Missing Persons in Pennsylvania and Ohio (1969–2024)

Exploratory Data Analysis of Demographic, Temporal, and Spatial Patterns

Sharvari, Sumaiya, Yuze

2025-12-15

1 Introduction

1.1 Motivation and Context

For our group project, we had decided to look into missing persons data. Certain areas that were especially interesting to us were the potential for any pattern in reported missing persons cases, in addition, any potential relevance to our geographical area.

From these interests, we were motivated to look at missing persons data in Pennsylvania and Ohio, which are very close geographically. For accurate reporting, we also focused on finding data from credible and updated sources.

This report will offer exploratory data analysis of a medium sized data set of reported missing persons data with visualization and inferences/insight on any patterns.

1.2 Research Question

For clarity and focus, we generalized our research question to such:

Do missing persons cases in Pennsylvania and Ohio from 1969 to 2024 exhibit over-representation across specific demographic groups, time spans, and spatial contexts?

2 Data Provenance

We decided to use missing persons data from NamUS (the National Missing and Unidentified Persons System), which has accurate and well-maintained information and is made fairly easily accessible to the public. We trust NamUS because they're a federal, government supported system and have worked directly with other trustworthy entities in law enforcement.

NamUS's data is contributed by law enforcement agencies, medical examiners, and authorized forensic professionals, and is maintained by the NIJ (the National Institute of Justice).

To obtain the data-set, we needed to register for an account for NamUS. Afterwards, a csv can be downloaded from the website after filtering through down to a manageable size.

In our repository, the data will be found under "MissingPersons.csv". The dataframe we worked with in data wrangling and visualization will follow a similar naming scheme "missingPersons".

```
  Case.Number      DLC Legal.Last.Name Legal.First.Name Missing.Age
1      MP2620   8/8/04            Vega          Antonio    73 Years
2      MP2905  6/27/97         Fawcett          Michele    35 Years
3      MP2508 12/12/71           Lande        Elizabeth    21 Years
4      MP2477  9/27/74           Jones        Katherine    29 Years
5      MP2659  3/18/65            Shea         Kathleen     6 Years
6      MP2509  6/24/75          Thorne             Edna    15 Years
          City       County State Biological.Sex
1 Philadelphia        Adams    PA           Male
2      Donegal     Somerset    PA         Female
3 Philadelphia Philadelphia    PA         Female
4   Clearfield   Clearfield    PA         Female
5       Tyrone        Blair    PA         Female
6 Philadelphia Philadelphia    PA         Female
                      Race...Ethnicity Date.Modified
1 White / Caucasian, Hispanic / Latino       8/12/20
2                    White / Caucasian        4/1/20
3                    White / Caucasian       4/27/21
4                    White / Caucasian       9/28/23
5                    White / Caucasian       4/29/24
6                    White / Caucasian       2/21/23
```

As seen from the 6 example entries displayed above, NamUS offered 11 different attribute underneath each case (where each case is an individual reported case of missing persons). These attributes contain: case number, date of last contact, legal last name, legal first name, missing age, city, county, state, biological sex, race/ethnicity, and the date modifed.

3 Ethical frameworks: FAIR and CARE

3.1 Fair Principles

- Findable:

This project clearly states the source of the data (NamUS) and correctly cites the source in order to increase findability for the viewers of this report.

- Accessible:

Our data offered by NamUS was downloaded after we have met the terms of NamUS (registering for an account). The only data we will be using are ones that are publically accessible on the NamUS website that can be visited by anyone.

- Interoperable:

The developers of this project are able to operate with the data using the csv file located in the repository. The data is then moved to a data frame stored in Rdata and cleaned into different data frames for the convience of EDA.

- Reusable:

The coding process of this project has been meaningfully commented and documented for the goal of easy comprehension for viewers and reader of this project.

3.2 Care Principles

- Collective Benefit:

The purpose of this project is for the collective benefit of the public. Designed to generate and offer visualizations that may strength public understanding of missing persons cases from the perspective of demographic, over time, and geographically.

- Authority to Control:

The data comes from proper law enforcement and other proper sources, which obtained consent and permission from related individuals of the missing persons to register the case and enter the information into the database. This analysis also acknowledges that all data were premitted by the right authorities and respects these authorities.

- Responsibility:

All responsibilities of this project falls upon the 3 developers/authors. Responsibilities include any misconduct or misrepresentation of the data and the corresponding analysis.

- Ethics:

Interpretation and presentation of results consider contextual factors like population growth and demographic disparities. Throughout the process of creating this project, developers try to avoiding deterministic or causal claims where evidence is insufficient and ensuring that findings are framed in ways that do not reinforce harmful stereotypes.

4 Data and Attributes

The analysis uses nearly all attributes of the data, with the except of case number (which is simply just used as an identifier), and date modified (where we could not find a useful interruptation related to our research question),

4.1 Original attributes (from dataset):

Geogrpahic attributes:

- state
- county
- city

Demographic attributes:

- biological sex
- race/ ethnicity
- missing age

Time-span attributes:

- DLC

4.2 Derived attributes

From the original listed attributes of the database, there were a couple of attributes that we need to derive for further analysis. In order to adequately categorize the missing cases by the range of the missing ages, we had to create a new attribute representing the age group of the missing age (categorized as a demographic attribute). The ranges we have decided upon are:

- below 20
- between 20-40
- between 40-60
- above 60

5 Data Wrangling

Since there were many factors to consider in our data analysis, the data wrangling process consisted of wrangling simple summary tables.

Here are different ways our data is wrangled:

- Each case is defined by a combination of sex, race, and age group
- Each case is organized by state, county and city
- Each case is associated with a specific date of last contact

```
# A tibble: 2 x 2
  `Biological Sex` `Number of Missing People`
  <chr>                                 <int>
1 Female                                  440
2 Male                                    519
```

```
# A tibble: 6 x 2
  `Race / Ethnicity`                          `Number of Missing People`
  <chr>                                                            <int>
1 Asian                                                               14
2 Black / African American                                           209
3 Black / African American, Asian                                      1
4 Black / African American, Hispanic / Latino                          2
5 Black / African American, White / Caucasian                          5
6 Hispanic / Latino                                                   43
```

```
# A tibble: 4 x 2
  age_group     `Number of Missing People`
  <chr>                              <int>
1 above 60                              93
2 below 20                             185
3 between 20-40                        439
4 between 40-60                        242
```

```
# A tibble: 6 x 4
  State County    City         `Number of Missing People`
  <chr> <chr>     <chr>                             <int>
1 OH    Allen     Cridersville                          1
2 OH    Allen     Lima                                  1
3 OH    Ashland   Ashland                               1
4 OH    Ashtabula Andover                               1
5 OH    Ashtabula Ashtabula                             1
6 OH    Ashtabula Cleveland                             1
```

```
# A tibble: 6 x 2
  DLC    `Number of Missing People`
  <chr>                       <int>
1 1/1/02                          1
2 1/1/03                          1
3 1/1/05                          2
4 1/1/08                          1
5 1/1/11                          1
6 1/1/17                          1
```

Complex summary tables.

```
# A tibble: 6 x 4
  `Race / Ethnicity` `Biological Sex` age_group     `Number of Missing People`
  <chr>              <chr>            <chr>                                <int>
1 Asian              Female           above 60                                 0
2 Asian              Female           below 20                                 2
3 Asian              Female           between 20-40                            4
4 Asian              Female           between 40-60                            0
5 Asian              Male             above 60                                 3
6 Asian              Male             below 20                                 0
```

```
# A tibble: 6 x 7
  State County City  `Race / Ethnicity` `Biological Sex` age_group
  <chr> <chr>  <chr> <chr>              <chr>            <chr>
1 OH    Adams  Akron Asian              Female           above 60
2 OH    Adams  Akron Asian              Female           below 20
3 OH    Adams  Akron Asian              Female           between 20-40
4 OH    Adams  Akron Asian              Female           between 40-60
5 OH    Adams  Akron Asian              Male             above 60
6 OH    Adams  Akron Asian              Male             below 20
# i 1 more variable: `Number of Missing People` <int>
```

```
# A tibble: 6 x 5
  DLC    `Race / Ethnicity` `Biological Sex` age_group    Number of Missing Pe~1
  <chr>  <chr>              <chr>            <chr>                         <int>
1 1/1/02 Hispanic / Latino  Male             between 40-~                      1
2 1/1/03 White / Caucasian  Female           between 20-~                      1
3 1/1/05 White / Caucasian  Female           between 20-~                      1
4 1/1/05 White / Caucasian  Female           between 40-~                      1
5 1/1/08 Other              Male             between 20-~                      1
6 1/1/11 White / Caucasian  Male             above 60                          1
# i abbreviated name: 1: `Number of Missing People`
```

6 Geographic Breakdown

For the spatial breakdown of our explanatory data analysis, we chose to examine missing persons cases at the county level rather than at the state or city level. Counties provide a useful middle scale for a geographic analysis as they are more spatially detailed than state-level summaries, while avoiding many of the inconsistencies and missing values in city-level reporting. Moreover, city boundaries are difficult to represent spatially, as cities are typically mapped as point locations (such as maps package used for this visualization). In contrast, counties are defined by clear polygon boundaries, allowing missing persons counts to be spatially contained and visualized more clearly. In addition, counties are commonly used units in U.S. demographic and public policy analysis, making them well-suited for spatial comparison.

To generate the county-level map, we used the maps package in R, which provides built-in polygon boundary data for U.S. states and counties; the structure of this workflow was adapted from the county mapping approach demonstrated in Eriq Ande's *Making Maps with R* tutorial, allowing missing persons counts to be joined directly to county polygons for clear spatial visualization. Using county boundaries for Pennsylvania and Ohio, we aggregated the total number of reported missing persons cases per county from 1969 to 2024. These counts were then visualized using a choropleth map, where darker shading represents counties with higher numbers of reported cases.

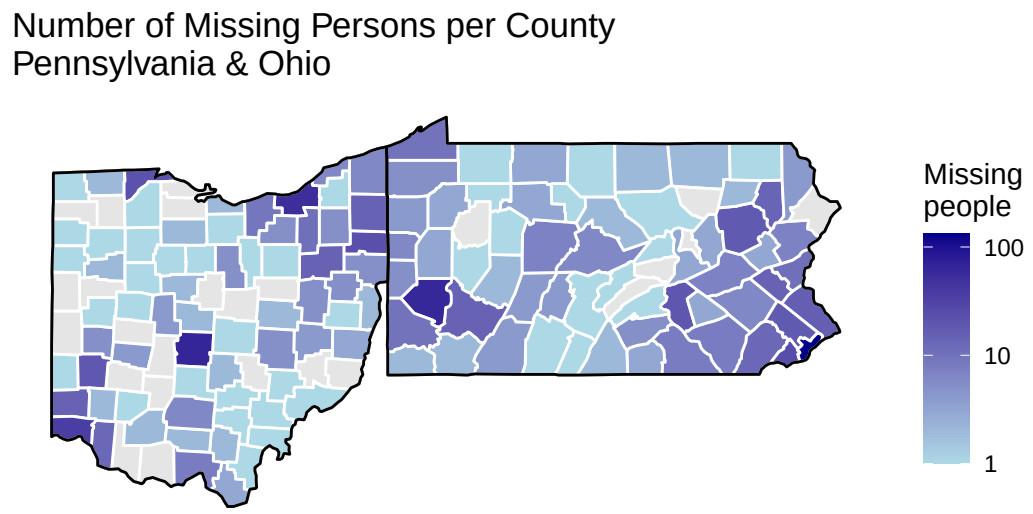
The resulting map reveals that missing persons cases are not evenly distributed across space. Instead, there is clear spatial variation, with certain counties exhibiting noticeably higher counts than others. In both Pennsylvania and Ohio, counties containing or adjacent to metropolitan areas (e.g. Philadelphia) tend to display higher numbers of reported cases. This spatial clustering suggests a form of over-representation in specific geographic contexts.

However, it is important to interpret this pattern cautiously. Higher case counts at the county level do not necessarily indicate a higher underlying risk of disappearance. Instead, these patterns may reflect:

- Larger population sizes in urban counties
- Greater reporting capacity and law enforcement infrastructure
- Higher likelihood that cases are formally documented and entered into national systems such as NamUS

Conversely, counties with very low counts may not reflect fewer incidents, but rather underreporting or differences in administrative capacity.

Overall, this county-level spatial analysis helps identify **where reported missing persons cases are concentrated**, highlighting geographic unevenness across Pennsylvania and Ohio. While the map does not support causal conclusions, it provides important spatial context that complements the demographic and time-series analyses in this report and helps frame future work, such as population-normalized rates or urban–rural comparisons.

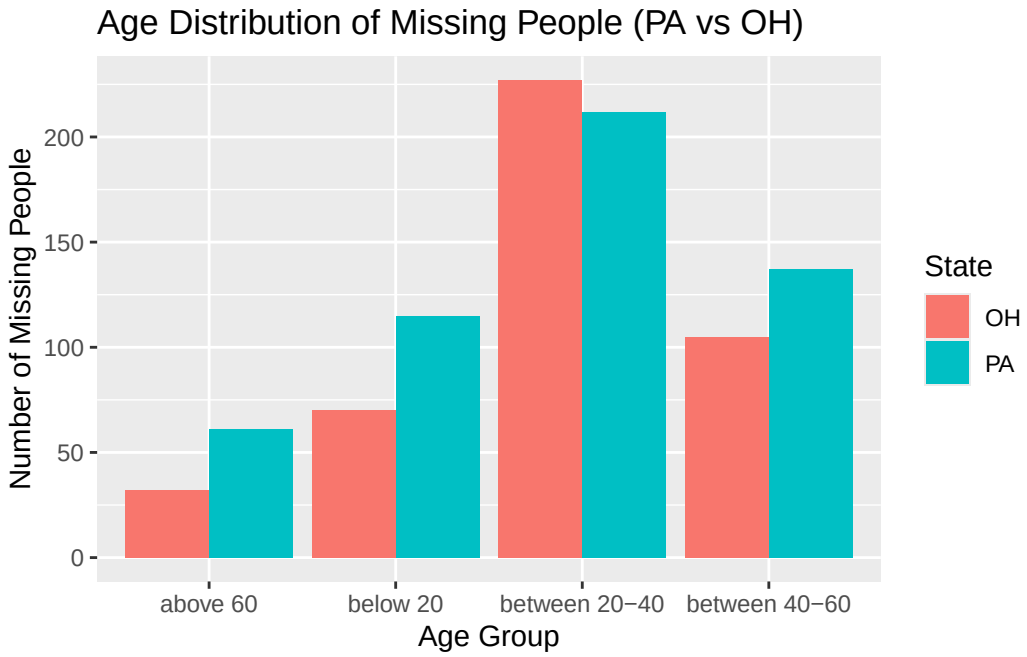


7 Demographic Breakdown

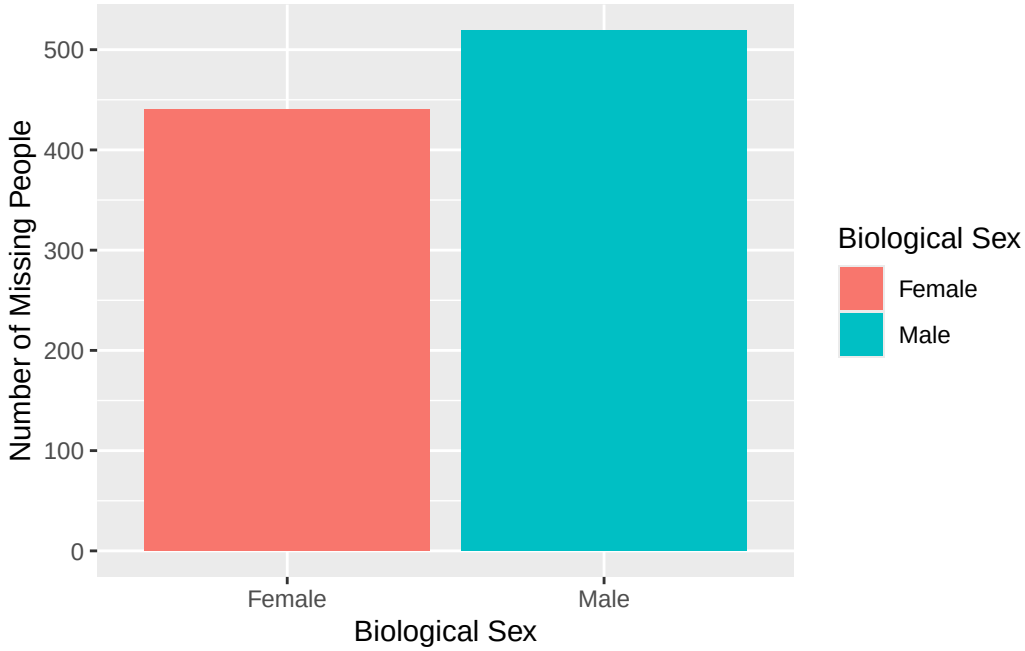
Age:

```
# A tibble: 6 x 13
  `Case Number` DLC     `Legal Last Name` `Legal First Name` `Missing Age` City
  <chr>         <chr>   <chr>             <chr>              <chr>         <chr>
1 MP2620        8/8/04  Vega              Antonio            73 Years      Phil~
2 MP2905        6/27/97 Fawcett           Michele            35 Years      Done~
3 MP2508        12/12/~ Lande             Elizabeth          21 Years      Phil~
4 MP2477        9/27/74 Jones             Katherine          29 Years      Clea~
5 MP2659        3/18/65 Shea              Kathleen           6 Years       Tyro~
6 MP2509        6/24/75 Thorne            Edna               15 Years      Phil~
# i 7 more variables: County <chr>, State <chr>, `Biological Sex` <chr>,
#   `Race / Ethnicity` <chr>, `Date Modified` <chr>, age_num <dbl>,
#   age_group <chr>
```

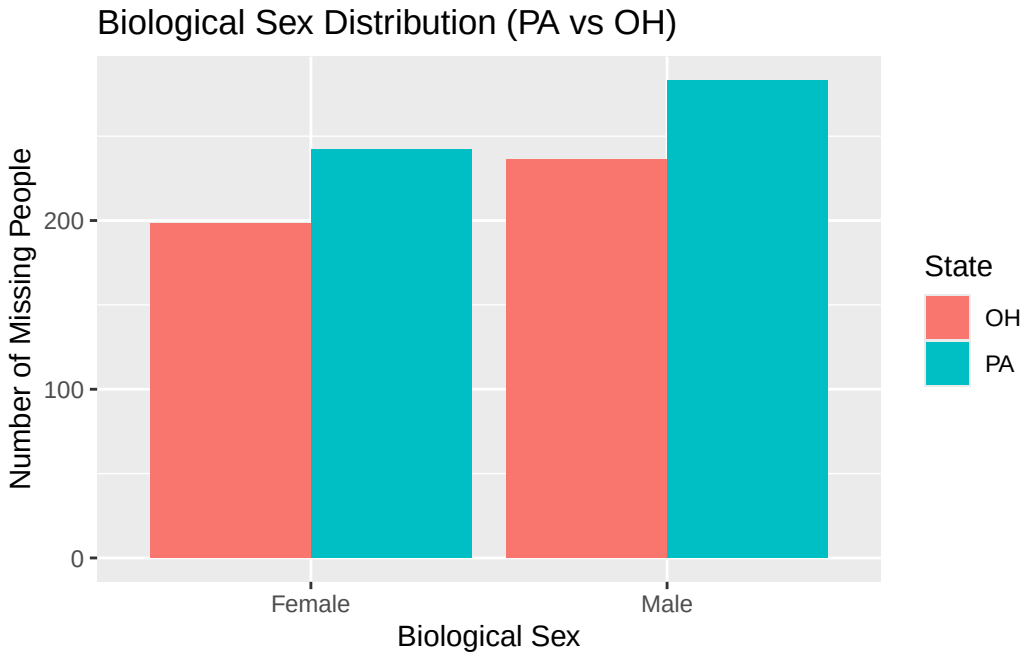
Age distribution by state



Biological Sex:

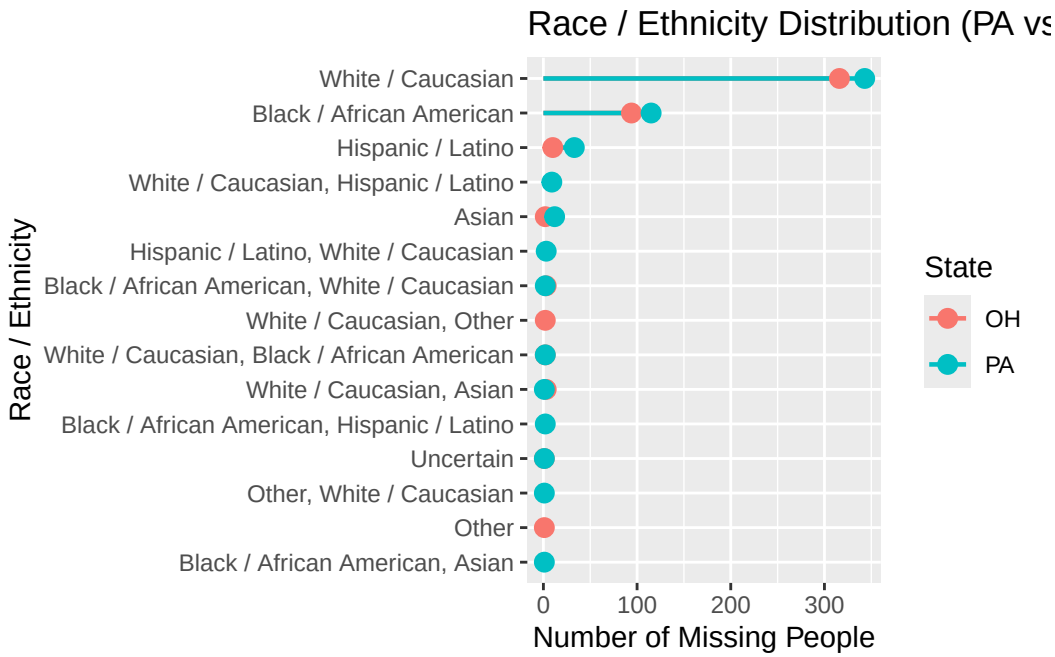


Biological sex distribution by state



Race/Ethnicity:

race distribution by state - lolipop plot



finding the single demographic group with the highest missing count.

```
# A tibble: 1 x 4
  `Race / Ethnicity` `Biological Sex` age_group     `Number of Missing People`
  <chr>              <chr>            <chr>                               <int>
1 White / Caucasian  Female           between 20-40                         160
```

8 Time-span Breakdown

For time-series analysis, we had an issue of not being able to graph the DLC as a variable on a graph due it's data type being just strings of characters. Fortunately, R has a built in Date data type that can extract date information from a given format of strings.

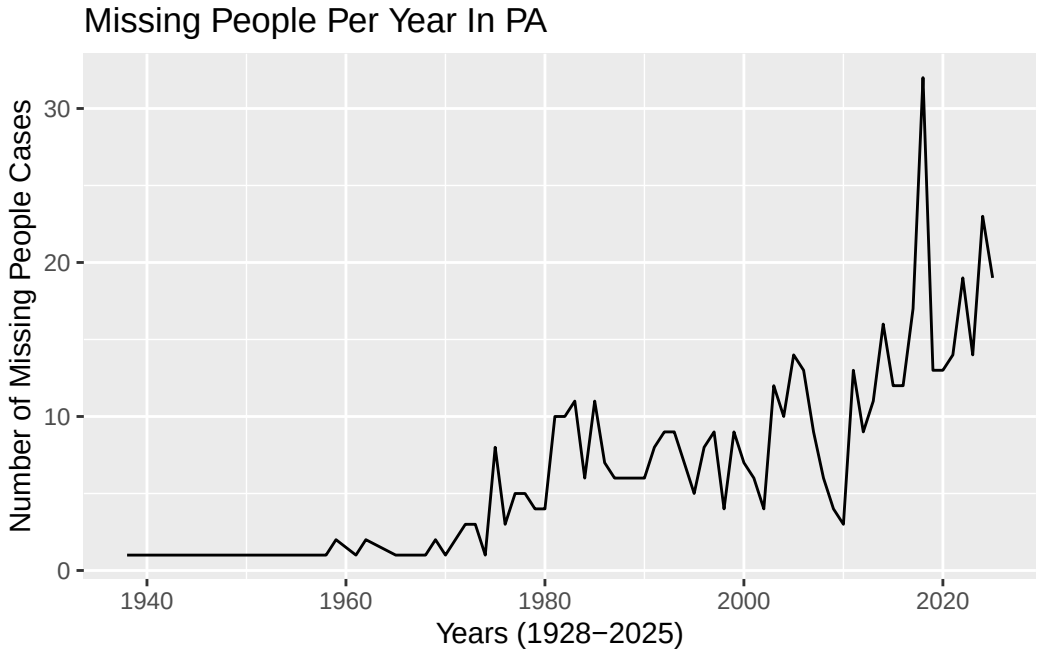
Unfortunately, there were issues that we ran into due to incomplete information. Dates were formatted like so in the original data frame: dd/mm/yy

Since the year has section has incomplete information, a year like 1960 is written as /60, which the as.Date function in R recognized as 2060. This problem is then sorted out by updating all years past the current year (2025) to its intended year.

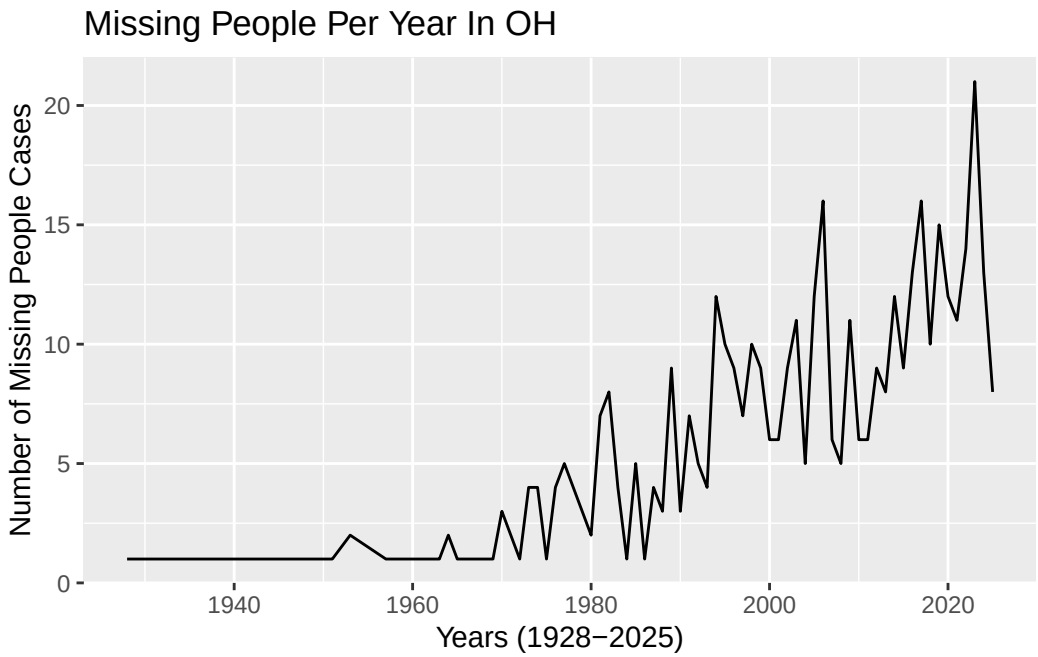
Otherwise, the final dataframe should compose of information with the year missing as a date data type and the count of missing persons in each recorded year. Therefore, each case in the dataframe time_sensitive_missing is a recorded year in the data.

```
# A tibble: 6 x 3
  year       State missing_count
  <date>     <chr>         <int>
1 1969-01-01 OH                1
2 1969-01-01 PA                2
3 1970-01-01 OH                3
4 1970-01-01 PA                1
5 1971-01-01 PA                2
6 1972-01-01 OH                1
```

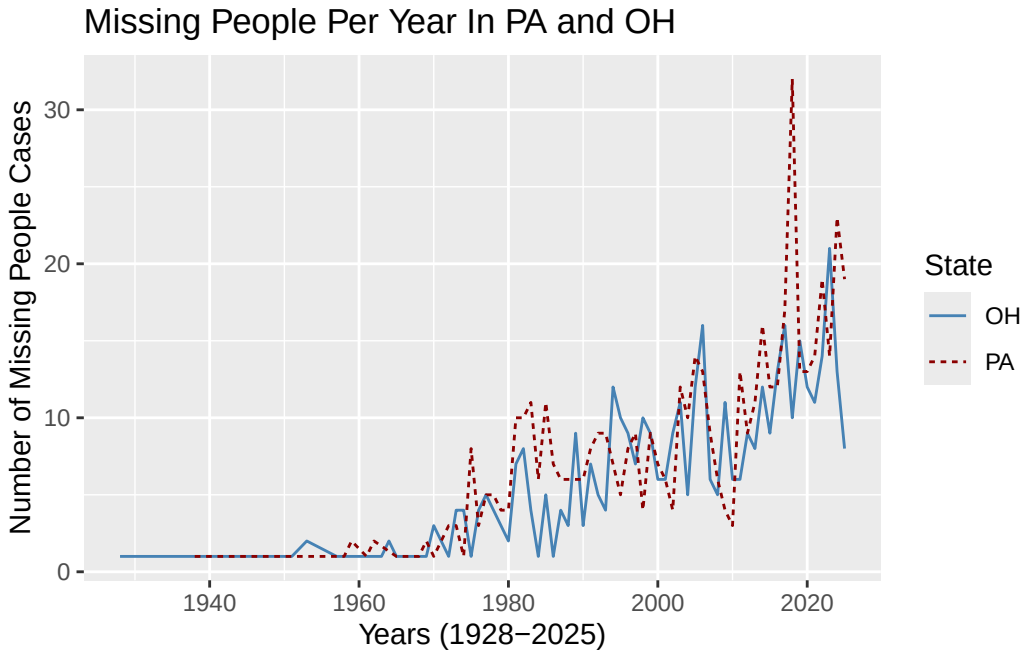
After data wrangling, a time-series graph of missing count in PA over time was created:



The process is then repeated for OH:



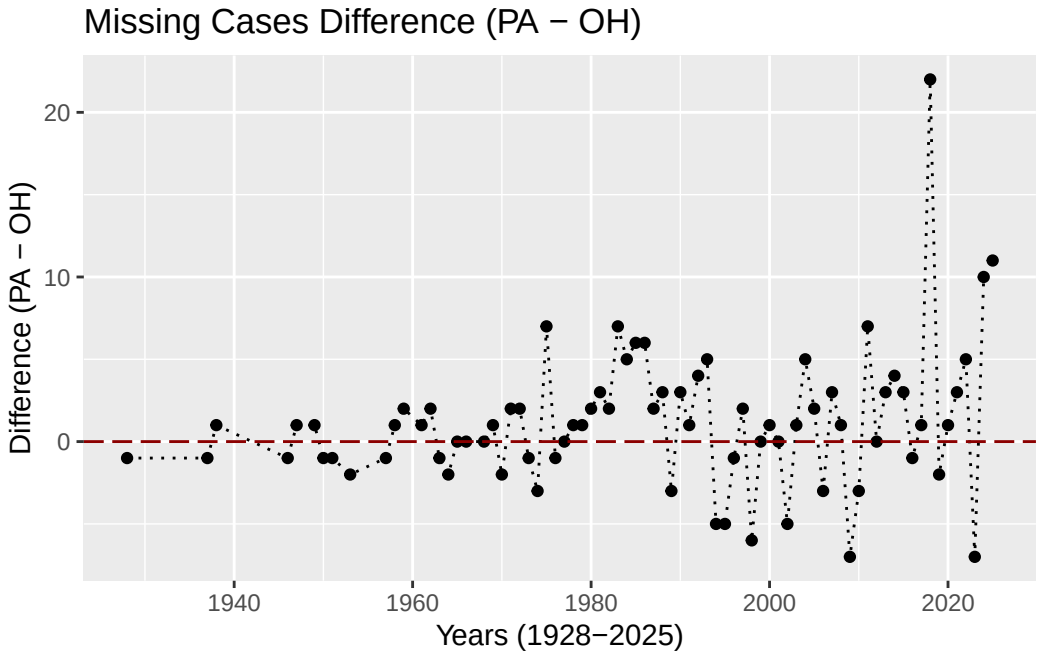
For closer inspection and comparison of both graphs, the 2 are then plotted against each other:



From @time-series-both, we can see that the trend of growth in missing persons count are very similar between the two very geographically close states. Along with the high level of variable

we can observe from the line, it is important to note that there are also little difference in the population of these states (OH: ~11.9 million, PA: ~13 million).

As a more focused indication of the similar growth and high variable of the time-series analysis, one more graph was completed showing the year-to-year difference in the missing count (PA - OH):



The average difference of PA missing count minus OH missing count is slightly positive/just above 1. Meaning that on average, PA has a slightly higher missing count during this time frame.

9 Code Appendix

```
library(tidyverse)

missingPersons <- read.csv("MissingPersons.csv")

# preview the data frame
head(missingPersons)
#library(tidyverse)

missingPersons <- read_csv("MissingPersons.csv")

#Simple summary tables

# 1) Sort by sex
gender_table <- missingPersons %>%
  count(`Biological Sex`, name = "Number of Missing People")

head(gender_table)


# 2) Sort by race/ethnicity
race_table <- missingPersons %>%
  count(`Race / Ethnicity`, name = "Number of Missing People")

head(race_table)


# 3) Sort by age group
age_table <- missingPersons %>%
  mutate(
    age_num = parse_number(`Missing Age`),
    age_num = if_else(is.na(age_num), 0, age_num),
    age_group = case_when(
      age_num < 20 ~ "below 20",
      age_num < 40 ~ "between 20-40",
      age_num < 60 ~ "between 40-60",
      TRUE         ~ "above 60"
    )
  ) %>%
  count(age_group, name = "Number of Missing People")
```

```r
head(age_table)


# 4) Sort by location
location_table <- missingPersons %>%
  count(State, County, City, name = "Number of Missing People") %>%
  arrange(State, County, City)

head(location_table)


# 5) Sort by date of last contact
date_table <- missingPersons %>%
  count(DLC, name = "Number of Missing People") %>%
  arrange(DLC)

head(date_table)
# Complex summary tables

# 1) Sort by gender, race/ethnicity and age group
missing_table_1 <- missingPersons %>%
  mutate(
    age_num = readr::parse_number(`Missing Age`),
    age_num = if_else(is.na(age_num), 0, age_num),
    age_group = case_when(
      age_num < 20 ~ "below 20",
      age_num < 40 ~ "between 20-40",
      age_num < 60 ~ "between 40-60",
      TRUE         ~ "above 60"
    )
  ) %>%
  count(
    `Race / Ethnicity`,
    `Biological Sex`,
    age_group,
    name = "Number of Missing People") %>%
  complete(
    `Race / Ethnicity`,
    `Biological Sex`,
    age_group,
    fill = list("Number of Missing People" = 0)
  )
```

```r
head(missing_table_1)


# 2) Sort by gender, race/ethnicity and age group by location, drop missing counts
missing_table_2 <- missingPersons %>%
  mutate(
    age_num = parse_number(`Missing Age`),
    age_num = if_else(is.na(age_num), 0, age_num),
    age_group = case_when(
      age_num < 20 ~ "below 20",
      age_num < 40 ~ "between 20-40",
      age_num < 60 ~ "between 40-60",
      TRUE         ~ "above 60"
    )
  ) %>%
  count(
    State,
    County,
    City,
    `Race / Ethnicity`,
    `Biological Sex`,
    age_group, name = "Number of Missing People") %>%
  complete(
    State,
    County,
    City,
    `Race / Ethnicity`,
    `Biological Sex`,
    age_group,
    fill = list("Number of Missing People" = 0)) %>%
  arrange(State, County, City) %>%
  filter("Number of Missing People" > 0)

head(missing_table_2)


# 3) Sort by gender, race/ethnicity and age group by date of last contact, drop missing count
missing_table_3 <- missingPersons %>%
  mutate(
    age_num = parse_number(`Missing Age`),
    age_num = if_else(is.na(age_num), 0, age_num),
    age_group = case_when(
```

```
      age_num < 20 ~ "below 20",
      age_num < 40 ~ "between 20-40",
      age_num < 60 ~ "between 40-60",
      TRUE         ~ "above 60"
    )
  ) %>%
  count(
    DLC,
    `Race / Ethnicity`,
    `Biological Sex`,
    age_group,
    name = "Number of Missing People"
  ) %>%
  complete(
    DLC,
    `Race / Ethnicity`,
    `Biological Sex`,
    age_group,
    fill = list(`Number of Missing People` = 0)
  ) %>%
  arrange(
    DLC,
    `Race / Ethnicity`,
    `Biological Sex`,
    age_group
  ) %>%
  filter(`Number of Missing People` > 0)

head(missing_table_3)
#Load packages
library(tidyverse)
library(maps)

#Filter by County
county_summary <- missingPersons %>%
  filter(State %in% c("PA", "OH")) %>%
  group_by(State, County) %>%
  summarise(
    NumberMissing = n(),
    .groups = "drop"
  )
```

```r
#Extract states and counties data
states   <- map_data("state")
counties <- map_data("county")

ohpa_states   <- states   %>% filter(region %in% c("pennsylvania", "ohio"))
ohpa_counties <- counties %>% filter(region %in% c("pennsylvania", "ohio"))

#Attach every point of polygon to missing persons numbers by county
county_summary_for_join <- county_summary %>%
  mutate(
    region    = recode(State,
                       "PA" = "pennsylvania",
                       "OH" = "ohio"),
    subregion = tolower(County)
  )

#Redefine data by region (state) and subregion (county) -> for working with map package
ohpa_map_df <- ohpa_counties %>%
  left_join(county_summary_for_join,
            by = c("region", "subregion"))

#Plot missing persons data by county
ditch_the_axes <- theme(
  axis.text  = element_blank(),
  axis.line  = element_blank(),
  axis.ticks = element_blank(),
  panel.border = element_blank(),
  panel.grid   = element_blank(),
  axis.title   = element_blank()
)

ohpa_base <- ggplot(ohpa_states,
                    aes(x = long, y = lat, group = group)) +
  coord_fixed(1.3) +
  geom_polygon(color = "black", fill = "gray80")

elbow_room <- ohpa_base +
  geom_polygon(
    data  = ohpa_map_df,
    aes(fill = NumberMissing),
    color = "white"
  ) +
```

```
  geom_polygon(color = "black", fill = NA) +
  theme_bw() +
  ditch_the_axes

elbow_room +
  scale_fill_gradient(
    trans = "log10",
    low = "lightblue",
    high = "darkblue",
    na.value = "grey90"
  ) +
  labs(
    title = "Number of Missing Persons per County\nPennsylvania & Ohio",
    fill  = "Missing\npeople"
  )

eda_df <- missingPersons |>
  # create numeric age + age groups
  mutate(
    age_num = readr::parse_number(`Missing Age`),
    age_num = if_else(is.na(age_num), 0, age_num),
    age_group = case_when(
      age_num < 20 ~ "below 20",
      age_num < 40 ~ "between 20-40",
      age_num < 60 ~ "between 40-60",
      TRUE         ~ "above 60"
    )
  )

head(eda_df)
### age distribution by state (PA vs OH)
age_state <- eda_df |>
  count(State, age_group, name = "Number of Missing People")

ggplot(
  data = age_state,
  aes(x = age_group,
      y = `Number of Missing People`,
      fill = State)
) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(
```

```
    title = "Age Distribution of Missing People (PA vs OH)",
    x = "Age Group",
    y = "Number of Missing People"
  )
library(tidyverse)

# Biological Sex Inference ----
## Biological Sex Frequency Bar Graph ----
ggplot(
  data = gender_table,
  mapping = aes(
    x = `Biological Sex`,
    y = `Number of Missing People`,
    fill = `Biological Sex`
  )
) +
  geom_bar(stat = "identity")
### sex distribution by state
sex_state <- eda_df |>
  count(State, `Biological Sex`, name = "Number of Missing People")

ggplot(
  data = sex_state,
  aes(x = `Biological Sex`,
      y = `Number of Missing People`,
      fill = State)
) +
  geom_bar(stat = "identity", position = "dodge") +
  labs(
    title = "Biological Sex Distribution (PA vs OH)",
    x = "Biological Sex",
    y = "Number of Missing People"
  )

### race distribution by state - lolipop plot
race_state <- eda_df |>
  count(State, `Race / Ethnicity`, name = "Number of Missing People")

ggplot(race_state,
       aes(x = reorder(`Race / Ethnicity`, `Number of Missing People`),
           y = `Number of Missing People`,
           color = State)) +
```

```
  geom_segment(aes(xend = `Race / Ethnicity`,
                   y = 0,
                   yend = `Number of Missing People`),
               linewidth = 0.8) +
  geom_point(size = 3) +
  coord_flip() +
  labs(
    title = "Race / Ethnicity Distribution (PA vs OH)",
    x = "Race / Ethnicity",
    y = "Number of Missing People"
  )

### find the single demographic group with the highest missing count
missing_table_1 %>%
  arrange(desc(`Number of Missing People`)) %>%
  slice(1)
library(tidyverse)

# Data wrangling ----
## Creating a time sensitive data frame via dates
time_sensitive_missing <- missingPersons |>
  mutate(date = as.Date(DLC, format = "%m/%d/%y")) |>
## Summarize the data frame where each case is a year and an attribute is missing count
### cut() rounds every date down to year-01-01; all dates in a year will have that label
  mutate(year = as.Date(cut(date, breaks = "year"))) |>
  group_by(year, State) |>
  summarize(missing_count = n(),
            .groups = "drop")

## Pythonian caveman function that solves problems with as.Date()
### Function takes parameter dates (column of a data frame)
futureDeleter <- function(dates) {
  strings = as.character(dates)
  for (i in seq_along(strings)) {
    d = strings[i]
    if (substr(d, 1, 2) == "20" && as.numeric(substr(d, 3, 4)) > 25) {
      new_str = paste0("19", substr(d, 3, 10))
      strings[i] = new_str
    }
  }
  return(as.Date(strings))
}
```

```
## Make sure there's no data from the future
  time_sensitive_missing$year <- futureDeleter(time_sensitive_missing$year)

# Viewing the wrangled data frame in qmd file
head(time_sensitive_missing)
# Time Series for PA ----
## Create PA only time series df
time_sensitive_missing |>
  filter( State == "PA") |>
    ggplot(
      mapping = aes(
        x = year,
        y = missing_count
      )
    ) +
    geom_line() +
    labs(
      title = "Missing People Per Year In PA",
      x = "Years (1928-2025)",
      y = "Number of Missing People Cases"
    )
# Time Series for OH ----
## Create OH only time series df
time_sensitive_missing |>
  filter(State == "OH") |>
    ggplot(
      mapping = aes(
        x = year,
        y = missing_count
      )
    ) +
    geom_line() +
    labs(
      title = "Missing People Per Year In OH",
      x = "Years (1928-2025)",
      y = "Number of Missing People Cases"
    )
# Time Series for both ----
ggplot() +
  geom_line(
    data = time_sensitive_missing,
    mapping = aes(
```

```
      x = year,
      y = missing_count,
      color = State,
      linetype = State
    )
  ) +
  scale_color_discrete(palette = c("steelblue", "darkred")) +
  labs(
    title = "Missing People Per Year In PA and OH",
    x = "Years (1928-2025)",
    y = "Number of Missing People Cases"
  )
# Create df with difference in missing persons (PA - OH)----
difference_time_series <- time_sensitive_missing |>
  pivot_wider(names_from = State, values_from = missing_count) |>
## Change all NA Values to 0
  mutate(PA = replace_na(PA, 0)) |>
  mutate(OH = replace_na(OH, 0)) |>
  mutate(difference = PA - OH)

# Time Series for difference (PA - OH) ----
ggplot(
  data = difference_time_series,
  mapping = aes(
    x = year,
    y = difference
  )
) +
  geom_point() +
  geom_line(linetype = "dotted") +
  geom_hline(yintercept = 0, color = "darkred", linetype = "longdash") +
  labs(
    title = "Missing Cases Difference (PA - OH)",
    x = "Years (1928-2025)",
    y = "Difference (PA - OH)"
  )
```