Beshaw Y Assignment 2 SURV727

Yael Beshaw

2024-10-01

Load Necessary Packages

```
library(tidyverse)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.2

## Warning: package 'tidyr' was built under R version 4.3.2

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

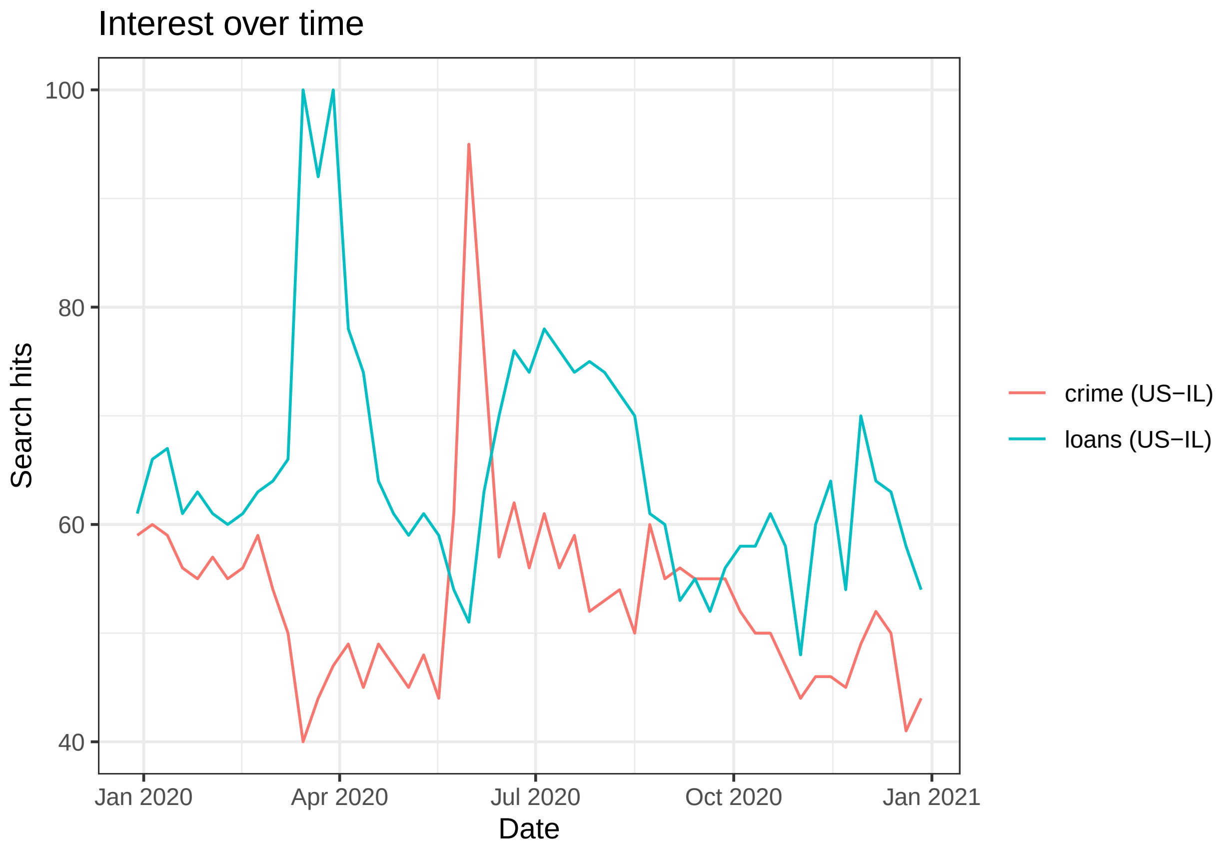
```
library(gtrendsR)
library(censusapi)
```

```
##
## Attaching package: 'censusapi'
##
## The following object is masked from 'package:methods':
##
##     getFunction
```

```
library(dplyr)
```

Part One: Pulling from API's

```
res <- gtrends(c("crime","loans"),
               geo="US-IL",
               time= "2020-01-01 2020-12-31",
               low_search_volume = TRUE)
plot(res)
```



#a) Find the mean, median and variance of the search hits for the keywords.

```
#Here, I create a function that will give us these descriptive statistics
#for the search hits we are looking to find in any given dataframe that
#has the column name "hits".

res_time <- as_tibble(res$interest_over_time) #turn the df into a tibble
res_time %>%
  drop_na(.) %>%
  dim(.) #no NA's, the tibble's dimensions remain the same
```

```
## [1] 106   7
```

```
descriptive_hits <- function(df) {
  df%>%
    summarize(
      mean= mean(df$hits),
      median= median(df$hits),
      variance= var(df$hits)
  )
}

descriptive_hits(res_time)
```

```
## # A tibble: 1 x 3
##    mean median variance
##   <dbl>  <dbl>    <dbl>
## 1  59.2     58     133.
```

The mean is 58.58 hits, the median is 58.00 hits, and the variance is 118.25.

#b) Which cities (locations) have the highest search frequency for loans? Note that there might be multiple rows for each city if there were hits for both "crime" and "loans" in that city. It might be easier to answer this question if we had the search hits info for both search terms in two separate variables. That is, each row would represent a unique city.

```
#utilizing the spread() function
res_city <- as_tibble(res$interest_by_city) #turn the df into a tibble; 400:5

res_city <- spread(res_city, key = keyword, value = hits) #dim become 345:5

highfreq_loans <- res_city %>%
  arrange(desc(loans)) %>% #arrange the loans column from greatest to least
  head(5) #output the first five values

highfreq_loans
```

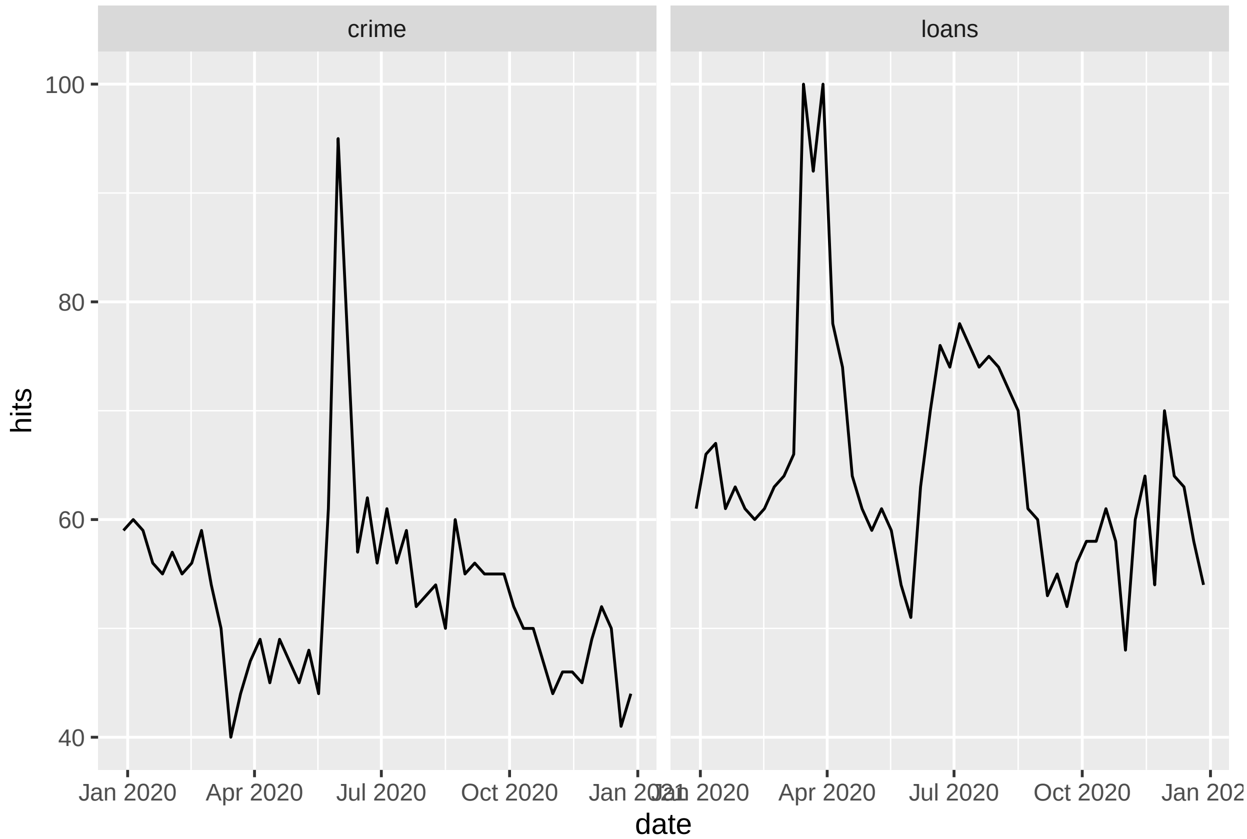
```
## # A tibble: 5 x 5
##   location     geo   gprop crime loans
##   <chr>        <chr> <chr> <int> <int>
## 1 Long Lake    US-IL web      NA   100
## 2 Rosemont     US-IL web      NA    80
## 3 Ford Heights US-IL web      NA    79
## 4 Peotone      US-IL web      NA    78
## 5 Dolton       US-IL web      NA    76
```

The top five cities with the highest search frequency for 'loans' are; Long Lake, East Saint Louis, Peotone, Rosemont, and Ford Heights.

#c) Is there a relationship between the search intensities between the 2 keywords?

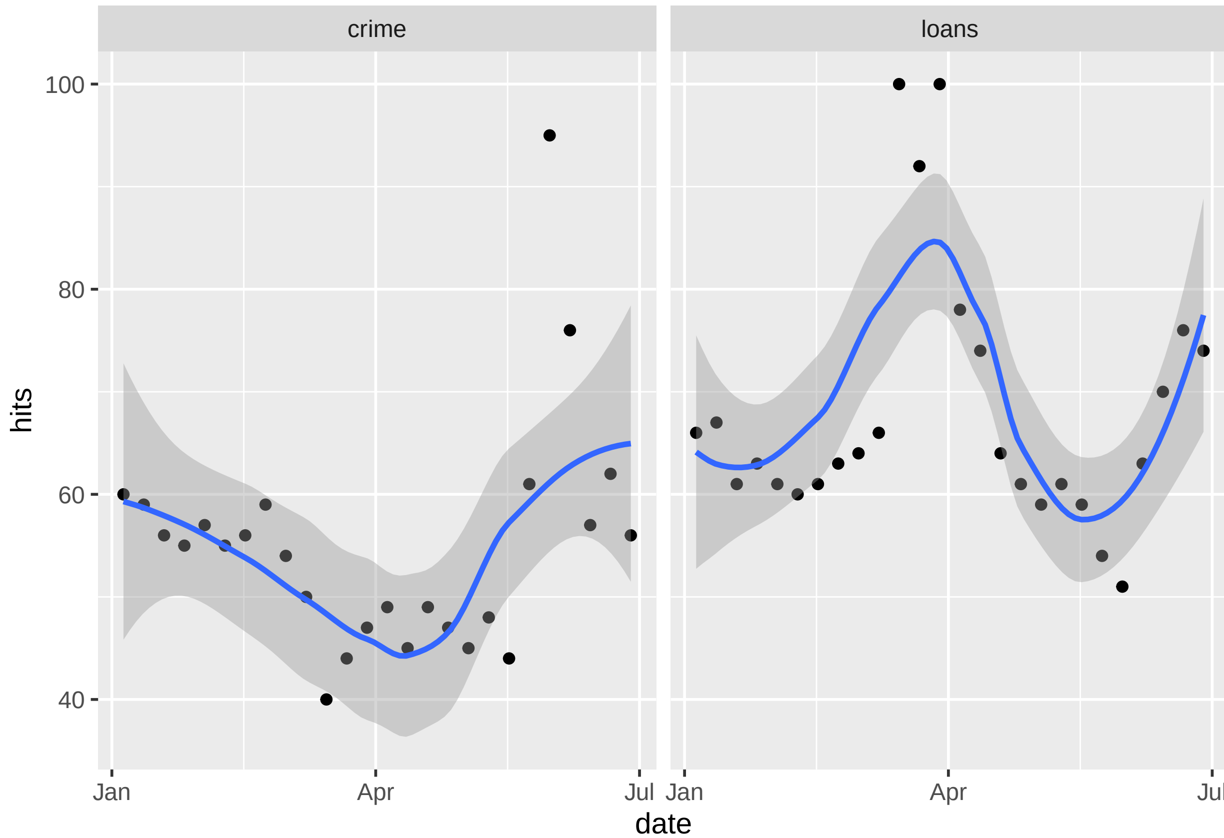
```
res_time %>%
  qplot(x = date, y = hits, data = .,
        geom = "line", facets = . ~ keyword)
```

```
## Warning: `qplot()` was deprecated in ggplot2 3.4.0.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



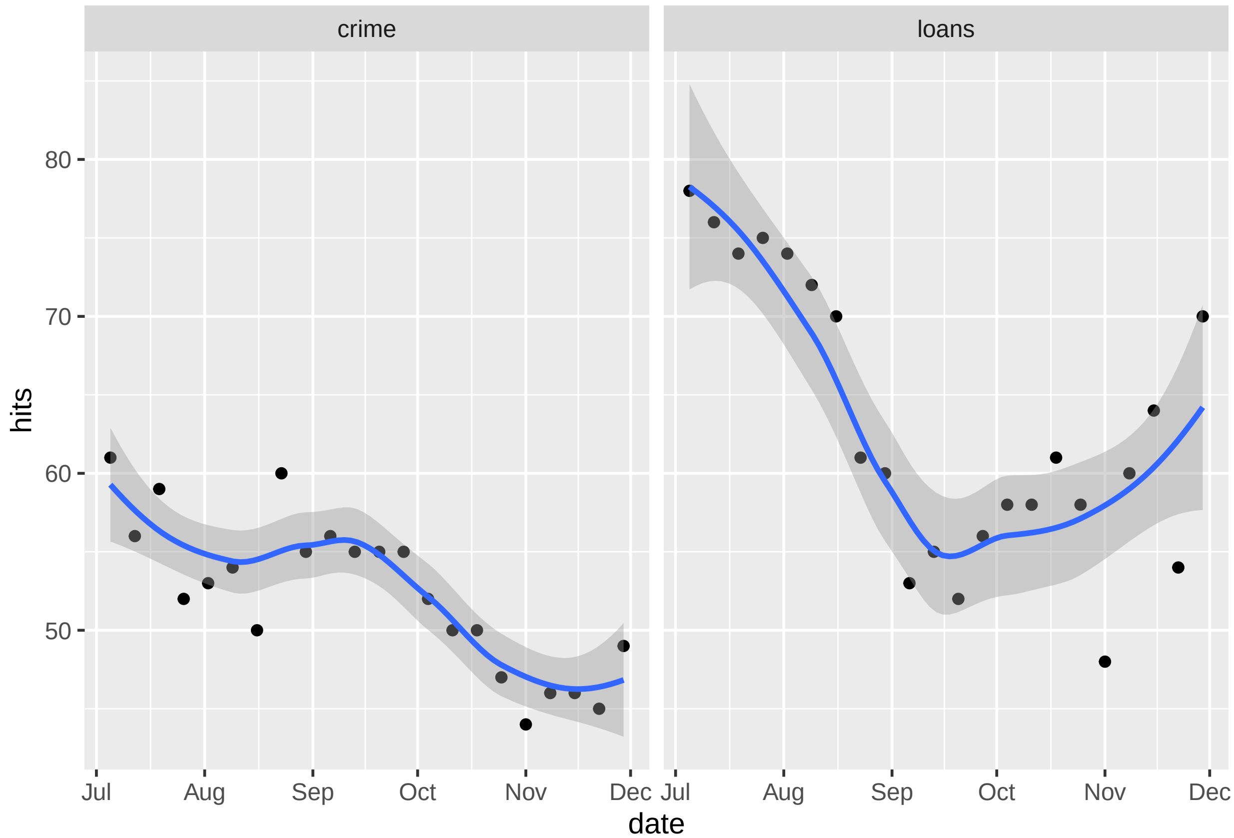

```r
res_time %>%
  separate(date, c("year", "month", "day"),
           sep = "-", remove = FALSE) %>%
  filter(month %in% c("01","02","03", "04", "05", "06")) %>%
  qplot(x = date, y = hits, data = .,
        geom = c("point", "smooth"), facets = . ~ keyword)
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




```
res_time %>%
  separate(date, c("year", "month", "day"),
           sep = "-", remove = FALSE) %>%
  filter(month %in% c("07","08","09", "10", "11")) %>%
  qplot(x = date, y = hits, data = .,
        geom = c("point", "smooth"), facets = . ~ keyword)
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```

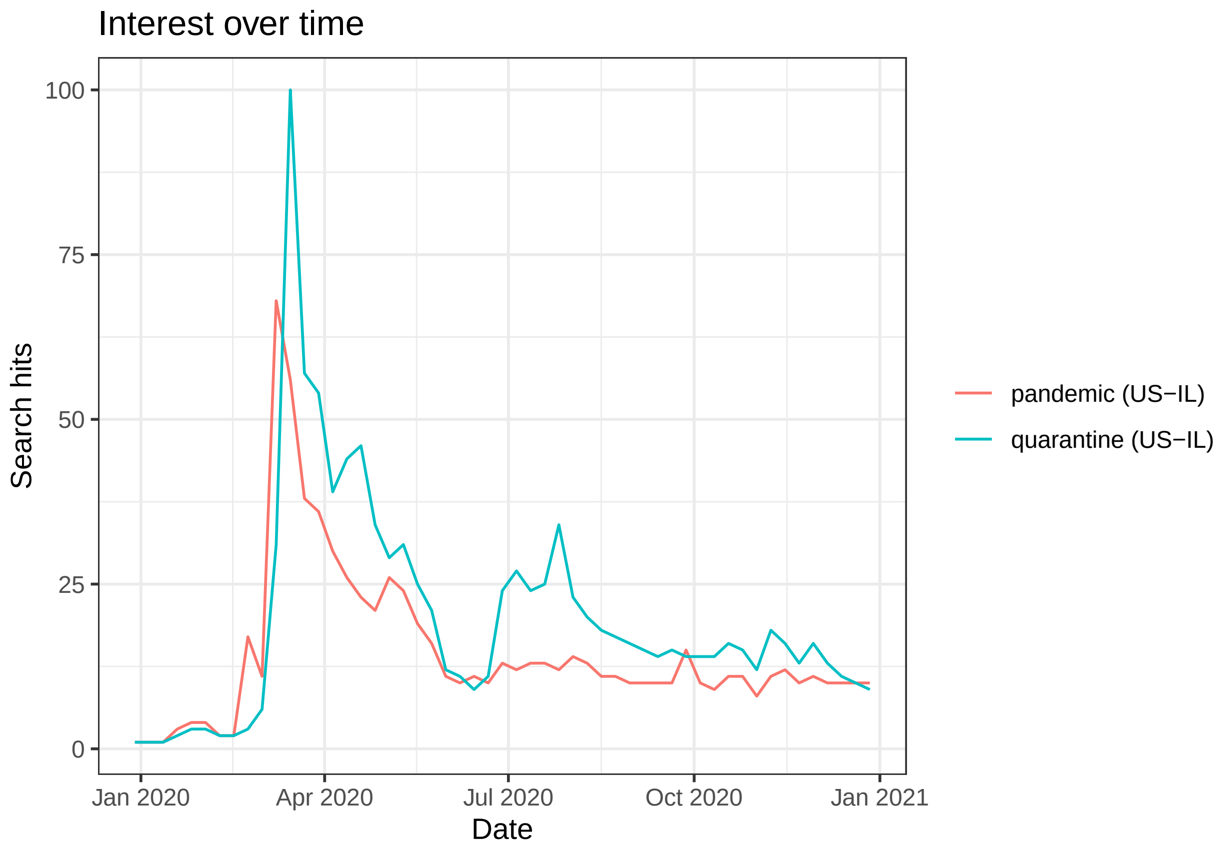


Utilizing the qplot() function allows us to see patterns within the dataset according to the keywords "crime" and "loans". We see that in the plots ranging across the year, that there seems to be a pattern where the hits in loans increase while the hits in crime decrease. When we separate these plots between the first six and last six months of the year, this pattern is evident especially in the first six months. Between the months of February to April and May to July we see the strongest evidence that there may be an inverse relationship. Starting August, this relationship is less straightforward and we see a decrease in both topics with the troughs for both being in November 2020, around the time of the 2020 presidential election.

#d) Repeat the above for keywords related to covid. Make sure you use multiple keywords like we did above. Try several different combinations and think carefully about words that might make sense within this context.

##Pandemic and Quarantine

```
covid <- gtrends(c("pandemic","quarantine"),
               geo="US-IL",
               time= "2020-01-01 2020-12-31",
               low_search_volume = TRUE)
plot(covid)
```




```
covid_time <- as_tibble(covid$interest_over_time) #turn the df into a tibble

#View(covid_time) there are results for hits that say <1, turn these to NA's
covid_time <- covid_time %>%
  mutate(hits = as.numeric(hits))%>% #turn all into numeric, coerce NA's
  mutate(hits = replace_na(hits, 0)) #turn NA's into 0 since its <1
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `hits = as.numeric(hits)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
#double-check that this works
#str(covid_time)
#covid_time$hits


descriptive_hits <- function(df) {
  df%>%
    summarize(
      mean= mean(df$hits),
      median= median(df$hits),
      variance= var(df$hits)
  )
}
descriptive_hits(covid_time)
```

```
## # A tibble: 1 x 3
##    mean median variance
##   <dbl>  <dbl>    <dbl>
## 1  17.1     13     233.
```

The mean is 17.22 hits, the median is 13 hits, and the variance is 236.67.

```
#utilizing the spread() function
covid_city <- as_tibble(covid$interest_by_city) #turn the df into a tibble; 400:5

#There is a duplicate found here in covid_city for location= Windsor
duplicate_rows <- covid_city %>%
  group_by(location, keyword) %>%
  filter(n() > 1)
duplicate_rows
```

```
## # A tibble: 0 x 5
## # Groups:   location, keyword [0]
## # i 5 variables: location <chr>, hits <int>, keyword <chr>, geo <chr>,
## #   gprop <chr>
```

```
#Remove the duplicate
covid_city <- covid_city %>%
  distinct(location, keyword, .keep_all = TRUE)

#Analyze high frequency for "pandemic"
covid_city <- spread(covid_city, key = keyword, value = hits)

highfreq_pandemic <- covid_city %>%
  arrange(desc(pandemic)) %>% #arrange the loans column from greatest to least
  head(5) #output the first five values

highfreq_pandemic
```

```
## # A tibble: 5 x 5
##   location        geo   gprop pandemic quarantine
##   <chr>           <chr> <chr>    <int>      <int>
## 1 Lake Barrington US-IL web        100         NA
## 2 Highland Park   US-IL web         99         82
## 3 Riverdale       US-IL web         97         NA
## 4 Sugar Grove     US-IL web         95         NA
## 5 La Grange Park  US-IL web         94         82
```

```
#Winnetka, Highland Park, Lake Barrington, La Grange Park, Oak Park

#Analyze high frequency for "quarantine"
highfreq_quarantine <- covid_city %>%
  arrange(desc(quarantine)) %>% #arrange the loans column from greatest to least
  head(5) #output the first five values

highfreq_quarantine
```

```
## # A tibble: 5 x 5
##   location          geo   gprop pandemic quarantine
##   <chr>             <chr> <chr>    <int>      <int>
## 1 Willow Springs    US-IL web         90        100
## 2 South Barrington US-IL web         NA         96
```

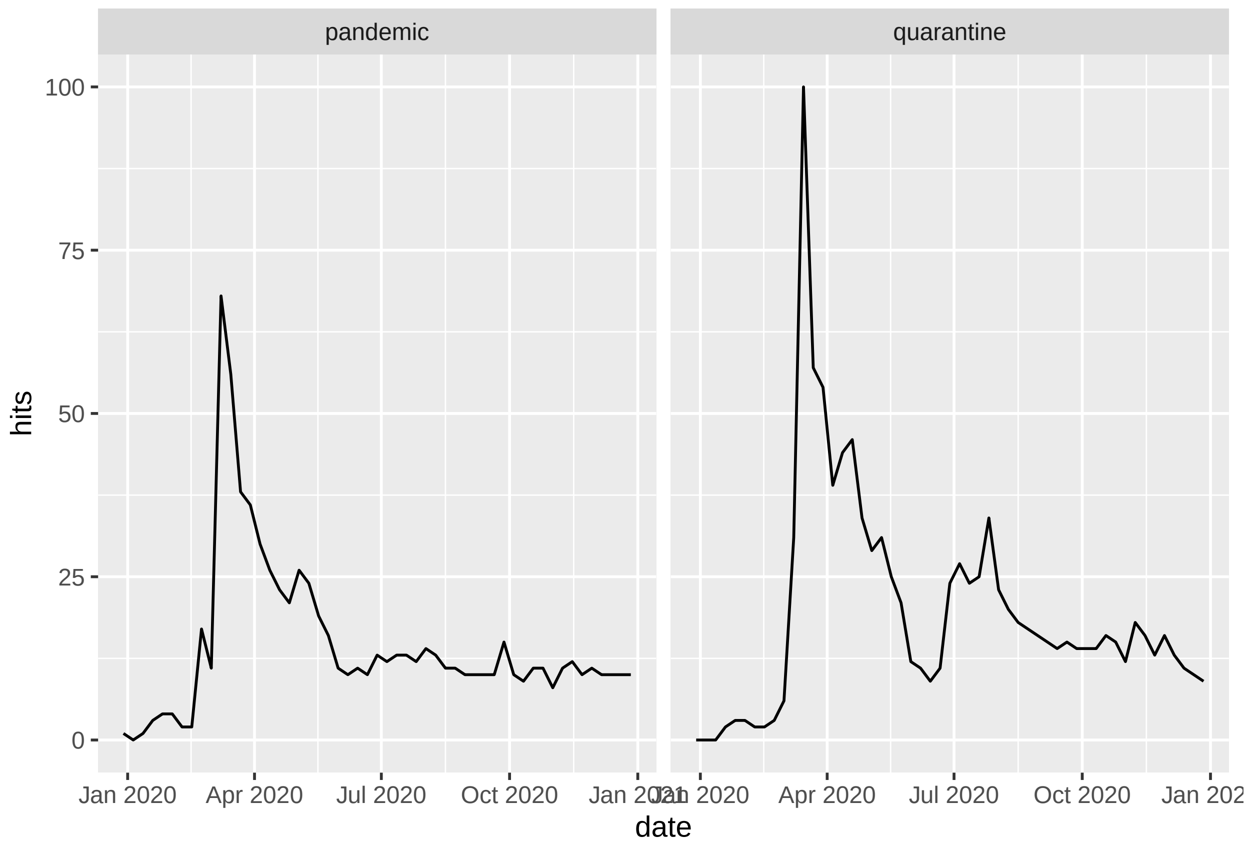
```
## 3 Winfield         US-IL web         NA        96
## 4 Western Springs  US-IL web         NA        94
## 5 Hawthorn Woods   US-IL web         NA        87
```

```
#Winfield, Barrington Hills, Hodgkins, Lemont, Rolling Meadows
```

The top five cities with the highest search frequency for ‘pandemic’ are; Winnetka, Highland Park, Lake Barrington, La Grange Park, and Oak Park.

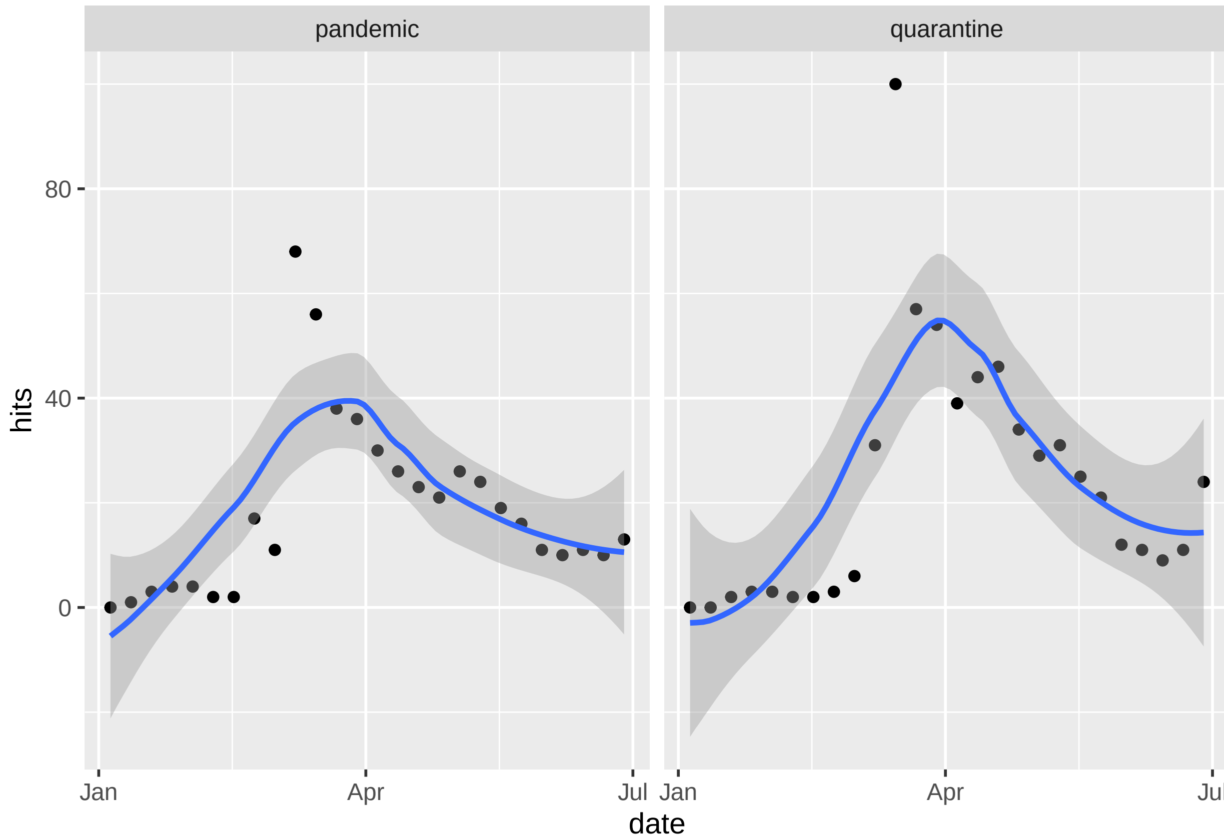
The top five cities with the highest search frequency for ‘quarantine’ are; Winfield, Barrington Hills, Hodgkins, Lemont, and Rolling Meadows

```
covid_time %>%
  qplot(x = date, y = hits, data = .,
        geom = "line", facets = . ~ keyword)
```



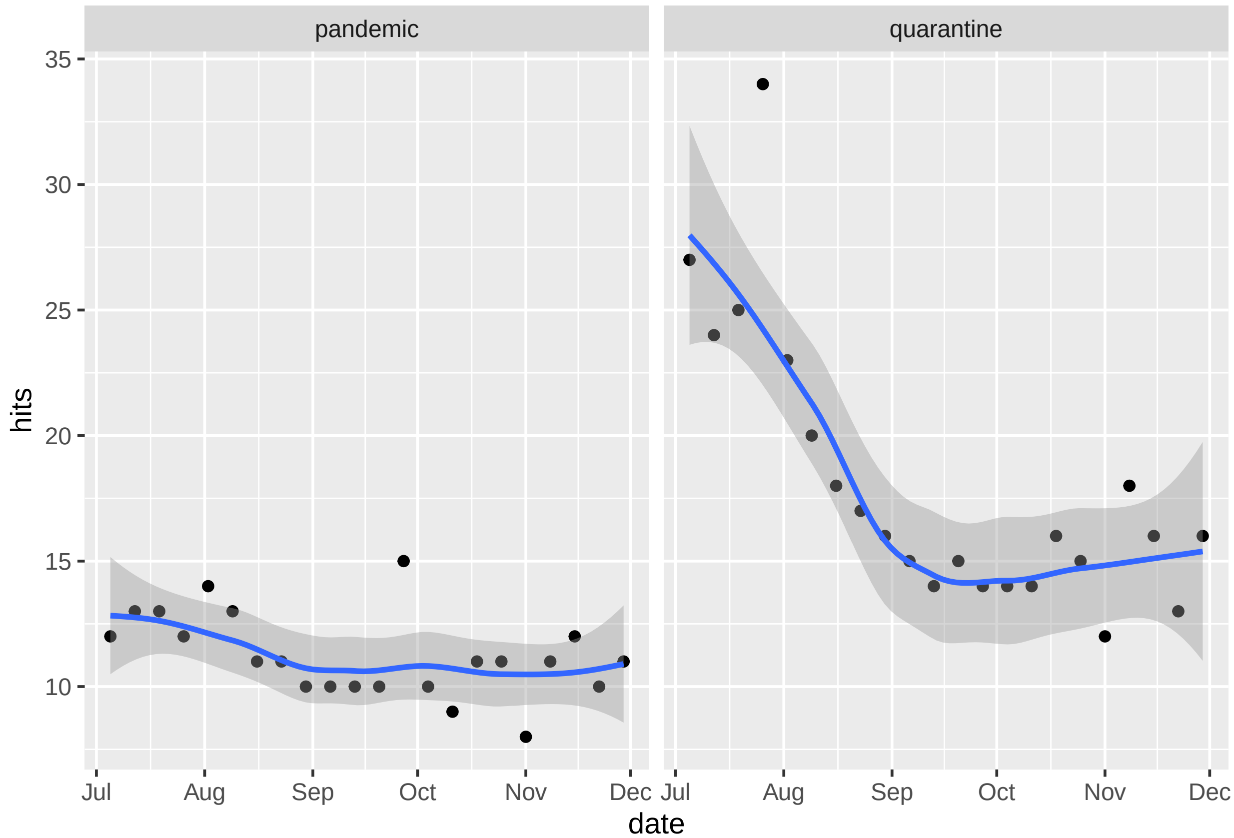

```
covid_time %>%
  separate(date, c("year", "month", "day"),
           sep = "-", remove = FALSE) %>%
  filter(month %in% c("01","02","03", "04", "05", "06")) %>%
  qplot(x = date, y = hits, data = .,
         geom = c("point", "smooth"), facets = . ~ keyword)
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




```
covid_time %>%
  separate(date, c("year", "month", "day"),
           sep = "-", remove = FALSE) %>%
  filter(month %in% c("07","08","09", "10", "11")) %>%
  qplot(x = date, y = hits, data = .,
        geom = c("point", "smooth"), facets = . ~ keyword)
```

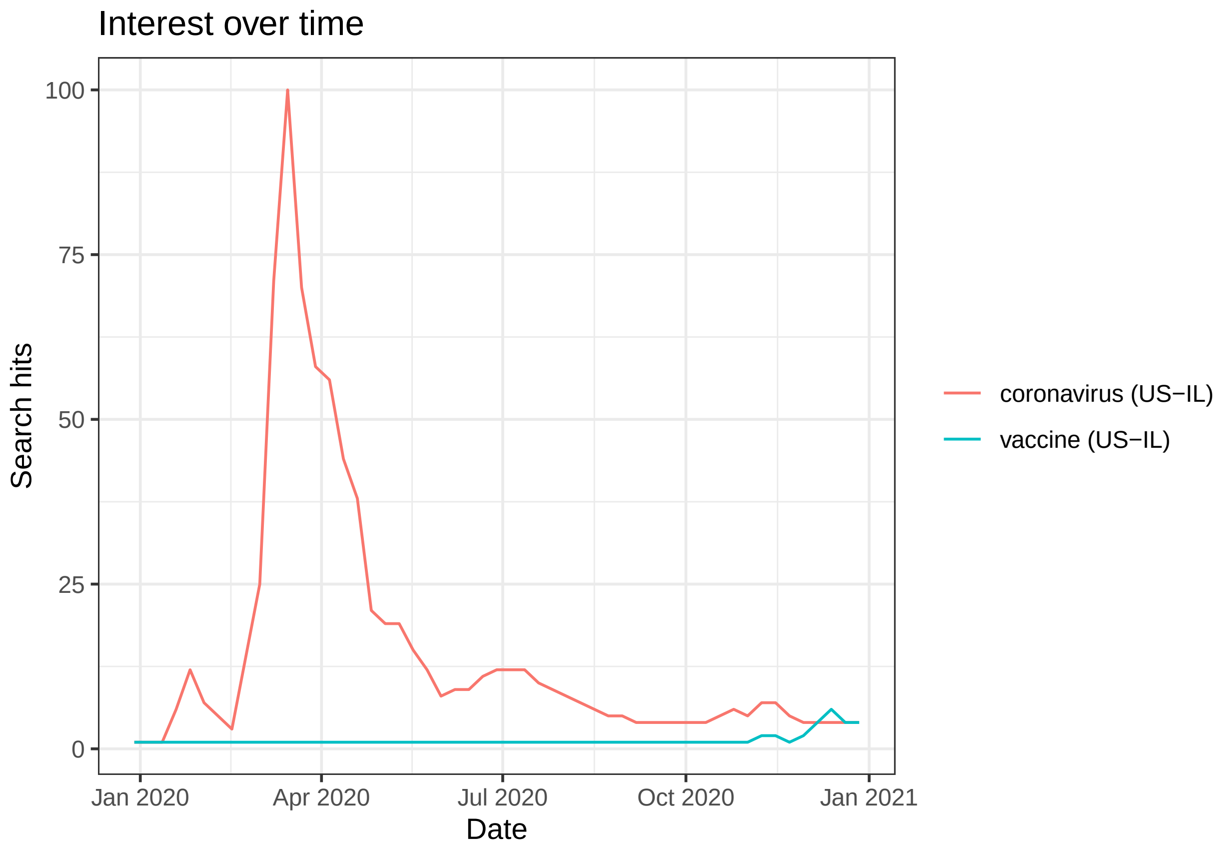
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



For the keywords "pandemic" and "quarantine" seem to have a proportional relationship. As the hits in "pandemic" increase and decrease so do that of "quarantine". The magnitude in which they increase and decrease is different, we see a very steep decrease in "quarantine" from July to September. The context of these results makes sense as we see that many quarantine restrictions decreased or were lifted as this time, however, this jump in July is reflective of the introduction of variants and "super-spreader" events. The mean of 17.22, median of 13, and the variance of 236.67, this shows us there is a large variation in these hits as there are peaks and troughs indicating times of relevance for these topics.

Coronavirus and Vaccine

```
covid2 <- gtrends(c("coronavirus","vaccine"),
               geo="US-IL",
               time= "2020-01-01 2020-12-31",
               low_search_volume = TRUE)
plot(covid2)
```




```
covid2_time <- as_tibble(covid2$interest_over_time) #turn the df into a tibble

#View(covid2_time) #there are results for hits that say <1, turn these to NA's
covid2_time <- covid2_time %>%
  mutate(hits = as.numeric(hits))%>% #turn all into numeric, coerce NA's
  mutate(hits = replace_na(hits, 0)) #turn NA's into 0 since its <1
```

```
## Warning: There was 1 warning in `mutate()`.
## i In argument: `hits = as.numeric(hits)`.
## Caused by warning:
## ! NAs introduced by coercion
```

```
#double-check that this works
#str(covid2_time)
#covid2_time$hits


descriptive_hits <- function(df) {
  df%>%
    summarize(
      mean= mean(df$hits),
      median= median(df$hits),
      variance= var(df$hits)
  )
}
descriptive_hits(covid2_time)
```

```
## # A tibble: 1 x 3
##    mean median variance
##   <dbl>  <dbl>    <dbl>
## 1  8.09    3.5     259.
```

The mean is 8.09 hits, the median is 3.5, and the variance is 258.75.

```
#utilizing the spread() function
covid2_city <- as_tibble(covid2$interest_by_city) #turn the df into a tibble; 400:5

#Analyze high frequency for "coronavirus"
covid2_city <- spread(covid2_city, key = keyword, value = hits)

highfreq_coronavirus <- covid2_city %>%
  arrange(desc(coronavirus)) %>% #arrange the loans column from greatest to least
  head(5) #output the first five values

highfreq_coronavirus
```

```
## # A tibble: 5 x 5
##   location         geo   gprop coronavirus vaccine
##   <chr>            <chr> <chr>       <int>   <int>
## 1 Clarendon Hills US-IL web           100      NA
## 2 London Mills    US-IL web            91      NA
## 3 Farmersville    US-IL web            87      NA
## 4 Buffalo Grove   US-IL web            82      81
## 5 Wheeling        US-IL web            81      39
```

```
#Clarendon Hills, Warren, London Mills, Farmersville, Wheeling

#Analyze high frequency for "vaccine"
highfreq_vaccine <- covid2_city %>%
  arrange(desc(vaccine)) %>% #arrange the loans column from greatest to least
  head(5) #output the first five values

highfreq_vaccine
```

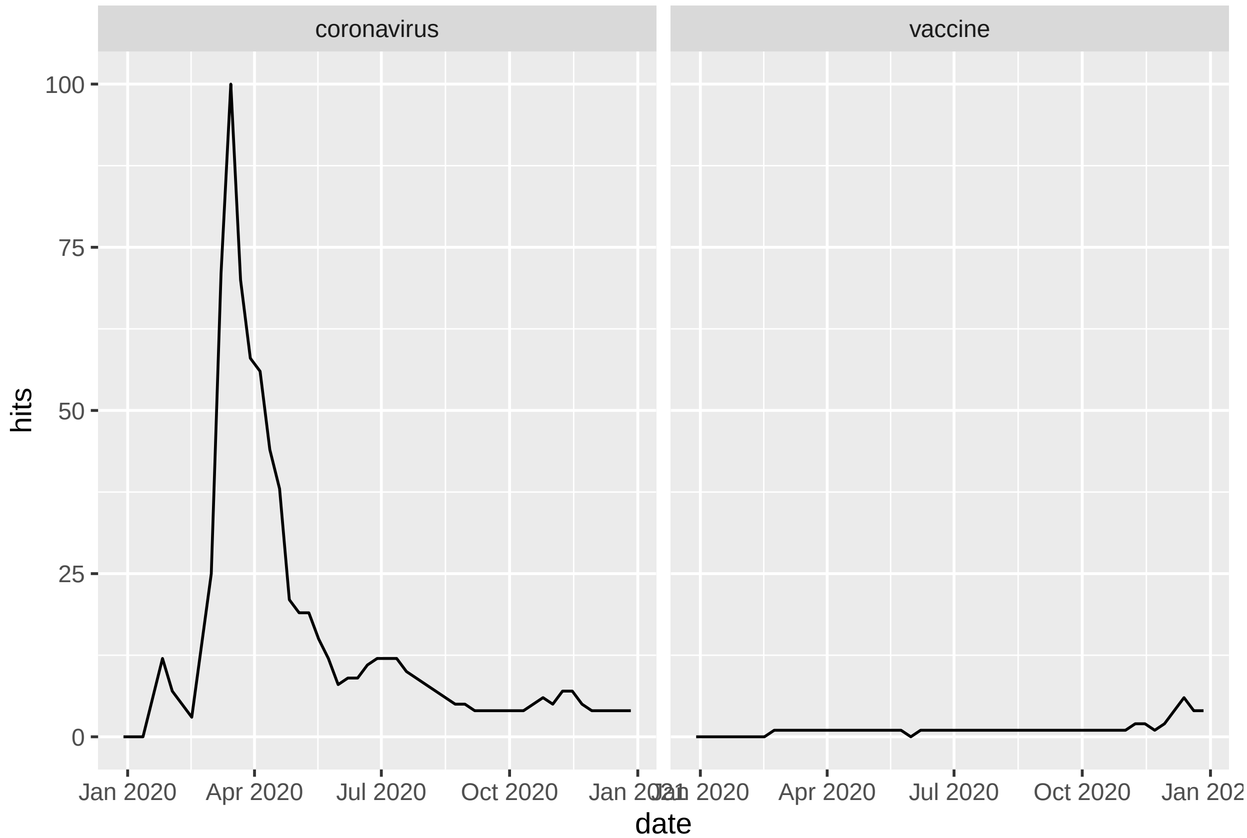
```
## # A tibble: 5 x 5
##   location      geo   gprop coronavirus vaccine
##   <chr>         <chr> <chr>       <int>   <int>
## 1 Hurst         US-IL web            NA     100
## 2 Buffalo Grove US-IL web            82      81
## 3 North Aurora  US-IL web            NA      52
## 4 Hinsdale      US-IL web            60      50
## 5 Deer Park     US-IL web            62      48
```

```
#Hurst, North Aurora, Deer Park, Hinsdale, Mokena
```

The top five cities with the highest search frequency for ‘coronavirus’ are; Clarendon Hills, Warren, London Mills, Farmersville, and Wheeling

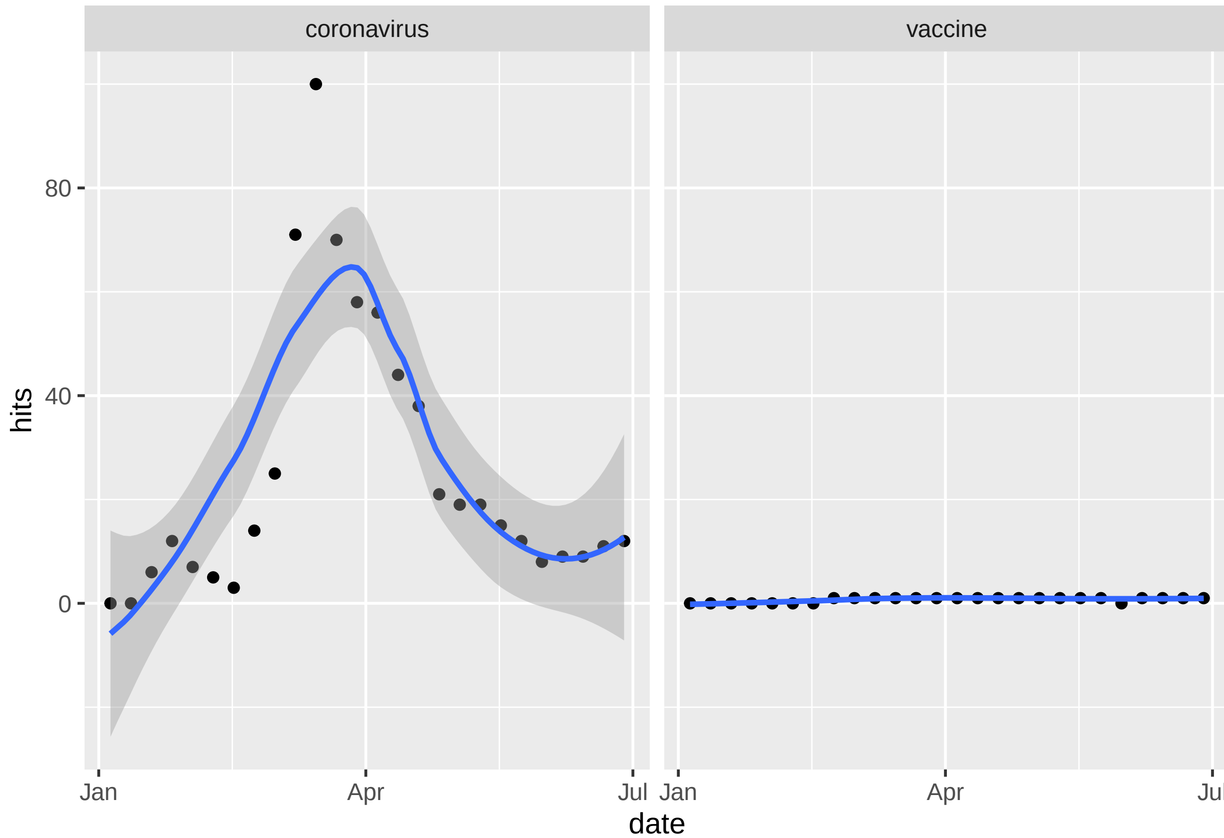
The top five cities with the highest search frequency for ‘vaccine’ are; Hurst, North Aurora, Deer Park, Hinsdale, and Moken

```
covid2_time %>%
  qplot(x = date, y = hits, data = .,
        geom = "line", facets = . ~ keyword)
```



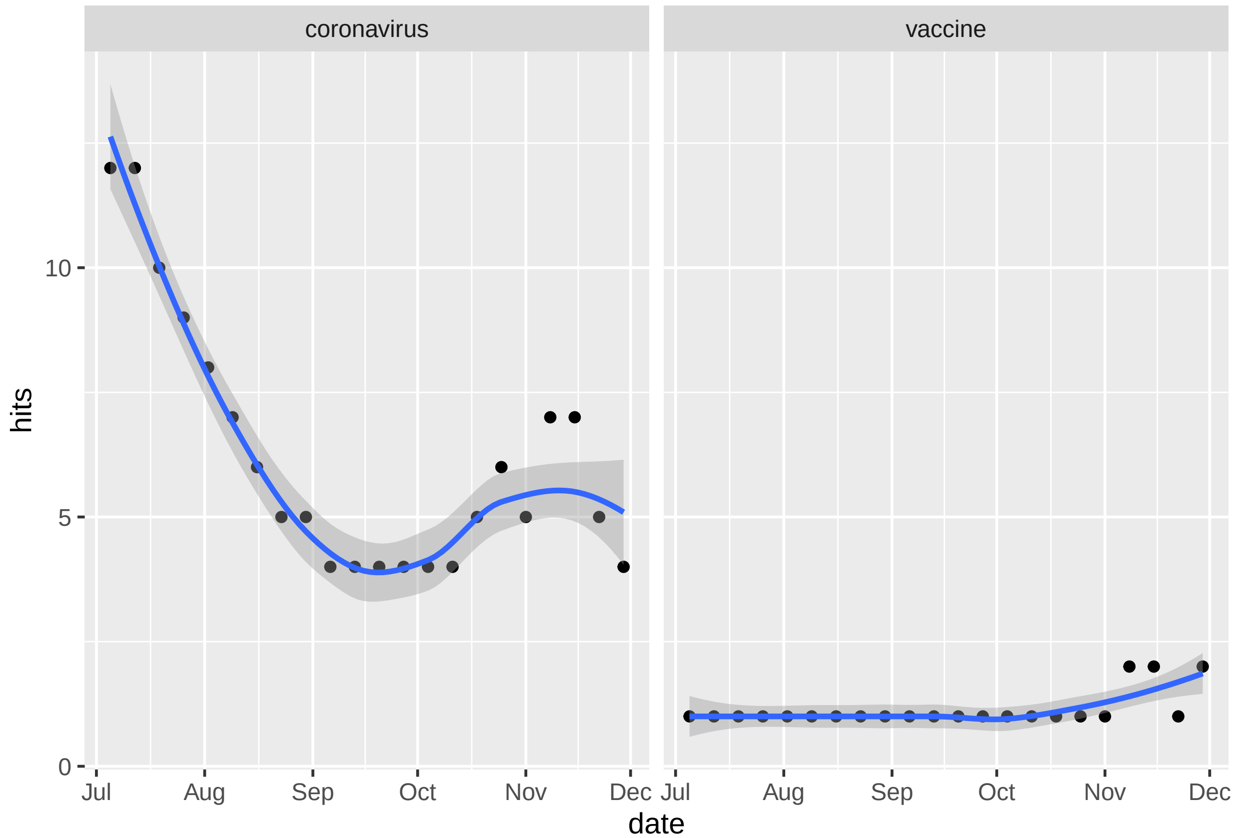

```
covid2_time %>%
  separate(date, c("year", "month", "day"),
           sep = "-", remove = FALSE) %>%
  filter(month %in% c("01","02","03", "04", "05", "06")) %>%
  qplot(x = date, y = hits, data = .,
        geom = c("point", "smooth"), facets = . ~ keyword)
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




```
covid2_time %>%
  separate(date, c("year", "month", "day"),
           sep = "-", remove = FALSE) %>%
  filter(month %in% c("07","08","09", "10", "11")) %>%
  qplot(x = date, y = hits, data = .,
        geom = c("point", "smooth"), facets = . ~ keyword)
```

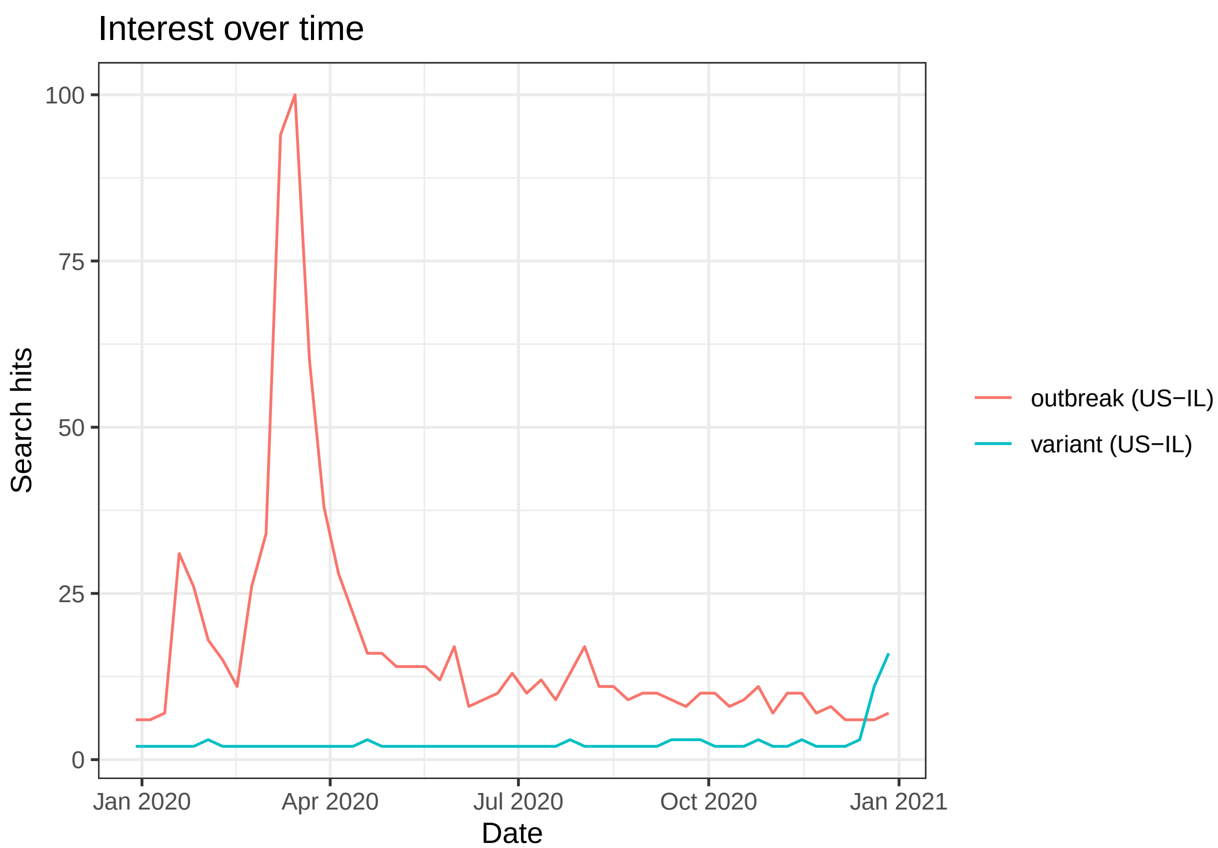
```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



For the keywords "coronavirus" and "vaccine" seem to not have a relationship. As the hits in "coronavirus" increase and decrease there is no change in "vaccine". The only time we see an increase in the hits for "vaccine" is around November 2020 but not particularly high. Interestingly enough, there is a steep decrease of "coronavirus" hits after April. This makes sense but ultimately, I would elect to utilize the first combination for further analysis.

Variant and Outbreak

```
covid3 <- gtrends(c("variant","outbreak"),
                  geo="US-IL",
                  time= "2020-01-01 2020-12-31",
                  low_search_volume = TRUE)
plot(covid3)
```




```r
covid3_time <- as_tibble(covid3$interest_over_time) #turn the df into a tibble

#View(covid3_time) #there are results for hits that say <1, turn these to NA's
covid3_time <- covid3_time %>%
  mutate(hits = as.numeric(hits))%>% #turn all into numeric, coerce NA's
  mutate(hits = replace_na(hits, 0)) #turn NA's into 0 since its <1

#double-check that this works
#str(covid3_time)
#covid3_time$hits


descriptive_hits <- function(df) {
  df%>%
    summarize(
      mean= mean(df$hits),
      median= median(df$hits),
      variance= var(df$hits)
  )
}
descriptive_hits(covid3_time)
```

```
## # A tibble: 1 x 3
##    mean median variance
##   <dbl>  <dbl>    <dbl>
## 1  9.88      6     229.
```

The mean is 9.88 hits, the median is 6, and the variance is 229.29.

```
#utilizing the spread() function
covid3_city <- as_tibble(covid3$interest_by_city) #turn the df to tibble; 400:5

#There is a duplicate found here in covid3_city for location= Windsor
duplicate_rows <- covid3_city %>%
  group_by(location, keyword) %>%
  filter(n() > 1)
duplicate_rows
```

```
## # A tibble: 2 x 5
## # Groups:   location, keyword [1]
##   location  hits keyword  geo   gprop
##   <chr>    <int> <chr>    <chr> <chr>
## 1 Windsor     NA outbreak US-IL web
## 2 Windsor     NA outbreak US-IL web
```

```
#Remove the duplicate
covid3_city <- covid3_city %>%
  distinct(location, keyword, .keep_all = TRUE)

#Analyze high frequency for "variant"
covid3_city <- spread(covid3_city, key = keyword, value = hits)

highfreq_variant <- covid3_city %>%
  arrange(desc(variant)) %>% #arrange the loans column from greatest to least
  head(5) #output the first five values

highfreq_variant
```

```
## # A tibble: 5 x 5
##   location      geo   gprop outbreak variant
##   <chr>         <chr> <chr>    <int>   <int>
## 1 Bartlett      US-IL web         NA     100
## 2 Oak Brook     US-IL web         NA      52
## 3 Lincolnshire  US-IL web         NA      43
## 4 North Chicago US-IL web         NA      42
## 5 Waterloo      US-IL web         NA      41
```

```
#Bartlett, Vernon Hills, Lake Zurich, Lisle, River Forest

#Analyze high frequency for "outbreak"
highfreq_outbreak <- covid3_city %>%
  arrange(desc(outbreak)) %>% #arrange the loans column from greatest to least
  head(5) #output the first five values

highfreq_outbreak
```

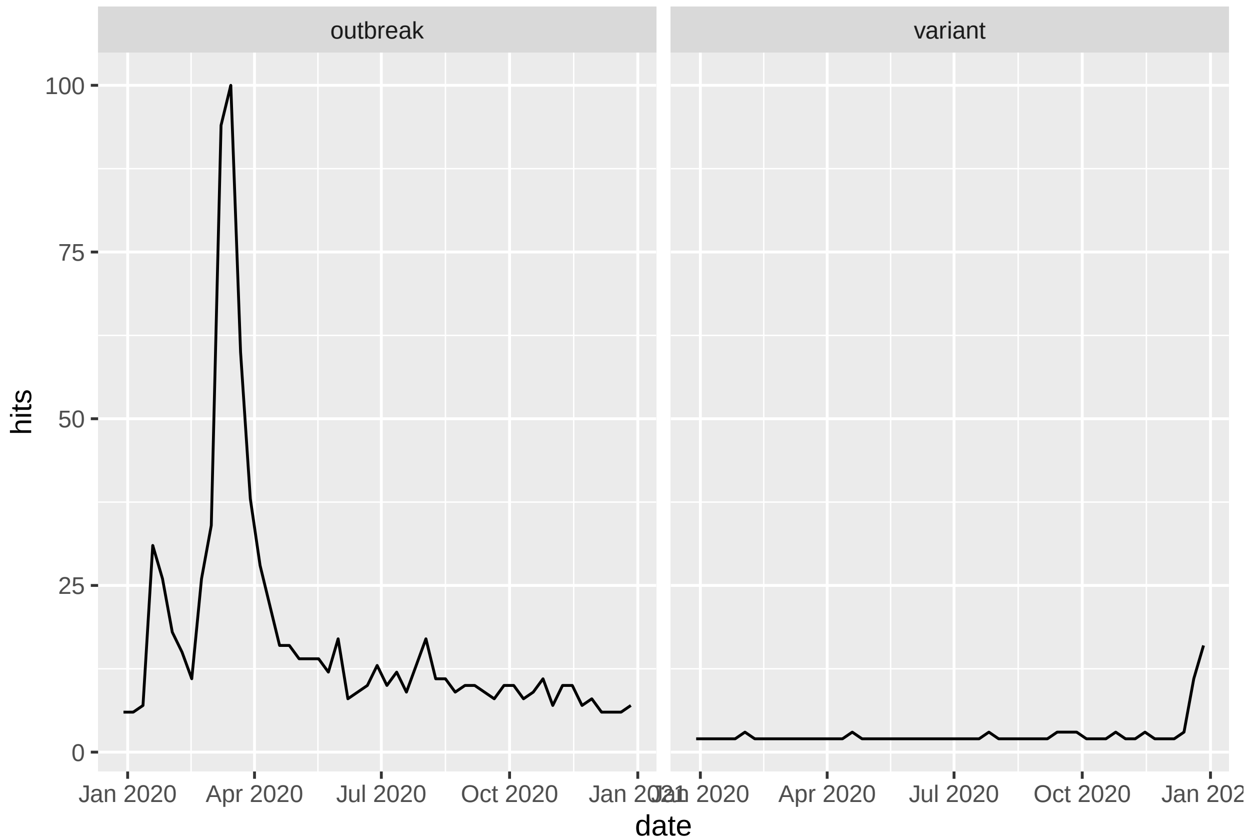
```
## # A tibble: 5 x 5
##   location  geo   gprop outbreak variant
##   <chr>     <chr> <chr>    <int>   <int>
## 1 Albion    US-IL web         NA      NA
## 2 Aledo     US-IL web         NA      NA
## 3 Alexis    US-IL web         NA      NA
## 4 Algonquin US-IL web         NA      NA
```

```
## 5 Alhambra  US-IL web          NA       NA
#No information available about locations and hits grouped by outbreak
```

The top five cities with the highest search frequency for 'variant' are; Bartlett, Vernon Hills, Lake Zurich, Lisle, and River Forest

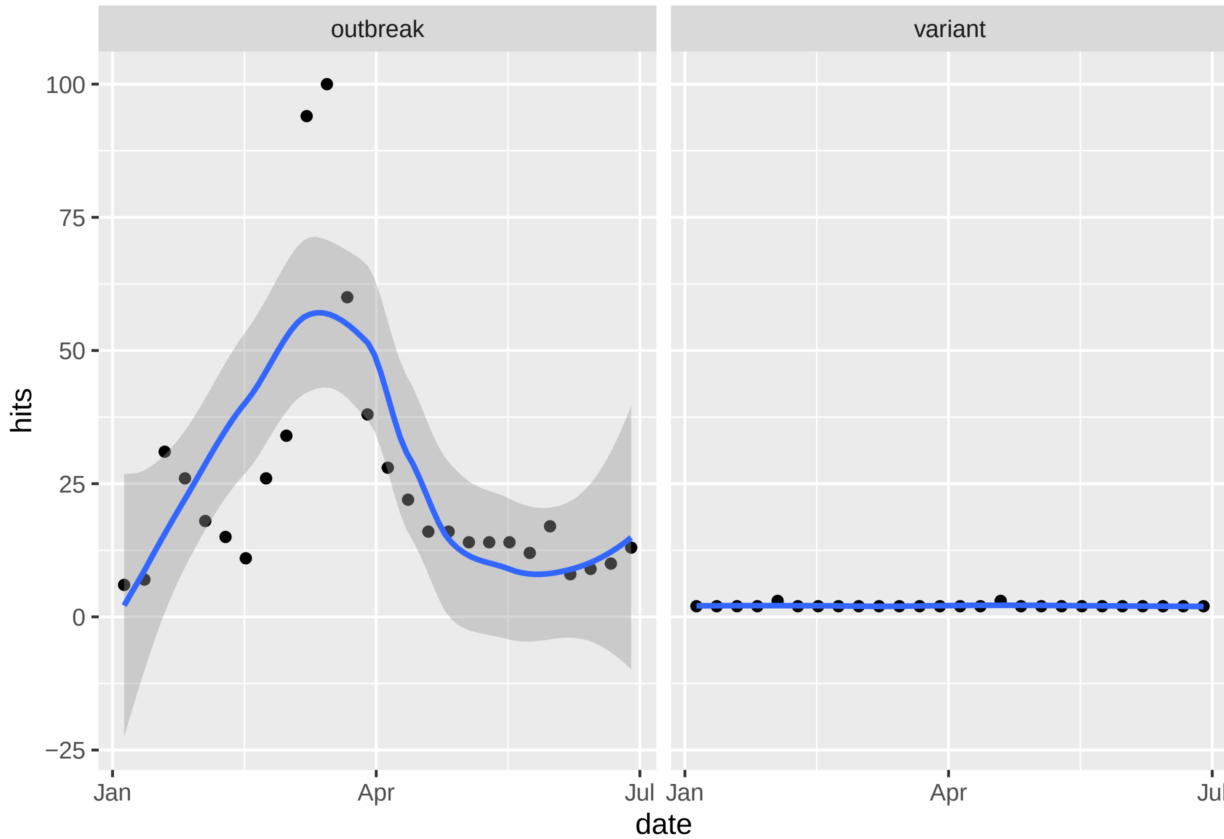
The data for the top five cities with the highest search frequency for 'outbreak' are not available.

```
covid3_time %>%
  qplot(x = date, y = hits, data = .,
        geom = "line", facets = . ~ keyword)
```



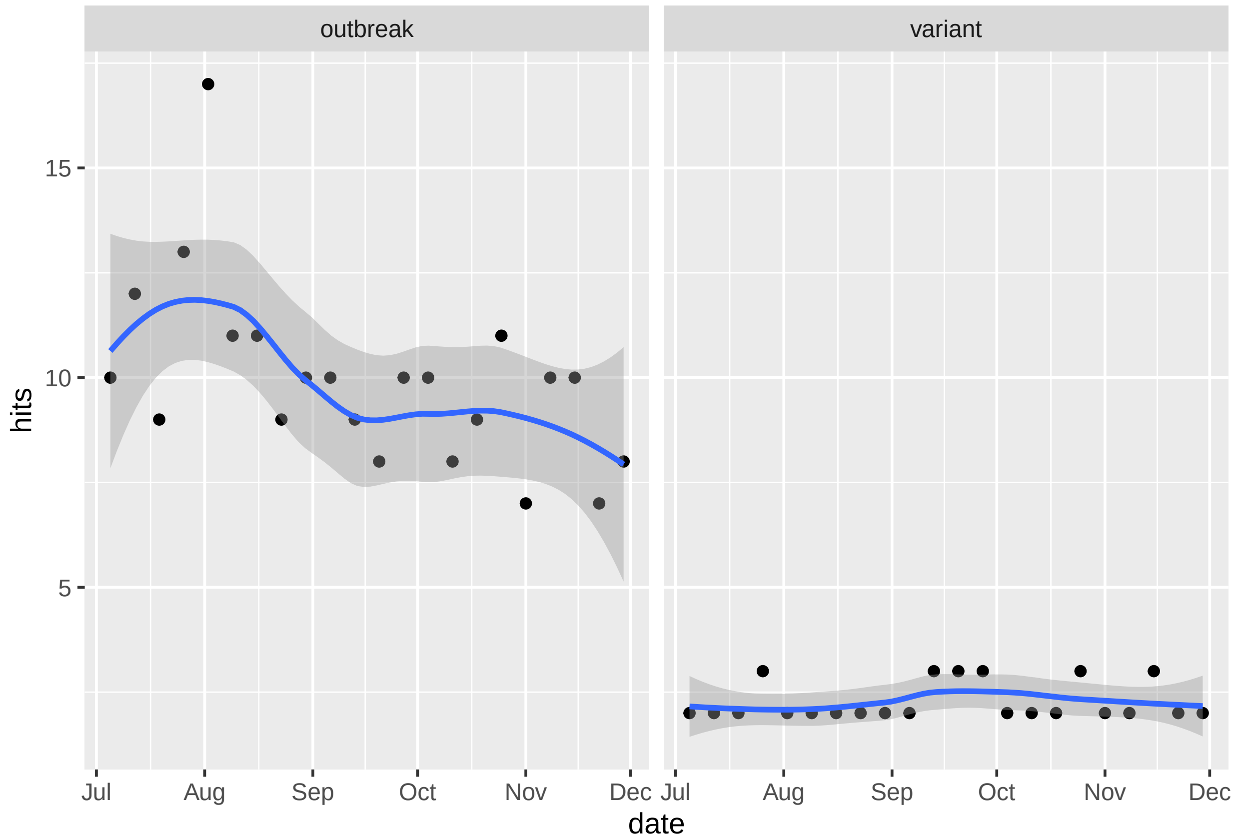

```
covid3_time %>%
  separate(date, c("year", "month", "day"),
           sep = "-", remove = FALSE) %>%
  filter(month %in% c("01","02","03", "04", "05", "06")) %>%
  qplot(x = date, y = hits, data = .,
        geom = c("point", "smooth"), facets = . ~ keyword)
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```




```
covid3_time %>%
  separate(date, c("year", "month", "day"),
           sep = "-", remove = FALSE) %>%
  filter(month %in% c("07","08","09", "10", "11")) %>%
  qplot(x = date, y = hits, data = .,
        geom = c("point", "smooth"), facets = . ~ keyword)
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



Similar to the keywords "coronavirus" and "vaccine"

For the keywords "outbreak" and "variant" seem to not have a relationship. As the hits in "outbreaks" increase and decrease there is no change in "variant". The only time we see an increase in the hits for "variant" is around November 2020 but not particularly high. Again, similar to the past combination, there is a steep decrease of "coronavirus" hits after April. This makes also sense as the term "outbreak" becomes less utilized. Variants of covid began to be introduced later in the year 2020.

#Discussion of Part One: Pulling from API's Between the three combinations attempted, I will continue to use the first one "pandemic" and "quarantine". This combination has highest hits, the middle variance, and most evident relationship.

#Part Two: Google Trends + ACS

Read the census key in the cs_key object

```
cs_key <- read_file("census-key.txt")
```

Request Basic Socio-Demographic Information for State of Illinois

```
acs_il <- getCensus(name = "acs/acs5",
                    vintage = 2020,
                    vars = c("NAME",
                             "B01001_001E",
                             "B06002_001E",
                             "B19013_001E",
                             "B19301_001E"),
                    region = "place:*",
                    regionin = "state:17",
                    key = cs_key)
```

```r
head(acs_il)
```

```
##   state place                           NAME B01001_001E B06002_001E B19013_001E
## 1    17 15261 Coatsburg village, Illinois         180        35.6       55714
## 2    17 15300    Cobden village, Illinois        1018        44.2       38750
## 3    17 15352      Coffeen city, Illinois         640        33.4       35781
## 4    17 15378   Colchester city, Illinois        1347        42.2       43942
## 5    17 15469    Coleta village, Illinois         230        27.7       56875
## 6    17 15495    Colfax village, Illinois        1088        32.5       58889
##   B19301_001E
## 1       27821
## 2       19979
## 3       26697
## 4       24095
## 5       23749
## 6       24861
```

Convert values representing missingness to NAs

```r
acs_il[acs_il == -666666666] <- NA
```

Rename the socio-demographic variables and assign meaningful names

```r
acs_il <-acs_il %>%
  rename(pop = B01001_001E, #population
         age = B06002_001E, #MEDIAN age
         hh_income = B19013_001E, #MEDIAN household income
         income = B19301_001E) #income per capita
```

#a) Clean NAMES and add location It seems like we could try to use this location information listed above to merge this data set with the Google Trends data. However, we first have to clean NAME so that it has the same structure as location in the search interest by city data. Add a new variable location to the ACS data that only includes city names.

```r
#View(acs_il$NAME) - locations are structured as name, city/town/village/CDP, IL
# 1466 locations
#View(res_city$location) - locations are only mentioned by name
# 346 locations

#First separate the name + designation from the state, we see that the
#designation options are village, city, CDP, and town. Remove these words
#from the name and designation column to only have the names left

acs_il_clean <- acs_il %>%
  separate(NAME, c("namedes", "state"),
           sep = ",", remove = FALSE)

acs_il_clean <- acs_il_clean %>%
  separate(namedes, c("location"),
           sep = "\\s*( village| city| town| CDP)\\s*", remove = FALSE)
```

```
## Warning: Expected 1 pieces. Additional pieces discarded in 1466 rows [1, 2, 3, 4, 5, 6,
## 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, ...].
```

```r
acs_il_clean <- acs_il_clean %>%
  select(-namedes, -state)
```

#b) Check how many cities don't appear in both data sets, cannot be matched.

```
not_matching <- acs_il_clean %>%
  filter(!location %in% res_city$location)

not_matching2 <- res_city %>%
  filter(!location %in% acs_il_clean$location)

nrow(not_matching) # 1125 locations in acs_il_clean not in res_city
```

```
## [1] 1121
```

```
nrow(not_matching2) # 8 locations in res_city not in acs_il_clean
```

```
## [1] 8
```

#c) Create a new data set by joining the Google Trends and the ACS data. Keep only cities that appear in both data sets.

```
merged <- acs_il_clean %>%
  inner_join(res_city, by="location")

#double-checked to make sure cities in not_matching 2 did not show up
dim(merged) # 341 11
```

```
## [1] 345  11
```

#d) Compute mean of search popularity / keyword grouped by avg median hh_income.

For both keywords for cities that have an above average median household income and for those that have an below average median household income.

When building your pipe, start with creating the grouping variable and then proceed with the remaining tasks.

```
#begin the pipe by creating the grouping variable
mean_hits <- merged %>%
  mutate(hhi_group = case_when( #utilize case_when to set the grouping var.
    hh_income > mean(hh_income, na.rm = TRUE) ~ "above_average",
    hh_income < mean(hh_income, na.rm = TRUE) ~ "below_average"
  )) %>%
  group_by(hhi_group) %>%
  summarise(mean_crime = mean(crime, na.rm = TRUE), #mean of crime hits
            mean_loans = mean(loans, na.rm = TRUE), #mean of loan hits
            count= n()) #verify this worked correctly, should be 341 total
mean_hits
```

```
## # A tibble: 3 x 4
##   hhi_group     mean_crime mean_loans count
##   <chr>              <dbl>      <dbl> <int>
## 1 above_average       56.9       66.6   143
## 2 below_average       60.7       64.9   198
## 3 <NA>               NaN        NaN       4
```

What conclusions might you draw from this?

For cities with above average median household income, the mean of search popularity (hits) are 54.95 for crime and 65.22 for loans. For cities with below average median household income, the mean of search popularity for crime is 60.94 while for loans it is 63.55.

Some conclusions we can draw from this is that overall the means for loans were higher than that of crime between both house hold income groups. This indicates that for those living in Illinois during 2020, there was more interest in loans and finances than crime. However we see that there is a stark contrast between the hits for the above and below average groups. We can conclude that in below average cities, there is more interest for crime than that of above_average cities. This may indicate that crime is a more salient issue for those in below average cities while loans are more salient for cities of above average household income.

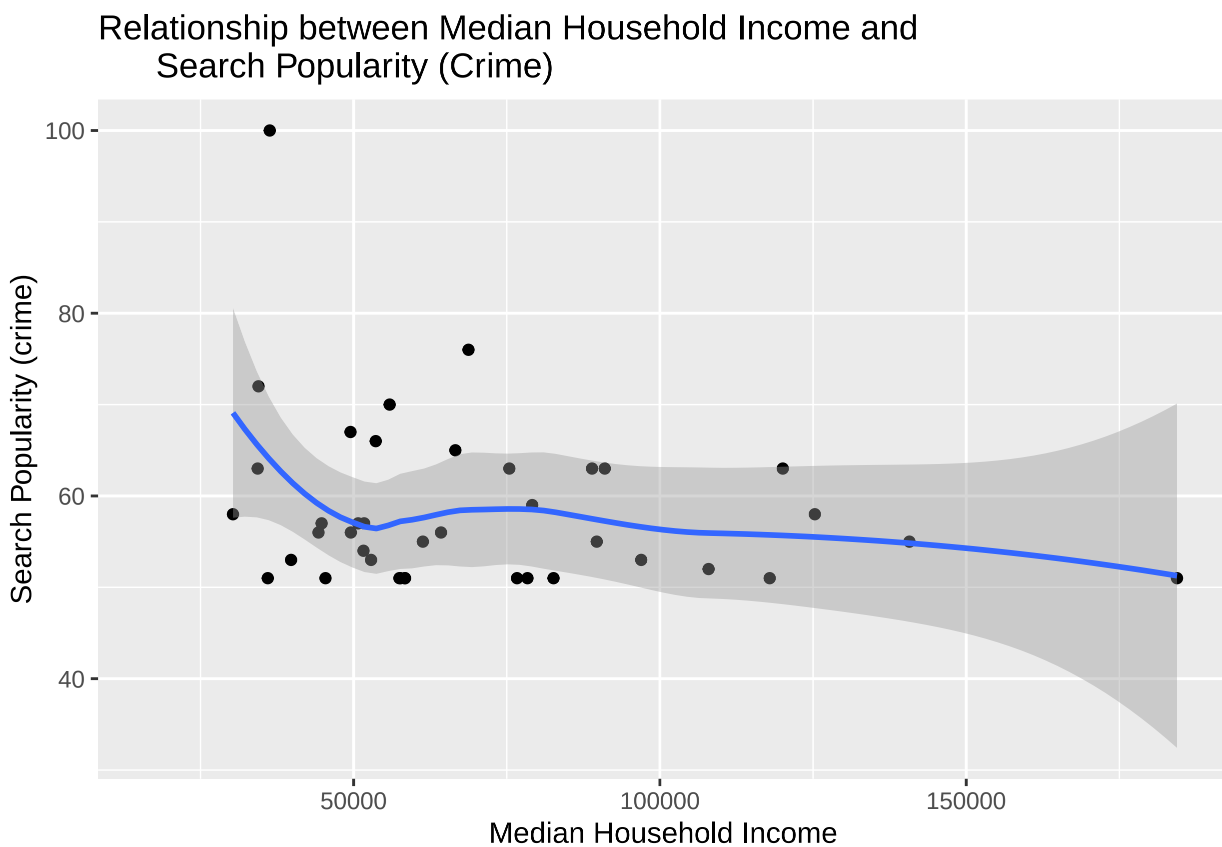
#e) Is there a relationship between the median hh_income & search popularity ? of the Google trends terms? Describe the relationship and use a scatterplot with qplot().

```
# Scatterplot for crime
qplot(x = hh_income, y = crime, data = merged,
      xlab = "Median Household Income",
      ylab = "Search Popularity (crime)",
      main = "Relationship between Median Household Income and
      Search Popularity (Crime)",
      geom = c("point", "smooth"))
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 305 rows containing non-finite outside the scale range
## (`stat_smooth()`).

## Warning: Removed 305 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



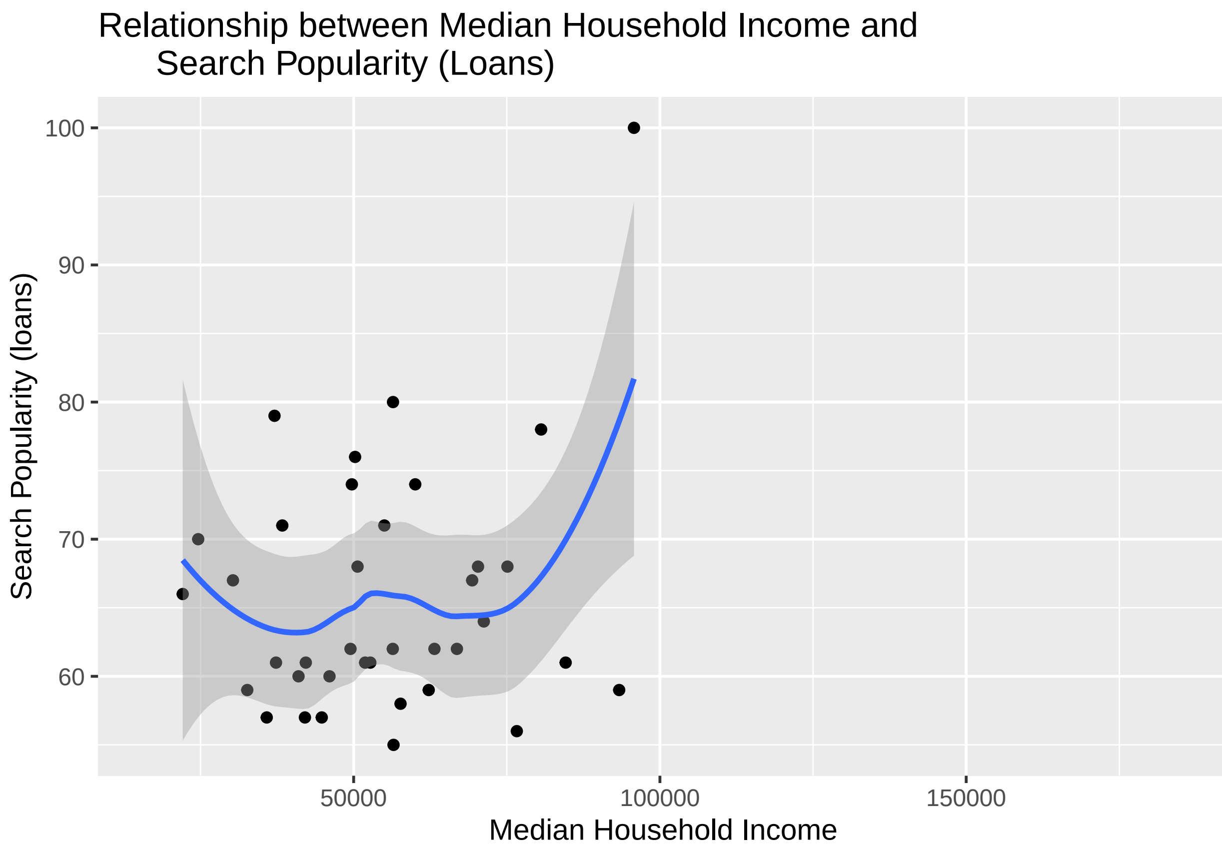

```
# Scatterplot for loans
qplot(x = hh_income, y = loans, data = merged,
      xlab = "Median Household Income",
```

```
ylab = "Search Popularity (loans)",
main = "Relationship between Median Household Income and
Search Popularity (Loans)",
geom = c("point", "smooth"))
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 308 rows containing non-finite outside the scale range
## (`stat_smooth()`).

## Warning: Removed 308 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



The relationship we see here is that for the case of keyword "crime", as median household income increases, the search popularity decreases. In the case of keyword "loans", as median household income increases, the search popularity increases until we see a subtle decrease past around $115,000USD. This inverse relationship is evident as discussed in part one.

#f) Repeat the above steps using the covid data and the ACS data.

```
#first using acs_il_clean and covid_city, covid_city is the preferred
#combination of "pandemic" and "quarantine".

not_matching_covid <- acs_il_clean %>%
  filter(!location %in% covid_city$location)

not_matching2_covid <- covid_city %>%
  filter(!location %in% acs_il_clean$location)
```

```r
nrow(not_matching_covid) # 1125 locations in acs_il_clean not in res_city
```

```
## [1] 1130
```

```r
nrow(not_matching2_covid) # 11 locations in res_city not in acs_il_clean
```

```
## [1] 12
```

```r
merged_covid <- acs_il_clean %>%
  inner_join(covid_city, by="location")

#double-checked to make sure cities in not_matching 2 did not show up
dim(merged_covid) # 341 11
```

```
## [1] 336  11
```

```r
mean_hits_covid <- merged_covid %>%
  mutate(hhi_group = case_when( #utilize case_when to set the grouping var.
    hh_income > mean(hh_income, na.rm = TRUE) ~ "above_average",
    hh_income < mean(hh_income, na.rm = TRUE) ~ "below_average"
  )) %>%
  group_by(hhi_group) %>%
  summarise(mean_pandemic = mean(pandemic, na.rm = TRUE), #mean of crime hits
            mean_quarantine = mean(quarantine, na.rm = TRUE), #mean of loan hits
            count= n()) #verify this worked correctly, should be 341 total
mean_hits_covid
```

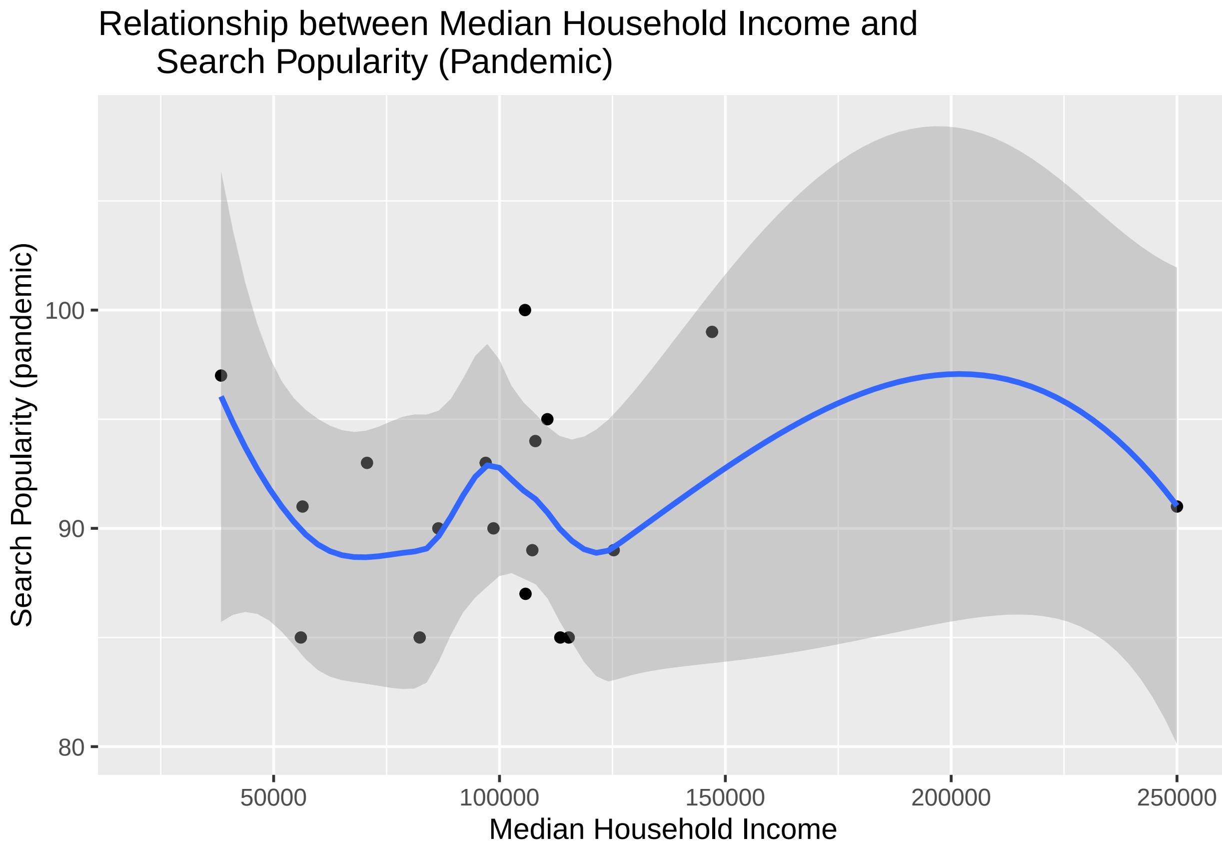
```
## # A tibble: 3 x 4
##   hhi_group     mean_pandemic mean_quarantine count
##   <chr>                 <dbl>           <dbl> <int>
## 1 above_average            91            80.8   120
## 2 below_average            91            79     213
## 3 <NA>                    NaN           NaN       3
```

```r
# Scatterplot for pandemic
qplot(x = hh_income, y = pandemic, data = merged_covid,
      xlab = "Median Household Income",
      ylab = "Search Popularity (pandemic)",
      main = "Relationship between Median Household Income and
      Search Popularity (Pandemic)",
      geom = c("point", "smooth"))
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 318 rows containing non-finite outside the scale range
## (`stat_smooth()`).

## Warning: Removed 318 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

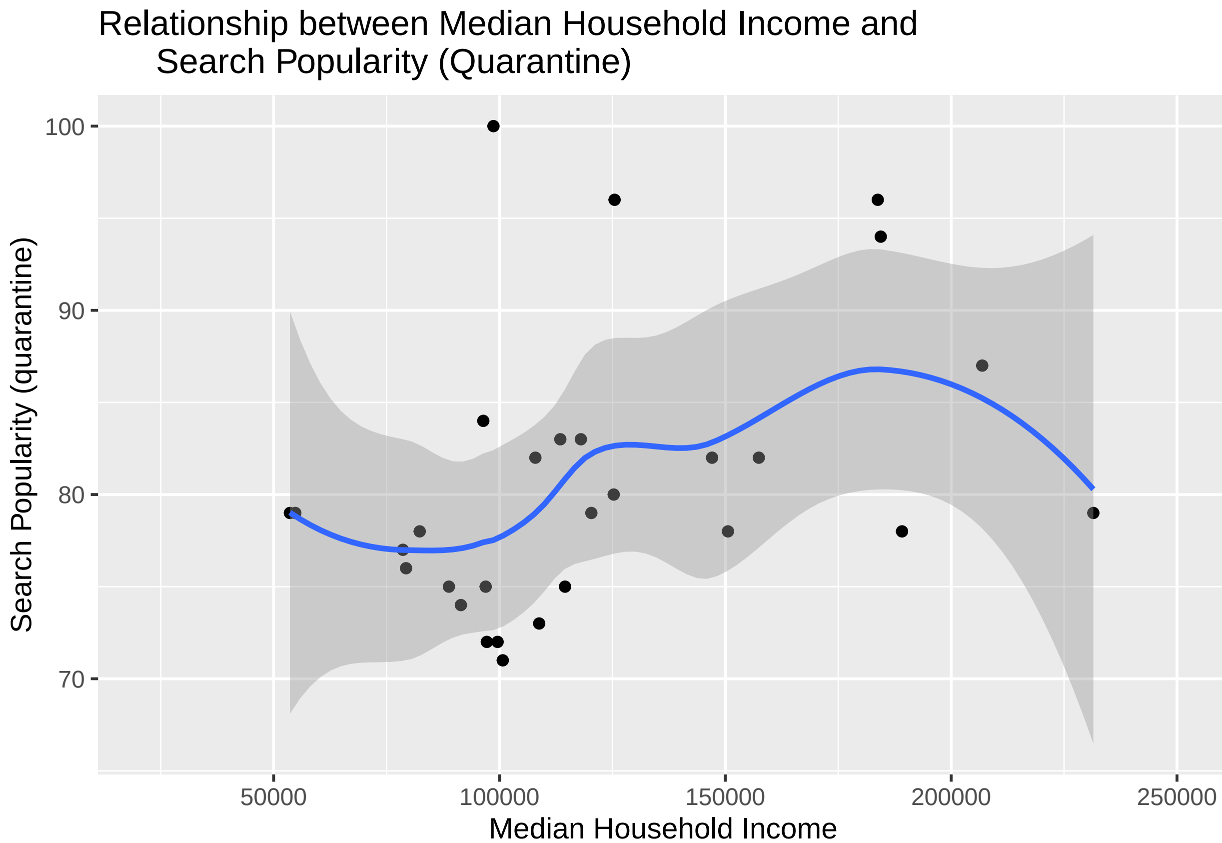



```
# Scatterplot for quarantine
qplot(x = hh_income, y = quarantine, data = merged_covid,
      xlab = "Median Household Income",
      ylab = "Search Popularity (quarantine)",
      main = "Relationship between Median Household Income and
      Search Popularity (Quarantine)",
      geom = c("point", "smooth"))
```

```
## `geom_smooth()` using method = 'loess' and formula = 'y ~ x'

## Warning: Removed 307 rows containing non-finite outside the scale range
## (`stat_smooth()`).

## Warning: Removed 307 rows containing missing values or values outside the scale range
## (`geom_point()`).
```



What conclusions might you draw from this?

For cities with above average median household income, the mean of search popularity (hits) are 89.00 for pandemic and 80.24 for quarantine. For cities with below average median household income, the mean of search popularity for crime is 85.33 while for loans it is 76.00.

Some conclusions we can draw from this is that overall the means for pandemic were higher than that of quarantine between both house hold income groups. This indicates that for those living in Illinois during 2020, there was more interest in pandemic information than quarantine. The difference between the keywords across both groups is around 9 points which indicates that the interest for “pandemic” is greater than “quarantine” on a similar order for both groups. We see that there is a much greater mean for above average cities and we can conclude that this may be a part of internet access during a time of mass lock downs and inaccessibility to spaces that provide free internet. Additionally, we now know that cities and areas with below average median household income were most impacted by covid and struggled with quarantine efforts, this is reflected in the means. Looking at the scatter plots, we see that the general shape for both keywords is similar, this is also reflected in the proportional relationship of both keywords. As median household income increases, the search popularity also increases until we see a decrease at around $200,000 USD.