FirstQMD

Andrew Hoang

2025-11-11

Armed Forces Data Wrangling (Activities #08 and #10)

NOTE: You will need to have the csv loaded in the same directory as this QMD in order to run the code (as well as the same name for csv, check code appendix for more info)

```
# A tibble: 486 x 11
   branch       pay_grade sex    count x                ranks_by_branch_of_ser~1
   <chr>        <chr>     <chr>  <dbl> <chr>            <chr>
 1 Army         E1        Male    7429 Enlisted Members Private
 2 Army         E1        Female  1326 Enlisted Members Private
 3 Army         E1        Total   8755 Enlisted Members Private
 4 Navy         E1        Male    8903 Enlisted Members Private
 5 Navy         E1        Female  3434 Enlisted Members Private
 6 Navy         E1        Total  12337 Enlisted Members Private
 7 Marine Corps E1        Male    7849 Enlisted Members Private
 8 Marine Corps E1        Female   655 Enlisted Members Private
 9 Marine Corps E1        Total   8504 Enlisted Members Private
10 Air Force    E1        Male    8537 Enlisted Members Private
# i 476 more rows
# i abbreviated name: 1: ranks_by_branch_of_service
# i 5 more variables: ranks_by_branch_of_service_2 <chr>,
#   ranks_by_branch_of_service_3 <chr>, ranks_by_branch_of_service_4 <chr>,
#   ranks_by_branch_of_service_5 <chr>, ranks_by_branch_of_service_6 <chr>
```

```
[1] 15337944
```

Visualization for the Armed Forces

Table 1: Two-way frequency table of sex by pay grade for enlisted soldiers in the U.S. Army.

pay_grade	Female	Male
E1	1326	7429
E2	4336	22338

pay_grade	Female	Male
E3	10229	43775
E4	15143	79234
E5	10954	54803
E6	7363	49502
E7	4410	30264
E8	1472	9482
E9	394	2865

Explanation For Visualization

The table I created for the Armed Forces data looks at how sex and pay grade relate within the group of enlisted soldiers serving in the Army. What it shows most clearly is that the distribution of men and women is not even across the different ranks. We see that across all pay grades, there are noticeable more males than females within the US-Army. The table suggests a Gaussian distribution with a right skew since there are more lower pay grade enlisted soldiers.

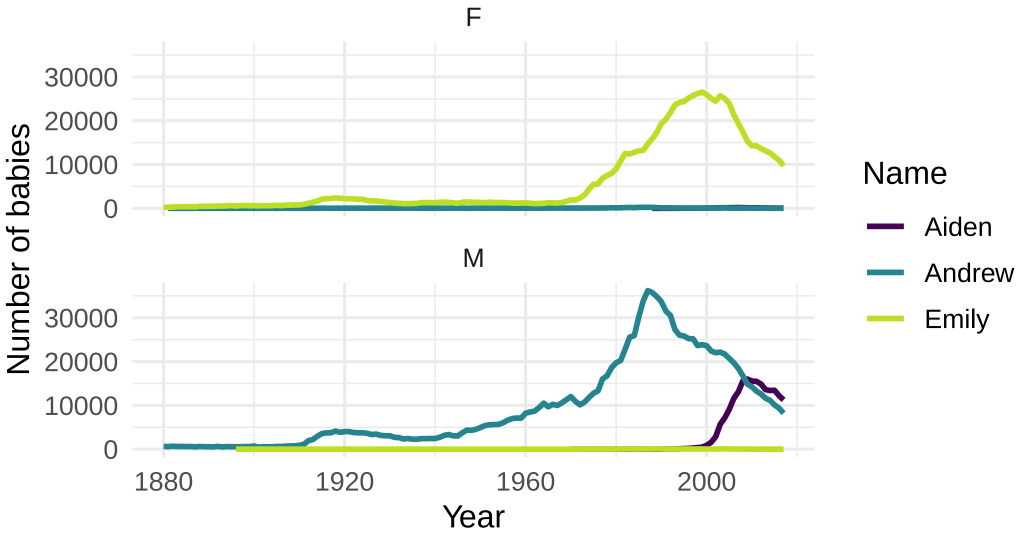
This kind of table helps make the relationship visible without relying on a graph. Each cell counts the number of enlisted men or women for a certain pay grade so you can easily scan the rows and see where the most and least concentrated levels are. For instance, the E-1 through E-4 ranks make up the bulk of enlisted personnel.

Overall, the data give a snapshot of how gender composition changes along the Army's enlisted structure. It invites further questions about what drives these differences—such as recruitment trends or occupational roles in society, but on its own it shows that sex and rank within the army is similarly distributed across both females and males.

Popularity of Baby Names (Activity #13)

Figure 1. Popularity of Selected Baby Names Over Time

Counts of babies given each name by year, separated by sex



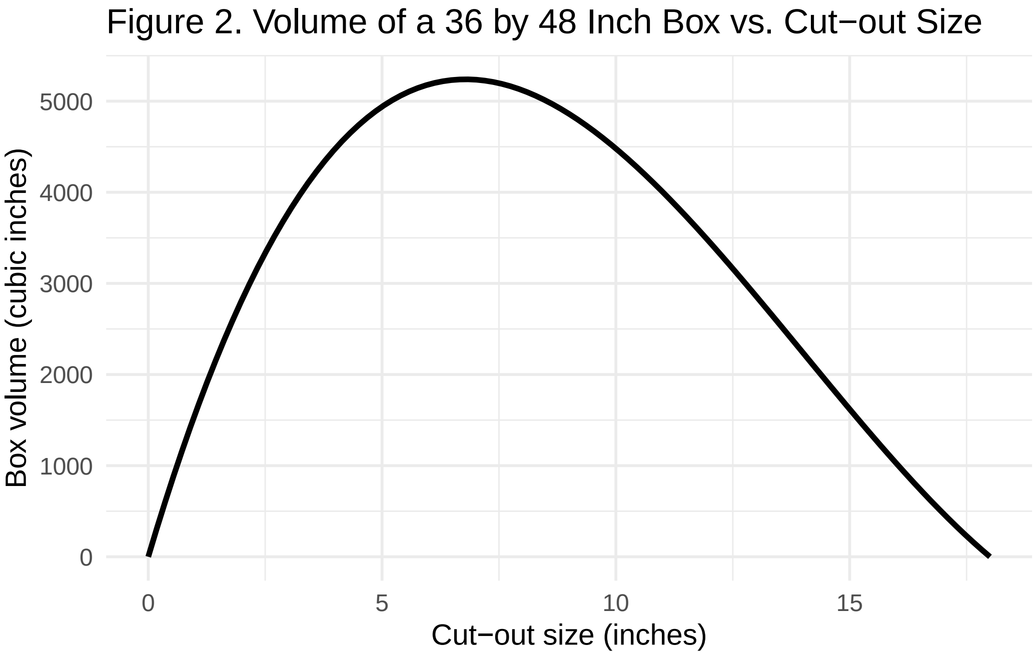
Explanation for Baby Names

For this part of the assignment I decided to look at the names Andrew, Emily, and Aiden, since they seem to belong to different “naming” eras and that makes the story that the plot tells a little more interesting to read. Andrew is a more traditional name dating back to old saints in history (as well as my name). Emily feels like a classic name too but from what I have seen from personal experience is that it really took off for girls in the late twentieth century and stayed popular for a while before tapering off. Lastly, Aiden feels like a name that exploded more recently especially in the 2000s, so I wanted to see that rise in the data to see if my hunch is correct.

In the visualization, each line tracks the number of babies given that name in a particular year, and the plot is split into two panels by sex so it is easier to compare patterns for boys and girls without everything getting mangled together. You can see Andrew mostly concentrated among boys, which matches who the name is usually used for. Emily appears almost entirely in the female panel and you can see a clear hump where it peaks in popularity (2000s) and then slowly fades. Aiden is in the male graph only and recently picked up in the more recent decades, which matches my hunch on it’s popularity. By putting these three names on the same graph, the plot gives a quick visual story of how naming preferences change over time, and it helps explain why some names feel timeless while others feel tied to particular generations.

Plotting a Mathematical Funciton (Activity #04)

Plot



Describing the Plot

The plot shows how the volume of the box changes as the size of the corner cutouts increases for a sheet that is 36 inches by 48 inches. The curve starts at zero when no squares are cut out, we see that it rises and peaks around 7 inches, and then falls back down to zero as the cutouts get too large and the box collapses. From the graph, the maximum volume happens at about a 7-inch cutout. At that point, the box reaches a volume of around 5,200 cubic inches.

The pattern makes sense since cutting out small corners doesn't create much depth, but cutting out too much makes the base too small. The middle area gives the best trade-off between height and base area. We see that the plot helps visualize this balance and shows that there is one clear point where the box is most efficient in holding volume before it quickly drops again as the cuts become too large.

What I Feel I've Learned So Far

In this course, I feel as if I have learned a lot not only involving R but also about the world in which R is used. From learning about tidying data, to learning about how to properly present your R scripts for professional settings using quarto, and even about what good data visualization is, the amount of knowledge I've gained about the peripheral world of R is immense. Furthermore, before coming in this class I had little experince involving using R, this class has also taught us about the intricacies of how to use different tools in R which is amazing.

Code Appendix

Armed Forces Wrangling Code

```r
# Armed Forces Data Transformation ----

# Install required packages if not already installed
required_packages <- c("tidyverse", "janitor", "rvest", "stringr", "knitr")

for(pkg in required_packages){
  if(!require(pkg, character.only = TRUE)){
    install.packages(pkg, dependencies = TRUE)
    library(pkg, character.only = TRUE)
  }
}

# Goal: Convert pivot table into group_df (group level) and individual_df (individual level)

library(tidyverse)
library(janitor)
library(rvest)

# Load data (skip first two header rows)
armed <- read_csv("US_Armed_Forces_(6_2025) - Sheet1.csv", skip = 2)

# Clean names
armed <- armed %>% clean_names()
# pay_grade, male, female, total, male_1, female_1, total_1, ...

# Keep only rows with a pay grade (ignore any stray header rows)
armed <- armed %>% filter(!is.na(pay_grade))

# Identify all columns except pay_grade
cols_to_pivot <- names(armed)[names(armed) != "pay_grade"]

# Map each numeric column to a branch + sex
branch_levels <- c("Army", "Navy", "Marine Corps", "Air Force", "Space Force", "Total")

col_info <- tibble(
  col = cols_to_pivot,
  branch = rep(branch_levels, each = 3),
  sex = rep(c("Male", "Female", "Total"), times = length(branch_levels))
)

# Convert comma-separated numbers to numeric
armed <- armed %>%
  mutate(across(all_of(cols_to_pivot), ~ as.numeric(gsub(",", "", as.character(.)))))

# Reshape to long format with explicit branch and sex
armed_long <- armed %>%
  pivot_longer(
    cols = all_of(cols_to_pivot),
    names_to = "col",
    values_to = "count"
  ) %>%
  left_join(col_info, by = "col") %>%
  filter(!is.na(count))

# Group-level data frame: one row per branch / pay grade / sex
group_df <- armed_long %>%
  select(branch, pay_grade, sex, count)

# Individual-level data frame: one row per individual
individual_df <- group_df %>% uncount(weights = count)

# Scrape Pay Grade and Rank data
rank_url <- "https://neilhatfield.github.io/Stat184_PayGradeRanks.html"
rank_table <- read_html(rank_url) %>%
  html_table() %>%
  .[[1]] %>%
  clean_names()

# Merge Pay Grade info
group_df <- left_join(group_df, rank_table, by = "pay_grade")
individual_df <- left_join(individual_df, rank_table, by = "pay_grade")

# Verify totals
# print(sum(group_df$count) == nrow(individual_df))

# Preview
print((group_df))
print(nrow(individual_df))
```

Armed Forces Frequency Table Code

```r
library(stringr)
library(knitr)

# Add a rank category based on pay_grade
group_df <- group_df %>%
  mutate(
    rank_category = case_when(
      str_starts(pay_grade, "E") ~ "Enlisted",
      str_starts(pay_grade, "W") ~ "Warrant Officer",
      str_starts(pay_grade, "O") ~ "Officer",
      TRUE ~ "Other"
    )
  )

# Choose subgroup: Enlisted soldiers in the Army, only Male/Female (not "Total")
army_enlisted <- group_df %>%
  filter(
    branch == "Army",
    rank_category == "Enlisted",
    sex %in% c("Male", "Female")
  )

# Two-way frequency table: rows = pay_grade, columns = sex
army_enlisted_table <- army_enlisted %>%
  count(pay_grade, sex, wt = count, name = "frequency") %>%
  tidyr::pivot_wider(
    names_from = sex,
    values_from = frequency,
    values_fill = 0
  ) %>%
  arrange(pay_grade)

# formatted table with caption, suitable for PDF
kable(
  army_enlisted_table,
  caption = "Two-way frequency table of sex by pay grade for enlisted soldiers in the U.S. Army
)
```

Baby Names Visualization Code

```
# Install required packages if not already installed
required_packages <- c("babynames", "tidyverse", "viridis")

for(pkg in required_packages){
  if(!require(pkg, character.only = TRUE)){
    install.packages(pkg, dependencies = TRUE)
    library(pkg, character.only = TRUE)
  }
}

# Load packages and data
```

```r
library(babynames)
library(tidyverse)
library(viridis)

# 1. Data wrangling for selected names ------------------------------

selectedNames <- c("Andrew", "Emily", "Aiden")

babyNamesSelected <- babynames %>%
  filter(name %in% selectedNames) %>%         # keep only chosen names
  group_by(year, name, sex) %>%              # group by year, name, and sex
  summarise(totalBirths = sum(n), .groups = "drop")

# 2. Data visualization of popularity over time ---------------------

ggplot(babyNamesSelected, aes(x = year, y = totalBirths, color = name)) +
  geom_line(linewidth = 1) +
  facet_wrap(~ sex, ncol = 1) +
  scale_color_viridis_d(option = "D", end = 0.9) +  # color-blind-friendly
  labs(
    title = "Figure 1. Popularity of Selected Baby Names Over Time",
    subtitle = "Counts of babies given each name by year, separated by sex",
    x = "Year",
    y = "Number of babies",
    color = "Name"
  ) +
  theme_minimal(base_size = 12)
```

Mathematical Function Code (Box volume)

```r
# Box volume function for a 36 by 48 inch sheet ----
getVol <- function(x, Len = 48, Wth = 36){
  # x is the cut-out size from each corner
  # Len and Wth are the original length and width of the paper (in inches)
  x * (Wth - 2 * x) * (Len - 2 * x)
}
```

Box Volume Plot Code

```r
library(ggplot2)

# For a 36 by 48 inch sheet, the cut-out size must be between 0 and 18 inches
cut_range <- data.frame(x = c(0, 18))
```

```r
ggplot(cut_range, aes(x = x)) +
  stat_function(
    fun = getVol,
    args = list(Len = 48, Wth = 36),
    linewidth = 1
  ) +
  labs(
    title = "Figure 2. Volume of a 36 by 48 Inch Box vs. Cut-out Size",
    x = "Cut-out size (inches)",
    y = "Box volume (cubic inches)"
  ) +
  theme_minimal()
```