Exploring NYC Complaints

Dav King

2022-04-22

```
library(tidyverse)
library(tidycensus)
library(tigris)
library(socviz)
library(nycomplaints)
library(viridis)
library(ggridges)
library(cowplot)
library(knitr)
options(tigris_use_cache = T)
```

```
theme_nymap <- function(base_size=9, base_family="") {
    require(grid)
    theme_bw(base_size=base_size, base_family=base_family) %+replace%
        theme(axis.line=element_blank(),
              axis.text=element_blank(),
              axis.ticks=element_blank(),
              axis.title=element_blank(),
              panel.background=element_blank(),
              panel.border=element_blank(),
              panel.grid=element_blank(),
              panel.spacing=unit(0, "lines"),
              plot.background=element_blank(),
              legend.justification = c(0,0),
              legend.position = c(0.1, 0.6),
              legend.direction = "horizontal"
        )
}
#There is no such thing as code theft, only code repurposing.
```

1. Briefly Describe the Dataset

This dataset is an aggregate of cases raised by constituents across every NYC council district office, with complaints about a wide variety of topics. Though different offices gather their data differently, it has been aggregated into a fairly specific set of overarching lables. There are eleven variables in the dataset: a unique identifier for each case (`unique_key`), a listing of which city council district each complaint was raised within (`account`), the date that the case was opened (`opendate`) and the date that it was closed (`closedate`), the type of complaint being raised (`complaint_type`) and a more specific indicator of what was being targeted in the complaint (`descriptor`), and the `zip` code, `borough`, `city`, council district (`council_dist`), and community board (`community_board`) in which the complaint was raised.

2. Look at each column/variable in more detail

```
Hmisc::describe(nycomplaints)
```

```
## nycomplaints
##
##  11  Variables      240027  Observations
## --------------------------------------------------------------------------------
## unique_key
##        n  missing distinct
##   240027        0   239958
##
## lowest : NYCC01506329 NYCC01506331 NYCC01506332 NYCC01506333 NYCC01506334
## highest: NYCC51505771 NYCC51505772 NYCC51505773 NYCC51505775 NYCC51505776
## --------------------------------------------------------------------------------
## account
##        n  missing distinct
##   240027        0       51
##
## lowest : NYCC01 NYCC02 NYCC03 NYCC04 NYCC05, highest: NYCC47 NYCC48 NYCC49 NYCC50 NYCC51
## --------------------------------------------------------------------------------
## opendate
##          n    missing   distinct       Info       Mean        Gmd        .05
##     240027          0       2596          1 2017-12-29        783 2015-04-24
##        .10        .25        .50        .75        .90        .95
## 2015-07-28 2016-06-22 2017-10-02 2019-06-05 2020-08-28 2021-06-28
##
## lowest : 2015-01-01 2015-01-02 2015-01-03 2015-01-04 2015-01-05
## highest: 2022-04-03 2022-04-04 2022-04-05 2022-04-06 2022-04-07
## --------------------------------------------------------------------------------
## closedate
##          n    missing   distinct       Info       Mean        Gmd        .05
##     223981      16046       2107          1 2018-03-23      803.5 2015-05-28
##        .10        .25        .50        .75        .90        .95
## 2015-09-08 2016-08-08 2018-01-29 2019-08-07 2021-02-02 2021-10-15
##
## lowest : 2015-01-02 2015-01-05 2015-01-06 2015-01-07 2015-01-08
## highest: 2022-04-02 2022-04-04 2022-04-05 2022-04-06 2022-04-07
## --------------------------------------------------------------------------------
## complaint_type
##        n  missing distinct
##   234967     5060       56
##
## lowest : @getxlate.Udf_Code_Genera Aging                     CIVIL AND HUMAN RIGHTS    Civil Service
## highest: SPAM MAIL                 Transportation            Utilities                 Veterans Affa
## --------------------------------------------------------------------------------
## descriptor
##        n  missing distinct
##   226770    13257     1002
##
## lowest : --Select--          001                 003                 004                 006
## highest: WRAP                WRONGFUL TERMINATION YOUTH ENGAGEMENT    ZOMBIE HOUSE        Zoning
```

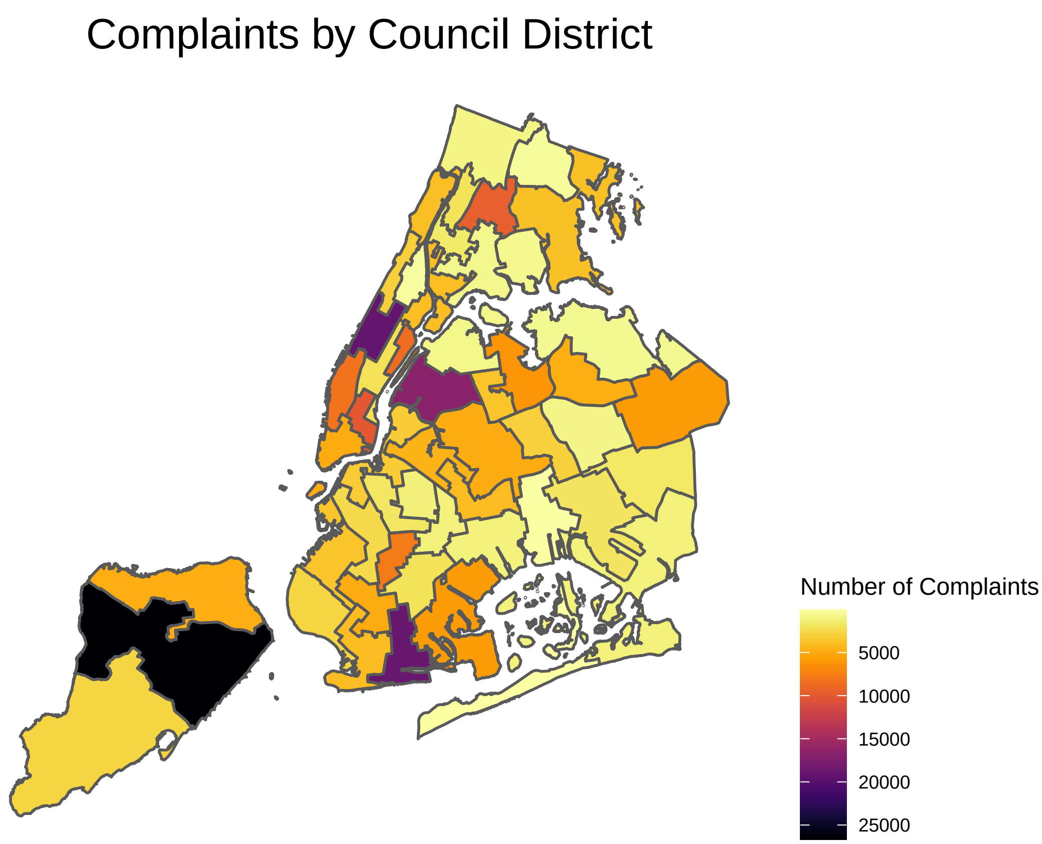
```
## ------------------------------------------------------------------------------
## zip
##        n  missing distinct
##   240027        0     1046
##
## lowest : 00000 00001 00002 00194 00347, highest: 98276 98332 98516 98682 99163
## ------------------------------------------------------------------------------
## borough
##        n  missing distinct
##   223869    16158        5
##
## lowest : Bronx         Brooklyn      Manhattan     Queens        Staten Island
## highest: Bronx         Brooklyn      Manhattan     Queens        Staten Island
##
## Value              Bronx      Brooklyn     Manhattan        Queens
## Frequency          18507         63718         61000         47610
## Proportion         0.083         0.285         0.272         0.213
##
## Value      Staten Island
## Frequency          33034
## Proportion         0.148
## ------------------------------------------------------------------------------
## city
##        n  missing distinct
##   239038      989      992
##
## lowest : 0                 1 Montague Street 10009             10010             10028
## highest: Yonkers, New York Yonkers, Ny       Yorktown          Yorktown Heights  Your City
## ------------------------------------------------------------------------------
## council_dist
##        n  missing distinct
##   230299     9728      154
##
## lowest : NYCC         NYCC001      NYCC002      NYCC003      NYCC004
## highest: NYCC51       NYCC7        NYCC8        NYCC9        NYCCNew York
## ------------------------------------------------------------------------------
## community_board
##        n  missing distinct
##   223936    16091       94
##
## lowest : 01            01 Bronx      01 Brooklyn   01 Manhattan  01 Queens
## highest: Bronx         Brooklyn      CB3           Queens        Staten Island
## ------------------------------------------------------------------------------
```

Using the `describe` function from the `Hmisc` package, we get a very extensive description of every variable: the number of observations it has, the number of missing (`NA`) values that it contains, and the number of distinct observations that it contains. It also shows the lowest and highest values for each variable (which are alphabetical, in character strings), and for numeric variables it also includes several quantile denotations. This is a great way to get a view of a complete dataset with minimal work.

`unique_key`: there's not much to say here. This variable is an indicator of each separate case being raised to the NYC city council district offices. It starts with the code of the specific `account` with which it was raised, and then appears to be a random, mostly sequential string after that. It is present for every variable and almost entirely unique across all cases, `n` = 240027 and `n_distinct` = 239958.

`account`: this is a descriptor of which of NYC's 51 different districts the complaint was raised within. It begins with the string NYCC followed by a two-digit number from 01 to 51, enumerating the city district. It is present for all variables. In the graph below, I count and visualize the number of complaints raised in each different council district in order to see whether there are any geographic trends to speak of. It does not appear that there is much of note.

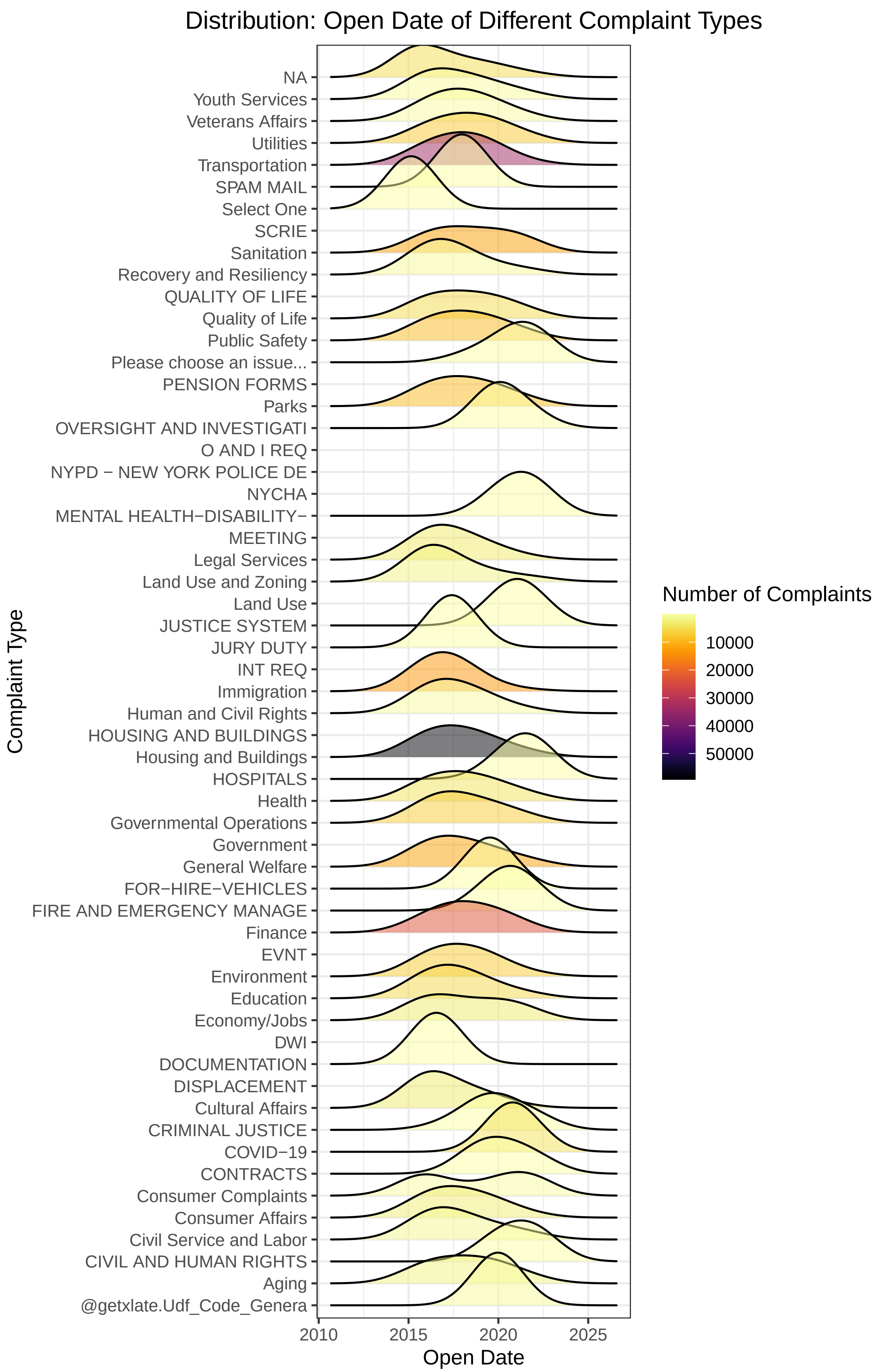
```
council_dist <-
  sf::read_sf(dsn = "C:/Users/Dav/Downloads/nycc_22a/nycc_22a/nycc.shp")
#Note: my apologies that this isn't fully reproducible, I didn't have time to
#figure out how to trace the file from its online .zip folder. This shapefile
#is included in my Sakai dropbox.

nycomplaints %>%
  transform(NYCC = substr(account, 1, 4), CounDist = substr(account, 5, 6)) %>%
  mutate(CounDist = as.integer(CounDist)) %>%
  count(CounDist) %>%
  inner_join(council_dist) %>%
  ggplot(aes(fill = n, geometry = geometry)) +
  geom_sf() +
  theme_nymap() +
  labs(title = "Complaints by Council District",fill = "Number of Complaints") +
  theme(legend.position = "right", legend.direction = "vertical",
        plot.title = element_text(hjust = 0.5, size = 16)) +
  scale_fill_viridis(option = "B", trans = "reverse")
```



`opendate`: this variable lists the date that each complaint was opened with an NYC council district office. The first complaint in this dataset was opened on January 1st, 2015, while the most recent complaint was opened on April 7th, 2022. The total number of distinct opening dates (2596) suggests that a complaint was opened almost but not quite every day between those two dates. The mean date for opening a complaint was December 29, 2017, but the median date for opening a complaint was October 2nd, 2017, suggesting a slight left-skew in opening dates for complaints (that is, that people have been opening complaints more frequently recently than they used to).

The graph below shows the distribution of open dates for different complaint types, allowing us to identify any trends in the data structure and any possible outliers. By shading the data according to the total number of complaints in each complaint type, we can identify that the most common complaints had much more broadly distributed peaks in terms of when they were most frequently filed, with medians around the median date of the whole dataset. Though the ridgeline plot smooths out a lot of the detail from this dataset, it still shows that each different complaint type saw a peak at some point in time, with some most common as early as 2015 and others peaking as late as 2021. There is some rhyme and reason to this distribution: for example, complaints about Covid-19, Hospitals, and the Justice System all peaked in the past couple of years, when those issues were notably more salient than they had been in recent times before that.

```
nycomplaints %>%
  group_by(complaint_type) %>%
  mutate(n = n()) %>%
  ggplot(aes(x = opendate, y = complaint_type, fill = n)) +
  geom_density_ridges(alpha = 0.5) +
  labs(title = "Distribution: Open Date of Different Complaint Types",
       x = "Open Date", y = "Complaint Type", fill = "Number of Complaints") +
  theme_bw() +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_fill_viridis(option = "B", trans = "reverse")
```

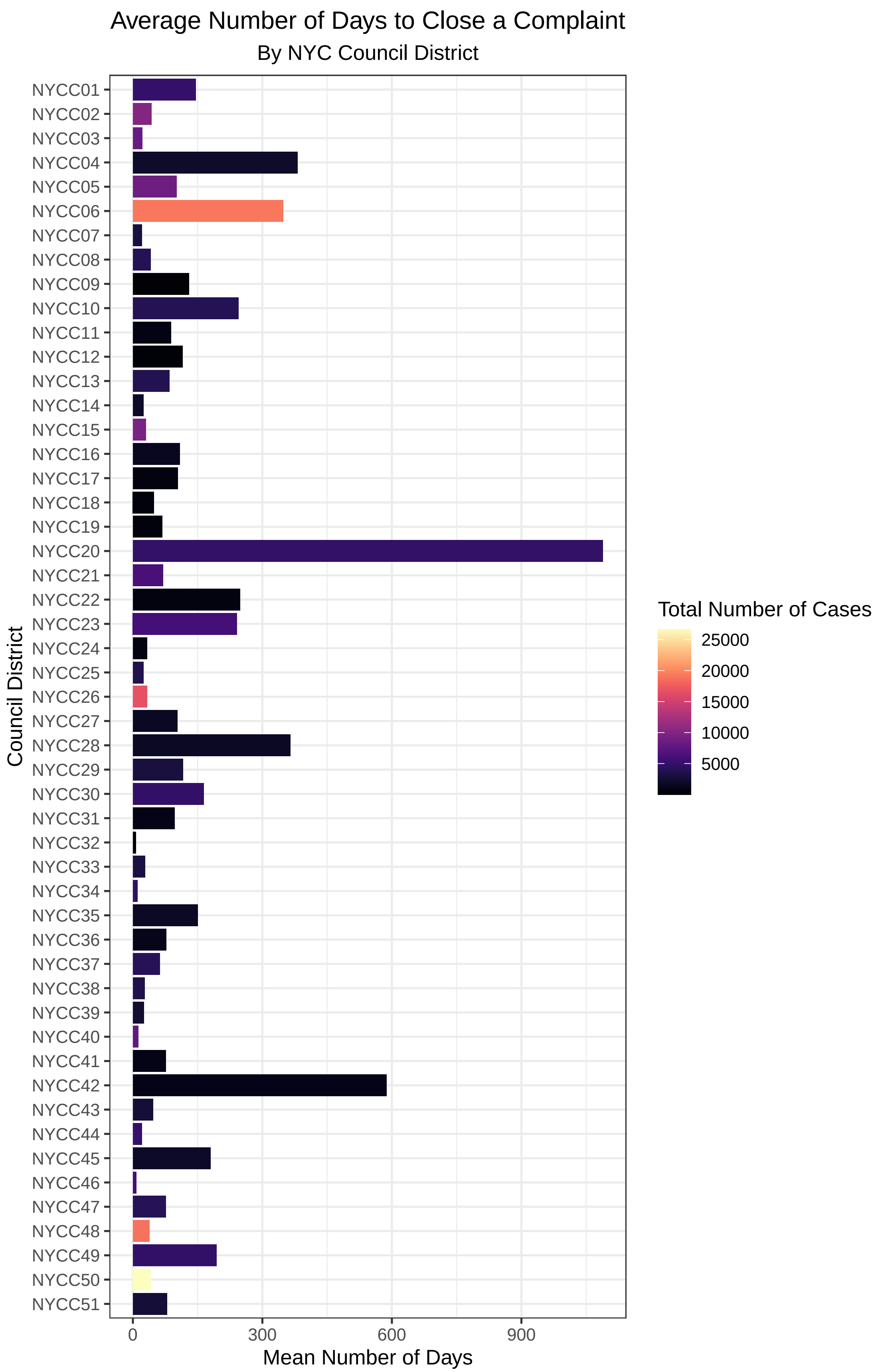

Distribution: Open Date of Different Complaint Types
Complaint Type
NA
Youth Services
Veterans Affairs
Utilities
Transportation
SPAM MAIL
Select One
SCRIE
Sanitation
Recovery and Resiliency
QUALITY OF LIFE
Quality of Life
Public Safety
Please choose an issue...
PENSION FORMS
Parks
OVERSIGHT AND INVESTIGATI
O AND I REQ
NYPD − NEW YORK POLICE DE
NYCHA
MENTAL HEALTH−DISABILITY−
MEETING
Legal Services
Land Use and Zoning
Land Use
JUSTICE SYSTEM
JURY DUTY
INT REQ
Immigration
Human and Civil Rights
HOUSING AND BUILDINGS
Housing and Buildings
HOSPITALS
Health
Governmental Operations
Government
General Welfare
FOR−HIRE−VEHICLES
FIRE AND EMERGENCY MANAGE
Finance
EVNT
Environment
Education
Economy/Jobs
DWI
DOCUMENTATION
DISPLACEMENT
Cultural Affairs
CRIMINAL JUSTICE
COVID−19
CONTRACTS
Consumer Complaints
Consumer Affairs
Civil Service and Labor
CIVIL AND HUMAN RIGHTS
Aging
@getxlate.Udf_Code_Genera
2010
2015
2020
2025
Open Date
Number of Complaints
10000
20000
30000
40000
50000

`closedate`: this variable lists the date that each complaint was closed. Though every complaint had an open date (meaning all of these close dates are for complaints that were opened on or after January 1st, 2015), not all complaints have a close date - it's missing for 16046 of these complaints. Similarly, there are far fewer distinct dates in here, suggesting that on some 500 days between 1/1/2015 and 4/7/2022 there were zero complaints closed citywide. The first date that any complaint was closed was January 2nd, 2015 (all of which happened to be opened on that same date), while the last date that any complaint was closed was April 7th, 2022 - the last date of the dataset. Despite how quickly some complaints were closed on January 2nd, the mean close date was March 23rd, 2018 (a little less than 3 months after the mean open date) and the median close date was January 29th, 2018 (almost four months after the median open date). Not only does this suggest that most complaints take a while to process, it also suggests a right skew in how long they take to close - that is, that most cases are closed relatively quickly, but some take a very long time to be closed.

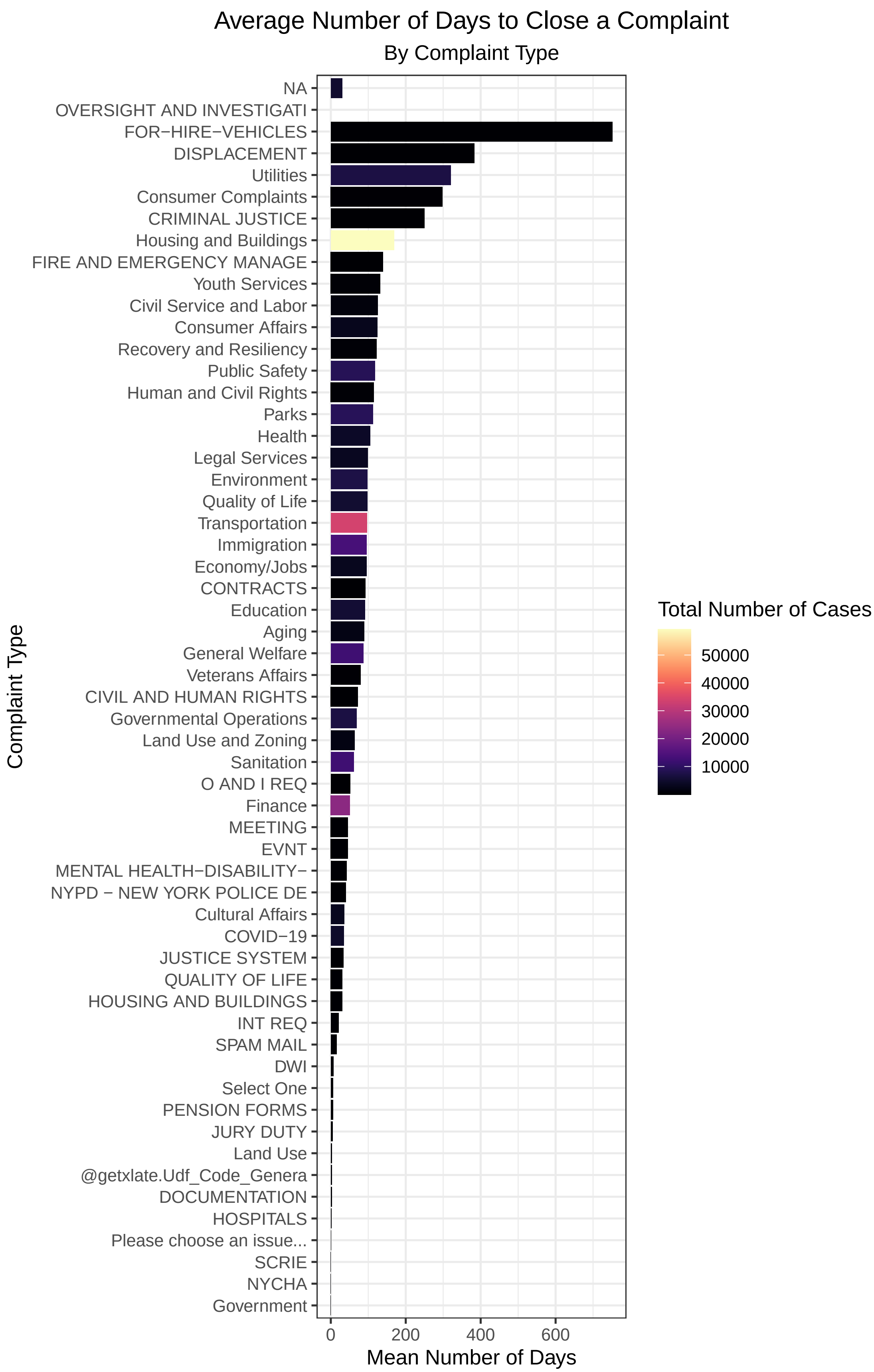
In the graphs below, we can consider the average number of days to close a complaint grouped by both council district and complaint type, looking for trends and outliers. There is a lot of variability in how long it took council districts to close complaints on average, with some as short as perhaps a couple of weeks and others (very notably the 20th district) taking over 1,000 days to do so on average. This appears to be largely unrelated to the total number of cases that each district office received, though the districts receiving the most cases did tend to resolve them in the shortest amounts of time (probably both because they were receiving much more generic, easier complaints and because they were more prepared to handle them).

When grouping by complaint type, the distribution made much more sense - the complaints that took the longest to resolve on average were *much*, much longer than most other types of complaints, and were very low in terms of raw case counts. There is one notable exception however - the sixth-longest complaint type to resolve on average was also the one that had the outright largest number of cases - housing and buildings. That seems logical, as many people file complaints about their housing but landlord protections are incredibly strong compared to tenant protections. This graph also reveals that no `OVERSIGHT AND INVESTIGATI` complaints were ever resolved, while all `NYCHA`, `SCRIE`, or `Government` complaints were either never resolved or resolved the day they were filed.

```
cd <- nycomplaints %>%
  mutate(timeopen = closedate-opendate) %>%
  separate(timeopen, c("nDays", "days"), " ") %>%
  select(-days) %>%
  mutate(nDays = as.integer(nDays))
cd %>%
  group_by(account) %>%
  summarize(mDays = mean(nDays, na.rm = T), n = n()) %>%
  ggplot(aes(x = mDays, y = account, fill = n)) +
  geom_col() +
  scale_y_discrete(limits = rev) +
  scale_fill_viridis(option = "A") +
  theme_bw() +
  labs(title = "Average Number of Days to Close a Complaint",
       subtitle = "By NYC Council District", x = "Mean Number of Days",
       y = "Council District", fill = "Total Number of Cases") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```


Average Number of Days to Close a Complaint
By NYC Council District
Council District
NYCC01
NYCC02
NYCC03
NYCC04
NYCC05
NYCC06
NYCC07
NYCC08
NYCC09
NYCC10
NYCC11
NYCC12
NYCC13
NYCC14
NYCC15
NYCC16
NYCC17
NYCC18
NYCC19
NYCC20
NYCC21
NYCC22
NYCC23
NYCC24
NYCC25
NYCC26
NYCC27
NYCC28
NYCC29
NYCC30
NYCC31
NYCC32
NYCC33
NYCC34
NYCC35
NYCC36
NYCC37
NYCC38
NYCC39
NYCC40
NYCC41
NYCC42
NYCC43
NYCC44
NYCC45
NYCC46
NYCC47
NYCC48
NYCC49
NYCC50
NYCC51
0
300
600
900
Mean Number of Days
Total Number of Cases
25000
20000
15000
10000
5000

```r
cd %>%
  group_by(complaint_type) %>%
  summarize(mDays = mean(nDays, na.rm = T), n = n()) %>%
  ggplot(aes(x = mDays, y = reorder(complaint_type, mDays), fill = n)) +
  geom_col() +
  scale_fill_viridis(option = "A") +
  theme_bw() +
  labs(title = "Average Number of Days to Close a Complaint",
       subtitle = "By Complaint Type", x = "Mean Number of Days",
       y = "Complaint Type", fill = "Total Number of Cases") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```



Average Number of Days to Close a Complaint
By Complaint Type
NA
OVERSIGHT AND INVESTIGATI
FOR−HIRE−VEHICLES
DISPLACEMENT
Utilities
Consumer Complaints
CRIMINAL JUSTICE
Housing and Buildings
FIRE AND EMERGENCY MANAGE
Youth Services
Civil Service and Labor
Consumer Affairs
Recovery and Resiliency
Public Safety
Human and Civil Rights
Parks
Health
Legal Services
Environment
Quality of Life
Transportation
Immigration
Economy/Jobs
CONTRACTS
Education
Aging
General Welfare
Veterans Affairs
CIVIL AND HUMAN RIGHTS
Governmental Operations
Land Use and Zoning
Sanitation
O AND I REQ
Finance
MEETING
EVNT
MENTAL HEALTH−DISABILITY−
NYPD − NEW YORK POLICE DE
Cultural Affairs
COVID−19
JUSTICE SYSTEM
QUALITY OF LIFE
HOUSING AND BUILDINGS
INT REQ
SPAM MAIL
DWI
Select One
PENSION FORMS
JURY DUTY
Land Use
@getxlate.Udf_Code_Genera
DOCUMENTATION
HOSPITALS
Please choose an issue...
SCRIE
NYCHA
Government
Complaint Type
0
200
400
600
Mean Number of Days
Total Number of Cases
50000
40000
30000
20000
10000

`complaint_type` is the first truly messy variable in this dataset. The code below briefly allows us to inspect it. On initial inspection it contains 56 different types of complaints, but when reducing it to lowercase that number drops to 55. While most of these variables make sense, some are problematic. 5060 are `NA` values, which drops the number of types of complaints to 54. 607 are labeled `@getxlate.udf_code_genera`, which is clearly a tabulation error - as are the 143 complaints labeled `please choose an issue` and the 5 labeled `select one`. Additonally, some of the variables look like they should arguably be merged together, and many of the complaints at very low frequencies are riddled with misspellings or are oddly specific enough that they seem functionally useless within this framework.

```
nycomplaints %>%
  mutate(complaint_type = tolower(complaint_type)) %>%
  count(complaint_type) %>%
  arrange(desc(n)) %>%
  #print(n = Inf) %>%
  kable()
```

complaint_type	n
housing and buildings	59209
transportation	33916
finance	23395
immigration	13486
sanitation	12336
general welfare	12322
parks	9019
public safety	8935
environment	7352
utilities	7158
governmental operations	7070
education	5629
quality of life	5332
NA	5060
covid-19	4533
health	4247
legal services	3470
cultural affairs	3308
economy/jobs	3221
consumer affairs	2987
aging	2040
land use and zoning	1899
civil service and labor	1219
@getxlate.udf_code_genera	607
recovery and resiliency	598
youth services	501
human and civil rights	432
consumer complaints	234
veterans affairs	200
please choose an issue…	143
criminal justice	32
fire and emergency manage	24
contracts	14
justice system	14
mental health-disability-	14

complaint_type	n
documentation	11
hospitals	10
civil and human rights	7
spam mail	7
for-hire-vehicles	6
jury duty	5
oversight and investigati	5
select one	5
nypd - new york police de	2
o and i req	2
scrie	2
displacement	1
dwi	1
evnt	1
government	1
int req	1
land use	1
meeting	1
nycha	1
pension forms	1

`descriptor` is a series of more specific explanations for what the complaint being filed was about. With just over 1,000 distinct responses, it's impossible to try to view and clean all of them - but it's very clear that a lot of these data points are serious issues, as some are just numbers and others simply say `--Select--`. Additionally, while not a large proportion in the grand scheme of things, over 13,000 of these complaints are missing a specific descriptor from this column. The code below allows us to view the most common descriptors (anything seen over 1,000 times in the dataset) along with the complaint type that they were housed under in order to identify trends in the meaningful chunk of the data. Looking at the list, we can see that most of the common descriptors came in the `Housing and Buildings` complaint type, with `Finance` and `Transportation` also rather popular complaint types. As the number goes down, the complaint types diversity (which is to be expected).

```
nycomplaints %>%
  mutate(descriptor = tolower(descriptor)) %>%
  count(complaint_type, descriptor) %>%
  arrange(desc(n)) %>%
  filter(n > 1000) %>%
  #print(n = Inf) %>%
  kable()
```

complaint_type	descriptor	n
Finance	tax preparation assi	7174
Immigration	u.s. citizenship	6652
Housing and Buildings	seeking an affordable housing/apartment	5508
Transportation	street resurfacing	5475
NA	NA	4971
Housing and Buildings	lease issue	4923
Housing and Buildings	landlord/managing agent issues	4469
Housing and Buildings	nycha building maintenance	4063
General Welfare	food stamps	3935

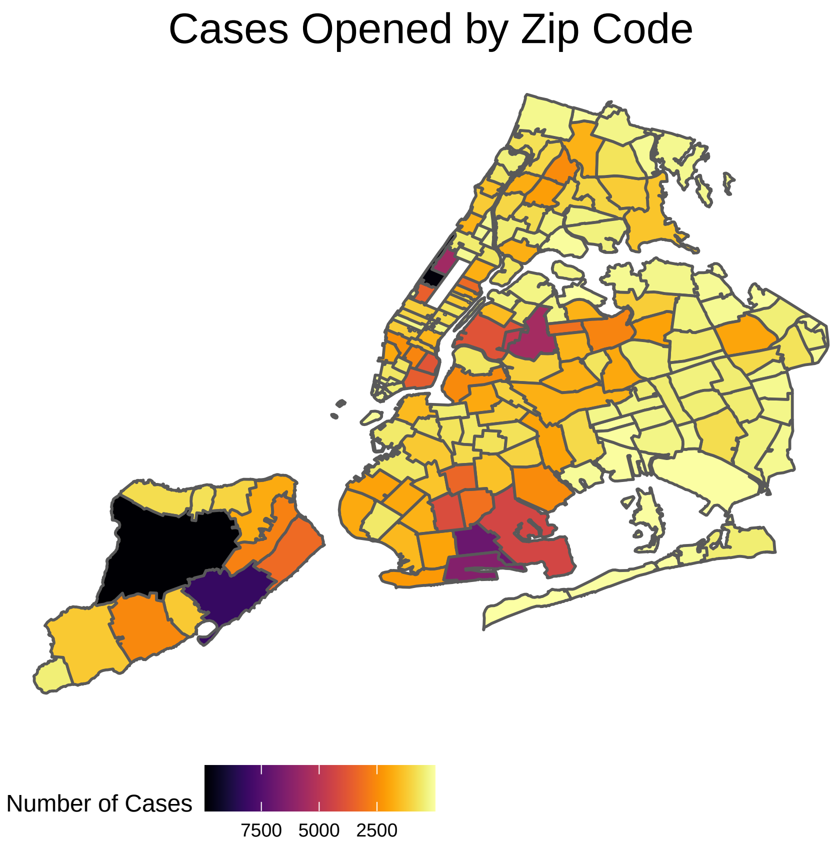
complaint_type	descriptor	n
Finance	sche	3580
Housing and Buildings	nycha misc.	3024
Finance	scrie	2993
Housing and Buildings	eviction	2728
Transportation	bicycles	2712
Finance	tax exemption	2638
Parks	tree planting/pruning	2492
Housing and Buildings	building maintenance	2340
Utilities	con edison	2229
Housing and Buildings	nycha new application	2222
Housing and Buildings	section 8	2160
Parks	park maintenance	2110
Sanitation	litter	2031
Transportation	street signs	1973
Immigration	green card	1969
Utilities	lifeline	1880
Cultural Affairs	libraries	1874
Transportation	sidewalks	1866
Transportation	traffic signals	1864
Housing and Buildings	construction general	1847
COVID-19	health	1837
Sanitation	recycling	1816
Housing and Buildings	heat/hot water	1806
Housing and Buildings	NA	1706
Immigration	literary services for immigrants	1629
Transportation	pot holes	1598
Transportation	street lights	1584
Sanitation	dirty sidewalk	1486
Legal Services	immigration	1480
General Welfare	public assistance	1477
Finance	property tax bill	1474
Parks	tree sidewalk prog	1469
Sanitation	e-waste pickups	1455
Housing and Buildings	hpd housing lottery	1420
Legal Services	housing	1418
Transportation	parking restrictions	1417
Housing and Buildings	repair assistance	1389
Transportation	buses	1380
Utilities	verizon	1347
Public Safety	police precinct	1309
Housing and Buildings	nycha transfers	1302
Governmental Operations	voting information	1285
Housing and Buildings	senior housing	1284
General Welfare	food pantry	1234
Housing and Buildings	maintenance	1227
Transportation	pedestrian safety	1180
Quality of Life	noise - other	1172
Transportation	parking permits	1169
Immigration	city services and benefits	1145
Economy/Jobs	seeking employment	1122
Transportation	speed bump request	1104
Utilities	national grid	1089

complaint_type	descriptor	n
Sanitation	snow removal	1086
General Welfare	ssi and social security	1052
Public Safety	community policing	1027
Housing and Buildings	rent increases	1004

`zip` gives the zipcode of the address where the person filing the complaint resided. Fortunately, every observation has a zipcode attached to it, and the total number of distinct zipcodes (1046) is far less than the total number of zipcodes found within New York City. Unfortunately, not all of these zipcodes are within NYC. Though the lowest zipcodes in this dataset are 00000, 00001 and 00002, the lowest zipcode in the country is 00001 and it is found in rural Alaska (as is 00002). However, we can match this with the zip codes found in the `nycdogs` package, and by using a convenient `inner_join` determine exactly which zip codes are actually within NYC.

The map below counts the number of cases opened in each zip code and displays them in a map. This shows the relatively low number of cases opened in Queens, especially as compared to areas in South Brooklyn or Manhattan. Overall, there is not a super large trend in geographic distribution visible on this map.

```
zips <- nycdogs::nyc_zips %>%
  mutate(zip = as.character(zip_code))

nycomplaints %>%
  count(zip) %>%
  inner_join(zips) %>%
  ggplot(aes(fill = n, geometry = geometry)) +
  geom_sf() +
  theme_nymap() +
  scale_fill_viridis(option = "B", trans = "reverse") +
  labs(title = "Cases Opened by Zip Code", fill = "Number of Cases") +
  theme(plot.title = element_text(hjust = 0.5, size = 16),
        legend.position = "bottom")
```



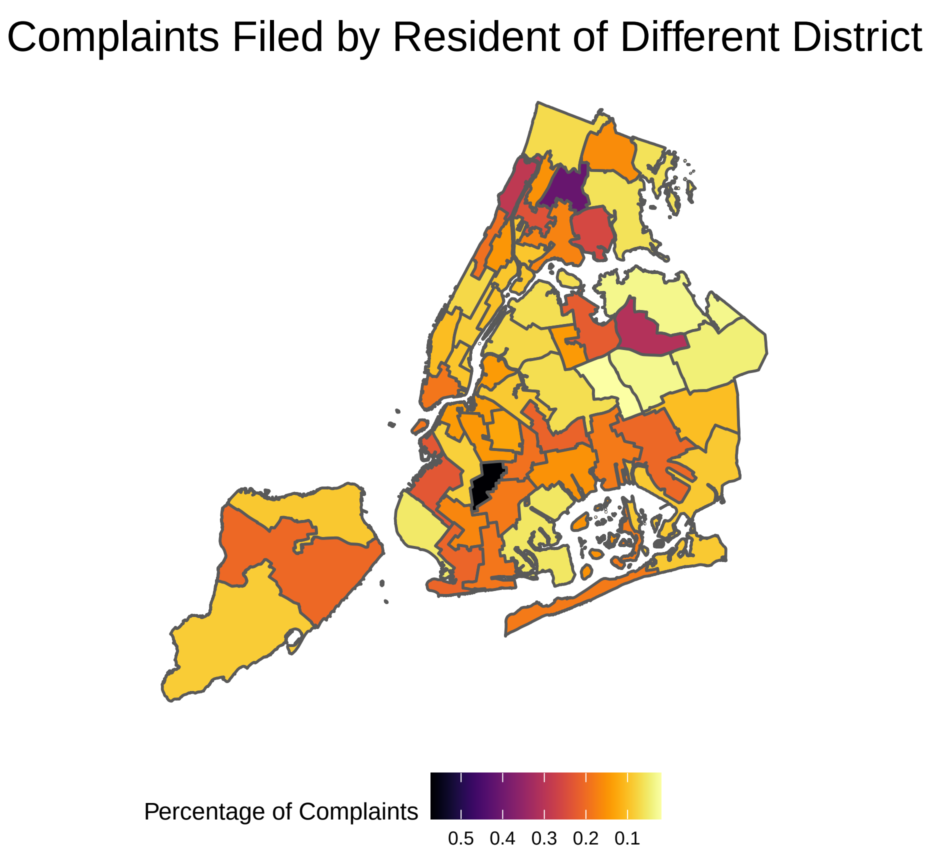
`borough` gives the borough in which the person filing the complaint resided. Though some 16 thousand observations are missing this variable, all of the rest do fall into the five boroughs of NYC. Brooklyn residents filed the highest proportion of complaints (28.5%), followed closely by Manhattan residents (27.2%), then Queens residents (21.3%) and Staten Island residents (14.8%) with Bronx residents bringing up the rear at a measly 8.3%. To some extent, this likely corresponds to the relative population of each borough.

`city` corresponds to the city in which the person filing the complaint resided. This data is missing for far fewer observations (just under a thousand), and has nearly a thousand distinct labels. Once again, some of these are not real cities (e.g., `0`, `1 Montague Street`, `Your City`), and others are duplicates (`Yonkers, New York` and `Yonkers, Ny`). These might be harder to verify than the zip codes, but they are still a valuable source of data to the extent that they can be trusted.

`council_dist` corresponds to the council district in which the person filing the complaint resided, which is not necessarily the same as the `account` variable corresponding to the council district in which the complaint was opened. This is a messy variable, as council district numbers have anywhere between 0 and 2 zeroes preceding them and thus there are serious overlap issues. NYC has 51 districts, but there are 154 distinct values within this variable. This variable can be cleaned for reasonable use without too much effort, though a substantial number of cases are lost when doing so (as some numbered higher than 51). There are nearly 10,000 missing observations, which is a substantial amount.

The code below mutates the `council_dist` variable into a usable format, identifies the proportion of cases in `account` that do not match up with `council_dist`, and displays this output in a map in order to identify which districts had a lot of complaints filed by out-of-district residents. Looking at the map below, we can see that a surprising number of complaints filed in the Bronx were filed by people who resided in a different council district. Far and away the district with the most complaints originating from residents of a different district, however, was District 40 - with 57.2% of its complaints being filed by residents of a different district (compared to the next highest, District 15, at 41.2%, and followed by district 20 at 30.7%). I can honestly find no reason for this to occur. District 40 is a majority-black district in Brooklyn, but there is only one thing of note about it - it is close to the geographical center of Brooklyn. District 15 is similarly located at the geographic center of the Bronx, but it makes more sense, as upstart Congressman Ritchie Torres' former district and the home of both the Bronx Zoo and the New York Botanical Garden.

```
nycomplaints %>%
  transform(NYCC = substr(council_dist, 1, 4),
            CounDist = substr(council_dist, 5, 7)) %>%
  mutate(CounDist = as.integer(CounDist)) %>%
  filter(!is.na(CounDist) & CounDist <= 51 & CounDist != 0) %>%
  transform(NYCC2 = substr(account, 1, 4),
            CounDistComp = substr(account, 5, 6)) %>%
  mutate(CounDistComp = as.integer(CounDistComp)) %>%
  group_by(CounDistComp) %>%
  mutate(perc = if_else(CounDistComp == CounDist, 1, 0)) %>%
  summarize(perc = 1 - mean(perc)) %>%
  inner_join(council_dist, by = c("CounDistComp" = "CounDist")) %>%
  ggplot(aes(fill = perc, geometry = geometry)) +
  geom_sf() +
  theme_nymap() +
  scale_fill_viridis(option = "B", trans = "reverse") +
  labs(title = "Complaints Filed by Resident of Different District",
       fill = "Percentage of Complaints") +
  theme(plot.title = element_text(hjust = 0.5, size = 16),
        legend.position = "bottom")
```



Finally, `community_board` designates the specific community board of the constituent. Though missing for just over 16,000 observations, there are only 94 distinct variables - thus, we can view all of them. Most of them appear to correspond to real community boards, but some 30 or so of these boards listed are just numbers or otherwise meaningless information (primarily corresponding to single-digit numbers of responses).

```
nycomplaints %>%
  count(community_board) %>%
  arrange(desc(n)) %>%
  #print(n = Inf) %>%
  kable()
```

community_board	n
07 Manhattan	18485
NA	16091
02 Staten Island	15717
15 Brooklyn	11481
02 Queens	10591
03 Staten Island	8950
03 Manhattan	8307
01 Staten Island	8264
08 Manhattan	8033
18 Brooklyn	6805
14 Brooklyn	6068
03 Queens	5764
13 Brooklyn	5125
12 Brooklyn	4964
01 Brooklyn	4902
01 Queens	4866
04 Manhattan	4667
06 Manhattan	4489
05 Queens	4091
07 Queens	3758
02 Manhattan	3670
12 Manhattan	3467
11 Queens	3265
05 Bronx	3137
01 Manhattan	3040
13 Queens	3014
04 Brooklyn	2928
17 Brooklyn	2912
04 Queens	2846
11 Bronx	2803
06 Bronx	2689
12 Queens	2581
07 Brooklyn	2552
06 Queens	2541
11 Manhattan	2485
11 Brooklyn	2449
10 Brooklyn	2426
10 Bronx	2337
02 Brooklyn	2304
05 Brooklyn	2169
09 Manhattan	2075

community_board	n
07 Bronx	2074
08 Queens	1863
06 Brooklyn	1807
05 Manhattan	1620
03 Brooklyn	1378
04 Bronx	1368
09 Brooklyn	1176
09 Queens	1144
16 Brooklyn	1052
08 Brooklyn	1039
08 Bronx	990
09 Bronx	913
03 Bronx	807
12 Bronx	792
10 Manhattan	662
14 Queens	631
10 Queens	512
02 Bronx	315
01 Bronx	268
Brooklyn	171
Queens	120
Staten Island	83
82 Queens	23
3 Staten Island	16
Bronx	12
15	10
07	9
01	7
3	6
6	6
8 Brooklyn	5
03	4
13	4
55 Brooklyn	4
95 Staten Island	4
8	3
0197992	2
0230458	2
2	2
0162998	1
0200787	1
0201066	1
0208188	1
0210527	1
0211258	1
0225656	1
0230206	1
0230244	1
1	1
17	1
26 Bronx	1
27 Bronx	1

community_board	n
5 Brooklyn	1
CB3	1

3. Explore in more depth: at least three polished plots or maps

- Show your work.
- Provide a motivation for and discussion of each plot.
- You can merge in external data if you like.

```
nycomplaints <- nycomplaints %>%
  mutate(complaint_type = tolower(complaint_type)) %>%
  filter(complaint_type != "@getxlate.udf_code_genera")
```

```
pop_inc <- get_acs(
  geography = "zcta",
  variables = c("B01003_001", "B06011_001", "B01002_001",
              "B01001_002", "B01001_026"),
  year = 2020#,
  #geometry = T
) %>%
  separate(NAME, into = c("name", "zip"), sep = " ") %>%
  pivot_wider(names_from = variable,
              values_from = c(estimate, moe)) %>%
  mutate(pop = estimate_B01003_001,
         pop_moe = moe_B01003_001,
         med_income = estimate_B06011_001,
         med_income_moe = moe_B06011_001,
         med_age = estimate_B01002_001,
         med_age_moe = moe_B01002_001,
         male = estimate_B01001_002,
         male_moe = moe_B01001_002,
         female = estimate_B01001_026,
         female_moe = moe_B01001_026) %>%
  mutate(mProp = male/(male + female),
         fProp = female/(male + female))
  #%>%
  #shift_geometry()

pop_names <- tribble(
    ~varname, ~clean_name,
    "B01003_001", "pop",
    "B01001B_001", "black",
    "B01001A_001", "white",
    "B01001H_001", "nh_white",
    "B01001I_001", "hispanic",
    "B01001D_001", "asian"
  )

race <- get_acs(geography = "zcta",
                    variables = pop_names$varname,
```

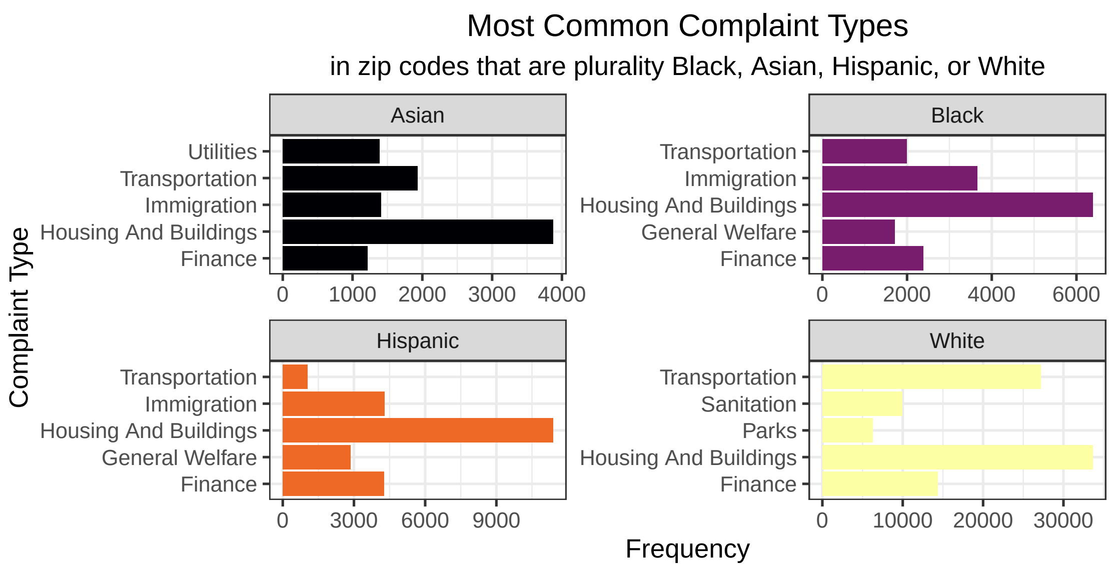
```r
                    cache_table = TRUE,
                    year = 2020) %>%
  mutate(variable = reduce2(pop_names$varname,
                            pop_names$clean_name,
                            str_replace,
                            .init = variable)) %>%
  select(-moe) %>%
  pivot_wider(names_from = variable, values_from = estimate) %>%
  rename(fips = GEOID, name = NAME) %>%
  separate(name, into = c("name", "zip"), sep = " ")

full <- nycomplaints %>%
  left_join(pop_inc) %>%
  left_join(race) %>%
  inner_join(zips) %>%
  mutate(complaint_type = stringr::str_to_title(complaint_type))
#Note: by getting rid of the observations that do not contain an NYC zip code,
#we lose 18,129 observations. However, this is probably a worthy loss, to
#consider all analysis within what we know firmly to be NYC residents.
```

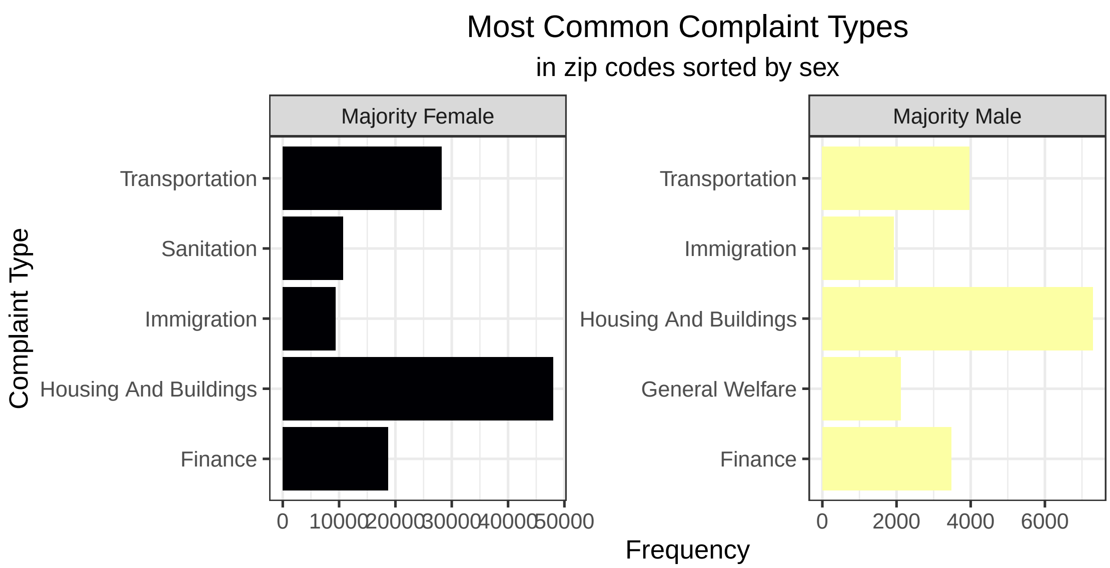
Plot 1: Complaints by Demographics

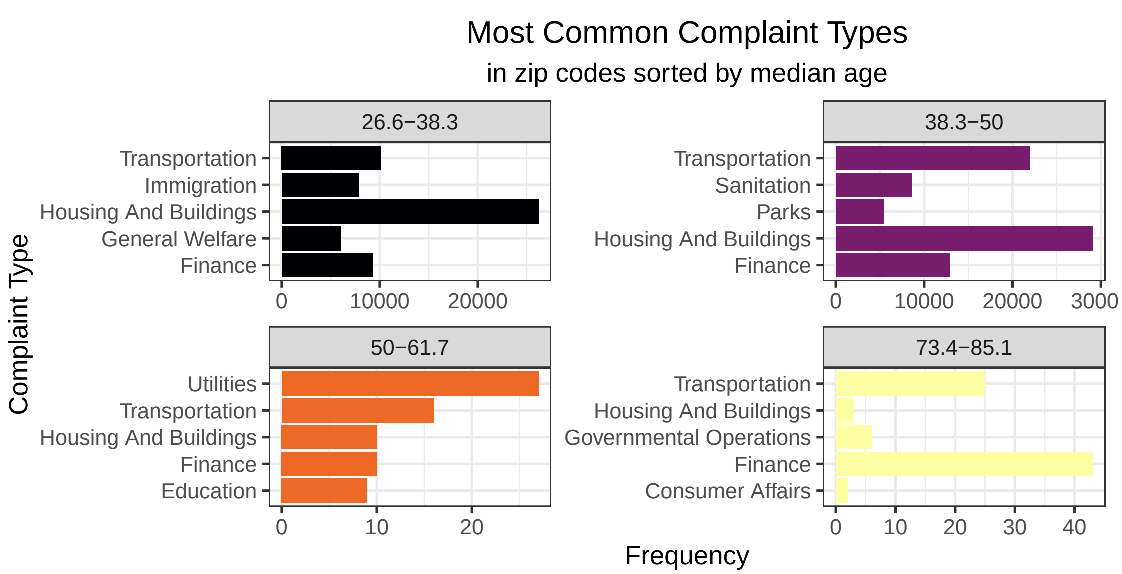
I was curious to see whether there were differences in complaints filed by the demographics of the filer. However, this information is not included in the dataset. While there is no way to fill in that information, there is a somewhat crude way to do it - by identifying the most common complaint types filed by residents of zip codes that are constituted of a majority of people belonging to various demographic groups. Drawing from the 2020 five-year ACS, the variables I chose to select were race, median age, and sex. Note that while no math is done to put the complaints in per-100,000 format, setting `scales = "free"` enables us to compare all of these data points on equal dimensions - essentially serving the same purpose.

```r
race <- full %>%
  mutate(race_plurality = pmax(black, white, nh_white, hispanic, asian)) %>%
  mutate(race_plurality = case_when(race_plurality == black ~ "Black",
                                    race_plurality == white ~ "White",
                                    race_plurality == nh_white ~ "nh_white",
                                    race_plurality == hispanic ~ "Hispanic",
                                    race_plurality == asian ~ "Asian")) %>%
  group_by(race_plurality, complaint_type) %>%
  mutate(complaints = n()) %>%
  select(complaints) %>%
  unique() %>%
  arrange(desc(complaints)) %>%
  ungroup() %>%
  group_by(race_plurality) %>%
  slice(1:5) %>%
  na.omit() %>%
  ggplot(aes(x = complaint_type, y = complaints, fill = race_plurality)) +
  geom_col() +
  facet_wrap(~ race_plurality, scales = "free") +
  coord_flip() +
  scale_fill_viridis(discrete = T, option = "B") +
  guides(fill = "none") +
```

```r
  theme_bw() +
  labs(title = "Most Common Complaint Types",
  subtitle = "in zip codes that are plurality Black, Asian, Hispanic, or White",
       y = "Frequency", x = "Complaint Type") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
age <- full %>%
  mutate(age_int = cut_interval(med_age, 5)) %>%
  group_by(age_int) %>%
  count(age_int, complaint_type) %>%
  na.omit() %>%
  arrange(desc(n)) %>%
  slice(1:5) %>%
  mutate(age_int = case_when(age_int == "[26.6,38.3]" ~ "26.6-38.3",
                             age_int == "(38.3,50]" ~ "38.3-50",
                             age_int == "(50,61.7]" ~ "50-61.7",
                             age_int == "(73.4,85.1]" ~ "73.4-85.1")) %>%
  ggplot(aes(x = n, y = complaint_type, fill = age_int)) +
  geom_col() +
  facet_wrap(~age_int, scales = "free") +
  scale_fill_viridis(discrete = T, option = "B") +
  guides(fill = "none") +
  theme_bw() +
  labs(title = "Most Common Complaint Types",
       subtitle = "in zip codes sorted by median age",
       x = "Frequency", y = "Complaint Type") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
sex <- full %>%
  mutate(sex = case_when(mProp > 0.5 ~ "Majority Male",
                         mProp < 0.5 ~ "Majority Female",
                         mProp == 0.5 ~ "Exactly Tied")) %>%
  group_by(sex) %>%
  count(complaint_type) %>%
  na.omit() %>%
  arrange(desc(n)) %>%
  slice(1:5) %>%
  ggplot(aes(x = n, y = complaint_type, fill = sex)) +
  geom_col() +
  facet_wrap(~sex, scales = "free") +
  guides(fill = "none") +
  theme_bw() +
  labs(title = "Most Common Complaint Types",
       subtitle = "in zip codes sorted by sex",
       x = "Frequency", y = "Complaint Type") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5)) +
  scale_fill_viridis(discrete = T, option = "B")
plot_grid(race, sex, age, ncol = 1)
```


Most Common Complaint Types
in zip codes that are plurality Black, Asian, Hispanic, or White
Asian
Utilities
Transportation
Immigration
Housing And Buildings
Finance
0
1000
2000
3000
4000
Black
Transportation
Immigration
Housing And Buildings
General Welfare
Finance
0
2000
4000
6000
Hispanic
Transportation
Immigration
Housing And Buildings
General Welfare
Finance
0
3000
6000
9000
White
Transportation
Sanitation
Parks
Housing And Buildings
Finance
0
10000
20000
30000
Complaint Type
Frequency


Most Common Complaint Types
in zip codes sorted by sex
Majority Female
Transportation
Sanitation
Immigration
Housing And Buildings
Finance
0
10000
20000
30000
40000
50000
Majority Male
Transportation
Immigration
Housing And Buildings
General Welfare
Finance
0
2000
4000
6000
Complaint Type
Frequency


Most Common Complaint Types
in zip codes sorted by median age
26.6–38.3
Transportation
Immigration
Housing And Buildings
General Welfare
Finance
0
10000
20000
38.3–50
Transportation
Sanitation
Parks
Housing And Buildings
Finance
0
10000
20000
3000
50–61.7
Utilities
Transportation
Housing And Buildings
Finance
Education
0
10
20
73.4–85.1
Transportation
Housing And Buildings
Governmental Operations
Finance
Consumer Affairs
0
10
20
30
40
Complaint Type
Frequency

Across all races, housing and buildings was still far and away the most common complaint type. Finance and transportation were also seen in the top five categories across all race groups. Perhaps unsurprisingly, one of the biggest sources of complaints was immigration issues for zip codes that were predominately every race *except* for White - particularly for Black and Hispanic plurality zip codes. Parks was a complaint seen most frequently in plurality White zip codes, likely because parks are often attached to gentrification in areas such as NYC. General Welfare, which includes many descriptors often related to poverty (such as food stamps, homeless encampments, and social security disability) was in the top 5 for Black and Hispanic plurality zip codes, while Utilities was common in Asian-plurality zip codes and Sanitation was in White plurality zip codes. While this obviously is still a crude measure of race in complaint types, a lot of these findings are rather sensical. This graph also gives some vague insight into the relative proportions of NYC residents who are Asian, Black, Hispanic, and White, though this is where putting the data in per-100K format would have served a useful purpose.

The graph dividing zip codes according to sex is probably functionally useless, as most zip codes are slightly majority female (reflecting the US sex divide as a whole) but none are greater than 55% in either direction, suggesting that this data breakdown is a poor proxy for the complaints filed by members of each sex. Still, there is one difference of note - one of the top five issues in female-dominated zip codes was Sanitation, while this was replaced by General Welfare in male-dominated zip codes. Other than that, these data show almost identical trends for both groups.

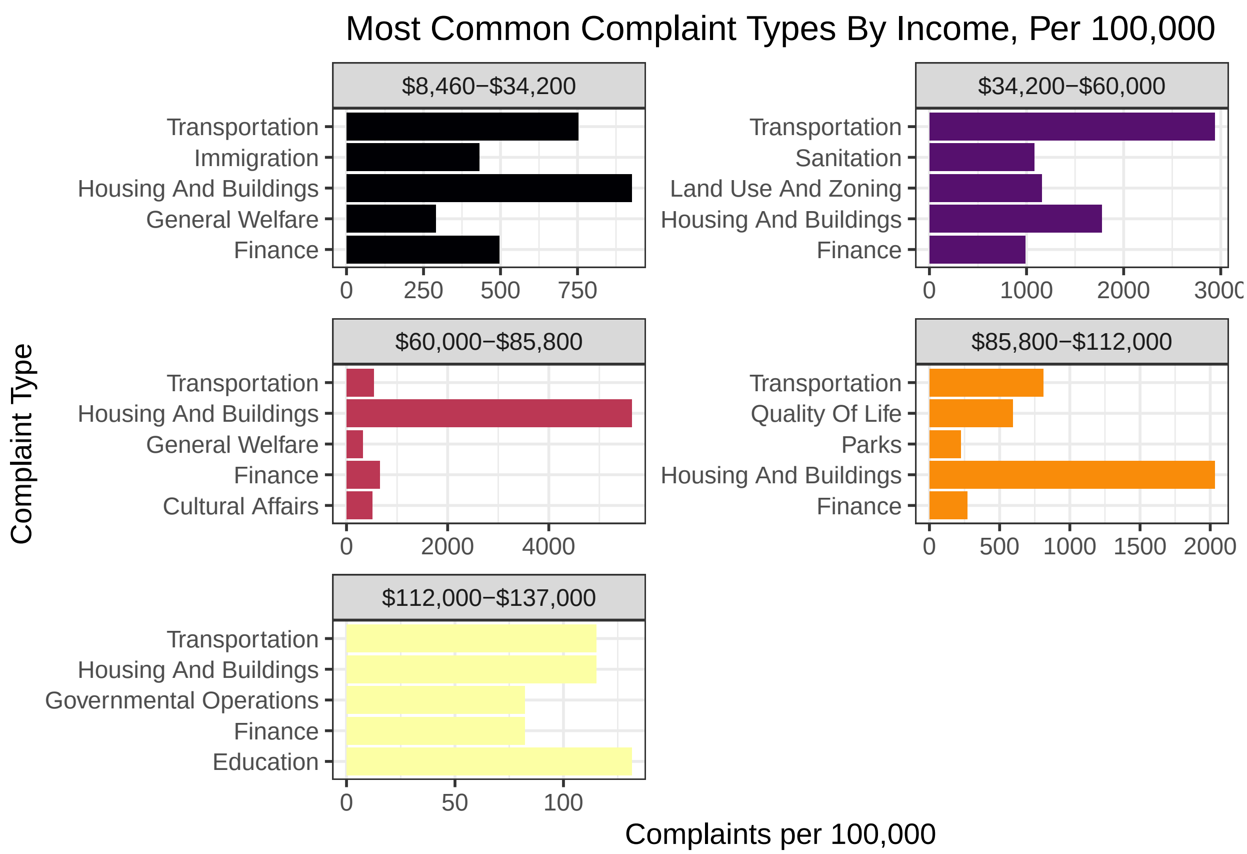
The graph dividing zip codes into different categories of median age is striking. To be clear, most zip codes have low median ages - though the data were split into five different intervals, zero zip codes fell into the 61.7-73.4 range, and almost every zip code had a median age below 50. However, there were seriously different trends in the different age groups. The first two groups again saw housing and buildings, transportation, and finance receive the largest number of complaints. However, zip codes with a median age of 26.6-38.3 saw immigration and general welfare round out their top five most common complaint types, while zip codes with a median age of 38.3-50 instead saw sanitation and parks fill that role. This is remarkably similar to the Black/Hispanic vs White split in the first of these graphs, and suggests that the prevalence of those specific complaint types may well be dictated largely by income. Meanwhile, zip codes with a median age from 50-61.7 were quite different - utilities were far and away the most common complaint type, followed by the usual transportation, housing and buildings (but not the most common this time!), and finance, with education rounding out the rear. Meanwhile, for zip codes with a median age of 73.4-85.1 (so, essentially, retirement homes), complaints about finance were far and away the most common (as people who are retired often have such issues), with a reasonable number of complaints about transportation (again, logical, as many old people slowly lose their ability and right to drive) and almost no other complaint types being filed regularly (with the most common remaining group being governmental operations, again related to retirement benefits).

While none of these plots can be said to truly display the actual trends by these demographics, as the proxy making up entire zip codes blunts a lot of the data, many of these trends indeed make a lot of sense. In the next plot, we will look more closely at the impact of income that was hypothesized as a result of some of these plots.

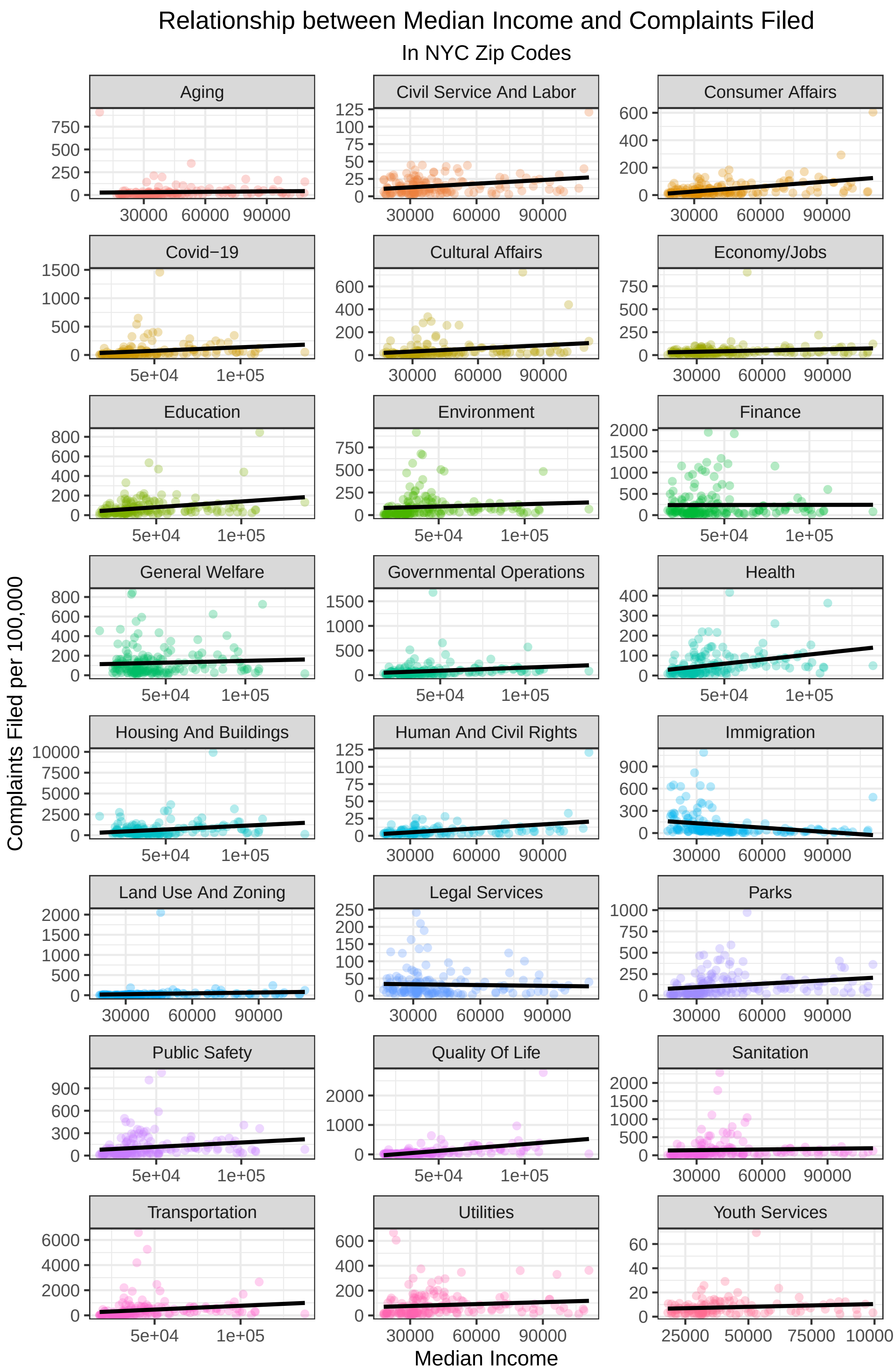
Plot 2: Complaints by Income

In order to consider how income affects these different complaint types, I drew two different plots (call it one and humor me, they're the same general idea). In the first, I split the data into five equal intervals of median income, selected the five most common complaint types per 100,000 in each of them, and displayed the output in a bar chart. This enables us to see what the most common complaint types are among people of different social classes in the same way as the previous demographic plots - not on a case-by-case basis, but still on an aggregate level. In the second, I calculated the per-100,000 complaint rate of each type of complaint according to zip code, filtered out any complaint type that showed up 100 or fewer times, and plotted it against median income. Though most of these trends are clearly not linear, the linear regression lines can still reveal something about the nature of these relationships - namely, whether richer zip codes tend to file each complaint type more or less frequently than poorer zip codes do.

```r
full %>%
  mutate(inc = cut_interval(med_income, 5)) %>%
  group_by(zip, complaint_type) %>%
  summarize(inc, pop, n = n()) %>%
  mutate(per100 = (n/pop)*100000) %>%
  ungroup() %>%
  group_by(inc, complaint_type) %>%
  summarize(n = n(), per100 = mean(per100, na.rm = T)) %>%
  #group_by(inc, complaint_type) %>%
  #summarize(n = n(), popsum = sum(pop)) %>%
  #mutate(per100 = (n/popsum)*100000) %>%
  arrange(desc(per100)) %>%
  na.omit() %>%
  slice(1:5) %>%
  mutate(income = case_when(inc == "[8.46e+03,3.42e+04]" ~ "$8,460-$34,200",
                            inc == "(3.42e+04,6e+04]" ~ "$34,200-$60,000",
                            inc == "(6e+04,8.58e+04]" ~ "$60,000-$85,800",
                            inc == "(8.58e+04,1.12e+05]" ~ "$85,800-$112,000",
                            inc == "(1.12e+05,1.37e+05]" ~ "$112,000-$137,000"))%>%
  mutate(across(income, factor, levels = c("$8,460-$34,200", "$34,200-$60,000",
                                           "$60,000-$85,800", "$85,800-$112,000",
                                           "$112,000-$137,000"))) %>%
  ggplot(aes(x = per100, y = complaint_type, fill = income)) +
  geom_col() +
  facet_wrap(~income, scales = "free", ncol = 2) +
  theme_bw() +
  guides(fill = "none") +
  labs(title = "Most Common Complaint Types By Income, Per 100,000",
       x = "Complaints per 100,000", y = "Complaint Type") +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_fill_viridis(discrete = T, option = "B")
```




```
full %>%
  group_by(zip, complaint_type) %>%
  summarize(n = n(), pop, med_income) %>%
  mutate(per100 = (n/pop)*100000) %>%
  ungroup() %>%
  group_by(complaint_type, med_income) %>%
  summarize(n = n(), per100 = mean(per100, na.rm = T)) %>%
  ungroup() %>%
  group_by(complaint_type) %>%
  mutate(nComplaint = n()) %>%
  filter(nComplaint > 100) %>%
  ggplot(aes(x = med_income, y = per100, color = complaint_type)) +
  geom_point(alpha = 0.3) +
  geom_smooth(method = "lm", se = F, color = "black") +
  facet_wrap(~complaint_type, scales = "free", ncol = 3) +
  theme_bw() +
  guides(color = "none") +
  labs(title = "Relationship between Median Income and Complaints Filed",
       subtitle = "In NYC Zip Codes", x = "Median Income",
       y = "Complaints Filed per 100,000") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```


Relationship between Median Income and Complaints Filed
In NYC Zip Codes
Aging
Civil Service And Labor
Consumer Affairs
Covid−19
Cultural Affairs
Economy/Jobs
Education
Environment
Finance
General Welfare
Governmental Operations
Health
Housing And Buildings
Human And Civil Rights
Immigration
Land Use And Zoning
Legal Services
Parks
Public Safety
Quality Of Life
Sanitation
Transportation
Utilities
Youth Services
Complaints Filed per 100,000
Median Income

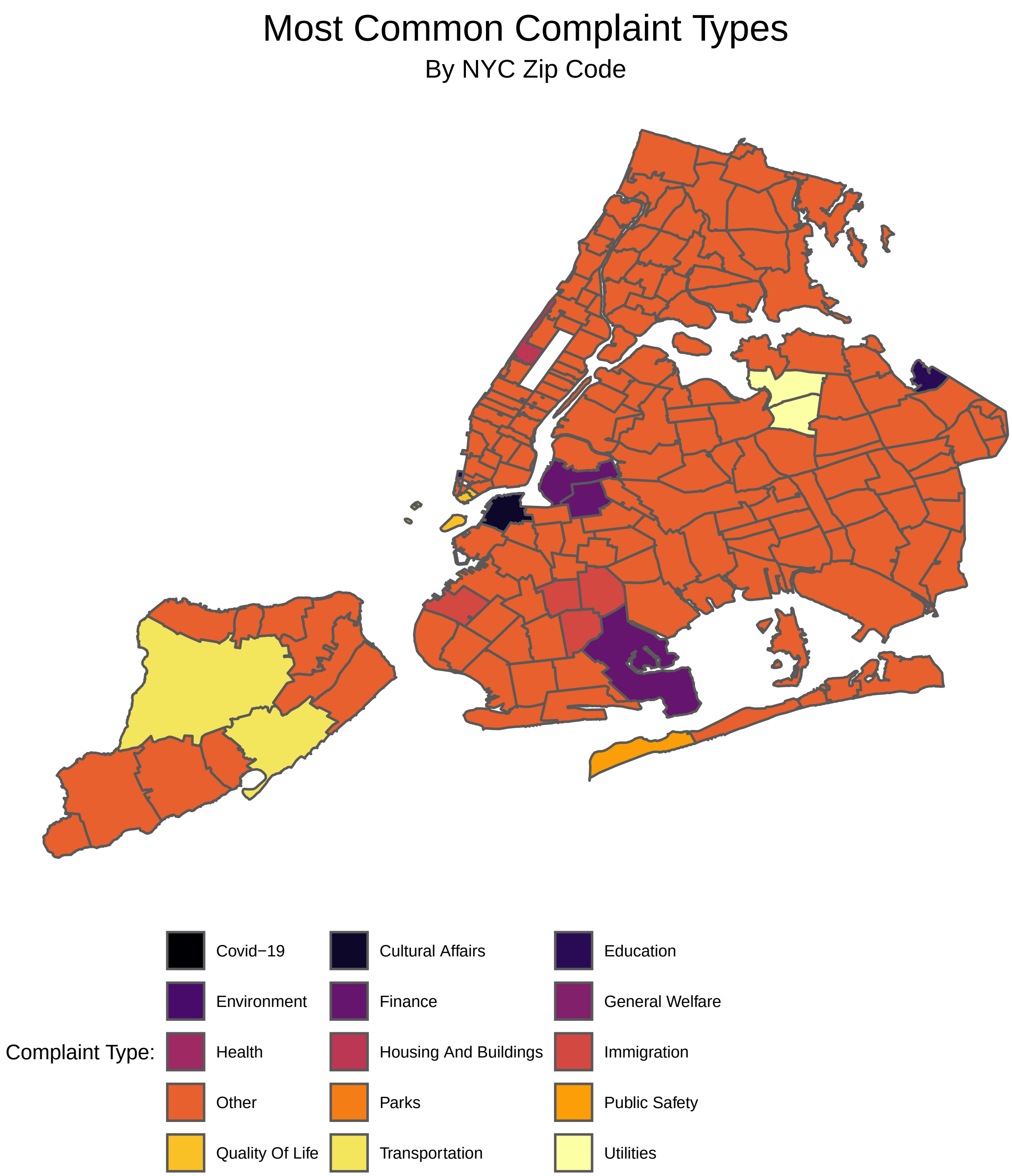
In the first graph, a couple of notable trends show up. First of all, even on a per-100K basis, almost no complaints are filed in zip codes falling under the richest bracket - probably logical, when considering the rate at which rich people actually use government services (instead of just paying for the problems themselves). Rates are also somewhat lower in the lowest bracket of zip codes. As was somewhat predicted by the takeaways from the previous set of plots, immigration was only an issue in the lowest bracket of incomes, while general welfare was only in that group and the $60K-$85.8K group. Like usual, housing and buildings, transportation, and finance were in the top five issues across all categories. There are some other notable trends in this plot, such as education and governmental operations only being issues in the richest zip codes or quality of life and parks only being an issue for the next-richest set.

The second graph suggests that a) almost all relationships between income and per-100K complaints are incredibly weak and b) that almost all of those relationships are positive - that is, that people at higher incomes file complaints at slightly higher rates than do people at lower incomes (once again, through the proxy of aggregate zip code median incomes). Notably, there are two complaint types for which there is a negative relationship - immigration and legal services. That is, similarly, a logical trend, given the nature of these complaint types in NYC.

Plot 3: Distribution of Complaints

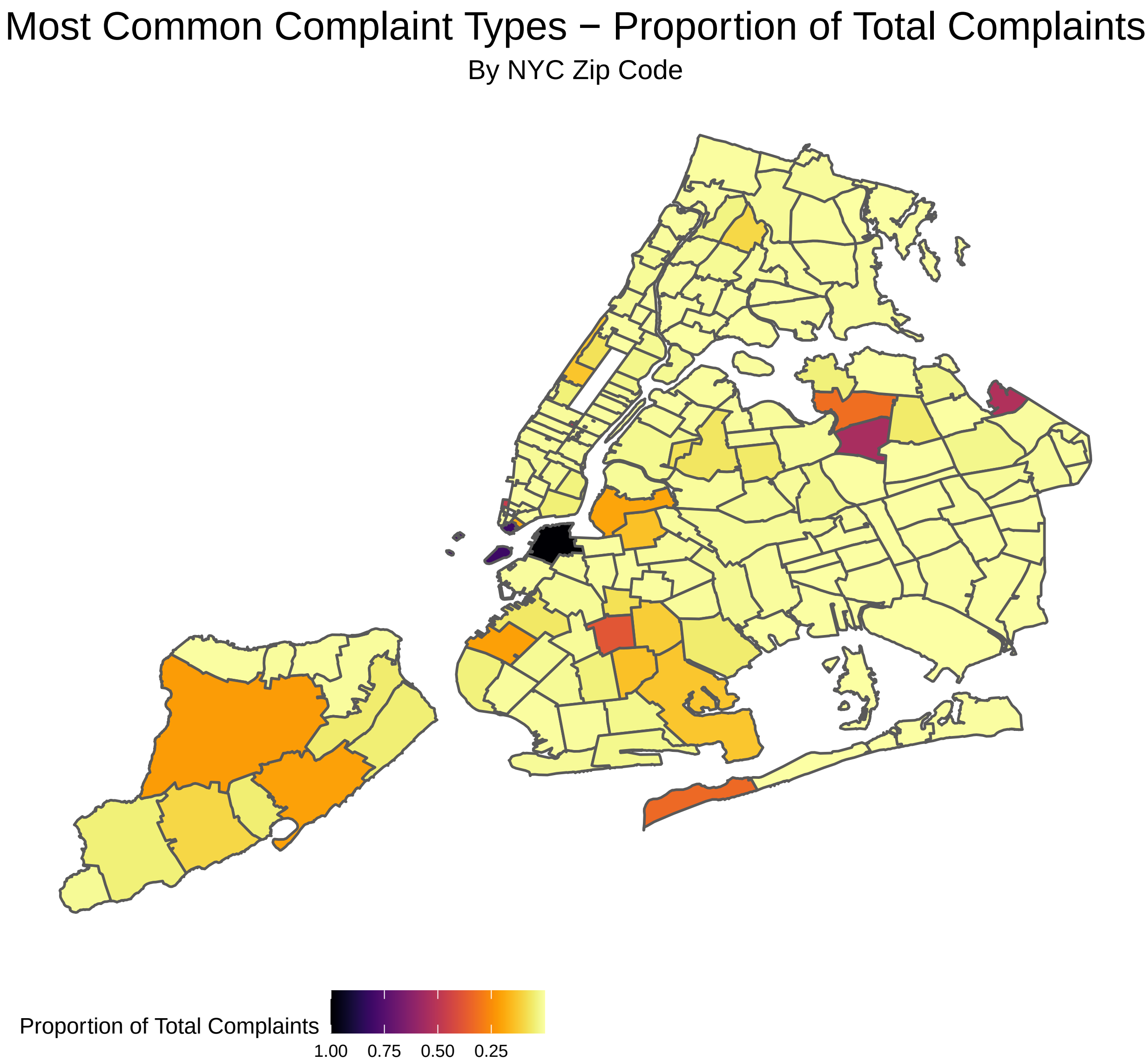
In the last set of plots, I wanted to look at the geographic distribution of these different complaint types. Thus, using the geography data from the `nycdogs` `nyczips` package, I drew a series of four maps of the city. The first is a map of the most common complaint types filed in each NYC zip code, marked as "Other" when no complaint type was over 10% of the total complaints in that zip code (which was the case for most of them). This enables us to see whether certain complaints are more common in different geographic areas of the city. The second map shows what percentage of the total number of complaints filed in each zip code were the most common complaint type, an addendum to the first map (originally, these were supposed to be one combined map, but there was no interpretable way to include this much information in the one plot). The third is simply a map of median incomes, allowing us to see geographic trends, and the fourth is an image of how many complaints per 100K were filed in each zip code.

```r
full %>%
  group_by(zip) %>%
  count(complaint_type) %>%
  arrange(desc(n)) %>%
  mutate(prop = n/sum(n)) %>%
  slice(1) %>%
  ungroup() %>%
  group_by(complaint_type) %>%
  mutate(complaint_prop = n/sum(n)) %>%
  mutate(complaint = case_when(complaint_prop < .1 ~ "Other",
                                T ~ complaint_type)) %>%
  inner_join(zips) %>%
  ggplot(aes(fill = complaint, geometry = geometry)) +
  geom_sf() +
  theme_nymap() +
  theme(legend.position = "bottom",
        plot.title = element_text(hjust = 0.5, size = 16),
        plot.subtitle = element_text(hjust = 0.5, size = 11),
        legend.text = element_text(size = 6.5)) +
  scale_fill_viridis(discrete = T, option = "B") +
  labs(title = "Most Common Complaint Types", subtitle = "By NYC Zip Code",
       fill = "Complaint Type:") +
  guides(fill = guide_legend(nrow = 5, byrow = T))
```



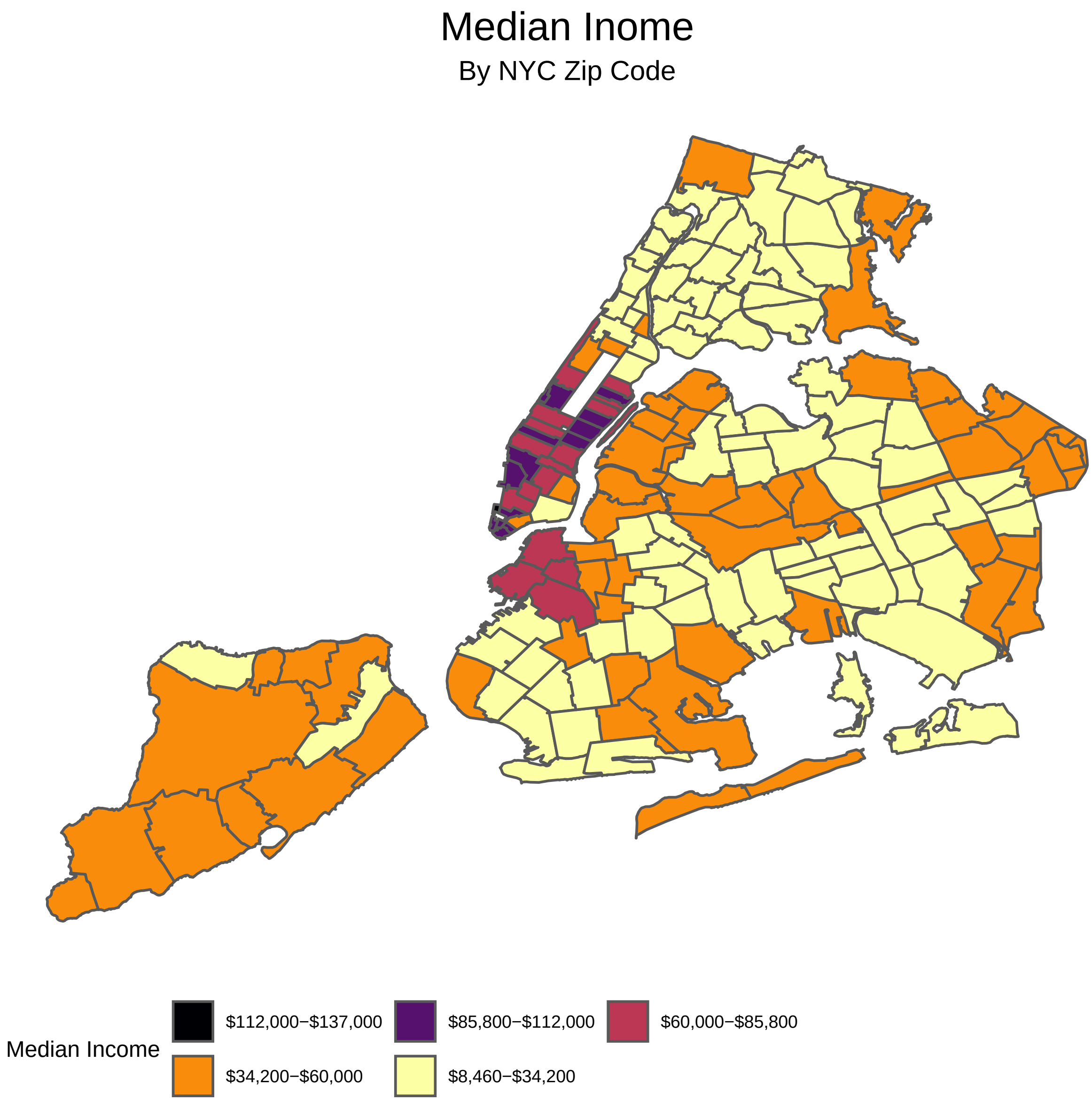

```
full %>%
  group_by(zip) %>%
  count(complaint_type) %>%
  arrange(desc(n)) %>%
  mutate(prop = n/sum(n)) %>%
```

```
slice(1) %>%
ungroup() %>%
group_by(complaint_type) %>%
mutate(complaint_prop = n/sum(n)) %>%
mutate(complaint = case_when(complaint_prop < .1 ~ "Other",
                             T ~ complaint_type)) %>%
inner_join(zips) %>%
ggplot(aes(fill = complaint_prop, geometry = geometry)) +
geom_sf() +
theme_nymap() +
scale_fill_viridis(option = "B", trans = "reverse") +
labs(title = "Most Common Complaint Types - Proportion of Total Complaints",
     subtitle = "By NYC Zip Code", fill = "Proportion of Total Complaints") +
theme(legend.position = "bottom",
      plot.title = element_text(hjust = 0.5, size = 16),
      plot.subtitle = element_text(hjust = 0.5, size = 11))
```



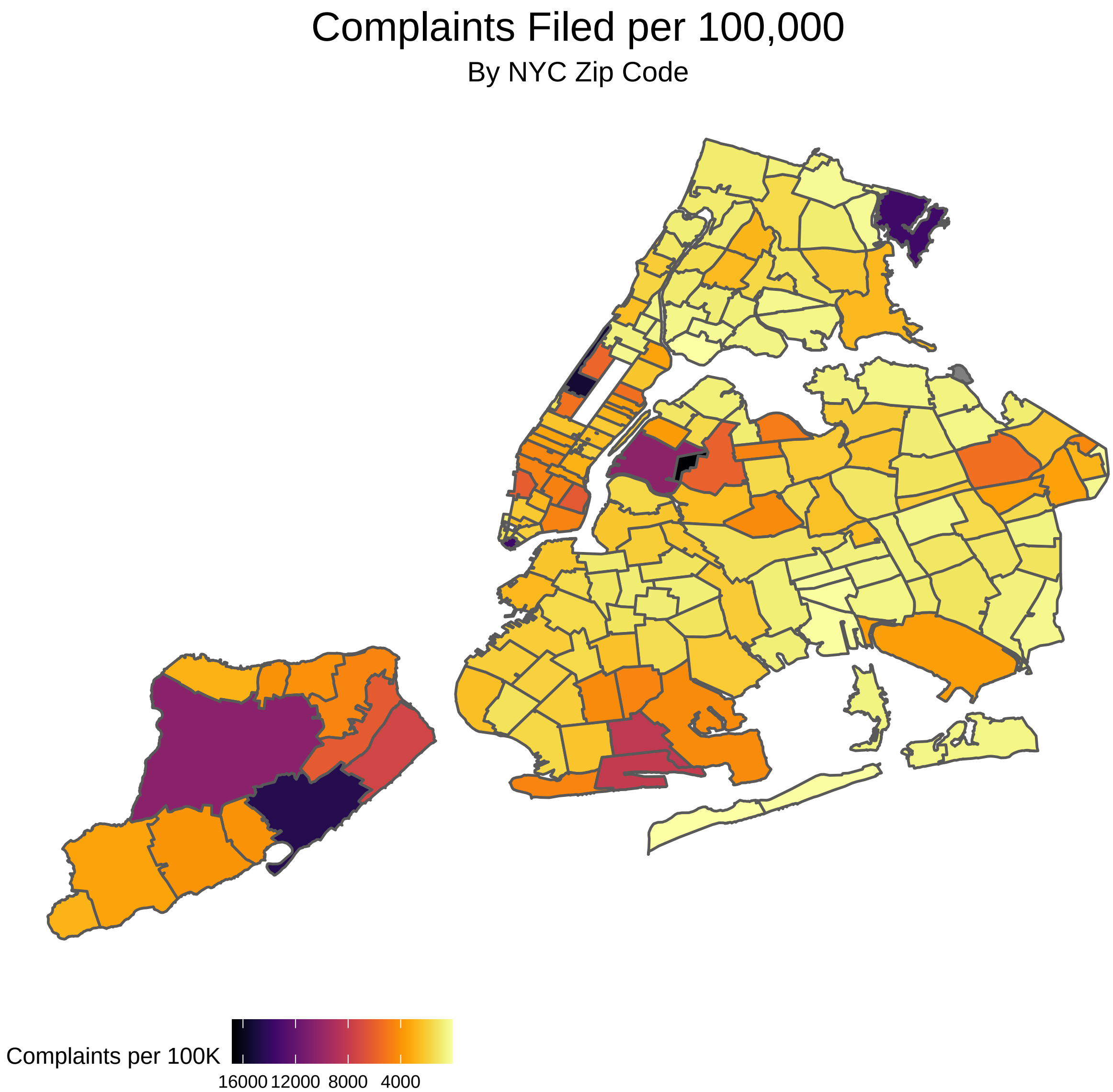

```r
full %>%
  mutate(inc = cut_interval(med_income, 5)) %>%
  mutate(income = case_when(inc == "[8.46e+03,3.42e+04]" ~ "$8,460-$34,200",
                            inc == "(3.42e+04,6e+04]" ~ "$34,200-$60,000",
                            inc == "(6e+04,8.58e+04]" ~ "$60,000-$85,800",
```

```
                        inc == "(8.58e+04,1.12e+05]" ~ "$85,800-$112,000",
                        inc == "(1.12e+05,1.37e+05]" ~ "$112,000-$137,000"))%>%
mutate(across(income, factor, levels = c("$8,460-$34,200", "$34,200-$60,000",
                                      "$60,000-$85,800", "$85,800-$112,000",
                                      "$112,000-$137,000"))) %>%
group_by(zip) %>%
summarize(income, geometry) %>%
slice(1) %>%
na.omit() %>%
ggplot(aes(fill = fct_rev(income), geometry = geometry)) +
geom_sf() +
theme_nymap() +
scale_fill_viridis(discrete = T, option = "B",
                   guide = guide_legend(reverse = T)) +
labs(title = "Median Inome", subtitle = "By NYC Zip Code",
     fill = "Median Income") +
theme(legend.position = "bottom",
      plot.title = element_text(hjust = 0.5, size = 16),
      plot.subtitle = element_text(hjust = 0.5, size = 11),
      legend.text = element_text(size = 6.5)) +
guides(fill = guide_legend(nrow = 2, byrow = T))
```




```r
full %>%
  group_by(zip) %>%
  summarize(n = n(), pop, geometry) %>%
  mutate(per100 = (n/pop)*100000) %>%
  slice(1) %>%
```

```r
ggplot(aes(fill = per100, geometry = geometry)) +
geom_sf() +
theme_nymap() +
scale_fill_viridis(option = "B", trans = "reverse") +
labs(title = "Complaints Filed per 100,000",
     subtitle = "By NYC Zip Code", fill = "Complaints per 100K") +
theme(legend.position = "bottom",
      plot.title = element_text(hjust = 0.5, size = 16),
      plot.subtitle = element_text(hjust = 0.5, size = 11))
```



The first map shows the most common complaint types in each zip code. Strikingly, almost no zip codes were dominated by a single complaint type. Even housing and buildings, which ran away with the title of most common complaint type, was strikingly missing from the map in which almost no zip codes were over 10% comprised of any one complaint type. The second map makes this very clear - it is visible that almost every zip code had an incredibly low proportion of complaints filed from their leading complaint type.

The third map, showing median income, is almost a damning tale of income inequality in NYC. All of the wealth is clustered in Manhattan (and a chunk of Brookly just across from it), while no zip code reaching even the third category of income is found outside of that area. Strikingly, however, this does not seem to match up very well with any other map considered yet - Manhattan did not stand out in the vast majority of complaint variables yet considered. This more or less continues in the fourth map, where there is a slightly higher proportion of per-100K complaints filed in Manhattan zip codes than the rest of the city but not really anything to speak of. The biggest anomaly here, actually, is Staten Island - which seems quite high in terms of per-100K complaints filed. In all fairness, if I lived in Staten Island, I too think I would file a substantial number of complaints.

4. Short final discussion: Scope and limits of this data

There are certainly some limits to these data. While these data are very thorough in terms of the location that the complaint originated from, what it pertained to, and when it was filed (all excluding `NA` values, of course), they are seriously lacking in some important other information, including the demographics of who filed the complaint. That would be important information for any government agency to include, as they should strive to see exactly who it is that uses their system and adapt their system to meet the people who are being underserved by it. Unfortunately, they are unable to do so.

That also serves as a major drawback in these plots. Instead of being able to aggregate complaints across specific demographics, we are reduced to looking at them in zip codes (or other geographic areas) composed of a *plurality* of each demographic group. Great amounts of information are lost in doing this, and a lot of assumptions are made. Though the plots generally seem to make sense in the context of these demographics, the specific information would bolster them greatly.