Class 17 Mini-Project: COVID-19 Vaccination Rates

Kira Jung

Date: 2023/06/12

The goal of this hands-on mini-project is to examine and compare the Covid-19 vaccination rates around San Diego.

Getting Started

First, we need to read the .csv that includes data about statewide COVID-19 vaccines administered by ZIP code:

```
vax <- read.csv("covid19vaccinesbyzipcode_test.csv")
head(vax)
```

```
  as_of_date zip_code_tabulation_area local_health_jurisdiction      county
1 2021-01-05                   94579                   Alameda     Alameda
2 2021-01-05                   93726                    Fresno      Fresno
3 2021-01-05                   94305               Santa Clara Santa Clara
4 2021-01-05                   93704                    Fresno      Fresno
5 2021-01-05                   94403                 San Mateo   San Mateo
6 2021-01-05                   93668                    Fresno      Fresno
  vaccine_equity_metric_quartile                vem_source
1                              3 Healthy Places Index Score
2                              1 Healthy Places Index Score
3                              4 Healthy Places Index Score
4                              1 Healthy Places Index Score
5                              4 Healthy Places Index Score
6                              1    CDPH-Derived ZCTA Score
  age12_plus_population age5_plus_population tot_population
```

```
1               19192.7                20872          21883
2               33707.7                39067          42824
3               15716.9                16015          16397
4               24803.5                27701          29740
5               37967.5                41530          44408
6                1013.4                 1199           1219
  persons_fully_vaccinated persons_partially_vaccinated
1                       NA                           NA
2                       NA                           NA
3                       NA                           NA
4                       NA                           NA
5                       NA                           NA
6                       NA                           NA
  percent_of_population_fully_vaccinated
1                                     NA
2                                     NA
3                                     NA
4                                     NA
5                                     NA
6                                     NA
  percent_of_population_partially_vaccinated
1                                         NA
2                                         NA
3                                         NA
4                                         NA
5                                         NA
6                                         NA
  percent_of_population_with_1_plus_dose booster_recip_count
1                                     NA                  NA
2                                     NA                  NA
3                                     NA                  NA
4                                     NA                  NA
5                                     NA                  NA
6                                     NA                  NA
  bivalent_dose_recip_count eligible_recipient_count
1                        NA                        4
2                        NA                        2
3                        NA                        8
4                        NA                        5
5                        NA                        7
6                        NA                        0
  eligible_bivalent_recipient_count
1                                 4
```

```
2                                  2
3                                  8
4                                  5
5                                  7
6                                  0
                                                                    redacted
1 Information redacted in accordance with CA state privacy requirements
2 Information redacted in accordance with CA state privacy requirements
3 Information redacted in accordance with CA state privacy requirements
4 Information redacted in accordance with CA state privacy requirements
5 Information redacted in accordance with CA state privacy requirements
6 Information redacted in accordance with CA state privacy requirements
```

Q1: What column details the total number of people fully vaccinated? *Answer*: `persons_fully_vaccinated`

Q2: What column details the Zip code tabulation area? *Answer*: `zip_code_tabulation_area`

Q3: What is the earliest date in this dataset? *Answer*: 2021-01-05

Q4:What is the latest date in this dataset? *Answer*: 2023-05-23

Call the `skim()` function from the skimr package to get a quick overview of this dataset.

```
skimr::skim_without_charts(vax)
```

Table 1: Data summary

Name	vax
Number of rows	220500
Number of columns	19
Column type frequency:	
character	5
numeric	14
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
as_of_date	0	1	10	10	0	125	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
local_health_jurisdiction	0	1	0	15	625	62	0
county	0	1	0	15	625	59	0
vem_source	0	1	15	26	0	3	0
redacted	0	1	2	69	0	2	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
zip_code_tabulation_area	0	1.00	93665.11	817.38	90001	92257.75	93658.50	95380.50	97635.0
vaccine_equity_metric_quartile	10877	0.95	2.44	1.11	1	1.00	2.00	3.00	4.0
age12_plus_population	0	1.00	18895.04	18993.87	0	1346.95	13685.10	31756.18	88556.7
age5_plus_population	0	1.00	20875.22	21105.97	0	1460.50	15364.00	34877.00	101902.0
tot_population	10750	0.95	23372.77	22628.50	12	2126.00	18714.00	38168.00	111165.0
persons_fully_vaccinated	17711	0.92	14272.72	15264.17	11	954.00	8990.00	23782.00	87724.0
persons_partially_vaccinated	17711	0.92	1711.05	2071.56	11	164.00	1203.00	2550.00	42259.0
percent_of_population_fully_vaccinated	22579	0.91	0.58	0.25	0	0.44	0.62	0.75	1.0
percent_of_population_partially_vaccinated	22579	0.91	0.08	0.09	0	0.05	0.06	0.08	1.0
percent_of_population_with_12_plus_dose	23732	0.89	0.64	0.24	0	0.50	0.68	0.82	1.0
booster_recip_count	74388	0.66	6373.43	7751.70	11	328.00	3097.00	10274.00	60022.0
bivalent_dose_recip_count	159956	0.27	3407.91	4010.38	11	222.00	1832.00	5482.00	29484.0
eligible_recipient_count	0	1.00	13120.40	15126.17	0	534.00	6663.00	22517.25	87437.0
eligible_bivalent_recipient_count	0	1.00	13016.51	15199.08	0	266.00	6562.00	22513.00	87437.0

Q5: How many numeric columns are in this dataset? *Answer*: There are 14 numeric columns.

Q6: Note that there are "missing values" in the dataset. How many NA values there in the persons_fully_vaccinated column?

```
sum(is.na(vax$persons_fully_vaccinated))
```

```
[1] 17711
```

Answer: There are 17711 missing values in the persons_fully_vaccinated column.

Q7: What percent of persons_fully_vaccinated values are missing (to 2 significant figures)?

```
(1-0.9196780)*100
```

```
[1] 8.0322
```

Answer: 8.03% of persons_fully_vaccinated values are missing (used the value found in column complete_rate).

Q8: [Optional]: Why might this data be missing?] *Answer*: The data could be missing because whatever method they are using to track every individual's data may not be efficient in collecting every peice of data in the table.

Working with dates

One of the "character" columns of the data is `as_of_date`, which contains dates in the Year-Month-Day format.

Using the **lubridate** package:

```
library(lubridate)
```

```
Attaching package: 'lubridate'

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

    date, intersect, setdiff, union
```

Today's date is:

```
today()
```

```
[1] "2023-06-12"
```

To make our `as_of_date` column usable for mathematical operations, we will convert our date data into a lubridate format.

```
vax$as_of_date <- ymd(vax$as_of_date)
```

How many days have passed since the first vaccination reported in this dataset?

```
today() - vax$as_of_date[1]
```

```
Time difference of 888 days
```

Using the last and the first date value we can now determine how many days the dataset span.

```
vax$as_of_date[nrow(vax)] - vax$as_of_date[1]
```

```
Time difference of 868 days
```

Q9: How many days have passed since the last update of the dataset?

```
today() - vax$as_of_date[220500]
```

```
Time difference of 20 days
```

Answer: 20 days have passed since the last update of the dataset (different from the answer '7' in the lab because our today dates are different).

Q10: How many unique dates are in the dataset (i.e. how many different dates are detailed)?

```
length(unique(vax$as_of_date))
```

```
[1] 125
```

Answer: There are 125 unique dates in the dataset.

Working with ZIP codes

One of the numeric columns in the dataset (namely vax$zip_code_tabulation_area) are actually ZIP codes - a postal code used by the United States Postal Service (USPS).

Using the **zipcodeR** package to make working with the zip codes easier.

```
library(zipcodeR)
```

```
The legacy packages maptools, rgdal, and rgeos, underpinning this package
will retire shortly. Please refer to R-spatial evolution reports on
https://r-spatial.org/r/2023/05/15/evolution4.html for details.
This package is now running under evolution status 0
```

Finding the centroid of the La Jolla 92037 ZIP code area:

```
geocode_zip('92037')
```

```
# A tibble: 1 x 3
  zipcode   lat   lng
  <chr>   <dbl> <dbl>
1 92037    32.8 -117.
```

Calculating the distance between the centroids of any two ZIP codes in miles:

```
zip_distance('92037','92109')
```

```
  zipcode_a zipcode_b distance
1     92037     92109     2.33
```

Pulling census data about ZIP code areas:

```
reverse_zipcode(c('92037', "92109"))
```

```
# A tibble: 2 x 24
  zipcode zipcode_type major_city post_office_city common_city_list county state
  <chr>   <chr>        <chr>      <chr>                     <blob> <chr>  <chr>
1 92037   Standard     La Jolla   La Jolla, CA           <raw 20 B> San D~ CA
2 92109   Standard     San Diego  San Diego, CA          <raw 21 B> San D~ CA
# i 17 more variables: lat <dbl>, lng <dbl>, timezone <chr>,
#   radius_in_miles <dbl>, area_code_list <blob>, population <int>,
#   population_density <dbl>, land_area_in_sqmi <dbl>,
#   water_area_in_sqmi <dbl>, housing_units <int>,
#   occupied_housing_units <int>, median_home_value <int>,
#   median_household_income <int>, bounds_west <dbl>, bounds_east <dbl>,
#   bounds_north <dbl>, bounds_south <dbl>
```

We can also use `reverse_zipcode()` to pull census data later on for any or all ZIP code areas we might be interested in:

```
zipdata <- reverse_zipcode(vax$zip_code_tabulation_area)
```

Focus on the San Diego area

Focus in on the San Diego County area by restricting ourselves first to vax$county == "San Diego" entries.

We can do this by using the **dplyr** package.

```
library(dplyr)
```

```
Attaching package: 'dplyr'

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

    filter, lag

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

    intersect, setdiff, setequal, union
```

```
sd <- filter(vax, county == "San Diego")

nrow(sd)
```

```
[1] 13375
```

Using **dplyr* is more convenient when subsetting across multiple criteria.

```
sd.10 <- filter(vax, county == "San Diego" & age5_plus_population > 10000)
```

Q11: How many distinct zip codes are listed for San Diego County?

```
sd %>% group_by(zip_code_tabulation_area) %>% summarise()
```

```
# A tibble: 107 x 1
   zip_code_tabulation_area
                      <int>
 1                    91901
 2                    91902
```

```
 3                    91905
 4                    91906
 5                    91910
 6                    91911
 7                    91913
 8                    91914
 9                    91915
10                    91916
# i 97 more rows
```

Answer: There are 107 distinct zip codes listed for San Diego County.

Q12: What San Diego County Zip code area has the largest population in this dataset?

```
which.max(sd$age12_plus_population)
```

```
[1] 87
```

```
sd[87, ]
```

```
   as_of_date zip_code_tabulation_area local_health_jurisdiction    county
87 2021-01-05                   92154                San Diego San Diego
   vaccine_equity_metric_quartile                vem_source
87                              2 Healthy Places Index Score
   age12_plus_population age5_plus_population tot_population
87               76365.2                82971          88979
   persons_fully_vaccinated persons_partially_vaccinated
87                       18                         1403
   percent_of_population_fully_vaccinated
87                               0.000202
   percent_of_population_partially_vaccinated
87                                  0.015768
   percent_of_population_with_1_plus_dose booster_recip_count
87                                 0.01597                 NA
   bivalent_dose_recip_count eligible_recipient_count
87                        NA                       18
   eligible_bivalent_recipient_count
87                                18
                                                                  redacted
87 Information redacted in accordance with CA state privacy requirements
```

Answer: 92154

Q13: What is the overall average (with 2 decimal numbers) "Percent of Population Fully Vaccinated" value for all San Diego "County" as of "2023-05-23"?

```
df_23.05.23 <- sd %>% filter(as_of_date == "2022-11-15")

df_23.05.23_clean <- df_23.05.23 %>% filter(!is.na(percent_of_population_fully_vaccinated)

mean(df_23.05.23_clean$percent_of_population_fully_vaccinated)
```

```
[1] 0.7392451
```

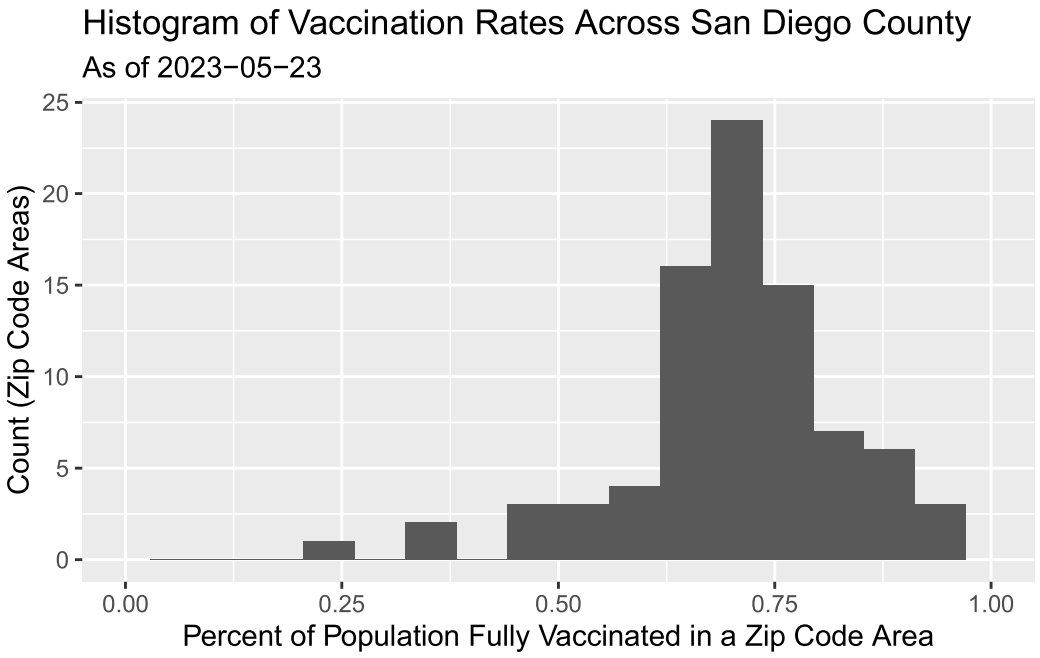
Answer: 0.74 is the overall average.

Q14: Using either ggplot or base R graphics make a summary figure that shows the distribution of Percent of Population Fully Vaccinated values as of "2023-05-23"?

```
library(ggplot2)

ggplot(df_23.05.23_clean) + aes(x = percent_of_population_fully_vaccinated) + geom_histogr
```

```
Warning: Removed 2 rows containing missing values (`geom_bar()`).
```



Focus on UCSD/La Jolla

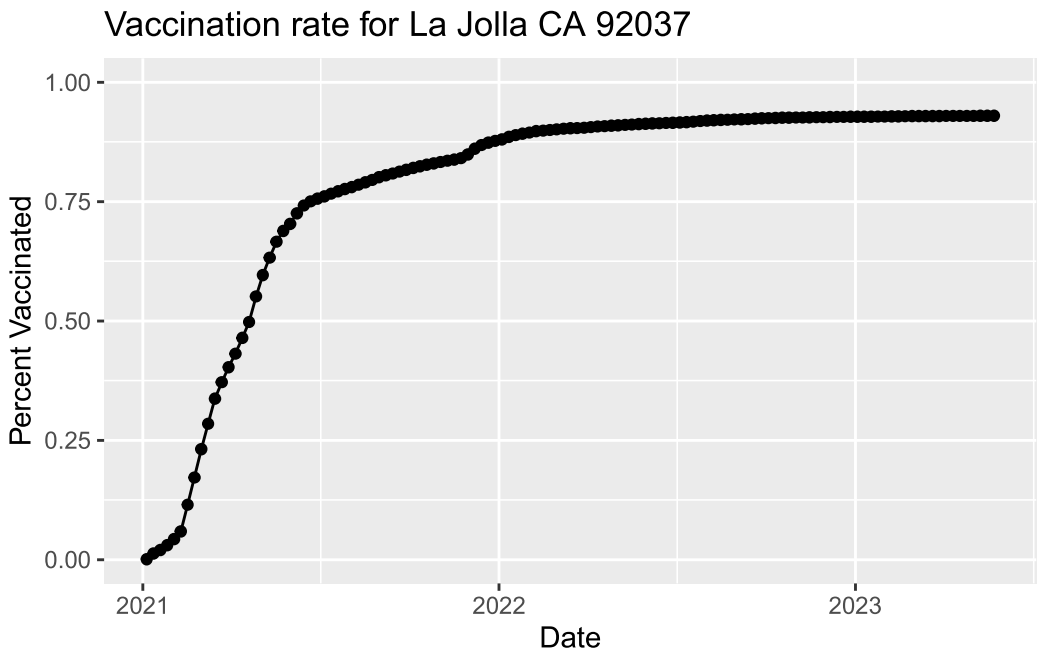
```
ucsd <- filter(sd, zip_code_tabulation_area=="92037")
ucsd[1,]$age5_plus_population
```

```
[1] 36144
```

Q15: Using ggplot make a graph of the vaccination rate time course for the 92037 ZIP code area:

```
ggplot(ucsd) + aes(x = as_of_date, y = ucsd$percent_of_population_fully_vaccinated) + geom
```

```
Warning: Use of `ucsd$percent_of_population_fully_vaccinated` is discouraged.
i Use `percent_of_population_fully_vaccinated` instead.
Use of `ucsd$percent_of_population_fully_vaccinated` is discouraged.
i Use `percent_of_population_fully_vaccinated` instead.
```



Comparing to similar sized areas

Return to the full dataset and look across every zip code area with a population at least as large as that of 92037 on as_of_date "2023-05-23".

```
vax.36 <- filter(vax, age5_plus_population > 36144 & as_of_date == "2023-05-23")
```

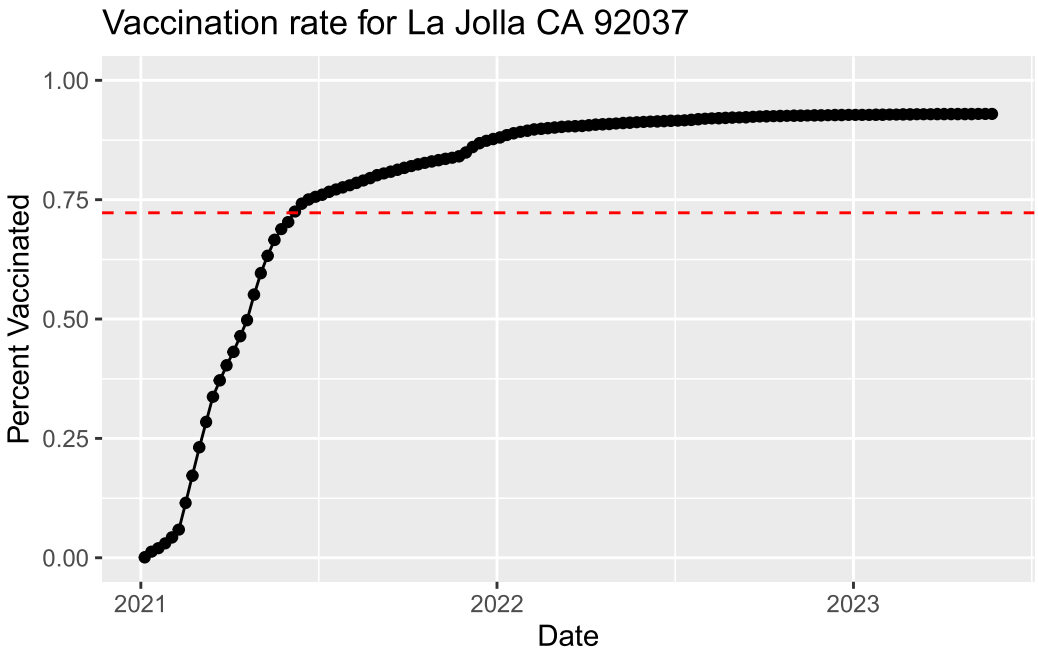
Q16: Calculate the mean "Percent of Population Fully Vaccinated" for ZIP code areas with a population as large as 92037 (La Jolla) as_of_date "2023-05-23". Add this as a straight horizontal line to your plot from above with the geom_hline() function:

```
mean(vax.36$percent_of_population_fully_vaccinated)
```

```
[1] 0.7225892
```

```
ggplot(ucsd) + aes(x = as_of_date, y = ucsd$percent_of_population_fully_vaccinated) + geom
```

```
Warning: Use of `ucsd$percent_of_population_fully_vaccinated` is discouraged.
i Use `percent_of_population_fully_vaccinated` instead.
Use of `ucsd$percent_of_population_fully_vaccinated` is discouraged.
i Use `percent_of_population_fully_vaccinated` instead.
```



Q17: What is the 6 number summary (Min, 1st Qu., Median, Mean, 3rd Qu., and Max) of the "Percent of Population Fully Vaccinated" values for ZIP code areas with a population as large as 92037 (La Jolla) as_of_date "2023-05-23"?

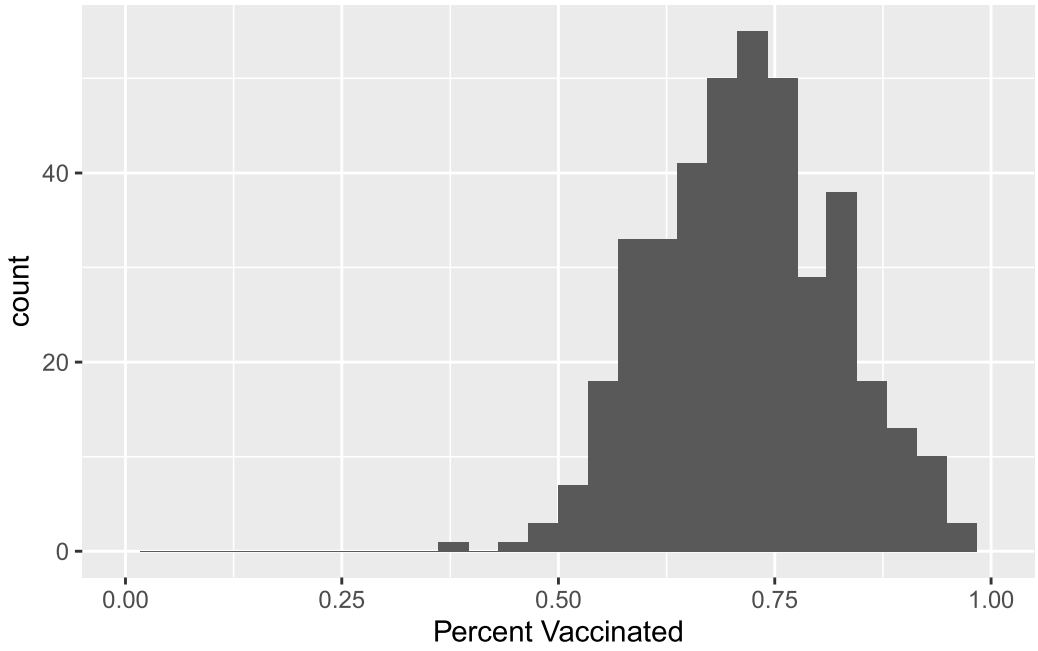
```
summary(vax.36$percent_of_population_fully_vaccinated)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.3816  0.6469  0.7207  0.7226  0.7924  1.0000
```

Q18: Using ggplot generate a histogram of this data.

```
ggplot(vax.36) + aes(x = percent_of_population_fully_vaccinated) + geom_histogram(bins = 3
```

```
Warning: Removed 2 rows containing missing values (`geom_bar()`).
```



Q19:Is the 92109 and 92040 ZIP code areas above or below the average value you calculated for all these above?

```
vax %>% filter(as_of_date == "2023-05-23") %>% filter(zip_code_tabulation_area=="92040") %
```

```
  percent_of_population_fully_vaccinated
1                               0.552434
```

```
vax %>% filter(as_of_date == "2023-05-23") %>% filter(zip_code_tabulation_area=="92109") %
```

```
  percent_of_population_fully_vaccinated
1                                0.69487
```

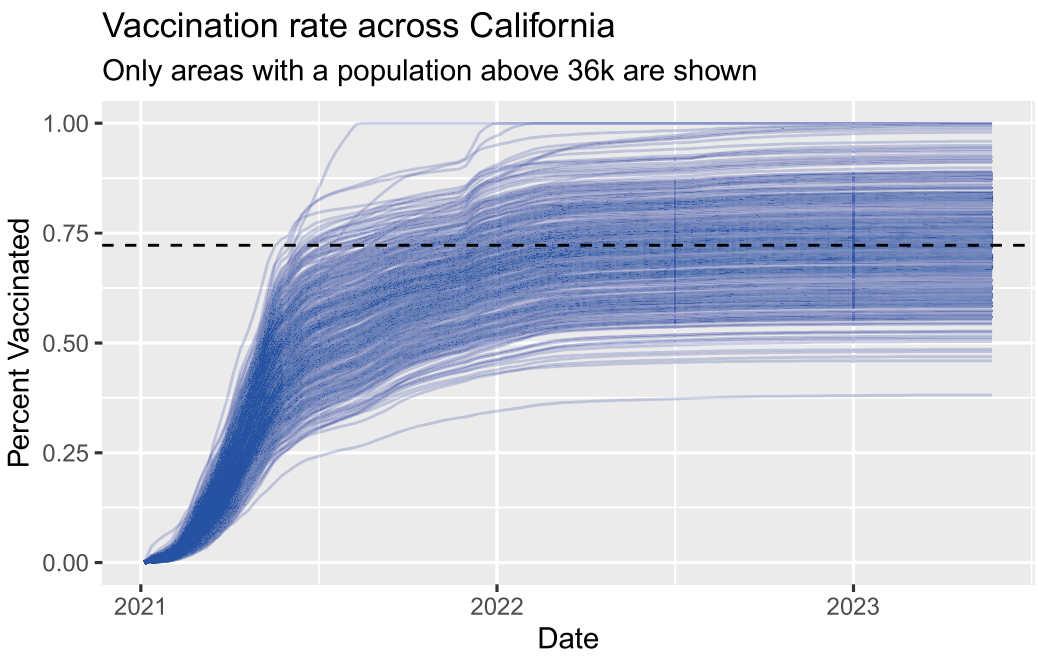
Answer: The percent of population fully vaccinated for 92040 is 0.55 which is below the average 0.7226. For 92109, the percent of population fully vaccinated in 0.69 which is also below the average 0.7226.

Q20: Finally make a time course plot of vaccination progress for all areas in the full dataset with a age5_plus_population > 36144.

```
vax.36.all <- filter(vax, age5_plus_population > 36144)

ggplot(vax.36.all) + aes(as_of_date, percent_of_population_fully_vaccinated, group = zip_c
```

```
Warning: Removed 185 rows containing missing values (`geom_line()`).
```



About this document

```
  sessionInfo()
```

```
R version 4.2.3 (2023-03-15 ucrt)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 10 x64 (build 22621)

Matrix products: default

locale:
[1] LC_COLLATE=English_United States.utf8
[2] LC_CTYPE=English_United States.utf8
[3] LC_MONETARY=English_United States.utf8
[4] LC_NUMERIC=C
[5] LC_TIME=English_United States.utf8

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] ggplot2_3.4.2   dplyr_1.1.2     zipcodeR_0.3.5  lubridate_1.9.2

loaded via a namespace (and not attached):
 [1] Rcpp_1.0.10         lattice_0.21-8      tidyr_1.3.0         class_7.3-21
 [5] digest_0.6.31       utf8_1.2.3          R6_2.5.1            repr_1.1.6
 [9] RSQLite_2.3.1       evaluate_0.21       e1071_1.7-13        httr_1.4.6
[13] pillar_1.9.0        rlang_1.1.0         curl_5.0.1          uuid_1.1-0
[17] rstudioapi_0.14     raster_3.6-20       blob_1.2.4          rmarkdown_2.22
[21] labeling_0.4.2      readr_2.1.4         stringr_1.5.0       munsell_0.5.0
[25] bit_4.0.5           proxy_0.4-27        compiler_4.2.3      xfun_0.39
[29] pkgconfig_2.0.3     tigris_2.0.3        base64enc_0.1-3     htmltools_0.5.5
[33] tidyselect_1.2.0    tibble_3.2.1        codetools_0.2-19    fansi_1.0.4
[37] crayon_1.5.2        tzdb_0.4.0          withr_2.5.0         sf_1.0-13
[41] tidycensus_1.4.1    rappdirs_0.3.3      grid_4.2.3          gtable_0.3.3
[45] jsonlite_1.8.5      lifecycle_1.0.3     DBI_1.1.3           magrittr_2.0.3
[49] scales_1.2.1        units_0.8-2         KernSmooth_2.23-20  cli_3.6.1
[53] stringi_1.7.12      cachem_1.0.8        farver_2.1.1        sp_1.6-1
[57] skimr_2.1.5         xml2_1.3.4          generics_0.1.3      vctrs_0.6.2
[61] tools_4.2.3         bit64_4.0.5         glue_1.6.2          purrr_1.0.1
[65] hms_1.1.3           fastmap_1.1.1       yaml_2.3.7          colorspace_2.1-0
```

```
[69] timechange_0.2.0   terra_1.7-29       classInt_0.4-9     rvest_1.0.3
[73] memoise_2.0.1      knitr_1.43
```