.02,
  labs(title = "Revenue per Capita by State",
       x = "State",
       y = "Revenue per Capita") +
  coord_flip() +
  theme(axis.text.y = element_text(size = 12),
    axis.title.y = element_text(size = 14),
    plot.title = element_text(size = 16),
    axis.text.x = element_text(size = 12),
    axis.title.x = element_text(size = 14))
```




From the bar chart the money clearly was more evenly distributed among the blue states. The red states had a more extreme discrepancy for the low population states. When looking on

average the red states received more money per capita, however the median more accurately reflects how the money was distributed.

```
ggplot(merged_df, aes(x = winner, y = per_capita, color = winner)) +
  scale_color_manual(values = c("BIDEN" = "blue", "TRUMP" = "red"))+
  geom_boxplot() +
  labs(title = "Revenue per Capita by Election Winner",
       x = "Winner",
       y = "Revenue per Capita")
```




The revenue per capita appeared to have an even distrubution among the majority of the states. States with low populations appear to have received more funding per person.

The median revenue per capita was very close for the red states and blue states. Despite a democrat president being in power it can be argued that the bias went towards the republican red states rather than the blue states. One way this would benefit politically is if the democrat party would want to increase their influence in the republican majority states. However the differences seem relatively minimal especially when comparing to the median revenue per capita between the red and blue states. How the money is allocated may be based on some other factor rather than political ideology. It does not appear that the Biden administration played "favorites" among the democrat majority states and if anything the opposite occurred.