Lab 2

Quynh Vu

2023-01-22

1. Using the `delay_2022` data, plot the five stations with the highest mean delays. Facet the graph by `line`

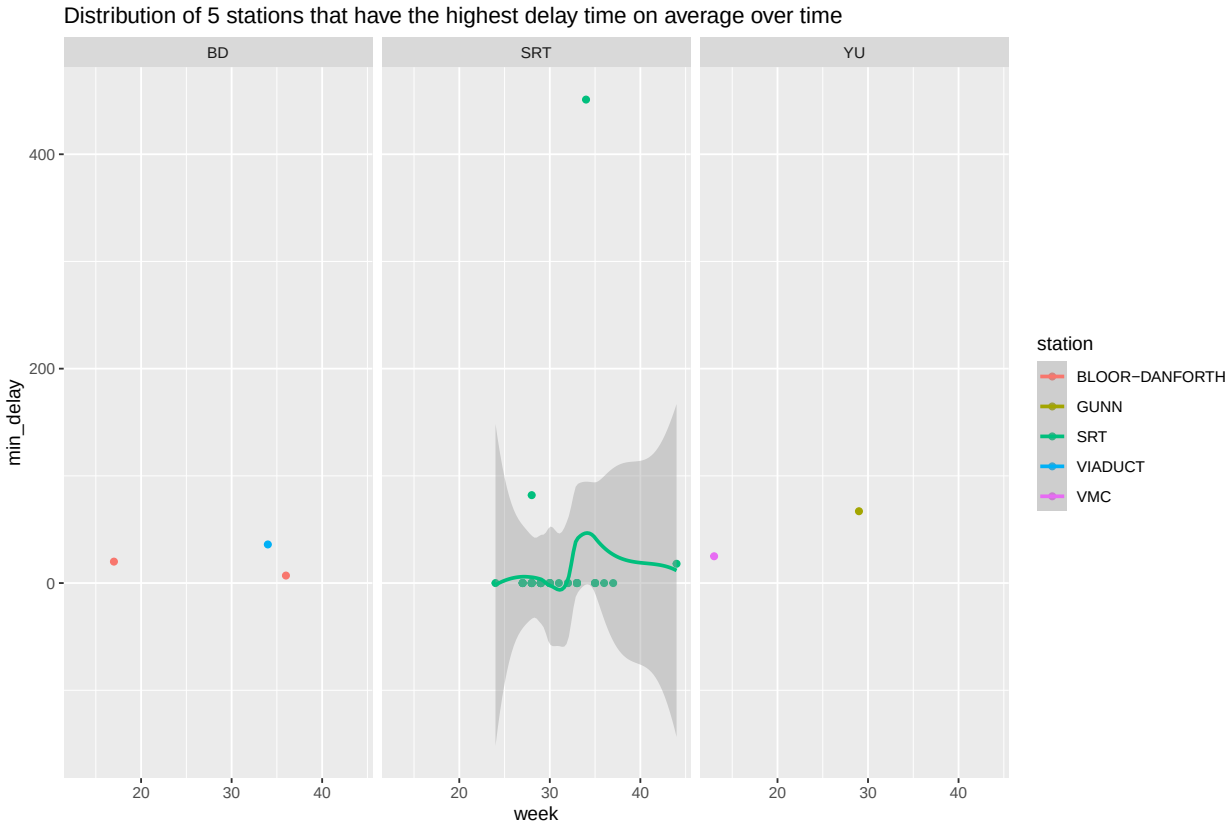
```
top_five_delays <- delay_2022 |>
                              group_by(station) |>
                              mutate(avgDelay = mean(min_delay)) |>
                              arrange(desc(avgDelay))|>
                              distinct(avgDelay)
```

After removing the observations that have non-standardized lines, we recoded station names to make them consistent, e.g. both `ST. GEORGE` and `ST GEORGE` to `ST.GEORGE` or `YONGE/UNIVERSITY` to `YONGE-UNIVERSITY`. The five stations with the highest mean delays are

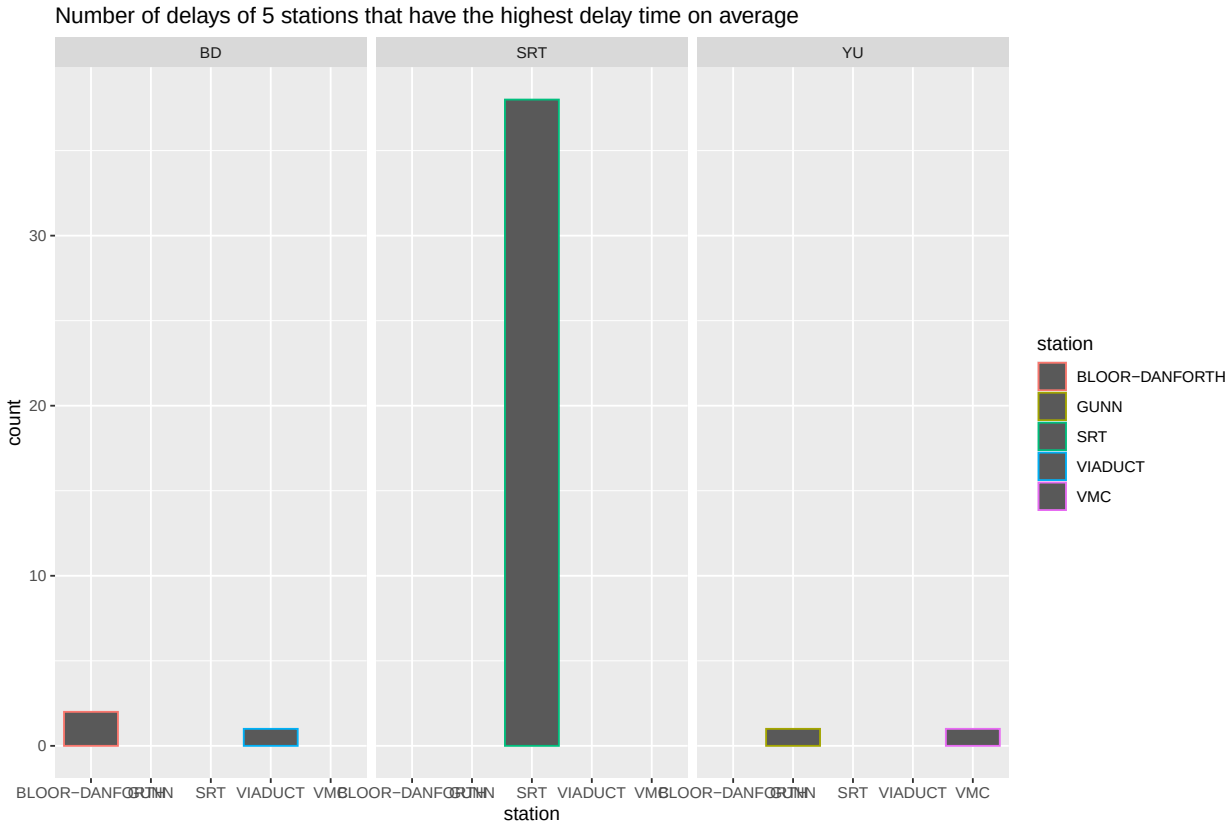
```
head(top_five_delays, 5)
```

```
## # A tibble: 5 x 2
## # Groups:   station [5]
##   station        avgDelay
##   <fct>             <dbl>
## 1 GUNN               67
## 2 VIADUCT            36
## 3 VMC                25
## 4 SRT                14.5
## 5 BLOOR-DANFORTH     13.5
```

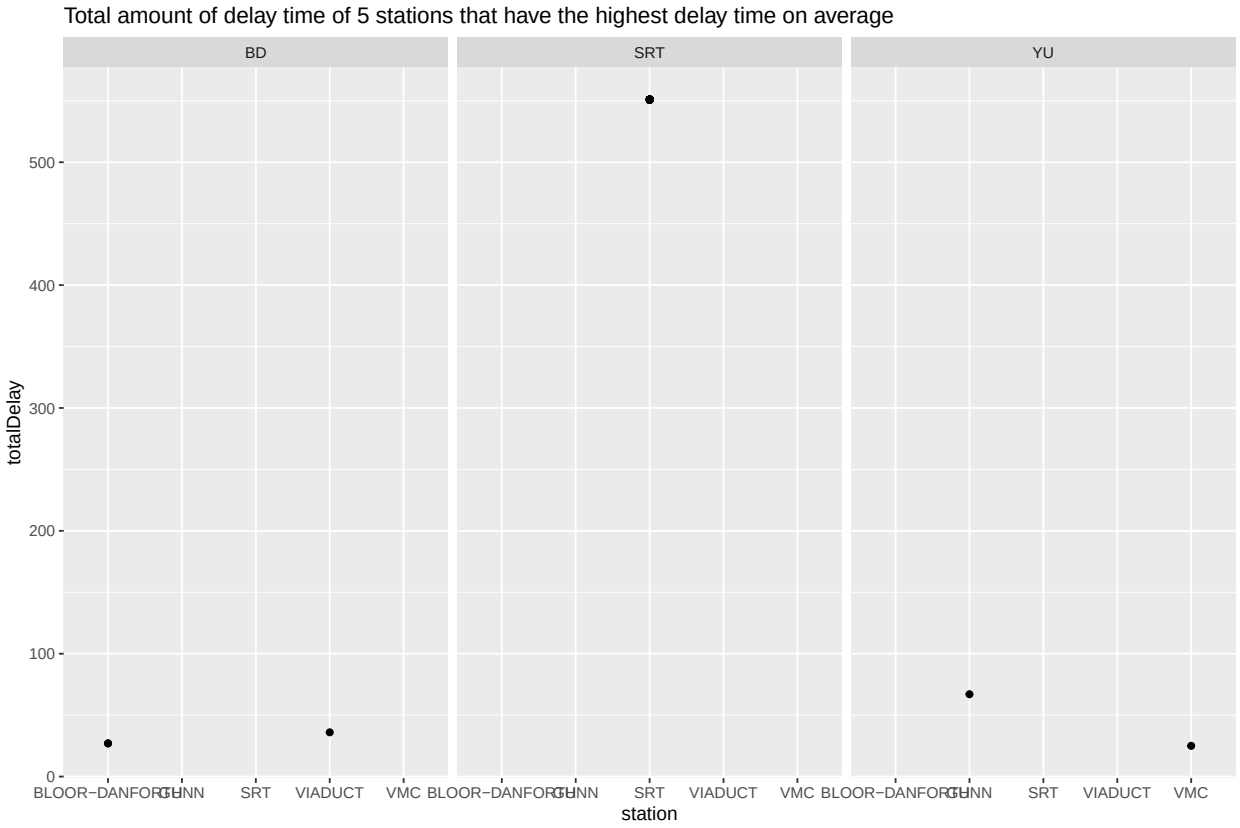
```
top5 <- delay_2022 |> filter(station %in% c("GUNN", "VIADUCT", "VMC", "SRT", "BLOOR-DANFORTH"))

top5 |> mutate(week = week(date)) |> # Get/set weeks component of a date-time
        group_by(week, line) |>
        ggplot(aes(week, min_delay, color = station)) +
               geom_point() +
               geom_smooth() +
               labs(title = "Distribution of 5 stations that have the highest delay time on average ove
               facet_grid(~line)
```




```
top5 |> group_by(station) |>
        ggplot(aes(station, color = station)) +
               geom_bar() +
               labs(title = "Number of delays of 5 stations that have the highest delay time on average
                facet_grid(~line)
```




```
top5 |> group_by(station) |>
        mutate(totalDelay = sum(min_delay)) |>
        ggplot(aes(y = totalDelay, x = station)) +
        geom_point() +
        labs(title = "Total amount of delay time of 5 stations that have the highest delay time on avera
        facet_grid(~line)
```



2. Using the `opendatatoronto` package, download the data on mayoral campaign contributions for 2014. Hints:

- find the ID code you need for the package you need by searching for 'campaign' in the `all_data` tibble above
- you will then need to `list_package_resources` to get ID for the data file

Note: the 2014 file you will get from `get_resource` has a bunch of different campaign contributions, so just keep the data that relates to the Mayor election

```
contribution <- list_package_resources("f6651a40-2f52-46fc-9e04-b760c16edd5c")
campaign <- get_resource("5b230e92-0a22-4a15-9572-0b19cc222985")
mayor2014 <- campaign[["2_Mayor_Contributions_2014_election.xls"]]
head(mayor2014)
```

```
## # A tibble: 6 x 13
##   2014 Munic~1 ...2  ...3  ...4  ...5  ...6  ...7  ...8  ...9  ...10 ...11 ...12
##   <chr>        <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr> <chr>
## 1 Contributor~ Cont~ Cont~ Cont~ Cont~ Good~ Cont~ Rela~ Pres~ Auth~ Cand~ Offi~
## 2 A D'Angelo,~ <NA>  M6A ~ 300   Mone~ <NA>  Indi~ <NA>  <NA>  <NA>  Ford~ Mayor
## 3 A Strazar, ~ <NA>  M2M ~ 300   Mone~ <NA>  Indi~ <NA>  <NA>  <NA>  Ford~ Mayor
## 4 A'Court, K ~ <NA>  M4M ~ 36    Mone~ <NA>  Indi~ <NA>  <NA>  <NA>  Chow~ Mayor
## 5 A'Court, K ~ <NA>  M4M ~ 100   Mone~ <NA>  Indi~ <NA>  <NA>  <NA>  Chow~ Mayor
## 6 A'Court, K ~ <NA>  M4M ~ 100   Mone~ <NA>  Indi~ <NA>  <NA>  <NA>  Chow~ Mayor
## # ... with 1 more variable: ...13 <chr>, and abbreviated variable name
## #   1: '2014 Municipal Election - List of Contributors to Mayoralty Candidates'
```

3. Clean up the data format (fixing the parsing issue and standardizing the column names using `janitor`)

```
#not_all_na <- function(x) any(!is.na(x))
#select_if(not_all_na) |> # remove columns with all NAs
names(mayor2014) <- as.matrix(mayor2014[1, ])
mayor2014 <- mayor2014[-1, ]  # make the first row the header

mayor2014 <- mayor2014 |> clean_names() |>
                          rename(contributor = contributors_name,
                                 contributor_type = contributor_type_desc,
                                 relationship = relationship_to_candidate,
                                 representative = authorized_representative,
                                 contribution_type = contribution_type_desc,
                                 manager = president_business_manager,
                                 services = goods_or_service_desc)
names(mayor2014)[1:5][-1] = str_sub(names(mayor2014)[-1], 14)
head(mayor2014)
```

```
## # A tibble: 6 x 13
##   contrib~1 address posta~2 amount type  servi~3 contr~4 relat~5 manager repre~6
##   <chr>     <chr>   <chr>   <chr>  <chr> <chr>   <chr>   <chr>   <chr>   <chr>
## 1 A D'Ange~ <NA>    M6A 1P5 300    Mone~ <NA>    Indivi~ <NA>    <NA>    <NA>
## 2 A Straza~ <NA>    M2M 3B8 300    Mone~ <NA>    Indivi~ <NA>    <NA>    <NA>
## 3 A'Court,~ <NA>    M4M 2J8 36     Mone~ <NA>    Indivi~ <NA>    <NA>    <NA>
## 4 A'Court,~ <NA>    M4M 2J8 100    Mone~ <NA>    Indivi~ <NA>    <NA>    <NA>
## 5 A'Court,~ <NA>    M4M 2J8 100    Mone~ <NA>    Indivi~ <NA>    <NA>    <NA>
## 6 Aaron, R~ <NA>    M6B 1H7 250    Mone~ <NA>    Indivi~ <NA>    <NA>    <NA>
## # ... with 3 more variables: candidate <chr>, office <chr>, ward <chr>, and
## #   abbreviated variable names 1: contributor, 2: postal_code, 3: services,
## #   4: contributor_type, 5: relationship, 6: representative
```

4. Summarize the variables in the dataset. Are there missing values, and if so, should we be worried about them? Is every variable in the format it should be? If not, create new variable(s) that are in the right format.

Summarize the variables in the dataset:

```
skim(mayor2014)
```

Table 1: Data summary

Name	mayor2014
Number of rows	10199
Number of columns	13
Column type frequency:	
character	13
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
contributor	0	1	4	31	0	7545	0
address	10197	0	24	26	0	2	0
postal_code	0	1	7	7	0	5284	0
amount	0	1	1	18	0	209	0
type	0	1	8	14	0	2	0
services	10188	0	11	40	0	9	0
contributor_type	0	1	10	11	0	2	0
relationship	10166	0	6	9	0	2	0
manager	10197	0	13	16	0	2	0
representative	10197	0	13	16	0	2	0
candidate	0	1	9	18	0	27	0
office	0	1	5	5	0	1	0
ward	10199	0	NA	NA	0	0	0

There are a lot of `NA`s in variables `address` (contributor's address), `services` (types of goods or services provided in place of monetary support), `relationship` (relationship between the contributor and the candidate), `manager` (name of president business manager), and `representative` (authorized representative). All of the values in the variable `ward` are missing.

How we deal with missing values depends on our analysis goal. For instance, if we want to investigate and compare the contribution values that individuals and cooperations supported the candidates from their favourable political party, then we should not be worried about the missing information on the six variables having the highest missing value. Also, the `postal code` variable provided more comprehensive information on the contributor's residency than the variable `address` if that topic of our interest. In case the variable we are interested in has lots of NAs, then steps such as interpolation, imputation, or adding missing indicator to encode "missingness" as a feature should be taken into consideration to obtain well-fitted models.

The variable `amount` should be of numeric instead of character type. We also recorded variables `type` (type of contribution) and `contributor_type` as factors with two levels as follows:

Variable	Levels of Factor
`type`	Goods/Services
	Monetary
`contributor_type`	Corporation
	Individual

```
str(mayor_2014)
```

```
## 'data.frame':    10199 obs. of  13 variables:
##  $ contributor     : chr  "A D'Angelo, Tullio" "A Strazar, Martin" "A'Court, K Susan" "A'Court, K Sus
##  $ address         : chr  NA NA NA NA ...
##  $ postal_code     : chr  "M6A 1P5" "M2M 3B8" "M4M 2J8" "M4M 2J8" ...
##  $ amount          : num  300 300 36 100 100 250 500 500 300 150 ...
##  $ type            : Factor w/ 2 levels "Goods/Services",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ services        : chr  NA NA NA NA ...
##  $ contributor_type: Factor w/ 2 levels "Corporation",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ relationship    : chr  NA NA NA NA ...
##  $ manager         : chr  NA NA NA NA ...
##  $ representative  : chr  NA NA NA NA ...
```

```
##  $ candidate       : chr  "Ford, Rob" "Ford, Rob" "Chow, Olivia" "Chow, Olivia" ...
##  $ office          : chr  "Mayor" "Mayor" "Mayor" "Mayor" ...
##  $ ward            : chr  NA NA NA NA ...
```

5. Visually explore the distribution of values of the contributions. What contributions are notable outliers? Do they share a similar characteristic(s)? It may be useful to plot the distribution of contributions without these outliers to get a better sense of the majority of the data.

```
summary(mayor_2014$amount)
```

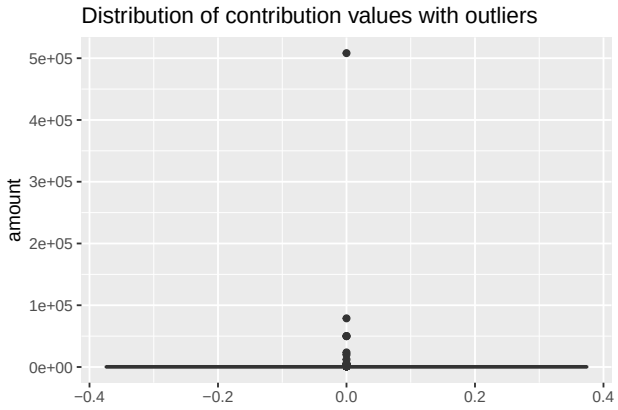
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1     100     300     608     500  508225
```

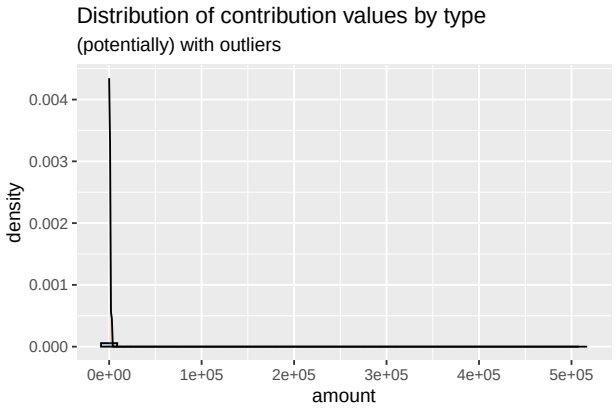
The maximum value of the contributions is very large compared to the mean, so we suspect some influential outliers exist in the dataset. We see that most substantial contributions come from the candidate themselves or their spouse. Some notable outliers are

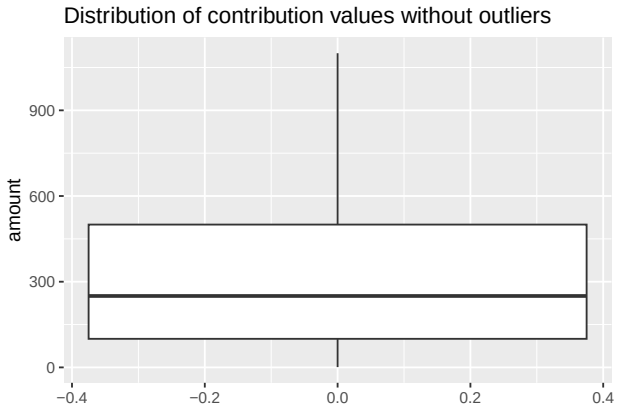
```
##        contributor amount     type contributor_type relationship
## 1     Chow, Olivia   2500 Monetary       Individual    Candidate
## 2 Hackett, Barbara   2500 Monetary       Individual       Spouse
## 3 Sniedzins, Erwin   2500 Monetary       Individual    Candidate
## 4   Thomson, Sarah   2500 Monetary       Individual    Candidate
## 5       Tory, John   2500 Monetary       Individual    Candidate
## 6       Yan, Flora   2500 Monetary       Individual       Spouse
##          candidate
## 1     Chow, Olivia
## 2       Tory, John
## 3 Sniedzins, Erwin
## 4   Thomson, Sarah
## 5       Tory, John
## 6 Sniedzins, Erwin
```

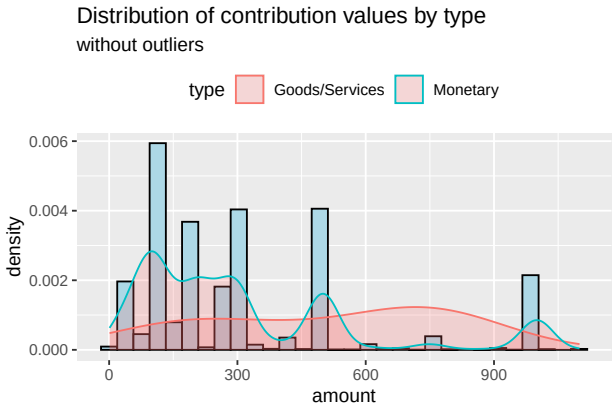
and also

```
##     contributor    amount     type contributor_type relationship     candidate
## 1    Ford, Doug 508224.73 Monetary       Individual    Candidate    Ford, Doug
## 2    Ford, Doug  50000.00 Monetary       Individual    Candidate    Ford, Doug
## 3     Ford, Rob  20000.00 Monetary       Individual    Candidate     Ford, Rob
## 4     Ford, Rob  50000.00 Monetary       Individual    Candidate     Ford, Rob
## 5     Ford, Rob  50000.00 Monetary       Individual    Candidate     Ford, Rob
## 6     Ford, Rob  78804.80 Monetary       Individual    Candidate     Ford, Rob
## 7     Ford, Rob  12210.00 Monetary       Individual    Candidate     Ford, Rob
## 8 Goldkind, Ari  23623.63 Monetary       Individual    Candidate Goldkind, Ari
```









6. List the top five candidates in each of these categories:

Total contributions

```
## # A tibble: 5 x 2
## # Groups:   candidate [5]
##   candidate     sumContr
##   <chr>            <dbl>
## 1 Tory, John    2767869.
## 2 Chow, Olivia  1638266.
## 3 Ford, Doug     889897.
## 4 Ford, Rob      387648.
## 5 Stintz, Karen  242805
```

Mean contribution

```
## # A tibble: 5 x 2
## # Groups:   candidate [5]
##   candidate        avgContr
##   <chr>               <dbl>
## 1 Sniedzins, Erwin    2025
## 2 Syed, Hïmy          2018
## 3 Ritch, Carlie       1887.
## 4 Ford, Doug          1456.
## 5 Clarke, Kevin       1200
```

Number of contributions

```
## # A tibble: 5 x 2
## # Groups:   candidate [5]
##   candidate       count
##   <chr>           <int>
## 1 Chow, Olivia     5708
## 2 Tory, John       2602
## 3 Ford, Doug        611
## 4 Ford, Rob         538
## 5 Soknacki, David   314
```

7. Repeat 6 but without contributions from the candidates themselves.

Total contributions

```
## # A tibble: 5 x 2
## # Groups:   candidate [5]
##   candidate     sumContr2
##   <chr>             <dbl>
## 1 Tory, John     2763989.
## 2 Chow, Olivia   1249285.
## 3 Ford, Doug      305810.
## 4 Stintz, Karen   234605
## 5 Ford, Rob       161414.
```

Mean contribution

```
## # A tibble: 5 x 2
## # Groups:   candidate [5]
##   candidate        avgContr2
##   <chr>                <dbl>
## 1 Sniedzins, Erwin     2025
## 2 Syed, Hïmy           2018
## 3 Ritch, Carlie        1887.
## 4 Clarke, Kevin        1200
## 5 Di Paola, Rocco      1174.
```

Number of contributions

```
## # A tibble: 5 x 2
## # Groups:   candidate [5]
##   candidate       count2
##   <chr>            <int>
## 1 Chow, Olivia      3533
## 2 Tory, John        2597
## 3 Ford, Doug         533
## 4 Ford, Rob          439
## 5 Soknacki, David    240
```

8. How many contributors gave money to more than one candidate?

184 contributors gave money to more than one candidate.

```
mayor_2014_new3 <- aggregate(mayor_2014_new2$contributor,
                             by=list(mayor_2014_new2$contributor, mayor_2014_new2$candidate),
```

```
                              FUN=length)
colnames(mayor_2014_new3) <- c("contributor", "candidate", "x")
dm <- as.data.frame(table(mayor_2014_new3$contributor))|> filter(Freq > 1)
colnames(dm) <- c("contributor", "number of candiates supported")
length(dm$contributor)
```

```
## [1] 184
```

```
## # A tibble: 378 x 2
## # Groups:   candidate [11]
##    contributor       candidate
##    <chr>             <chr>
##  1 Abadi, Babak      Tory, John
##  2 Abadi, Babak      Chow, Olivia
##  3 Adams, Michael    Soknacki, David
##  4 Adams, Michael    Chow, Olivia
##  5 Anga, John        Ford, Rob
##  6 Anga, John        Ford, Doug
##  7 Argyris, Katerina Ford, Rob
##  8 Argyris, Katerina Ford, Doug
##  9 Atkinson, Tom     Soknacki, David
## 10 Atkinson, Tom     Chow, Olivia
## # ... with 368 more rows
```