EDA and data visualization

Monica Alexander

22/01/23

Table of contents

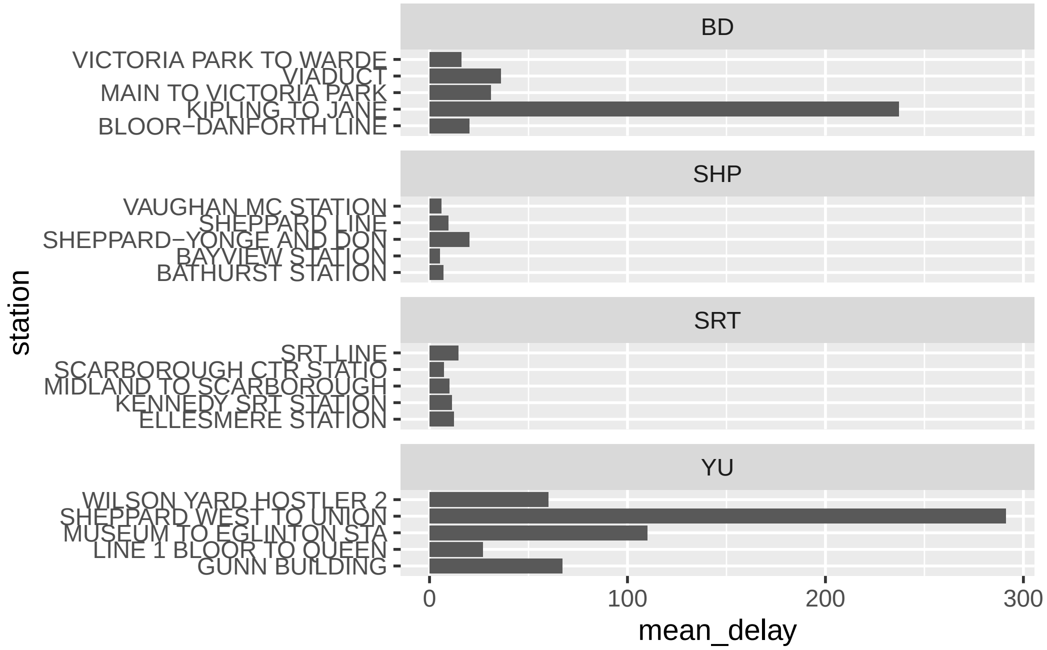
1 Lab Exercises

To be handed in via submission of quarto file (and rendered pdf) to GitHub.

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

```
delay_2022 |>
  group_by(line, station) |>
  summarise(mean_delay = mean(min_delay)) |>
  arrange(-mean_delay) |>
  slice(1:5) |>
  ggplot(aes(x = station,
             y = mean_delay)) +
  geom_col() +
  facet_wrap(vars(line),
             scales = "free_y",
             nrow = 4) +
  coord_flip()
```

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



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

```
cam <- list_package_resources("f6651a40-2f52-46fc-9e04-b760c16edd5c") # obtained code from
#res <- res |> mutate(year = str_extract(name, "202.?"))
#delay_2022_ids <- res |> filter(year==2022) |> select(id) |> pull()

camp_2014_1 <- get_resource("5b230e92-0a22-4a15-9572-0b19cc222985")
```

```
New names:
New names:
New names:
New names:
New names:
New names:
New names:
* `` -> `...2`
* `` -> `...3`
```

```
camp_2014 <- camp_2014_1[[2]]
names(camp_2014) <- camp_2014[1,]
camp_2014 <- camp_2014[-1,]

camp_2014 |>
  slice_head(n = 5)
```

```
# A tibble: 5 x 13
  Contributor'~1 Contr~2 Contr~3 Contr~4 Contr~5 Goods~6 Contr~7 Relat~8 Presi~9
  <chr>          <chr>   <chr>   <chr>   <chr>   <chr>   <chr>   <chr>   <chr>
1 A D'Angelo, T~ <NA>    M6A 1P5 300     Moneta~ <NA>    Indivi~ <NA>    <NA>
2 A Strazar, Ma~ <NA>    M2M 3B8 300     Moneta~ <NA>    Indivi~ <NA>    <NA>
3 A'Court, K Su~ <NA>    M4M 2J8 36      Moneta~ <NA>    Indivi~ <NA>    <NA>
4 A'Court, K Su~ <NA>    M4M 2J8 100     Moneta~ <NA>    Indivi~ <NA>    <NA>
5 A'Court, K Su~ <NA>    M4M 2J8 100     Moneta~ <NA>    Indivi~ <NA>    <NA>
# ... with 4 more variables: `Authorized Representative` <chr>,
#   Candidate <chr>, Office <chr>, Ward <chr>, and abbreviated variable names
#   1: `Contributor's Name`, 2: `Contributor's Address`,
#   3: `Contributor's Postal Code`, 4: `Contribution Amount`,
#   5: `Contribution Type Desc`, 6: `Goods or Service Desc`,
#   7: `Contributor Type Desc`, 8: `Relationship to Candidate`,
#   9: `President/ Business Manager`
```

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

```
camp_2014 <- clean_names(camp_2014)


camp_2014 |>
  slice_head(n = 5)
```

```
# A tibble: 5 x 13
  contributors~1 contr~2 contr~3 contr~4 contr~5 goods~6 contr~7 relat~8 presi~9
  <chr>          <chr>   <chr>   <chr>   <chr>   <chr>   <chr>   <chr>   <chr>
1 A D'Angelo, T~ <NA>    M6A 1P5 300     Moneta~ <NA>    Indivi~ <NA>    <NA>
2 A Strazar, Ma~ <NA>    M2M 3B8 300     Moneta~ <NA>    Indivi~ <NA>    <NA>
3 A'Court, K Su~ <NA>    M4M 2J8 36      Moneta~ <NA>    Indivi~ <NA>    <NA>
4 A'Court, K Su~ <NA>    M4M 2J8 100     Moneta~ <NA>    Indivi~ <NA>    <NA>
```

```
5 A'Court, K Su~ <NA>    M4M 2J8 100     Moneta~ <NA>    Indivi~ <NA>    <NA>
# ... with 4 more variables: authorized_representative <chr>, candidate <chr>,
#   office <chr>, ward <chr>, and abbreviated variable names
#   1: contributors_name, 2: contributors_address, 3: contributors_postal_code,
#   4: contribution_amount, 5: contribution_type_desc,
#   6: goods_or_service_desc, 7: contributor_type_desc,
#   8: relationship_to_candidate, 9: president_business_manager
```

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.

    ```
    skim(camp_2014)
    ```

Table 1: Data summary

Name	camp_2014
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
contributors_name	0	1	4	31	0	7545	0
contributors_address	10197	0	24	26	0	2	0
contributors_postal_code	0	1	7	7	0	5284	0
contribution_amount	0	1	1	18	0	209	0
contribution_type_desc	0	1	8	14	0	2	0
goods_or_service_desc	10188	0	11	40	0	9	0
contributor_type_desc	0	1	10	11	0	2	0
relationship_to_candidate	10166	0	6	9	0	2	0
president_business_manager	10197	0	13	16	0	2	0
authorized_representative	10197	0	13	16	0	2	0
candidate	0	1	9	18	0	27	0
office	0	1	5	5	0	1	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ward	10199	0	NA	NA	0	0	0

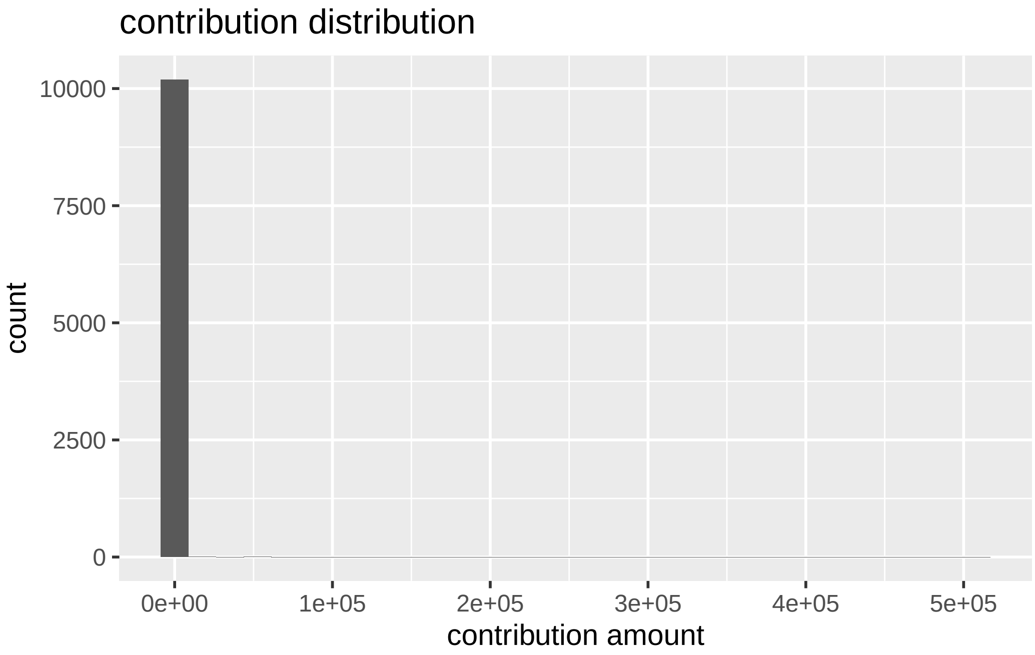
- contributors_address, goods_or_service_desc, relationship_to_candidate, president_business_manager, authorized_representative and ward are almost all missing. I think whether we should be worried about the missing data depends on what the data will be used for. After taking a look at the goal of the lab and the questions below, I think we don't need to worry about the missing values.
- Besides, for values like relationship_to_candidate, I think it is natural for missing values to occur (having no relationship), although we can assign a specific string to represent having no relationship instead of NA.
- Contribution_amount should be in numeric format instead.

```
camp_2014 <- camp_2014 |>
  mutate(num_contribution_amount = as.numeric(contribution_amount))
```

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.

```
ggplot(data = camp_2014) +
  geom_histogram(aes(x = num_contribution_amount)) +
  labs(title = 'contribution distribution',
       x = 'contribution amount')
```

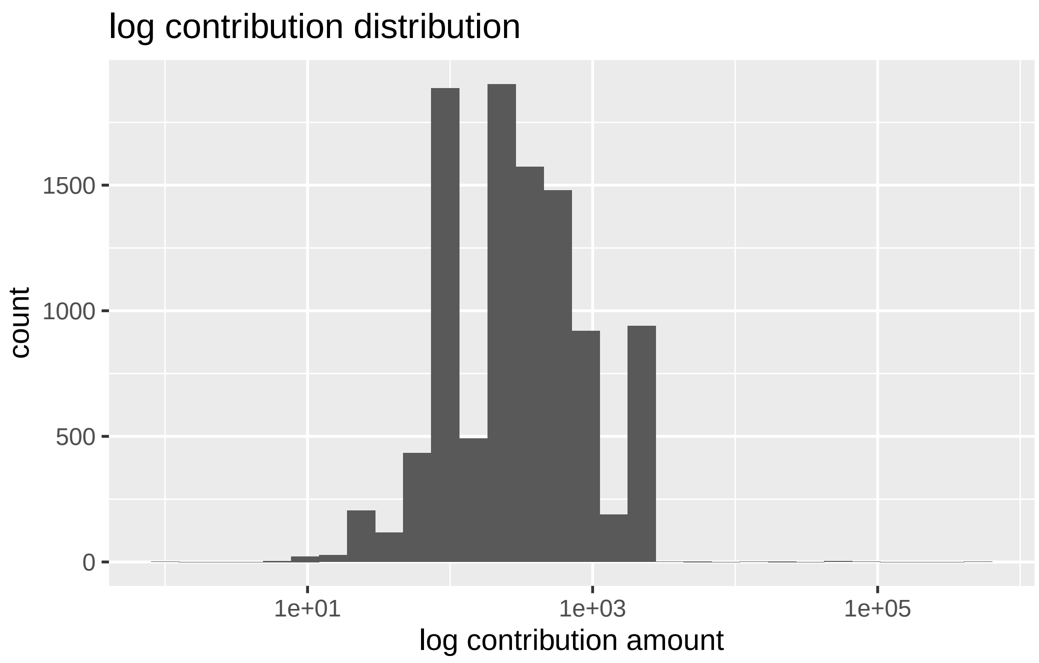
```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



- Looks like there is some very large outliers, trying log scale:

```
ggplot(data = camp_2014) +
  geom_histogram(aes(x = num_contribution_amount)) +
  scale_x_log10() +
  labs(title = 'log contribution distribution',
       x = 'log contribution amount')
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



- Found some outliers, now find them in the dataset:

```
camp_2014 |>
  arrange(-num_contribution_amount) |>
  slice_head(n=20)
```

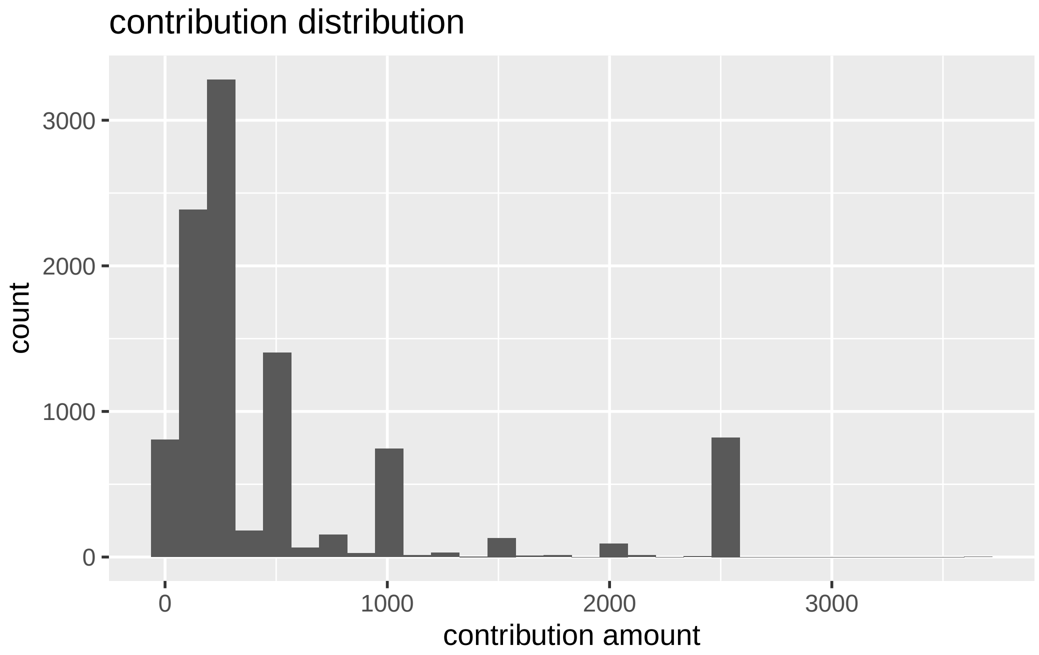
```
# A tibble: 20 x 14
   contributor~1 contr~2 contr~3 contr~4 contr~5 goods~6 contr~7 relat~8 presi~9
   <chr>         <chr>   <chr>   <chr>   <chr>   <chr>   <chr>   <chr>   <chr>
 1 Ford, Doug    <NA>    M9A 2C3 508224~ Moneta~ <NA>    Indivi~ Candid~ <NA>
 2 Ford, Rob     <NA>    M9A 3G9 78804.~ Moneta~ <NA>    Indivi~ Candid~ <NA>
 3 Ford, Doug    <NA>    M9A 2C3 50000   Moneta~ <NA>    Indivi~ Candid~ <NA>
 4 Ford, Rob     <NA>    M9A 3G9 50000   Moneta~ <NA>    Indivi~ Candid~ <NA>
 5 Ford, Rob     <NA>    M9A 3G9 50000   Moneta~ <NA>    Indivi~ Candid~ <NA>
 6 Goldkind, Ari <NA>    M5P 1P5 23623.~ Moneta~ <NA>    Indivi~ Candid~ <NA>
 7 Ford, Rob     <NA>    M9A 3G9 20000   Moneta~ <NA>    Indivi~ Candid~ <NA>
 8 Ford, Rob     <NA>    M9A 3G9 12210   Moneta~ <NA>    Indivi~ Candid~ <NA>
 9 Di Paola, Ro~ <NA>    M3H 2T1 6000    Moneta~ <NA>    Indivi~ Candid~ <NA>
10 Thomson, Sar~ <NA>    M4W 2X6 4425.5~ Moneta~ <NA>    Indivi~ Candid~ <NA>
11 kindred's Mu~ 723 Do~ M6H 2W7 3660    Goods/~ photog~ Corpor~ <NA>    Pharel~
12 Achber, Vern~ <NA>    M4N 3N6 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
13 Adam, Michael <NA>    M4W 3Y2 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
```

```
14 Aghaei, Saeid <NA>    M4N 3G1 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
15 Al Zaibak, M~ <NA>    M4V 2L7 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
16 Allan, David~ <NA>    M4X 1B2 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
17 Allen, Peter~ <NA>    M4T 1E2 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
18 Alper, Laura  <NA>    M4T 1B9 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
19 Alter, Robin  <NA>    M5N 2X6 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
20 Anderson, Ja~ <NA>    M4W 1X4 2500    Moneta~ <NA>    Indivi~ <NA>    <NA>
# ... with 5 more variables: authorized_representative <chr>, candidate <chr>,
#   office <chr>, ward <chr>, num_contribution_amount <dbl>, and abbreviated
#   variable names 1: contributors_name, 2: contributors_address,
#   3: contributors_postal_code, 4: contribution_amount,
#   5: contribution_type_desc, 6: goods_or_service_desc,
#   7: contributor_type_desc, 8: relationship_to_candidate,
#   9: president_business_manager
```

- I don't know much about how the mayoral campaign works, but it appears that the outliers are that candidates contributes to themselves.
- Remove the outliers by removing the cases where candidates contributes to themselves. This will remove a few more rows other than the outliers, but I think it is fair to remove all of them, because otherwise the remaining self-contributions cases in the dataset will be biased toward small values.

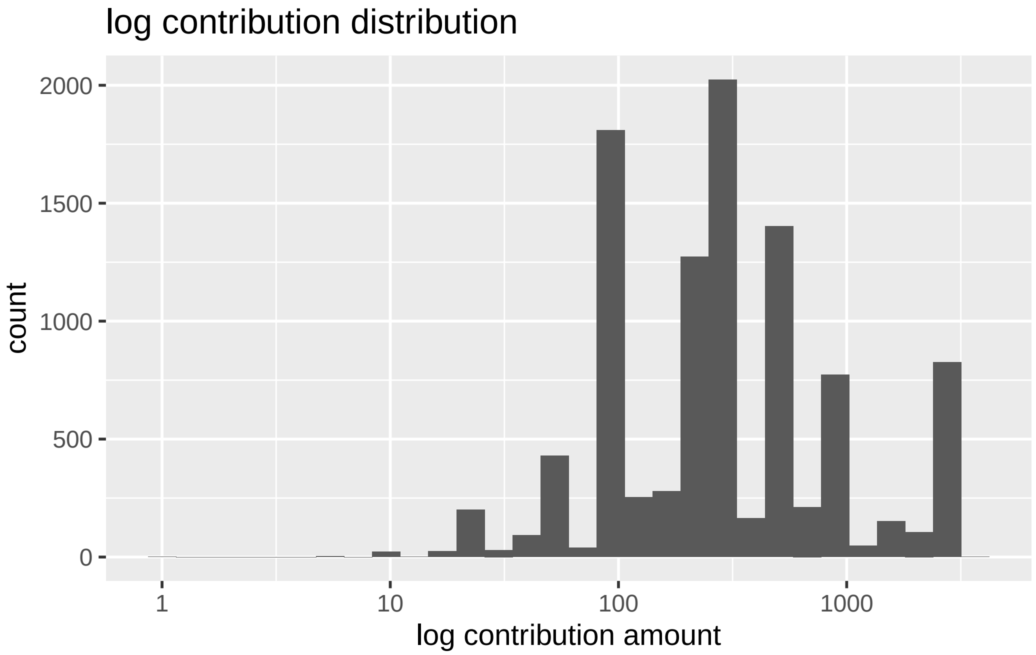
```
camp_2014 |>
  filter(relationship_to_candidate != "Candidate" | is.na(relationship_to_candidate)) |>
  ggplot() +
  geom_histogram(aes(x = num_contribution_amount)) +
  labs(title = 'contribution distribution',
       x = 'contribution amount')
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
camp_2014 |>
  filter(relationship_to_candidate != "Candidate" | is.na(relationship_to_candidate)) |>
  ggplot() +
  geom_histogram(aes(x = num_contribution_amount)) +
  scale_x_log10() +
  labs(title = 'log contribution distribution',
       x = 'log contribution amount')
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



- The majority of the contributions amount is between 100 to 1000.

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

    - total contributions
    - mean contribution
    - number of contributions

```
camp_2014 |>
  group_by(candidate) |>
  summarise(total_contributions = sum(num_contribution_amount)) |>
  arrange(-total_contributions) |>
  slice_head(n = 5)
```

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

```
camp_2014 |>
  group_by(candidate) |>
  summarise(mean_contributions = mean(num_contribution_amount)) |>
  arrange(-mean_contributions) |>
  slice_head(n = 5)
```

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

```
camp_2014 |>
  group_by(candidate) |>
  summarise(number_of_contributions = n()) |>
  arrange(-number_of_contributions) |>
  slice_head(n = 5)
```

```
# A tibble: 5 x 2
  candidate       number_of_contributions
  <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.

```
camp_2014 |>
  filter(relationship_to_candidate != "Candidate" | is.na(relationship_to_candidate)) |>
  group_by(candidate) |>
  summarise(total_contributions = sum(num_contribution_amount)) |>
  arrange(-total_contributions) |>
  slice_head(n = 5)
```

```
# A tibble: 5 x 2
  candidate     total_contributions
  <chr>                       <dbl>
1 Tory, John               2765369.
2 Chow, Olivia             1635766.
3 Ford, Doug                331173.
4 Stintz, Karen             242805
5 Ford, Rob                 174510.
```

```
camp_2014 |>
  filter(relationship_to_candidate != "Candidate" | is.na(relationship_to_candidate)) |>
  group_by(candidate) |>
  summarise(mean_contributions = mean(num_contribution_amount)) |>
  arrange(-mean_contributions) |>
  slice_head(n = 5)
```

```
# A tibble: 5 x 2
  candidate         mean_contributions
  <chr>                          <dbl>
1 Ritch, Carlie                  1887.
2 Sniedzins, Erwin               1867.
3 Tory, John                     1063.
4 Gardner, Norman                1000
5 Tiwari, Ramnarine              1000
```

```
camp_2014 |>
  filter(relationship_to_candidate != "Candidate" | is.na(relationship_to_candidate)) |>
  group_by(candidate) |>
  summarise(number_of_contributions = n()) |>
  arrange(-number_of_contributions) |>
  slice_head(n = 5)
```

```
# A tibble: 5 x 2
  candidate       number_of_contributions
  <chr>                             <int>
1 Chow, Olivia                       5707
2 Tory, John                         2601
3 Ford, Doug                          608
4 Ford, Rob                           531
5 Soknacki, David                     314
```

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

```
camp_2014 |>
  group_by(contributors_name) |>
  summarise(num_candidate_contributed = n_distinct(candidate)) |>
  filter(num_candidate_contributed > 1) |>
  nrow()
```

```
[1] 184
```

- 184 contributors gave money to more than one candidate.