Midterm Qs1

Anous Ahmed

2025-10-17

Link to Repository: https://github.com/aahme102/Midterm-.git

Part II: Web scraping

```
library(rvest)
library(httr)
library(jsonlite)
library(tidycensus)
```

```
## Warning: package 'tidycensus' was built under R version 4.4.2
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.4.3
```

```
## Warning: package 'ggplot2' was built under R version 4.4.2
```

```
## Warning: package 'tidyr' was built under R version 4.4.2
```

```
## Warning: package 'dplyr' was built under R version 4.4.3
```

```
## Warning: package 'forcats' was built under R version 4.4.2
```

```
## -- 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   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter()         masks stats::filter()
## x purrr::flatten()        masks jsonlite::flatten()
## x readr::guess_encoding() masks rvest::guess_encoding()
## x dplyr::lag()            masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```r
library(ggplot2)
```

```r
url <- "https://www.scrapethissite.com/pages/simple/"
html <- read_html(url)
```

```r
block <- html |>
  html_elements("div.col-md-4.country")
block
```

```
## {xml_nodeset (250)}
##  [1] <div class="col-md-4 country">\n                        <h3 class="count ...
##  [2] <div class="col-md-4 country">\n                        <h3 class="count ...
##  [3] <div class="col-md-4 country">\n                        <h3 class="count ...
##  [4] <div class="col-md-4 country">\n                        <h3 class="count ...
##  [5] <div class="col-md-4 country">\n                        <h3 class="count ...
##  [6] <div class="col-md-4 country">\n                        <h3 class="count ...
##  [7] <div class="col-md-4 country">\n                        <h3 class="count ...
##  [8] <div class="col-md-4 country">\n                        <h3 class="count ...
##  [9] <div class="col-md-4 country">\n                        <h3 class="count ...
## [10] <div class="col-md-4 country">\n                        <h3 class="count ...
## [11] <div class="col-md-4 country">\n                        <h3 class="count ...
## [12] <div class="col-md-4 country">\n                        <h3 class="count ...
## [13] <div class="col-md-4 country">\n                        <h3 class="count ...
## [14] <div class="col-md-4 country">\n                        <h3 class="count ...
## [15] <div class="col-md-4 country">\n                        <h3 class="count ...
## [16] <div class="col-md-4 country">\n                        <h3 class="count ...
## [17] <div class="col-md-4 country">\n                        <h3 class="count ...
## [18] <div class="col-md-4 country">\n                        <h3 class="count ...
## [19] <div class="col-md-4 country">\n                        <h3 class="count ...
## [20] <div class="col-md-4 country">\n                        <h3 class="count ...
## ...
```

```r
Country = block |>
  html_element("h3.country-name") |>
  html_text2()
head (Country, 10)
```

```
##  [1] "Andorra"              "United Arab Emirates" "Afghanistan"
##  [4] "Antigua and Barbuda"  "Anguilla"             "Albania"
##  [7] "Armenia"              "Angola"               "Antarctica"
## [10] "Argentina"
```

```r
Capital <- block |>
  html_element("span.country-capital") |>
  html_text2()
head (Capital, 10)
```

```
##  [1] "Andorra la Vella" "Abu Dhabi"        "Kabul"            "St. John's"
##  [5] "The Valley"       "Tirana"           "Yerevan"          "Luanda"
##  [9] "None"             "Buenos Aires"
```

```
Population <- block %>%
  html_element("span.country-population") %>%
  html_text2()
head (Population, 10)
```

```
##  [1] "84000"    "4975593"  "29121286" "86754"    "13254"    "2986952"
##  [7] "2968000"  "13068161" "0"        "41343201"
```

```
Area <- block %>%
  html_element("span.country-area") %>%
  html_text2()
head(Area, 10)
```

```
##  [1] "468.0"     "82880.0"   "647500.0"  "443.0"     "102.0"     "28748.0"
##  [7] "29800.0"   "1246700.0" "1.4E7"     "2766890.0"
```

```
df = tibble(
  Country,
  Capital,
  Population,
  Area
)
```

```
head(df, 10)
```

```
## # A tibble: 10 x 4
##    Country              Capital          Population Area
##    <chr>                <chr>            <chr>      <chr>
##  1 Andorra              Andorra la Vella 84000      468.0
##  2 United Arab Emirates Abu Dhabi        4975593    82880.0
##  3 Afghanistan          Kabul            29121286   647500.0
##  4 Antigua and Barbuda  St. John's       86754      443.0
##  5 Anguilla             The Valley       13254      102.0
##  6 Albania              Tirana           2986952    28748.0
##  7 Armenia              Yerevan          2968000    29800.0
##  8 Angola               Luanda           13068161   1246700.0
##  9 Antarctica           None             0          1.4E7
## 10 Argentina            Buenos Aires     41343201   2766890.0
```

PART III: API Access

Step 1: Identify Relevant Variables

https://api.census.gov/data/2023/acs/acs5/groups/B19013.html

https://api.census.gov/data/2023/acs/acs5/groups/B28002.html

Median household income (in the past 12 months): **B19013_001E** Households with broadband Internet: **B28002_004E** Total households with any type of internet access: **B28002_001E**

Step 2: Retrieve Data

```
readRenviron("~/.Renviron")
```

```
county_data <- get_acs(
  geography = "county",
  variables  = c(
  median_income = "B19013_001E",
  broadband_internet = "B28002_004E",
  internet_access = "B28002_001E"
  ),
  state = "IL",
  year = 2023,
  survey = "acs5"
)
```

```
## Getting data from the 2019-2023 5-year ACS
```

Step 3: Clean and Transform Data

```
county_data1a = county_data %>%
  select(NAME, variable, estimate) %>%
  pivot_wider(
    names_from = variable,
    values_from = estimate
  )
```

```
county_data1a = county_data1a %>%
  rename(
    income = B19013_001,
    broadband = B28002_004,
    total_households = B28002_001
  ) %>%
  separate(NAME, into = c("county", "state"), sep = ", ") %>%
  mutate(broadband_rate = (broadband / total_households) * 100) %>%
  arrange(desc(broadband_rate))
```

Step 4: Analyze patterns

a) Compute the mean and median broadband rate across all Illinois counties.

```
# mean for the broadband rate
county_data1a %>%
  summarize(mean(broadband_rate, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `mean(broadband_rate, na.rm = TRUE)`
##                                  <dbl>
## 1                                 84.8
```

```
# median for the broadband rate
county_data1a %>%
  summarize(median(broadband_rate, na.rm = TRUE))
```

```
## # A tibble: 1 x 1
##   `median(broadband_rate, na.rm = TRUE)`
##                                    <dbl>
## 1                                   85.4
```

b) Identify the top 5 counties with the highest broadband access and the bottom 5 counties with the lowest.

```
# Counties with highest broadband access
county_data1a %>%
  arrange(desc(broadband_rate)) %>%
  select(county, broadband_rate) %>%
  slice(1:5)
```

```
## # A tibble: 5 x 2
##   county         broadband_rate
##   <chr>                   <dbl>
## 1 Kendall County           95.2
## 2 McHenry County           95.1
## 3 DuPage County            94.4
## 4 Lake County              93.5
## 5 Will County              93.2
```

```
# Counties with lowest broadband access
county_data1a %>%
  arrange(broadband_rate) %>%
  select(county, broadband_rate) %>%
  slice(1:5)
```

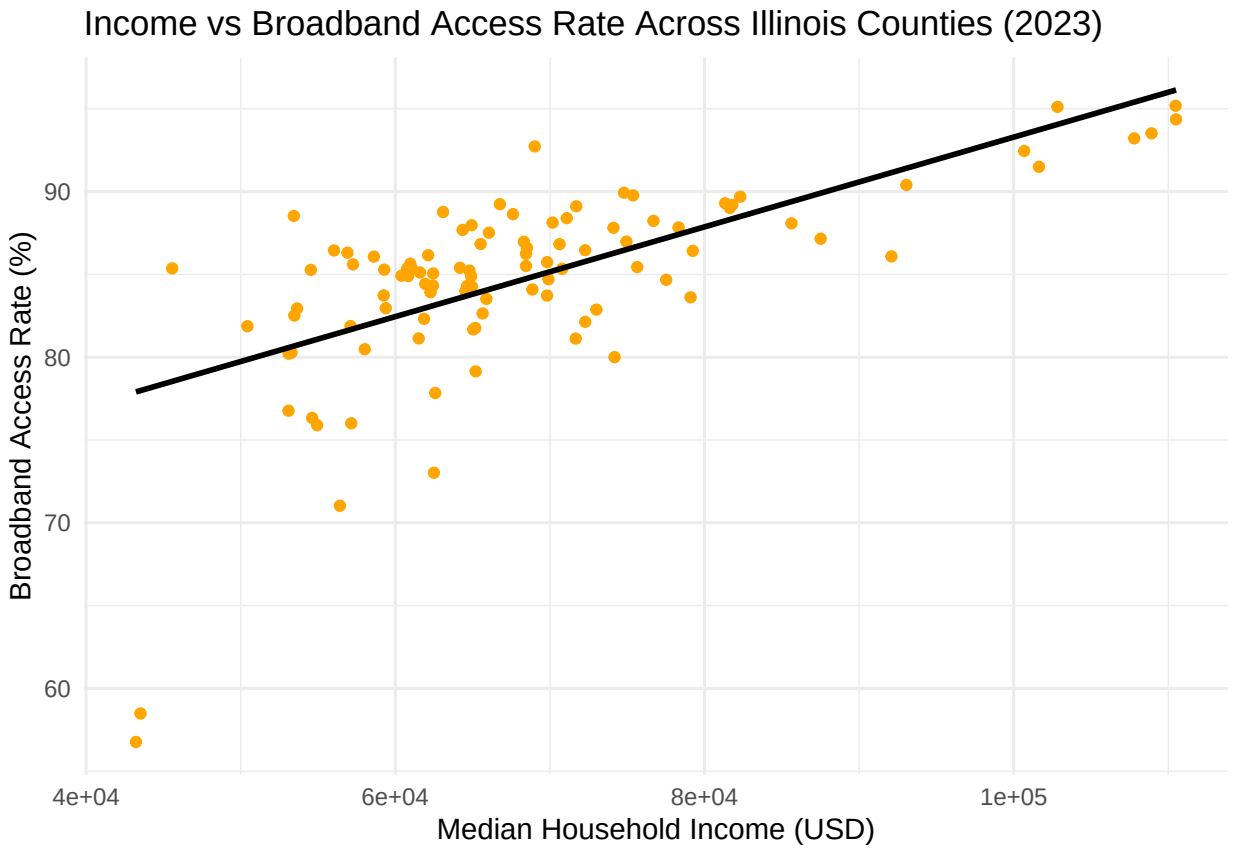
```
## # A tibble: 5 x 2
##   county           broadband_rate
##   <chr>                     <dbl>
## 1 Pulaski County             56.8
## 2 Alexander County           58.5
## 3 Union County               71.0
## 4 Pope County                73.0
## 5 Saline County              75.9
```

Step 5: Visualize the results

Scatterplot: Income vs Broadband Access Rate Across Illinois Counties (2023)

```
ggplot(county_data1a, aes(x = income, y = broadband_rate)) +
  geom_point(color = "orange") +
  geom_smooth(method = "lm", se = FALSE, color = "black") +
  labs(
    title = "Income vs Broadband Access Rate Across Illinois Counties (2023)",
    x = "Median Household Income (USD)",
    y = "Broadband Access Rate (%)"
  ) +
  theme_minimal()
```

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



Barplot

```
# top 10 broadband access
top_10 = county_data1a %>%
    slice_max(broadband_rate, n = 10)
top_10
```

```
## # A tibble: 10 x 6
##    county         state    income total_households broadband broadband_rate
##    <chr>          <chr>     <dbl>            <dbl>     <dbl>          <dbl>
##  1 Kendall County Illinois 110474            44526     42382           95.2
##  2 McHenry County Illinois 102836           116329    110646           95.1
##  3 DuPage County  Illinois 110502           349497    329798           94.4
##  4 Lake County    Illinois 108917           256660    240027           93.5
##  5 Will County    Illinois 107799           241310    224935           93.2
##  6 DeKalb County  Illinois  69022            39314     36455           92.7
##  7 Kane County    Illinois 100678           183196    169377           92.5
##  8 Monroe County  Illinois 101635            13830     12654           91.5
##  9 Grundy County  Illinois  93060            20518     18549           90.4
## 10 Madison County Illinois  74800           109385     98374           89.9
```

```
# bottom 10 broadband access
bottom_10 = county_data1a %>%
  slice_min(broadband_rate, n = 10)
bottom_10
```

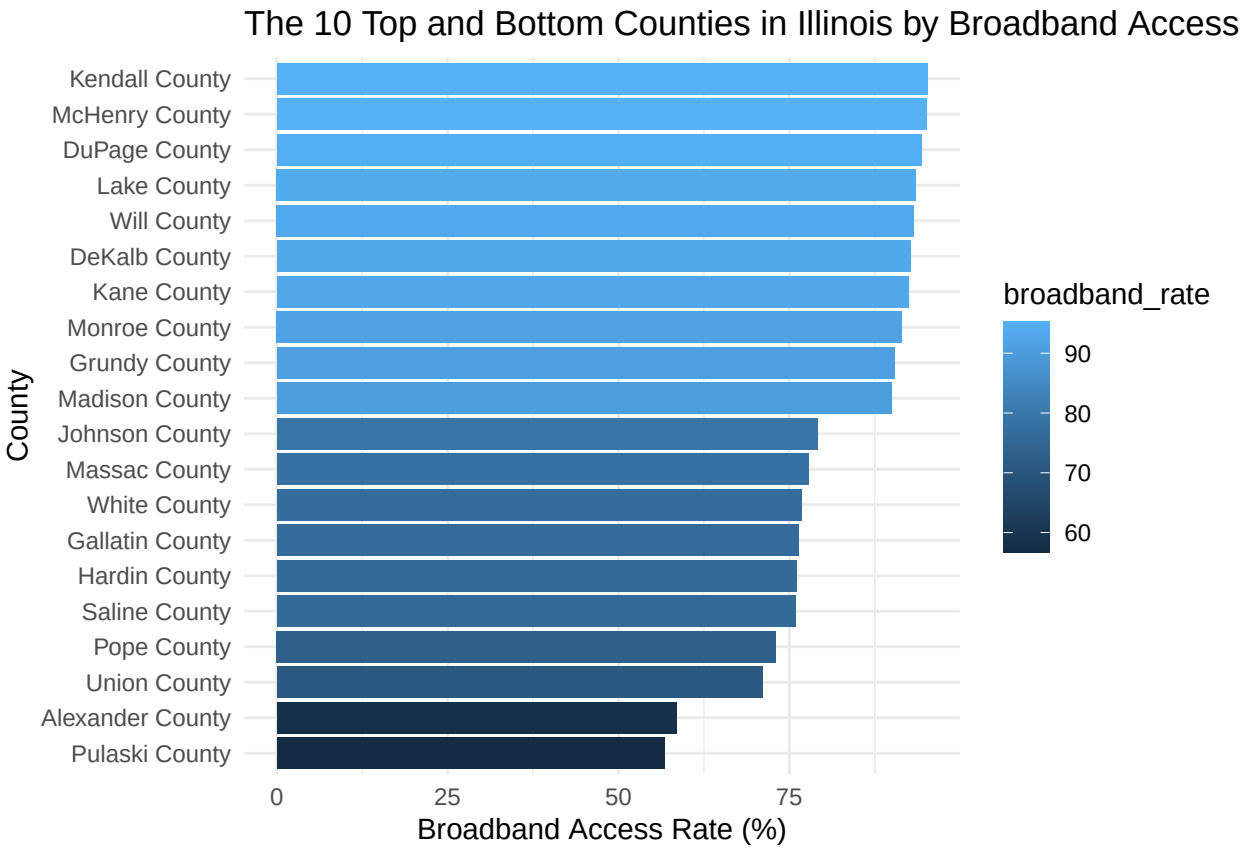
```
## # A tibble: 10 x 6
```

```
##    county           state    income total_households broadband broadband_rate
##    <chr>            <chr>     <dbl>            <dbl>     <dbl>          <dbl>
##  1 Pulaski County   Illinois  43227             1862      1057           56.8
##  2 Alexander County Illinois  43523             1826      1068           58.5
##  3 Union County     Illinois  56420             6914      4911           71.0
##  4 Pope County      Illinois  62500             1364       996           73.0
##  5 Saline County    Illinois  54945            10032      7614           75.9
##  6 Hardin County    Illinois  57155             1484      1128           76.0
##  7 Gallatin County  Illinois  54626             2096      1600           76.3
##  8 White County     Illinois  53097             5669      4352           76.8
##  9 Massac County    Illinois  62584             5482      4267           77.8
## 10 Johnson County   Illinois  65203             4133      3271           79.1
```

```
top_bottom_10 = rbind(top_10, bottom_10)
top_bottom_10
```

```
## # A tibble: 20 x 6
##    county           state    income total_households broadband broadband_rate
##    <chr>            <chr>     <dbl>            <dbl>     <dbl>          <dbl>
##  1 Kendall County   Illinois 110474            44526     42382           95.2
##  2 McHenry County   Illinois 102836           116329    110646           95.1
##  3 DuPage County    Illinois 110502           349497    329798           94.4
##  4 Lake County      Illinois 108917           256660    240027           93.5
##  5 Will County      Illinois 107799           241310    224935           93.2
##  6 DeKalb County    Illinois  69022            39314     36455           92.7
##  7 Kane County      Illinois 100678           183196    169377           92.5
##  8 Monroe County    Illinois 101635            13830     12654           91.5
##  9 Grundy County    Illinois  93060            20518     18549           90.4
## 10 Madison County   Illinois  74800           109385     98374           89.9
## 11 Pulaski County   Illinois  43227             1862      1057           56.8
## 12 Alexander County Illinois  43523             1826      1068           58.5
## 13 Union County     Illinois  56420             6914      4911           71.0
## 14 Pope County      Illinois  62500             1364       996           73.0
## 15 Saline County    Illinois  54945            10032      7614           75.9
## 16 Hardin County    Illinois  57155             1484      1128           76.0
## 17 Gallatin County  Illinois  54626             2096      1600           76.3
## 18 White County     Illinois  53097             5669      4352           76.8
## 19 Massac County    Illinois  62584             5482      4267           77.8
## 20 Johnson County   Illinois  65203             4133      3271           79.1
```

```
ggplot(top_bottom_10, aes(x = reorder(county, broadband_rate), y = broadband_rate, fill = broadband_rat
  geom_col() +
  coord_flip() +
  labs(
    title = "The 10 Top and Bottom Counties in Illinois by Broadband Access  Rate",
    x = "County",
    y = "Broadband Access Rate (%)"
  ) +
  theme_minimal()
```



Step 6: Reflection

1. What patterns do you observe between income and broadband access?

It can be illustrated from the scatterplot that median household income and the broadband access rate have a positive linear relationship. As the median household income increases for counties increases, the percentage of households with access to broadband also increases. Hence, illustrating how digital access grows in cases where economic resources increases. Therefore, wealthier households with greater economic resources are able to increase their access to digital connectivity.

2. What might explain the variation in broadband access across counties?

One of the major reasons why there is a variation in broadband access across counties is economic disparity. Since wealthier households have higher income, they are able to gain improved access to broadband services. However, lower income households lack the financial resources to attain such benefits. Moreover, the lower income households in certain counties may have poor broadband infrastructure in general due to the ignorance by such counties from making investments. On the other hand, there are also variations in digital literacy which explains as to why some counties have better access to broadband than others.

3. How could public administrators use this data to inform digital inclusion policies?

This data can be very beneficial for public administrators in directing affordable plans and digital inclusion policies toward counties that suffer from limited access to broadband facilties. Funds and grants can be allocated appropriately toward counties that have limited digital infrastructure. This can assist in tackling the barriers of digital divide and ensuring that there is equal access to digital technological solutions across different communities. Hence, enabling public trust in the government agencies.

4. What are some limitations of using ACS data for local decision-making?

There are several limitations associated with using ACS data for local decision-making. Since ACS depends on sampling, it has high margins of error (MOE). For instance, when comparing two local areas, there is a

large margin of error making it difficult to tell if the differences are statistically significant. Hence, undermining informed policy decisions. On the other hand, with ACS, data is self-reported causing inconsistencies across regions. Moreover, the ACS 5-year estimates are not designed for geographical areas that are undergoing rapid shifts like gentrification, natural disasters, or an economic boom. By the time the data would be released, it would already be outdated for planning purposes. For this reason, the ACS data should only be used to supplement public administrators when working with local surveys or administrative data for policy decisions.