Assignment2_727

Akari & Zhuoer

2023-10-03

Github link = https://github.com/ZuorW/SURV727.git

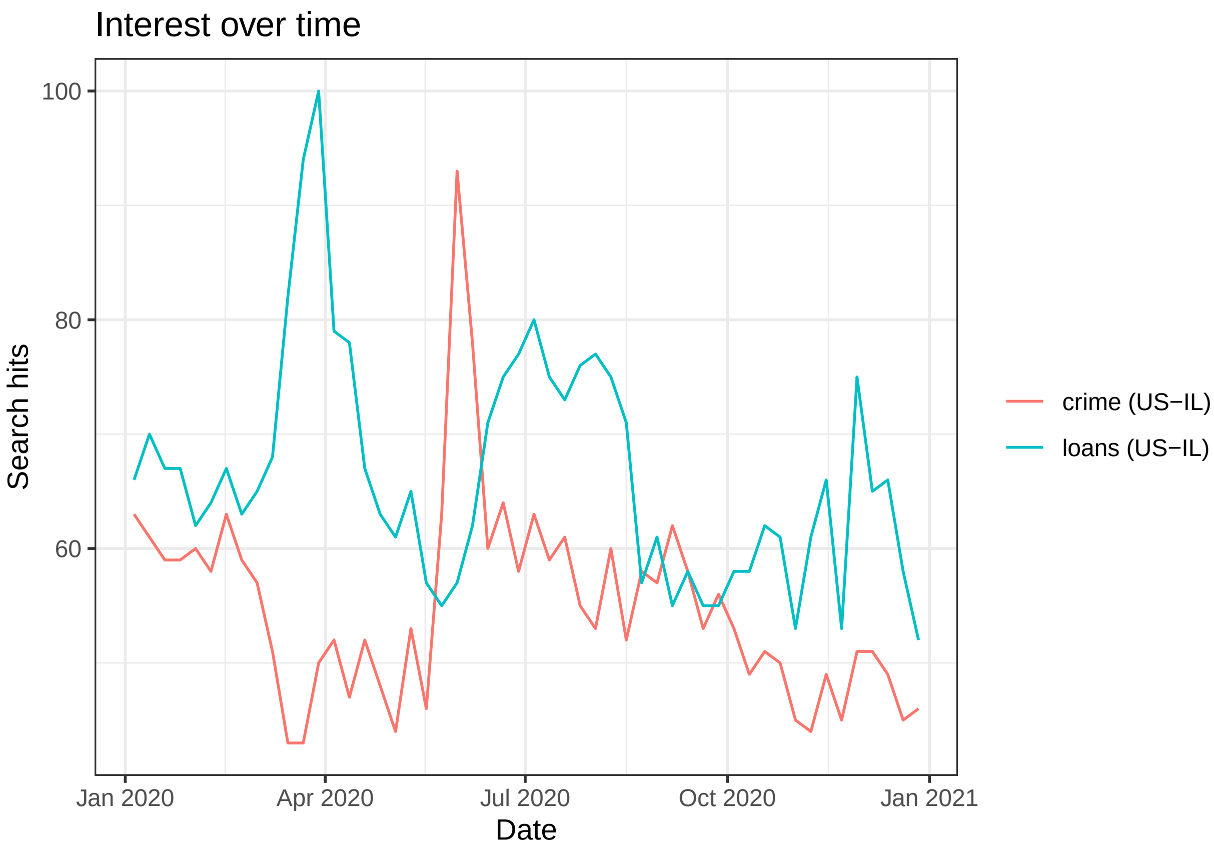
In this assignment, you will pull from APIs to get data from various data sources and use your data wrangling skills to use them all together. You should turn in a report in PDF or HTML format that addresses all of the questions in this assignment, and describes the data that you pulled and analyzed. You do not need to include full introduction and conclusion sections like a full report, but you should make sure to answer the questions in paragraph form, and include all relevant tables and graphics.

Whenever possible, use piping and `dplyr`. Avoid hard-coding any numbers within the report as much as possible.

Pulling from APIs

Our first data source is the Google Trends API. Suppose we are interested in the search trends for `crime` and `loans` in Illinois in the year 2020. We could find this using the following code:

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



Answer the following questions for the keywords "crime" and "loans".

- Find the mean, median and variance of the search hits for the keywords.

```
# Check what the data looks like
#res$interest_over_time %>% head()

#transform the data.frame into tibble
res_time = as_tibble(res$interest_over_time)

# Also compute the mean, sd, variance of each keyword
table1 <- res_time %>%
  group_by(keyword) %>%
  summarize(mean_hits = mean(hits),
            median = median(hits),
            var_hits = var(hits))
table1
```

```
## # A tibble: 2 x 4
##   keyword mean_hits median var_hits
##   <chr>       <dbl>  <dbl>    <dbl>
## 1 crime        55.0     53     78.1
## 2 loans        66.5     65    101.
```

The keyword `crime` had a mean search hit of 54.9807692307692 with a median of 53 and a variance of 78.1368778280543. The keyword `loans` had a mean search hit of 66.5 with a median of 65 and a variance of 101.392156862745.

- 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.

```
#transform the data.frame into tibble
rest_city <- tibble(res$interest_by_city)

#reshape the data & sort loans column in descending order
city_ranking <- rest_city %>%
  pivot_wider(names_from = keyword,
              values_from = hits) %>%
  arrange(., desc(loans))

#display first few rows of the ranking to find the highest searched
head(city_ranking)
```

```
## # A tibble: 6 x 5
##   location      geo   gprop crime loans
##   <chr>         <chr> <chr> <int> <int>
## 1 Hinckley      US-IL web      NA   100
## 2 Carrier Mills US-IL web      NA    96
## 3 Glasford      US-IL web      NA    94
## 4 Riverton      US-IL web      NA    88
## 5 Georgetown    US-IL web      NA    88
## 6 Rosemont      US-IL web      44    87
```

The city Hinckley has the highest search frequency for `loans`, followed by Carrier Mills, and Glasford.

- Is there a relationship between the search intensities between the two keywords we used?

```
# Run Pearson correlation test
cor1 <- cor.test(city_ranking$loans, city_ranking$crime)
cor1
```

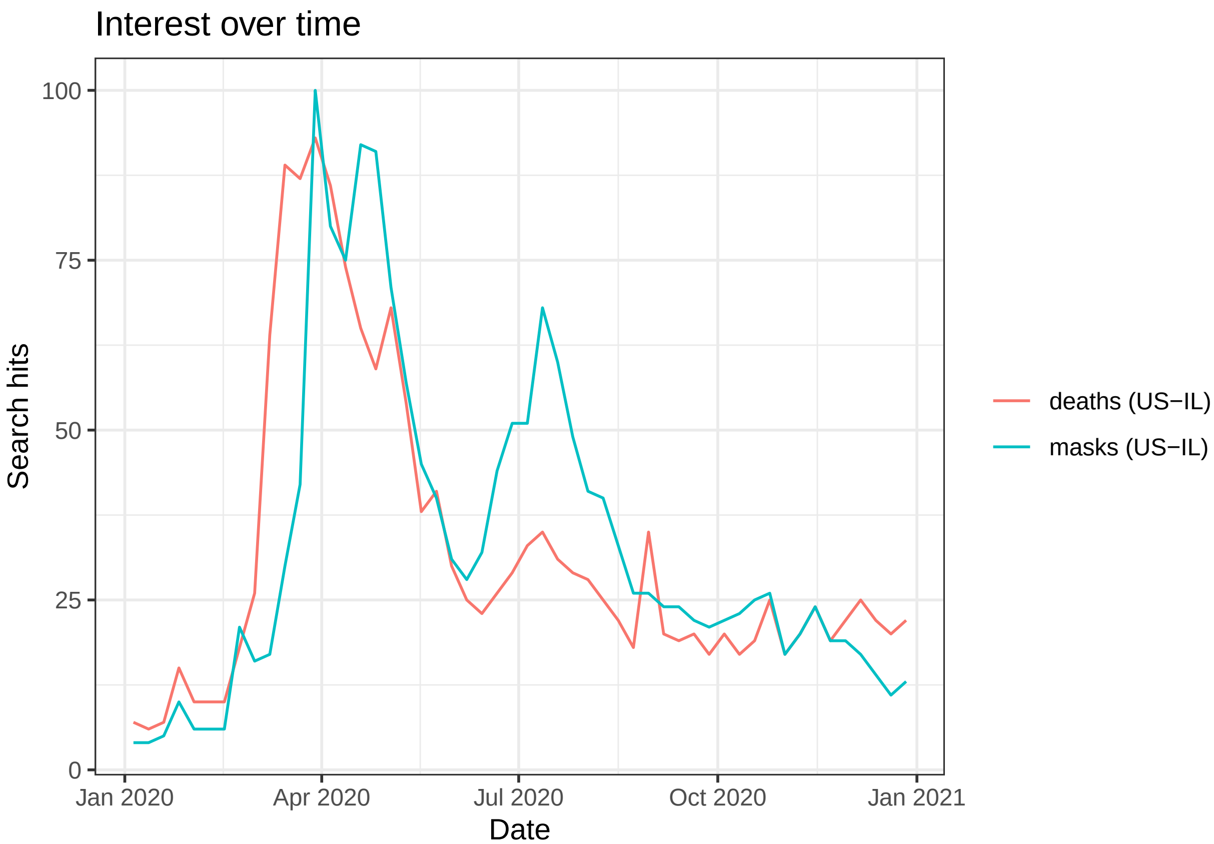
```
##
##  Pearson's product-moment correlation
##
## data:  city_ranking$loans and city_ranking$crime
## t = 0.49472, df = 14, p-value = 0.6285
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.3899644  0.5885433
## sample estimates:
##       cor
## 0.1310796
```

While `loans` had a higher mean search frequency over time, there does not seem to be a large difference compared to the search frequency of `crime`. However, patterns can be seen in the first plot. The two keywords seems to have an inverse relationship where search frequencies for `loans` are high when `crime` is low in the first peak/dip around April 2020. However, the pattern fades after around July 2020. We tested the relationship between the variables for interest by city to properly with a Pearson correlation test. The test suggests that there is no correlation between `loans` and `crime` ($r = 0.1310796$, $p = 0.6284683$).

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.

Keywords: masks & deaths

```
# Create another dataset with new keywords
res_2 <- gtrends(c("masks", "deaths"),
               geo = "US-IL",
               time = "2020-01-01 2020-12-31",
               low_search_volume = TRUE)
plot(res_2)
```



At first glance, the frequencies have a similar shape in the first half of 2020, but the pattern becomes less clear in the second half. The initial spike in search hits of both keywords understandably corresponds to near the beginning of the pandemic when mask mandates were placed and people may have been searching for information on the rising cases of death due to infections.

```
#check data
#res_2 %>% head()

#transform data into tibble
res_time2 <- tibble(res_2$interest_over_time)

# Compute the mean, standard deviation, and variance of search hits per keyword
table2 <- res_time2 %>%
  group_by(keyword) %>%
  summarize(mean_hits = mean(hits),
            median_hits = median(hits),
            var_hits = var(hits))
table2
```

```
## # A tibble: 2 x 4
##   keyword mean_hits median_hits var_hits
##   <chr>       <dbl>       <dbl>    <dbl>
## 1 deaths       32          24.5     515.
## 2 masks        33.4        25.5     585.
```

The search frequency for `masks` over time had a mean of 32 with a median of 24.5 and a variance of 514.627450980392. The search frequency for `deaths` over time had a mean of 33.4423076923077 with a median of 25.5 and a variance of 585.310331825038. Both keywords have a similar mean and a high variance.

```r
# Transform data into tibble
rest_city2 <- res_2$interest_by_city

# Check data
rest_city2 %>%
  arrange(desc(location)) %>%
  glimpse()
```

```
## Rows: 400
## Columns: 5
## $ location <chr> "Woodlawn", "Winslow", "Winnetka", "Winnetka", "Winnebago", "~
## $ hits     <int> NA, NA, 84, 55, 32, 60, 57, 55, 45, 58, NA, NA, NA, 55, 47, 3~
## $ keyword  <chr> "deaths", "masks", "masks", "deaths", "masks", "masks", "deat~
## $ geo      <chr> "US-IL", "US-IL", "US-IL", "US-IL", "US-IL", "US-IL", "US-IL"~
## $ gprop    <chr> "web", "web", "web", "web", "web", "web", "web", "web", "web"~
```

```r
#highest search frequency for "masks"
city_ranking2_masks <- rest_city2 %>%
  pivot_wider(names_from = keyword, values_from = hits) %>%
  arrange(., desc(masks))
head(city_ranking2_masks)
```

```
## # A tibble: 6 x 5
##   location      geo   gprop masks deaths
##   <chr>         <chr> <chr> <int>  <int>
## 1 Atlanta       US-IL web     100     NA
## 2 Waterman      US-IL web      92     NA
## 3 Geneva        US-IL web      88     NA
## 4 Hudson        US-IL web      85     NA
## 5 Winnetka      US-IL web      84     55
## 6 Highland Park US-IL web      70     45
```

```r
#highest search frequency for "deaths"
city_ranking2_deaths <- rest_city2 %>%
  pivot_wider(names_from = keyword, values_from = hits) %>%
  arrange(., desc(deaths))
head(city_ranking2_deaths)
```

```
## # A tibble: 6 x 5
##   location    geo   gprop masks deaths
##   <chr>       <chr> <chr> <int>  <int>
## 1 Carthage    US-IL web      NA    100
## 2 Galena      US-IL web      NA     90
## 3 Sherrard    US-IL web      37     76
## 4 Harvard     US-IL web      NA     75
## 5 Creve Coeur US-IL web      NA     70
```

```
## 6 Buffalo     US-IL web      NA     69
```

The city of Carthage had the highest search frequency for the keyword "deaths", followed by Galena and Sherrard. For the keyword "masks", Atlanta had the highest search frequency followed by Waterman and Geneva.

```
# Reshape data from long to wide format using keywords
wide_2 <-
  rest_city2 %>%
  pivot_wider(names_from = keyword,
           values_from = hits)

# Run Pearson correlation test
cor2 <- cor.test(wide_2$masks, wide_2$deaths)
cor2
```

```
##
##  Pearson's product-moment correlation
##
## data:  wide_2$masks and wide_2$deaths
## t = 0.42125, df = 38, p-value = 0.6759
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.2486127  0.3717900
## sample estimates:
##        cor
## 0.06817656
```

We conducted a Pearson correlation test to see if the search frequencies of the two keywords have a relationship. The test revealed that there is no significant correlation between masks and deaths ($r = 0.0681766$, $p = 0.6759468$).

Google Trends + ACS

Now lets add another data set. The `censusapi` package provides a nice R interface for communicating with this API. However, before running queries we need an access key. This (easy) process can be completed here:

https://api.census.gov/data/key_signup.html

Once you have an access key, store this key in the `cs_key` object. We will use this object in all following API queries.

```
cs_key <- "7b1cc9af0a42634e3ba57f9a8f5d0098cdedc5e4"
```

In the following, we request basic socio-demographic information (population, median age, median household income, income per capita) for cities and villages in the 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)
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 that represent missings to NAs.

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

Now, it might be useful to rename the socio-demographic variables (`B01001_001E` etc.) in our data set and assign more meaningful names.

```
acs_il <-
  acs_il %>%
  rename(pop = B01001_001E,
         age = B06002_001E,
         hh_income = B19013_001E,
         income = B19301_001E)
```

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.

```
# Check headers
# acs_il %>% head()

# Create new location variable without city/village
no_village <- gsub(' village, Illinois', '', acs_il$NAME) #remove "village, IL" from NAME and store
no_cityvill <- gsub(' city, Illinois', '', no_village) #take above and remove remaining "city, IL"
acs_with_loc <-
  acs_il %>%
  mutate(location = no_cityvill) #add new variable with only city names

# Check headers
# acs_with_loc %>%
#   head()
```

Answer the following questions with the "crime" and "loans" Google trends data and the ACS data.

- First, check how many cities don't appear in both data sets, i.e. cannot be matched. Then, create a new data set by joining the Google Trends and the ACS data. Keep only cities that appear in both data sets.

```r
# Merge ACS to gtrends data by city only keeping cases that match
merged <-
  city_ranking %>%
  inner_join(acs_with_loc, by = "location")

nrow(merged)
```

```
## [1] 329
```

```r
#cites not in both data sets
n = nrow(acs_with_loc) - nrow(merged) -(nrow(city_ranking)-nrow(merged))
n
```

```
## [1] 1120
```

1120 cities do not appear in both sets.

- Compute the mean of the search popularity 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. What conclusions might you draw from this?

```r
# If household income is greater than its median, name group as above average, if not, name group as ab
# Then compute mean by group
bymedv <- merged %>%
  group_by(
    hhinc_med =
      ifelse(hh_income > mean(hh_income, na.rm = TRUE),
                      "above", "below")) %>%
                      summarize(mean_crime = mean(crime, na.rm = TRUE),
                      mean_loans = mean(loans, na.rm = TRUE))
bymedv
```

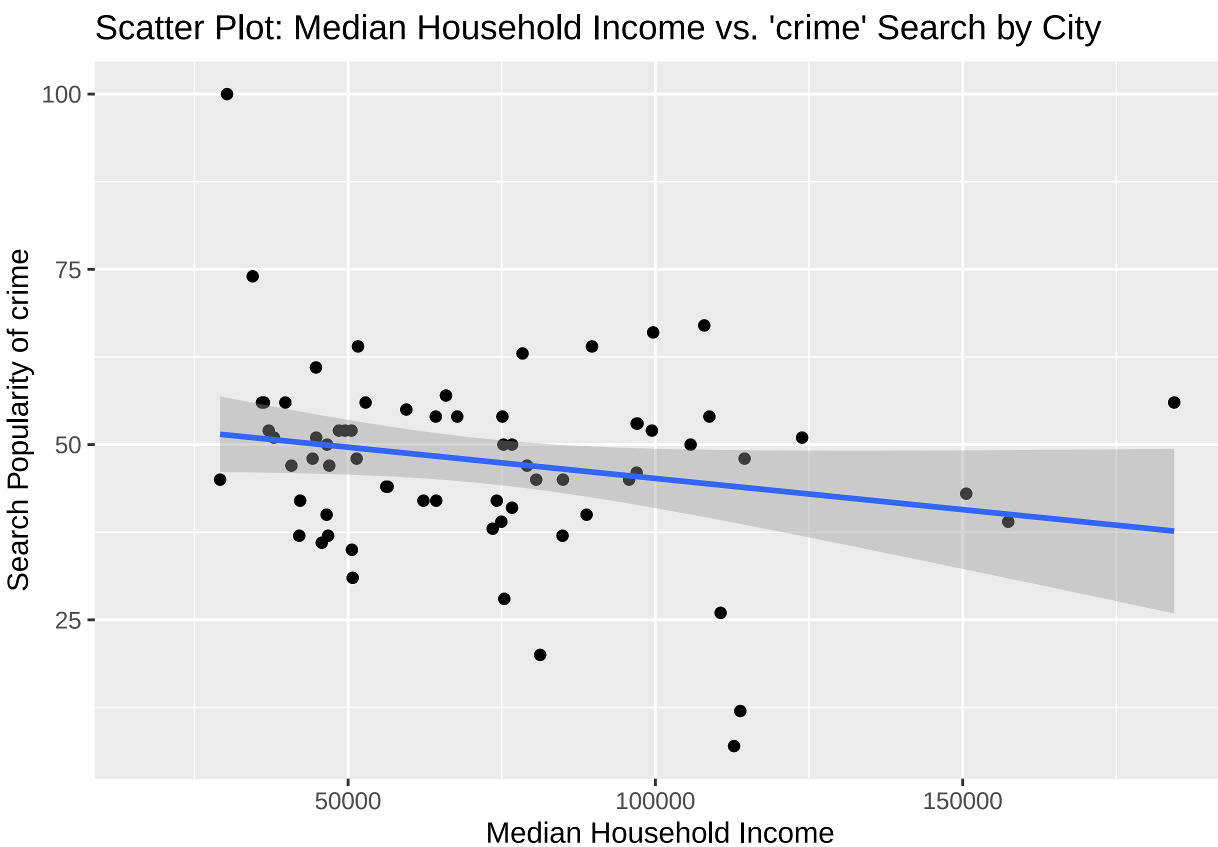
```
## # A tibble: 2 x 3
##   hhinc_med mean_crime mean_loans
##   <chr>          <dbl>      <dbl>
## 1 above           45.3       47.4
## 2 below           50.6       52.7
```

For cities that have an above average median household income, the search popularity of `crime` was 45.2631578947368 and 47.3518518518519 for `loans`. For cities that have a below average median household income, the search popularity of `crime` was 50.6333333333333 and 52.6543209876543 for `loans`. Cities with a below average household income had a higher search rate for both keywords. We conclude that crime rates may be higher in below average cities which may lead to more search hits for `crime`, and that people in these cities may search for `loans` more because there is a higher chance that they would take out loans to supplement their lower financial status.

- Is there a relationship between the median household income and the search popularity of the Google trends terms? Describe the relationship and use a scatterplot with `qplot()`.

```r
# Plot for crime
qplot(hh_income, crime, data = merged)+
  geom_point() +
  geom_smooth(method = lm) +
  labs(title = "Scatter Plot: Median Household Income vs. 'crime' Search by City",
       x = "Median Household Income",y = "Search Popularity of crime")
```

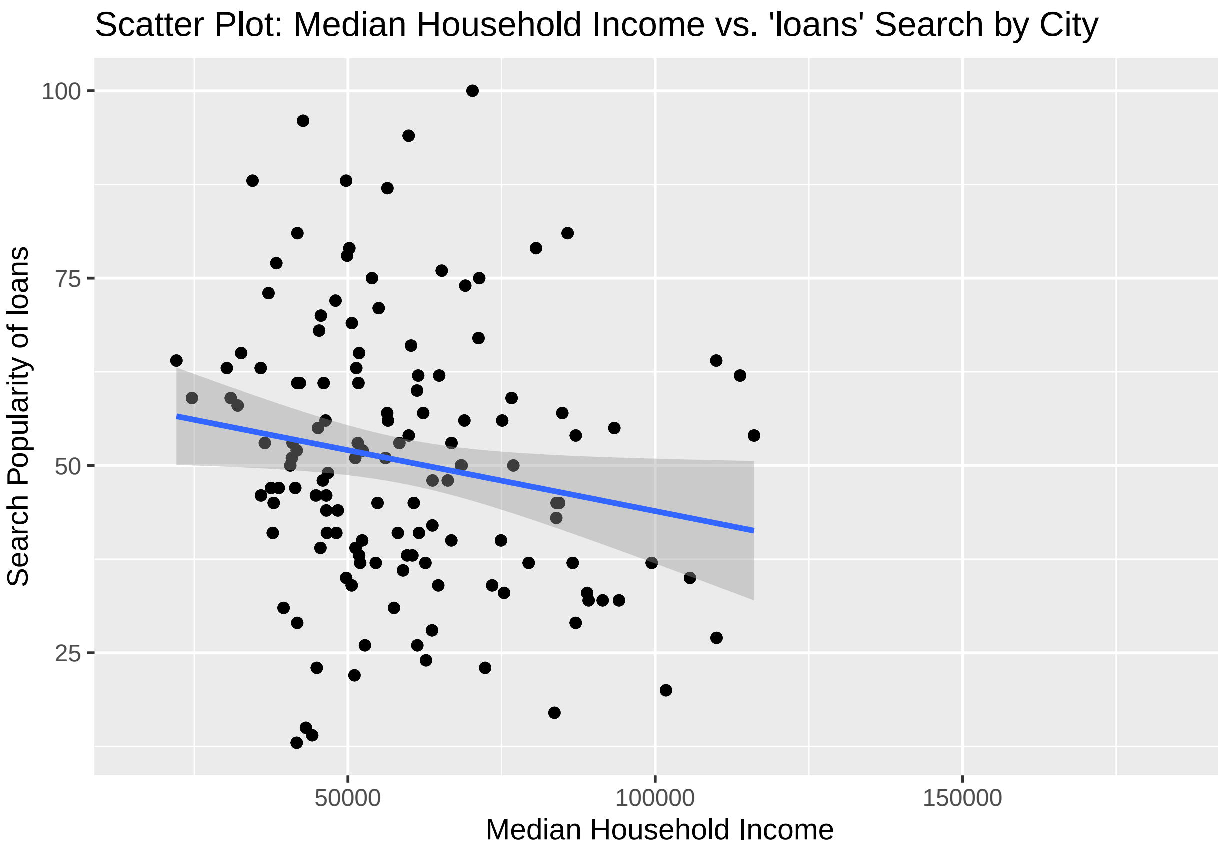
```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
# Correlation test
cor3 <- cor.test(merged$hh_income, merged$crime)

# Plot for loans
qplot(hh_income, loans, data = merged) +
  geom_point() +
  geom_smooth(method = lm) +
  labs(title = "Scatter Plot: Median Household Income vs. 'loans' Search by City",
       x = "Median Household Income",y = "Search Popularity of loans")
```

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




```
# Correlation test
cor4 <- cor.test(merged$hh_income, merged$loans)
```

In the plot for the median household income and the search popularity of `crime`, much of the data is gathered in the lower half of the median household income but there is no clear pattern. A Pearson correlation test supports this by showing that there is no correlation between the two variables (r = -0.2120137, p = 0.0826191). On the other hand, the plot for `loans` shows a clear pattern in which higher search hits are centered around the lower end of median household income, suggesting a relationship between the two variables. We tested this relationship using a Pearson correlation test. There was a significant negative correlation (r = -0.1770014, p < .001).

Repeat the above steps using the covid data and the ACS data.

```
# Merge ACS to gtrends data by city only keeping cases that match
merged_2 <-
  wide_2 %>%
  inner_join(acs_with_loc, by = "location")

merged_2 %>%
  head()
```

```
## # A tibble: 6 x 12
##   location       geo   gprop masks deaths state place NAME    pop   age hh_income
##   <chr>          <chr> <chr> <int>  <int> <chr> <chr> <chr> <dbl> <dbl>     <dbl>
## 1 Atlanta        US-IL web     100     NA 17    02752 Atla~  2156  35.5     55694
## 2 Waterman       US-IL web      92     NA 17    79163 Wate~  1738  36.5     82500
## 3 Geneva         US-IL web      88     NA 17    28872 Gene~ 21843  40.4    116083
## 4 Hudson         US-IL web      85     NA 17    36438 Huds~  2128  35       96538
```

```
## 5 Winnetka      US-IL web      84      55 17    82530 Winn~ 12361  42.1    250001
## 6 Highland Park US-IL web      70      45 17    34722 High~ 29596  47.2    147067
## # i 1 more variable: income <dbl>
```

```
nrow(merged_2)
```

```
## [1] 319
```

```
#cites not in both data sets
n2 = nrow(acs_with_loc) - nrow(merged) -(nrow(wide_2)-nrow(merged))
n2
```

```
## [1] 1126
```

1126 cities do not appear in both sets.

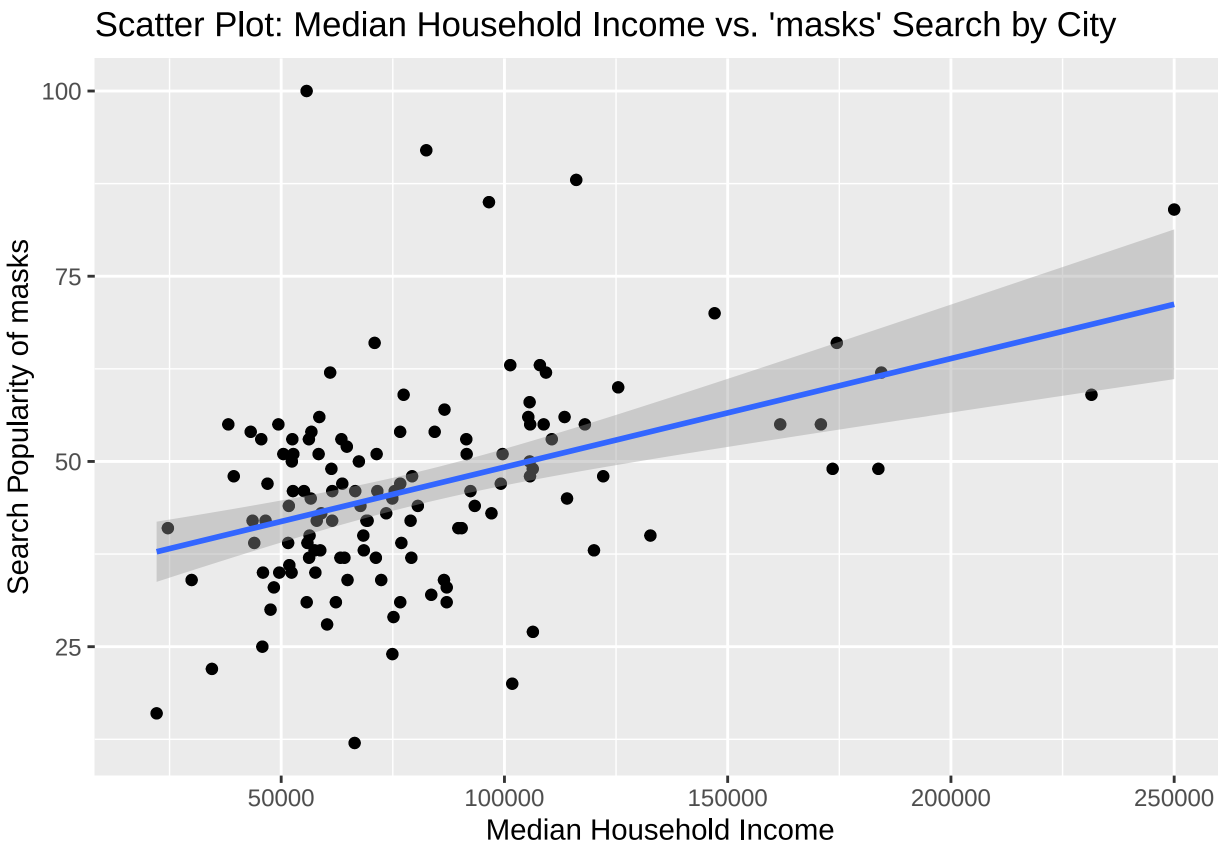
```
# If household income is greater than its median, name group as above average, if not, name group as ab
# Then compute mean by group
table_inc <- merged_2 %>%
  group_by(
    hhinc_med =
      ifelse(hh_income > median(hh_income, na.rm = TRUE),
                       "above", "below")) %>%
                       summarize(mean_masks = mean(masks, na.rm = TRUE),
                       mean_deaths = mean(deaths, na.rm = TRUE))
table_inc
```

```
## # A tibble: 2 x 3
##   hhinc_med mean_masks mean_deaths
##   <chr>          <dbl>       <dbl>
## 1 above           48.2        45.2
## 2 below           42.9        40.2
```

For cities that have an above average median household income, the search popularity of `masks` was 48.2235294117647 and 45.2272727272727 for `deaths`. For cities that have a below average median household income, the search popularity of `masks` was 42.8863636363636 and 40.2162162162162 for `deaths`. Those in cities with below average household income had a lower search rate for both keywords. We conclude there are more frequent searches of masks and death in cities with above average income. The possibility that people pay greater attention to protective gears and death cases related to the pandemic in richer areas may help to explain the observed difference in data.

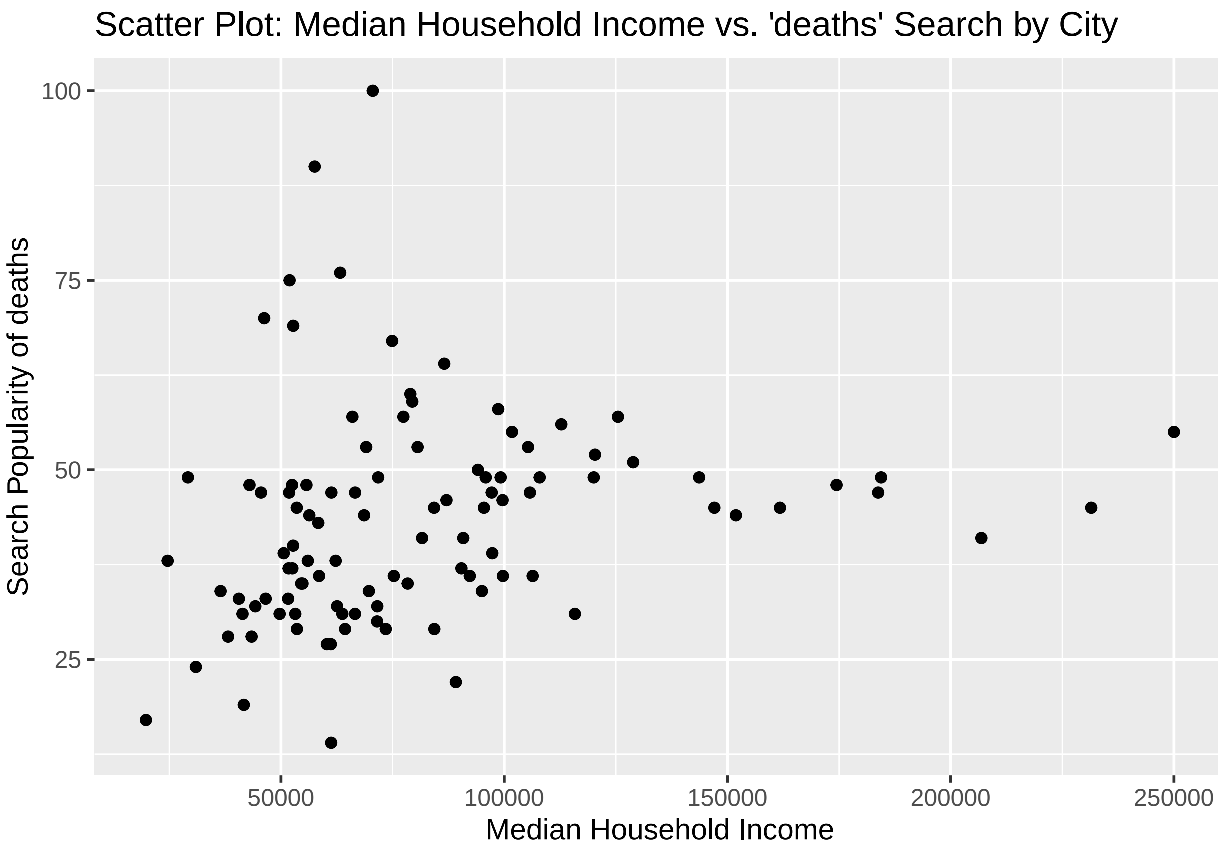
```
# Plot for masks
qplot(hh_income, masks, data = merged_2) +
  geom_point() +
  geom_smooth(method = lm) +
  labs(title = "Scatter Plot: Median Household Income vs. 'masks' Search by City",
       x = "Median Household Income", y = "Search Popularity of masks")
```

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




```
# Correlation test
cor_mask <- cor.test(merged_2$hh_income, merged_2$masks)

# Plot for deaths
qplot(hh_income, deaths, data = merged_2) +
  labs(title = "Scatter Plot: Median Household Income vs. 'deaths' Search by City",
       x = "Median Household Income", y = "Search Popularity of deaths")
```




```
# Correlation test
cor_dea <- cor.test(merged_2$hh_income, merged_2$deaths)
```

The Pearson correlation test shows that `masks` have a relationship with median household income ($r = 0.4040135$, $p < .001$). The data for `mask` search hits in the plot has less outliers with most of the data points gathering around the lower side of income. On the other hand, `deaths` did not have a relationship with median household income ($r = 0.2001904$, $p = 0.0426126$). This coincides with the data points in the plot for `deaths` being more spread out. Notably, people with lower household income who may be more at risk of being infected or spreading COVID-19 due to their socioeconomic status, may have searched for `masks` more frequently to buy or make them.