Applied Problem Set 2

Earnest Salgado, Guillermo Antonio Trefogli Wong

05/05/2020

```
library(tidyverse)
library(lubridate)
knitr::opts_chunk$set(fig.width=6, fig.height=3)
```

This submission is my work alone and complies with the 30535 integrity policy.

Add your initials to indicate your agreement: **ES__GATW**

Add your collaborators: **ES__GATW**

Late coins used this pset: 2. Late coins left: 7 (ES), 6 (GATW).

1 Part I

Read this (https://features.propublica.org/driven-into-debt/chicago-ticket-debt-bankruptcy/) article and this (https://www.propublica.org/article/chicago-vehicle-sticker-law-ticket-price-hike-black-drivers-debt) shorter article. Melissa Sanchez will be our second guest speaker (Monday May 10 at 3:30 pm). If you are curious to learn more, this (https://www.propublica.org/series/driven-into-debt) page has all of the articles that ProPublica has done on this topic.

1.1 Read in one percent sample (10 points)

1. To help you get started, the repository contains the file parking_tickets_one_percent.csv which gives you a one percent sample of tickets. We constructed the sample by selecting ticket numbers that end in 01. How long does it take to read in this file? (Don't time your code with your watch, find a function to measure how long it takes the command to run.) Use test_that to check that there are 287458 rows.

```
start <- Sys.time()
parking_tix_1 <- read.csv("parking_tickets_one_percent.csv")
end <- Sys.time()

read_time <- end - start
read_time
```

```
## Time difference of 6.635452 secs
```

```
library(testthat)
test_that('data dimensions correct', {
    expect_equal(ncol(parking_tix_1), 24)
    expect_equal(nrow(parking_tix_1), 287458)
})
```

```
## Test passed
```

2. How many megabytes is the file? Using math, how large would you predict the full data set is?

We know the file gives us a one percent sample of the full dataset of tickets. By running code to calculate 138.5 Mb, we multiply it by 100 to predict the full data set size and get 13850 Mb as an estimated size.

```
print(object.size(parking_tix_1), units="Mb")
```

```
## 138.5 Mb
```

3. How are the rows ordered?

The rows are ordered in terms of ticket issue date from oldest to newest. The dates range from January 2007 to May 2018.

4. For each column, how many rows are NA? Write a parsimonious command which calculates this. You will not get credit for a command which writes out every variable name.

```
parking_tix_1 %>%
  select(everything()) %>%
  summarise_all(list(~sum(is.na(.))))
```

```
##   X ticket_number issue_date violation_location license_plate_number
## 1 0             0          0                  0                     0
##   license_plate_state license_plate_type zipcode violation_code
## 1                  97               2054   54115              0
##   violation_description unit unit_description vehicle_make fine_level1_amount
## 1                     0   29                0            0                   0
##   fine_level2_amount current_amount_due total_payments ticket_queue
## 1                  0                  0              0            0
##   ticket_queue_date notice_level hearing_disposition notice_number officer
## 1                 0        84068              259899             0       0
##   address
## 1       0
```

5. Three variables are missing much more frequently than the others. Why? (Hint: look at some rows and read the data dictionary written by ProPublica, inside the repository data_dictionary.txt)

The three variables that have the most missing values are 1) hearing_disposition, 2) notice_level, and 3) zipcode. This may be the outcome because there are many tickets not contested, and not registered.

1.2 Cleaning the data and benchmarking (10 points)

1. How many tickets were issued in tickets_1pct in 2017? How many tickets does that imply were issued in the full data in 2017? How many tickets are issued each year according to the ProPublica article? Do you think that there is a meaningful difference?

In 2017, 22,364 tickets were issued according to this one percent sample. This implies over a one-hundred percent sample, roughly 2,236,400 total tickets were issued that year. According to the ProPublica article, over 3 million tickets are issued by the City of Chicago each year. The ~764,000 difference could be pointed at this sample only representing parking tickets, while the 3 million total represents a wide range of parking, vehicle compliance and automated traffic camera violations.

```
parking_tix_1 %>%
  filter(issue_date >= as.Date("2017-01-01") & issue_date <= as.Date("2017-12-31")) %>%
  summarise(n = length(X))
```

```
##       n
## 1 22364
```

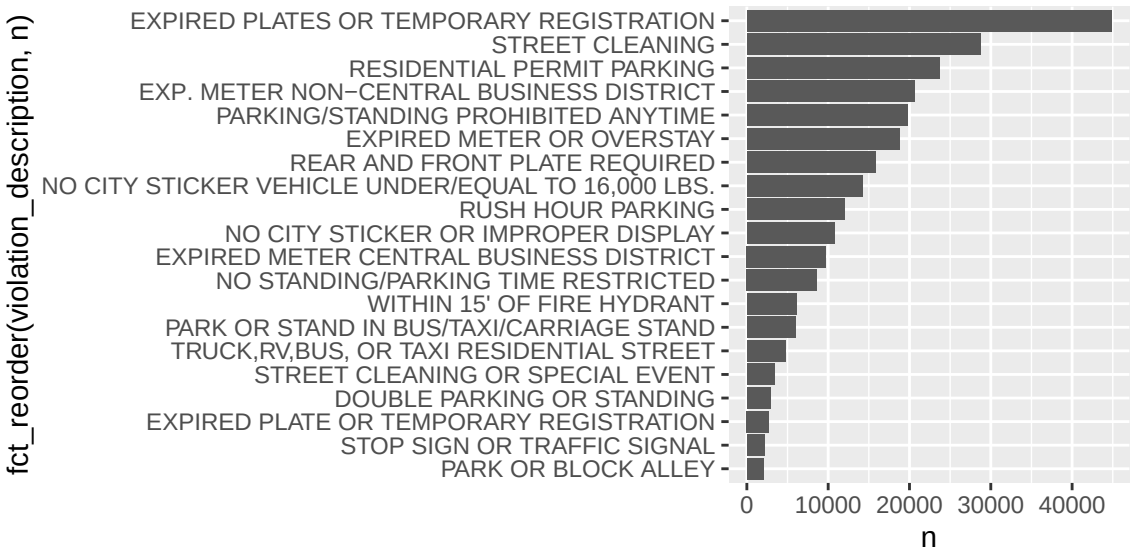
2. What are the top 20 most frequent violation types? Make a bar graph to show the frequency of these ticket types. Make sure to format the graph such that the violation descriptions are legible and no words are cut off.

```
tail(names(sort(table(parking_tix_1$violation_description))), 20)
```

```
##  [1] "PARK OR BLOCK ALLEY"
##  [2] "STOP SIGN OR TRAFFIC SIGNAL"
##  [3] "EXPIRED PLATE OR TEMPORARY REGISTRATION"
##  [4] "DOUBLE PARKING OR STANDING"
##  [5] "STREET CLEANING OR SPECIAL EVENT"
##  [6] "TRUCK,RV,BUS, OR TAXI RESIDENTIAL STREET"
##  [7] "PARK OR STAND IN BUS/TAXI/CARRIAGE STAND"
##  [8] "WITHIN 15' OF FIRE HYDRANT"
##  [9] "NO STANDING/PARKING TIME RESTRICTED"
## [10] "EXPIRED METER CENTRAL BUSINESS DISTRICT"
## [11] "NO CITY STICKER OR IMPROPER DISPLAY"
## [12] "RUSH HOUR PARKING"
## [13] "NO CITY STICKER VEHICLE UNDER/EQUAL TO 16,000 LBS."
## [14] "REAR AND FRONT PLATE REQUIRED"
## [15] "EXPIRED METER OR OVERSTAY"
## [16] "PARKING/STANDING PROHIBITED ANYTIME"
## [17] "EXP. METER NON-CENTRAL BUSINESS DISTRICT"
## [18] "RESIDENTIAL PERMIT PARKING"
## [19] "STREET CLEANING"
## [20] "EXPIRED PLATES OR TEMPORARY REGISTRATION"
```

```
top20_violations <- parking_tix_1 %>%
  group_by(violation_description) %>%
  count(violation_description) %>%
  arrange(desc(n)) %>%
  head(20)

ggplot(top20_violations) +
  geom_col(aes(x = fct_reorder(violation_description, n), y = n)) +
  coord_flip()
```



1.3 Joins - unit (10 points)

The data tell us what unit of city government issued the ticket, but we need to merge on a crosswalk.

1. For how many tickets is unit missing?

```
# NOTE: If we filter to find the string "NA", we get zero:

# missing_units <- parking_tix_1 %>%
#  filter(unit == "NA") %>%
#  count(n())

parking_tix_1 %>%
  select(unit) %>%
  summarise(sum(is.na((.))))
```

```
##   sum(is.na((.)))
## 1              29
```

2. Read in unit_key.csv. How many units are there?

```
unit_key <- read.csv("unit_key.csv", skip = 2)

unit_key %>%
  summarise(total_units = n_distinct(Reporting.District))
```

```
##   total_units
## 1         385
```

3. Join unit key to the tickets data. How many rows in the tickets data have a match in the unit table? How many rows are unmatched? How many rows in the unit table have a match in the tickets data? How many do not?

```
# use left join
unit_key1 <- unit_key %>%
  rename(unit = Reporting.District) %>%
  select(-X, -Department.Category.1) %>%
  mutate(unit = as.numeric(unit))
```

```
## Warning: Problem with `mutate()` input `unit`.
## i NAs introduced by coercion
## i Input `unit` is `as.numeric(unit)`.
```

```
## Warning in mask$eval_all_mutate(dots[[i]]): NAs introduced by coercion
```

```
parking_tix_2 <- parking_tix_1 %>%
  left_join(unit_key1, by = "unit")
```

```
library(dplyr)
# identifying rows that exist in the tickets data, but not in the unit table
anti_join(parking_tix_2, unit_key1, by = "unit")

# identifying rows that exist in the unit table but not in the tickets data
anti_join(unit_key1, parking_tix_2, by = "unit")
```

4. Who issues more tickets - Department of Finance or Chicago Police? Within Chicago Police, what are the top 5 departments that are issuing the most tickets? Be careful what your group by here and avoid columns with ambiguities.

We run code below that shows us that Department of Finance is responsible for the most tickets in our one percent sample, with 143909. CPD is close in second with 127,223 tickets. DOF and CPD combine to issue ~94% of all tickets in our dataset. Chicago Parking Meter, Miscellaneous departments, and Streets and Sanitation are the third, fourth, and fifth most in terms of issuing tickets. SERCO is a notable mention issuing 37,426 tickets which would be third most on our department list. The company, however, is under DOF.

```
parking_tix_2 %>%
  group_by(Department.Name) %>%
  summarise(n = n()) %>%
  arrange(desc(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 9 x 2
##   Department.Name      n
##   <chr>            <int>
## 1 DOF             143909
## 2 CPD             120712
## 3 Miscellaneous    16442
## 4 CPD-Other         3894
## 5 CPD-Airport       2617
## 6 CTA                 58
## 7 Red Light           58
## 8 Speed               29
## 9 Unidentified        29
```

```
parking_tix_2 %>%
  group_by(Department.Category) %>%
  summarise(n = n()) %>%
  arrange(desc(n))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 10 x 2
##    Department.Category        n
##    <chr>                  <int>
##  1 CPD                   127223
##  2 DOF                   106483
##  3 SERCO                  37426
##  4 Chicago Parking Meter   8064
##  5 Miscellaneous/Other     5063
##  6 Streets and San         2709
##  7 LAZ                      635
##  8 CTA                       58
##  9 Red light                 58
## 10 Speed                     29
```

1.4 Joins - ZIP code (15 points)

1. Download recent census data by ZIP for Chicago with population, share black and median household income. chi_zips.csv

```
# Census slides
library(tidycensus)

# DELETE BEFORE SUBMIT

census_data <- get_acs(
  geography = "zip code tabulation area",
  variables = c(population = "B01003_001",
                black = "B02001_003",
                medincome = "B19013_001"),
          year = 2018
  ) %>%
  arrange(NAME)
```

```
## Getting data from the 2014-2018 5-year ACS
```

```
census_data <- census_data %>%
  separate(NAME, into = c("ZCTA5", "zipcode"),
  sep = " ") %>%
  mutate(zipcode = as.numeric(zipcode)) %>%
  pivot_wider(names_from = variable, values_from = c(estimate, moe))
```

2. Clean vehicle registration ZIP and then join the Census data to the tickets data

```
chi_zips <- read.csv("chi_zips.csv")

parking_tix_2 <- parking_tix_2 %>%
  mutate(zipcode = strtrim(parking_tix_2$zipcode, 5))

tickets1pct <- parking_tix_2 %>%
  mutate(zipcode = as.numeric(zipcode)) %>%
  left_join(census_data, by = "zipcode")
```

3. Replicate the key findings in the ProPublica by ranking ZIPs by the number of unpaid tickets per resident by ZIP. What are the names of the three neighborhoods with the most unpaid tickets?

The three zip codes with the most unpaid tickets per resident are 60623, 60620, and 60651. Using a google search for the appropriate link, these neighborhoods are named South Lawndale, Auburn Gresham, and Humboldt Park respectively. (link: https://www.unitedstateszipcodes.org/60623/)

```
tickets1pct_unpaid <- tickets1pct %>%
  filter(ticket_queue != "Paid",  ticket_queue != "Dismissed") %>%
  group_by(zipcode, ticket_queue, GEOID) %>%
  summarise(n = n()) %>%
  arrange(desc(n)) %>%
  head(4)
```

```
## `summarise()` regrouping output by 'zipcode', 'ticket_queue' (override with `.groups` argument)
```

```
tickets1pct_unpaid <- tickets1pct_unpaid %>% filter((!is.na(zipcode)))
```

4. (extra credit) Make #3 into a map

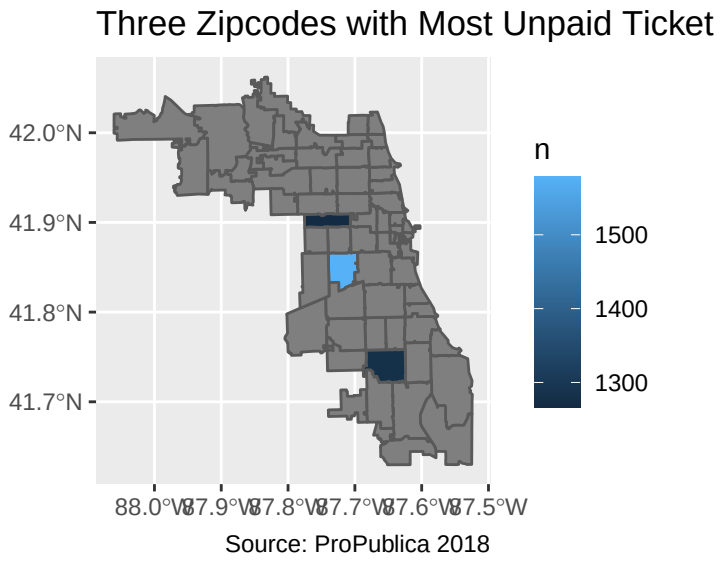
```
map <- get_acs(state = "IL", geography = "zcta",
                  variables = "B19013_001", geometry = TRUE) %>%
       separate(NAME, into = c("ZCTA5", "zipcode"),
       sep = " ") %>%
       mutate(zipcode = as.numeric(zipcode))
```

```
##   |                                                                              |
```

```
map <- left_join(map, tickets1pct_unpaid, by = "zipcode")

map2 <- map %>%
  filter(zipcode%in%chi_zips$ï..ZIP)

map2  %>%
  ggplot(aes(fill = n)) +
  geom_sf() +
  labs(title = "Three Zipcodes with Most Unpaid Ticket",
       caption = "Source: ProPublica 2018")
```



2 Part II

2.1 Understanding the structure of the data (20 points)

1. Most violation types double in price if unpaid. Does this hold for all violations? If not, find all violations with at least 100 citations that do not double. How much does each ticket increase if unpaid?

No, not all violations double in price if unpaid. We found four violations with at least 100 citations that do not double: Park or Block Alley, Disabled Parking Zone, Smoked/Tinted Windows Parked/Standing, and Block Access/Alley/Driveway/Firelane.

```
non_double_violations <- parking_tix_2 %>%
  filter(fine_level2_amount < 2*fine_level1_amount) %>%
  group_by(violation_description) %>%
  summarise(n = n()) %>%
  arrange(desc(n)) %>%
  filter(n > 100)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

Next, lets examine how much do tickets in our sample increase if left unpaid. By simply looking at the fine level 1 and 2 amounts compared to the current amount due, there seems to be a lot of variation. We quantified these observations by calculating fine ratio and percent increases. We confirmed most violation types double in price if unpaid. We also noted that level 2 fines varied in how they increased: either being a quarter, half, two-thirds, or double the price of level 1. Finally, there are also instances where people owe current balances that are more than 200-300% of the original fine!

```
tickets1pct_unpaid2 <- tickets1pct %>%
  filter(ticket_queue != "Paid",  ticket_queue != "Dismissed") %>%
  select(ticket_queue, fine_level1_amount, fine_level2_amount, current_amount_due) %>% mutate(
    fine_ratio_increase=round(fine_level2_amount/fine_level1_amount,2),
    fine_percent_increase=round((current_amount_due-fine_level1_amount)/(fine_level1_amount)*100,0))
```

```
tickets1pct_unpaid2 %>%
  group_by(ticket_queue, fine_ratio_increase) %>%
  summarise(n = n()) %>%
  arrange(desc(n))
```

```
## `summarise()` regrouping output by 'ticket_queue' (override with `.groups` argument)
```

```
## # A tibble: 17 x 3
## # Groups:   ticket_queue [5]
##    ticket_queue fine_ratio_increase     n
##    <chr>                      <dbl> <int>
##  1 Notice                         2 38661
##  2 Define                         2 26820
##  3 Bankruptcy                     2  3128
##  4 Court                          2   564
##  5 Notice                      1.67   355
##  6 Define                      1.67   278
##  7 Notice                      1.25   132
##  8 Notice                      1       91
##  9 Define                      1.25    80
## 10 Hearing Req                 2       56
## 11 Define                      1       50
## 12 Bankruptcy                  1.67    31
## 13 Notice                      1.55     8
## 14 Bankruptcy                  1        7
## 15 Bankruptcy                  1.25     7
## 16 Define                      1.55     4
## 17 Bankruptcy                  1.55     1
```

```
tickets1pct_unpaid2 %>%
  group_by(ticket_queue, fine_ratio_increase) %>%
  distinct(fine_percent_increase) %>%
  arrange(desc(fine_percent_increase))
```

```
## # A tibble: 479 x 3
## # Groups:   ticket_queue, fine_ratio_increase [17]
##    ticket_queue fine_ratio_increase fine_percent_increase
##    <chr>                      <dbl>                 <dbl>
##  1 Notice                         2                   354
##  2 Notice                         2                   310
##  3 Notice                         2                   280
##  4 Bankruptcy                     2                   236
##  5 Notice                         2                   227
##  6 Notice                         2                   226
##  7 Bankruptcy                     2                   226
##  8 Notice                         2                   213
##  9 Notice                         2                   212
## 10 Notice                         2                   208
## # ... with 469 more rows
```

2. Many datasets implicitly contain information about how a case can progress. Draw a diagram explaining the process of moving between the different values of notice_level (if you draw it on paper, take a

picture and include the image using knitr::include_graphics). Draw a second diagram explaining the different values of ticket_queue. If someone contests their ticket and is found not liable, what happens to notice_level and to ticket_queue? Include this in your drawings.

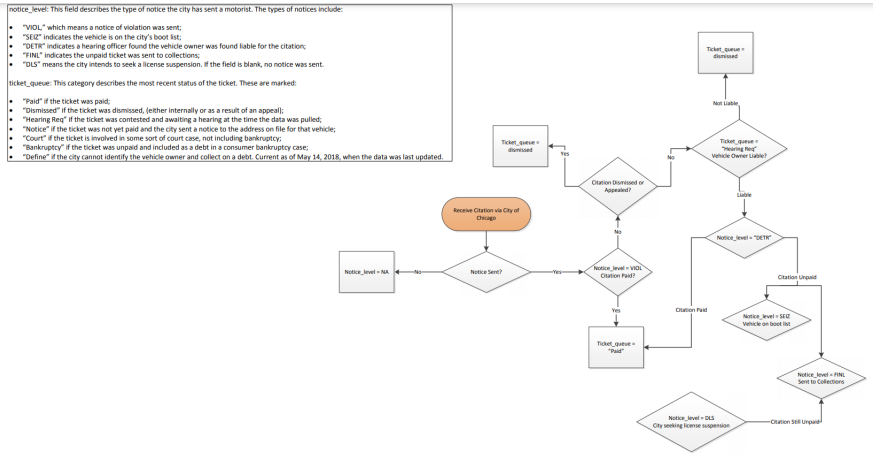


Figure 1: Diagram 1

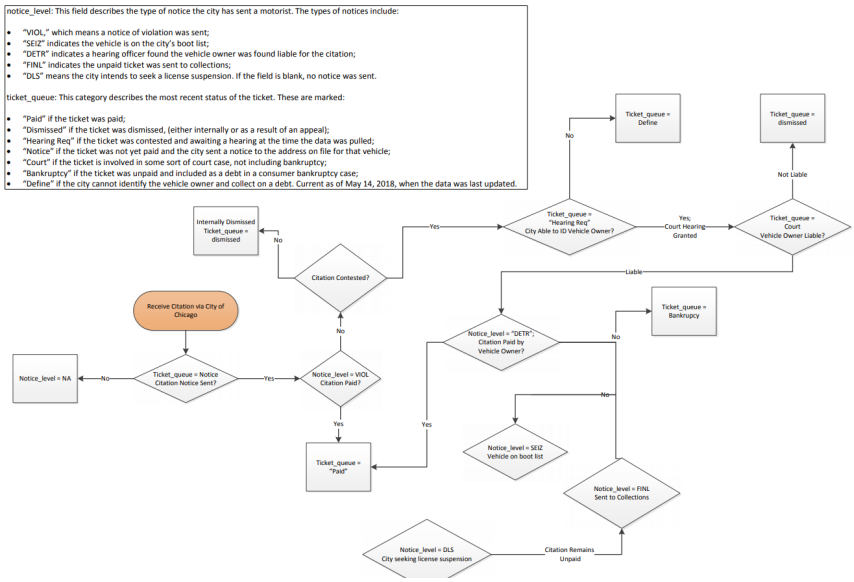


Figure 2: Diagram 2

3. Are any violation descriptions associated with multiple violation codes? If so, which descriptions have multiple associated codes and how many tickets are there in each description-code pair? (Hint: this can be done in just four lines of code)

```
tickets1pct %>%
  group_by(violation_description, violation_code) %>%
  summarise(n = n()) %>%
  arrange(violation_description)
```

```
## `summarise()` regrouping output by 'violation_description' (override with `.groups` argument)
```

```
## # A tibble: 124 x 3
## # Groups:   violation_description [119]
##    violation_description                   violation_code     n
##    <chr>                                   <chr>          <int>
##  1 2 REAR TRAILER LAMPS REQ'D VISIBLE 500' 0976050E           4
##  2 20'OF CROSSWALK                         0964100F         393
##  3 3-7 AM SNOW ROUTE                       0964060          827
##  4 3-7 AM SNOW ROUTE                       0964060B          12
##  5 ABANDONED VEH. FOR 7 DAYS OR INOPERABLE 0980110A        1104
##  6 BACK-UP LAMP LIT DURING OPERATION       0976070C           2
##  7 BLOCK ACCESS/ALLEY/DRIVEWAY/FIRELANE    0964100C        1579
##  8 BLOCK ALLEY                             0964130B          20
##  9 BRAKES REQUIRED DURING OPERATION        0976010A           1
## 10 BRAKES REQUIRED IN GOOD WORKING ORDER   0976020E           1
## # ... with 114 more rows
```

4. Are there any violation codes associated with multiple violation descriptions? If so, which codes have multiple associated descriptions and how many tickets are there in each description-code pair?

```
tickets1pct %>%
  group_by(violation_code, violation_description) %>%
  summarise(n = n()) %>%
  arrange(desc(violation_code))
```

```
## `summarise()` regrouping output by 'violation_code' (override with `.groups` argument)
```

```
## # A tibble: 124 x 3
## # Groups:   violation_code [116]
##    violation_code violation_description                          n
##    <chr>          <chr>                                      <int>
##  1 0980130C       PARK IN CITY LOT OVER 30 DAYS                  9
##  2 0980130B       PARK IN CITY LOT WHEN CLOSED                 225
##  3 0980130A       FAIL TO PAY OR OUTSIDE SPACE IN CITY LOT      90
##  4 0980120B       NO PARK IN PRIVATE LOT                       378
##  5 0980120A       NO PARK IN PUBLIC LOT                        119
##  6 0980110B       HAZARDOUS DILAPIDATED VEHICLE                148
##  7 0980110B       HAZARDOUS DILAPITATED VEHICLE                298
##  8 0980110A       ABANDONED VEH. FOR 7 DAYS OR INOPERABLE     1104
##  9 0980095        EXCESSIVE DIESEL POWERED VEHICLE ENGINE RUNNING   4
## 10 0980080C       PARK VEHICLE TO SELL MERCHANDISE             12
## # ... with 114 more rows
```

5. Review the 50 most common violation descriptions. Do any of them seem to be redundant? If so, can you find a case where what looks like a redundancy actually reflects the creation of a new violation code?

```
# Finding the 50 most common:
(descrip_50 <- tickets1pct %>%
  group_by(violation_description, violation_code) %>%
  summarise(n = n()) %>%
  arrange(desc(n)) %>%
  head(50))
```

```
## `summarise()` regrouping output by 'violation_description' (override with `.groups` argument)
```

```
## # A tibble: 50 x 3
## # Groups:   violation_description [50]
##    violation_description                              violation_code     n
##    <chr>                                              <chr>          <int>
##  1 EXPIRED PLATES OR TEMPORARY REGISTRATION           0976160F       44861
##  2 STREET CLEANING                                    0964040B       28712
##  3 RESIDENTIAL PERMIT PARKING                         0964090E       23703
##  4 EXP. METER NON-CENTRAL BUSINESS DISTRICT           0964190A       20600
##  5 PARKING/STANDING PROHIBITED ANYTIME                0964150B       19753
##  6 EXPIRED METER OR OVERSTAY                          0964190        18766
##  7 REAR AND FRONT PLATE REQUIRED                      0976160A       15839
##  8 NO CITY STICKER VEHICLE UNDER/EQUAL TO 16,000 LBS. 0964125B       14256
##  9 RUSH HOUR PARKING                                  0964080A       11965
## 10 NO CITY STICKER OR IMPROPER DISPLAY                0964125        10758
## # ... with 40 more rows
```

```
# Finding redundancy:
descrip_50 %>%
  filter(str_detect(violation_description, "NO CITY STICKER"))
```

```
## # A tibble: 2 x 3
## # Groups:   violation_description [2]
##   violation_description                              violation_code     n
##   <chr>                                              <chr>          <int>
## 1 NO CITY STICKER VEHICLE UNDER/EQUAL TO 16,000 LBS. 0964125B       14256
## 2 NO CITY STICKER OR IMPROPER DISPLAY                0964125        10758
```

2.2 Revenue increase from "missing city sticker" tickets (35 points)

Some of the other articles on the Propublica website discuss an increase in the dollar amount of the ticket for not having a city sticker.

1. What was the old violation code and what is the new violation code? How much was the cost of an initial offense under each code? (You can ignore the ticket for a missing city sticker on vehicles over 16,000 pounds.)

The old violation code was 0964125 and the new one is 0964125B. The initial offense under the old one was 120 and it is 200 foe the new one.

```r
missing_city <- tickets1pct %>%
  filter(str_detect(violation_description, "NO CITY STICKER"))

missing_city %>% group_by(violation_code, violation_description) %>% summarise(n=n())
```

```
## `summarise()` regrouping output by 'violation_code' (override with `.groups` argument)
```

```
## # A tibble: 4 x 3
## # Groups:   violation_code [4]
##   violation_code violation_description                                n
##   <chr>          <chr>                                            <int>
## 1 0964125        NO CITY STICKER OR IMPROPER DISPLAY                10758
## 2 0964125B       NO CITY STICKER VEHICLE UNDER/EQUAL TO 16,000 LBS. 14256
## 3 0964125C       NO CITY STICKER VEHICLE OVER 16,000 LBS.             131
## 4 0976170        NO CITY STICKER OR IMPROPER DISPLAY                   15
```

```r
missing_city  <-  missing_city %>% filter(violation_code != "0964125C")

missign <- missing_city %>%
  select(violation_code, issue_date, fine_level1_amount, everything()) %>%
  arrange(issue_date)

missign %>% head(1)
```

```
##   violation_code          issue_date fine_level1_amount  X ticket_number
## 1        0964125 2007-01-01 10:51:00                120 15      51262101
##   violation_location
## 1    4325 N BROADWAY
##                                          license_plate_number
## 1 33f570a3f9850393f316b97d476452faf76846ab6588e5980932acc9b2d237ea
##   license_plate_state license_plate_type zipcode
## 1                  IL                PAS   60613
##                violation_description unit unit_description vehicle_make
## 1 NO CITY STICKER OR IMPROPER DISPLAY   23              CPD         FORD
##   fine_level2_amount current_amount_due total_payments ticket_queue
## 1                240                   0            240         Paid
##   ticket_queue_date notice_level hearing_disposition notice_number officer
## 1        2007-08-31         SEIZ                <NA>    5078012670   17904
##                         address Department.Name Department.Description
## 1 4300 n broadway, chicago, il             CPD        3600 N. Halsted
##   Department.Category Reporting.District.1   GEOID ZCTA5 estimate_population
## 1                 CPD                  23 1760613 ZCTA5               50113
##   estimate_black estimate_medincome moe_population moe_black moe_medincome
## 1           2903              82321           1673       512          3210
```

```r
missign %>% tail(1)
```

```
##       violation_code          issue_date fine_level1_amount      X
## 25029       0964125B 2018-05-14 14:30:00                200 287453
##       ticket_number violation_location
## 25029      9.19e+09   1601 W CULLERTON
```

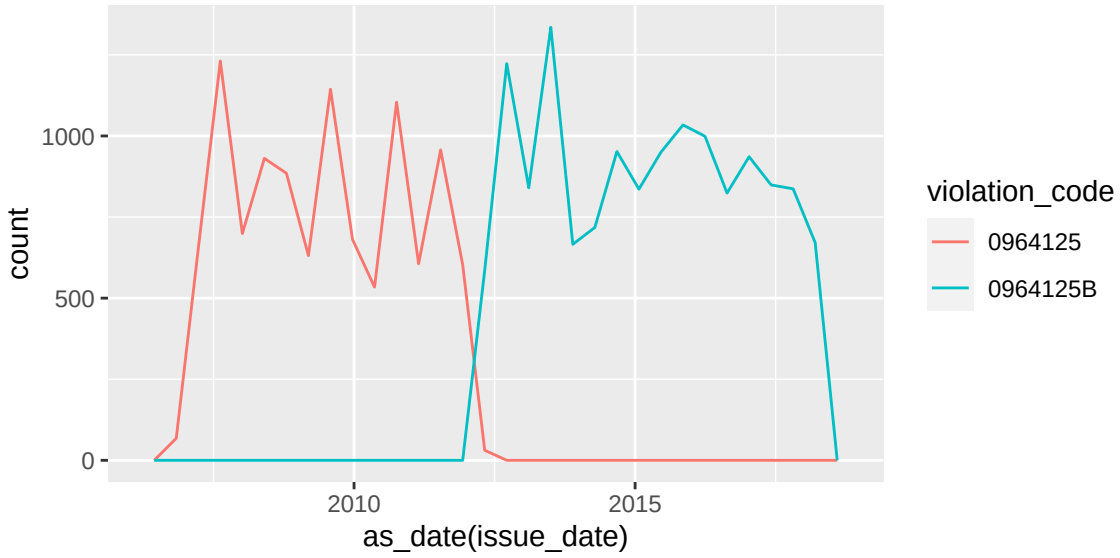
```
##                                                 license_plate_number
## 25029 593e6eeec22ad12b8e0d08c4d7bbd3de9c0688277d208174123e65513b66ffc0
##       license_plate_state license_plate_type zipcode
## 25029                  IL                PAS   60608
##                                violation_description unit unit_description
## 25029 NO CITY STICKER VEHICLE UNDER/EQUAL TO 16,000 LBS.  498              DOF
##       vehicle_make fine_level2_amount current_amount_due total_payments
## 25029         HOND               400                200              0
##       ticket_queue ticket_queue_date notice_level hearing_disposition
## 25029       Notice        2018-05-17         <NA>                <NA>
##       notice_number officer                       address Department.Name
## 25029     5.21e+09     826 1600 w cullerton, chicago, il             DOF
##       Department.Description Department.Category Reporting.District.1   GEOID
## 25029  Department of Finance                 DOF                 498 1760608
##       ZCTA5 estimate_population estimate_black estimate_medincome
## 25029 ZCTA5               79205          13982              44043
##       moe_population moe_black moe_medincome
## 25029           2460      1288          2459
```

2. Combining the two codes, how have the number of missing sticker tickets evolved over time?

```
missign_ev <- tickets1pct %>%
  filter(violation_code == "0964125" | violation_code == "0964125B")

missign_ev %>% ggplot(aes(x = as_date(issue_date))) +
  geom_freqpoly(aes(color = violation_code))
```

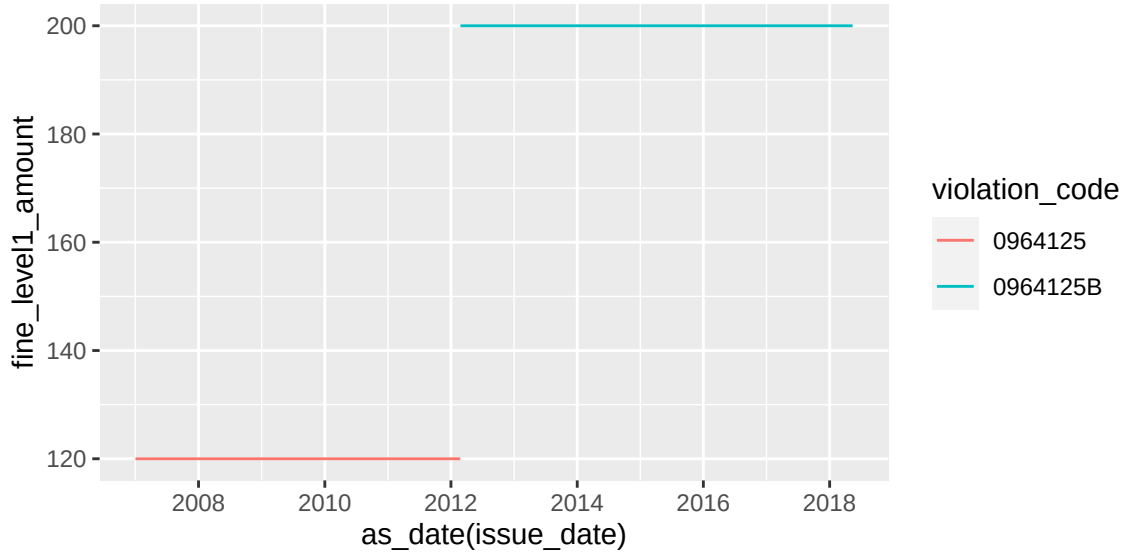
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



3. Using the dates on when tickets were issued, when did the price increase occur?

It was with the first type of "0964125B" violation code, which occurred in 2012-02-25 (at 02:00:00).

```
missign_ev %>% ggplot(aes(x = as_date(issue_date), y = fine_level1_amount)) +
  geom_line(aes(color = violation_code))
```




```
missign_ev %>% filter(violation_code == "0964125B") %>%
  arrange(issue_date) %>%
  head(1)
```

```
##        X ticket_number          issue_date violation_location
## 1 138605      61529401 2012-02-25 02:00:00  10130 S EVERGREEN
##                                             license_plate_number
## 1 f12ddc775c18cbbe06bb1630d0aea0db009bb63dbcfce9bd6201bd3b223a80f5
##   license_plate_state license_plate_type zipcode violation_code
## 1                  IL                PAS   60652       0964125B
##                             violation_description unit unit_description
## 1 NO CITY STICKER VEHICLE UNDER/EQUAL TO 16,000 LBS.   22              CPD
##   vehicle_make fine_level1_amount fine_level2_amount current_amount_due
## 1         CHEV                200                 400              -0.02
##   total_payments ticket_queue ticket_queue_date notice_level
## 1            488         Paid        2013-05-29         FINL
##   hearing_disposition notice_number officer                         address
## 1                <NA>      5.13e+09    8687 10100 s evergreen, chicago, il
##   Department.Name Department.Description Department.Category
## 1             CPD       1900 W. Monterey                 CPD
##   Reporting.District.1   GEOID ZCTA5 estimate_population estimate_black
## 1                   22 1760652 ZCTA5               43907          20699
##   estimate_medincome moe_population moe_black moe_medincome
## 1              68425           1398      1198          4563
```

4. The City Clerk said the price increase would raise by $16 million per year. Using only the data available in the calendar year prior to the increase, how much of a revenue increase should she have projected? Assume that the number of tickets of this type issued afterward would be constant and you can assume that there are no late fees or collection fees, so a ticket is either paid at is face value or is never paid.

We already know that the increase in the fine is from 120 to 200, only considering level 1 fine. Then, to estimate the increase in the raise in revenues, we will employ only level 1 fines increase, and will use the amount of the following variables in 2011, which is the year previous to the increase in the fine amount: number of fines, total fine amount and ratio of payment:

```
tickets_before <-  tickets1pct %>%
  mutate(year = str_sub(issue_date, 1, 4)) %>%
  filter(violation_code == "0964125" |
           violation_code == "0964125B"
         | violation_code == "0976170")

tickets_before <- tickets_before %>% filter(year == "2011")

tickets_before %>% summarise(n=n(),
            amount = sum(total_payments + current_amount_due),
            paid = sum(total_payments),
            ratio_pay = (sum(total_payments)/(sum(total_payments + current_amount_due))))
```

```
##      n   amount     paid ratio_pay
## 1 1935 372409.5 212393.1 0.5703213
```

Then, as counterfactual we get that, under the old fine amount, the total number of this type of fines was 1935, the annual amount in fines was 372,409.5 and the ratio of payment was 0.5703213. After adjusted by the ratio of payment, the collected amount in the year was 212,393.1 dollars. Finally, after adjusting it in terms of percentages, it turns 21,239,300.1 dollars.

Second, considering the same amount of annual fines (1935), at the cost of 200 dollars, we can estimate an annual amount in this type of fine of 387,000 dollars, which after adjusting in terms of ratio of payment turns 220,714.3. Finally, after adjusting it in terms of percentages, it turns 22,071,434 dollars.

Finally, calculating the raise implies contrasting the annual amount collected under the old fine amount with that projected under the new fine amount: 22,071,434 - 21,239,300.1, which is 832,133.9 dollars. Therefore, the projected increase in the revenues is by fay smaller than that indicated by the city clerk.

5. What happened to repayment rates on this type of ticket in the calendar year after the increase went into effect? If the City had not started issuing more of these tickets, what would its change in revenue have been?

As we have seen before, the payment rate in the year prior to the increase in the fine amount was 0.5703213.

To identify what the revenues would be had the city have not increased the number of fines, we need to see, first, what were the number of this type of tickets, the amount collected, and contrast it to the counterfactual scenario considering the same rate of payment.

```
tickets_before <-  tickets1pct %>%
  mutate(year = str_sub(issue_date, 1, 4)) %>%
  filter(violation_code == "0964125" |
           violation_code == "0964125B"
         | violation_code == "0976170")
```

```
tickets_before <- tickets_before %>% filter(year == "2012")

tickets_before %>% summarise(n=n(),
            amount = sum(total_payments + current_amount_due),
            paid = sum(total_payments),
            ratio_pay = (sum(total_payments)/(sum(total_payments + current_amount_due))))
```

```
##      n   amount     paid ratio_pay
## 1 2192 674172.4 350870.8 0.5204467
```

Considering the amount of tickets in 2011 and the same ratio of payment, it would had been: 1935 (number of tickets 2011) * 200 (new fine amount) * 0.5703213 (2011 ratio of payment), which is: 220,714.3. Finally, after adjusting it in terms of percentages, it turns 22,071,400.3 dollars. Then, in contrast to the amount collected in 2011, which was 21,239,300.1 dollars, they nuder this scenario, they would have collected 832,100.2 dollars more in revenues.

6. Make a plot with the repayment rates on not city sticker tickets and a vertical line at when the new policy was introduced. Interpret.

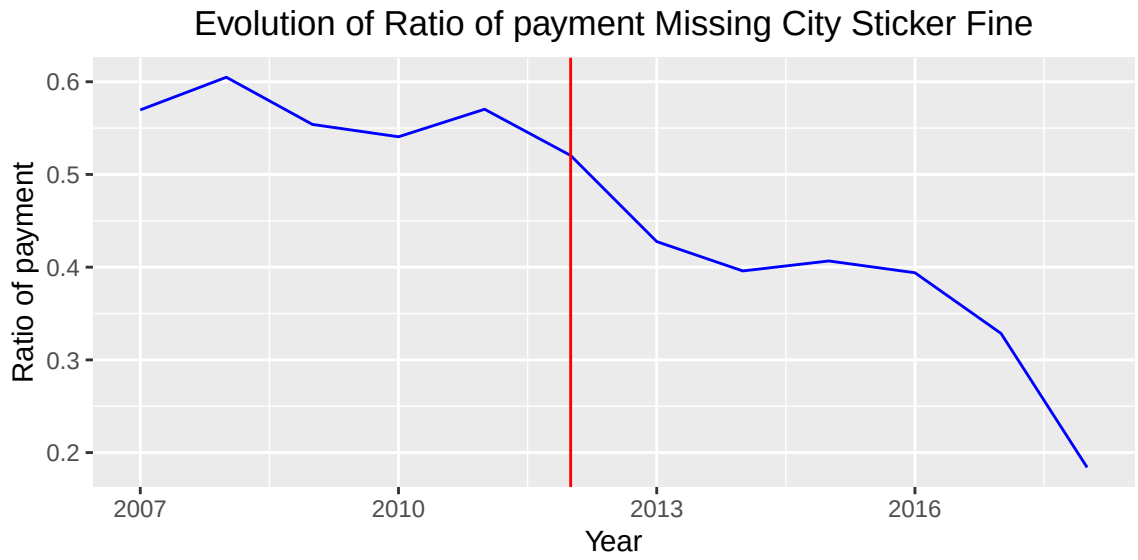
```
missign_repay <- tickets1pct %>%
  filter(violation_code == "0964125" | violation_code == "0964125B" | violation_code == "0976170")

missign_repay <- missign_repay %>% mutate(year = str_sub(issue_date, 1, 4))

missign_repay <- missign_repay %>%
  group_by(year) %>%
  summarise(pay = sum(total_payments),
            due = sum(current_amount_due),
            ratio = pay/(pay+due))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
missign_repay %>%
  ggplot(aes(x = as.numeric(year), y = ratio)) +
  geom_line(color = "blue") +
  labs(x="Year", y="Ratio of payment") +
  ggtitle("Evolution of Ratio of payment Missing City Sticker Fine") +
  theme(plot.title = element_text(hjust = 0.5)) +
geom_vline(aes(xintercept = 2012), color="red")
```



After the increase in the amount of the fine of missing city sticker, the rate of payment in this type of fine was already falling. Once the new policy started, the rate of payment decline increased until reaching a very low level in 2018.

7. Still focusing on the period before the policy change, suppose that the City Clerk were committed to getting revenue from tickets rather than other sources. What ticket types would you as an analyst have recommended she increase and why? Name up to three ticket types. Assume there is no behavioral response (i.e. people continue to commit violations at the same rate and repay at the same rate), but consider both ticket numbers and repayment rates.

```
fine_option <-  tickets1pct %>% mutate(year = str_sub(issue_date, 1, 4))

fine_option <- fine_option %>% filter(year == "2012")

fine_option %>%
  group_by(violation_description) %>%
  summarise(pay = sum(total_payments),
            due = sum(current_amount_due),
            ratio = pay/(pay+due)) %>%
  filter(ratio == "1") %>%
  arrange(desc(pay)) %>%
  head(3)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 3 x 4
##   violation_description               pay   due ratio
##   <chr>                             <dbl> <dbl> <dbl>
## 1 PARK IN FIRE LANE                  1116     0     1
## 2 PARK OR STAND IN VIADUCT/UNDERPASS  871     0     1
## 3 OUTSIDE DIAGONAL MARKINGS           666     0     1
```

8. In the previous question, the City Clerk was only optimizing gross revenue. Melissa Sanchez argue that ticketing is inherently regressive. Let's say the City Clerk took this critique to heart and determined

to raise ticket prices for violations that would affect households in high income zip codes more than low income zip codes.

a. What ticket types would you as an analyst recommend she increase and why? Make a data visualization to support your argument.

```
census_2 <- census_data %>%
    mutate(quantile = as.character(ntile(estimate_medincome, 10)))

census_2 <- census_2 %>% mutate(quantile_2 = str_c("quantile ", quantile))

high_fine <- parking_tix_2 %>%
  mutate(zipcode = as.numeric(zipcode)) %>%
  left_join(census_2, by = "zipcode")
```

```
## Warning: Problem with `mutate()` input `zipcode`.
## i NAs introduced by coercion
## i Input `zipcode` is `as.numeric(zipcode)`.
```

```
## Warning in mask$eval_all_mutate(dots[[i]]): NAs introduced by coercion
```

```
high_fine <-  high_fine %>% mutate(year = str_sub(issue_date, 1, 4))

high_fine <- high_fine %>% filter(year == "2017")

# choosing the 10 most important fines in terms of revenues

revenues_list <-  high_fine %>%
  group_by(violation_description) %>%
  summarise(amount = sum(total_payments) + sum(current_amount_due)) %>%
  arrange(desc(amount)) %>%
  head(10)
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
# targeting in terms of high-income zipcodes

high_fine_list <- high_fine %>%
  filter(violation_description%in%revenues_list$violation_description)

high_fine_list %>% group_by(violation_description) %>% summarise(n=n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 10 x 2
##    violation_description                              n
##    <chr>                                          <int>
##  1 EXP. METER NON-CENTRAL BUSINESS DISTRICT        2393
##  2 EXPIRED METER CENTRAL BUSINESS DISTRICT         1330
##  3 EXPIRED PLATE OR TEMPORARY REGISTRATION         1331
##  4 EXPIRED PLATES OR TEMPORARY REGISTRATION        2027
```

```
##  5 NO CITY STICKER VEHICLE UNDER/EQUAL TO 16,000 LBS.  2256
##  6 NO STANDING/PARKING TIME RESTRICTED                  777
##  7 PARKING/STANDING PROHIBITED ANYTIME                 1546
##  8 RESIDENTIAL PERMIT PARKING                          2099
##  9 RUSH HOUR PARKING                                    687
## 10 STREET CLEANING                                     2539
```

```
target_fines <-  high_fine_list %>%
  group_by(violation_description,quantile_2) %>%
  summarise(n_fines = n()) %>%
  pivot_wider(names_from = quantile_2, values_from = n_fines)
```

```
## `summarise()` regrouping output by 'violation_description' (override with `.groups` argument)
```

```
target_final <- target_fines %>% select(violation_description, `quantile 10`,`quantile 1`,everything())
    arrange(`quantile 1`)

# Then, the three suggested fines will be:

(suggested_fines <- target_fines %>% select(violation_description, `quantile 10`,`quantile 1`,everythin
    arrange(`quantile 1`) %>%
    head(3))
```

```
## # A tibble: 3 x 12
## # Groups:   violation_description [3]
##   violation_descr~ `quantile 10` `quantile 1` `quantile 2` `quantile 3`
##   <chr>                    <int>        <int>        <int>        <int>
## 1 RUSH HOUR PARKI~            86           59           50           91
## 2 NO STANDING/PAR~           133           69           55           68
## 3 EXPIRED METER C~           274          110           66          114
## # ... with 7 more variables: `quantile 4` <int>, `quantile 5` <int>, `quantile
## #   6` <int>, `quantile 7` <int>, `quantile 8` <int>, `quantile 9` <int>,
## #   `NA` <int>
```

b. If she raises the ticket price by $80 for each of these tickets, how much additional revenue can she expect? Assume there is no behavioral response (i.e. people continue to commit violations at the same rate and repay at the same rate).

It is 2393 (number of fines in a year, based on 2017) * 80 (increase) * 0.7243863 (ratio of repayment in 2017), which is 138,676.5. Adjusted by percentages, it is 13,867,600.5 dollars in the year.

```
high_fine %>% group_by(violation_description) %>%
  summarise(n=n(),
            rate_repay = sum(total_payments) / (sum(current_amount_due) + sum(total_payments))) %>%
  filter(violation_description == "EXP. METER NON-CENTRAL BUSINESS DISTRICT")
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 1 x 3
##   violation_description                        n rate_repay
##   <chr>                                    <int>      <dbl>
## 1 EXP. METER NON-CENTRAL BUSINESS DISTRICT  2393      0.724
```