Activity 14

First QMD File

Robert Pitsko

2025-11-16

1 Armed Forces Data

Table 1: Frequency Table: Sex by Rank in the Army (Percentages Calculated over Column)

Pay Grade	Rank	Female	Male
E1	Private	1,326 (2.4%)	7,429 (2.5%)
E2	Private	4,336 (7.8%)	22,338 (7.5%)
E3	Private First Class	10,229 (18.4%)	43,775 (14.6%)
E4	Corporal OR Specialist	15,143 (27.2%)	79,234 (26.4%)
E5	Sergeant	10,954 (19.7%)	54,803 (18.3%)
E6	Staff Sergeant	7,363 (13.2%)	49,502 (16.5%)
E7	Sergeant First Class	4,410 (7.9%)	30,264 (10.1%)
E8	First Sergeant OR Master Sergeant	1,472 (2.6%)	9,482 (3.2%)
E9	Sergeant Major OR Command Sergeant Major	394 (0.7%)	2,865 (1.0%)
Total	-	55,627 (100.0%)	299,692 (100.0%)

Figure 1

At first glance the rank distribution as seen in Figure 1 of Army Enlisted Members is similar across both genders with a majority of people in ranks Private First Class (E3), Corporal or Specialist (E4), Sergeant (E5), and Staff Sergeant (E6). Both distributions have a right skew, but Females appear to skew heavier with a higher percentage of Females in ranks First Class (E3), Corporal or Specialist (E4), Sergeant (E5) while a higher percentage of males have the rank Staff Sergeant (E6), Sergeant First Class (E7), First Sergeant or Master Sergeant (E8), and Sergeant Major or Command Sergeant Major (E9).

2 Baby Names

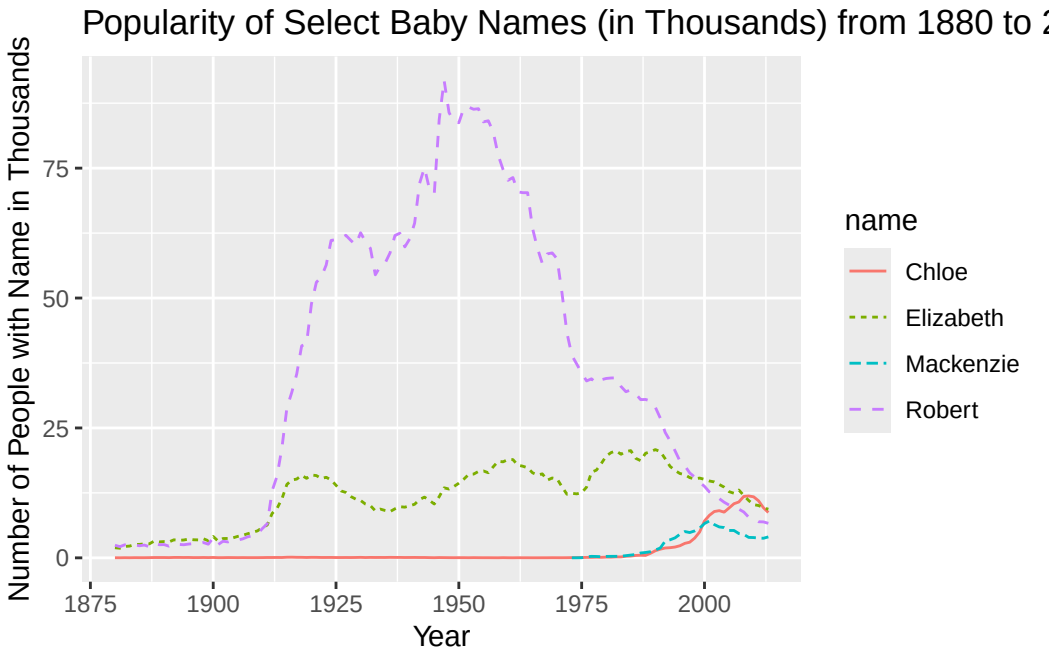

Figure 2: Baby Name Popularity

Figure 2 shows the popularity of select baby names from 1880 to 2013. These names where chosen after the author's close family members. From 1912 to 1990 Robert was the most popular name (out of the four selected) peaking around 90,000 in 1940 before dropping to under 10,000 by 2010. Elizabeth's popularity remained the most steady with counts between 8,000 and 16,000 since 1912. The popularity of Chloe and Elizabeth started later, with both names reaching counts around 6,000 for the first time in 2000.

3 Box Problem

The cut length problem is described as follows: Starting with a rectangular piece of paper, we can create an open-top box. We do this by cutting out squares from the four corners and folding up the sides. The goal is to find the side length of the cutout sections that maximize the volume of the open face box.

Figure 3 graphs volume by cut length for a 36 inch by 48 inch piece. The maximum volume for a box created from these dimensions is 5239.819 cubic inches. This is achieved with a cut length of 6.789 inches.

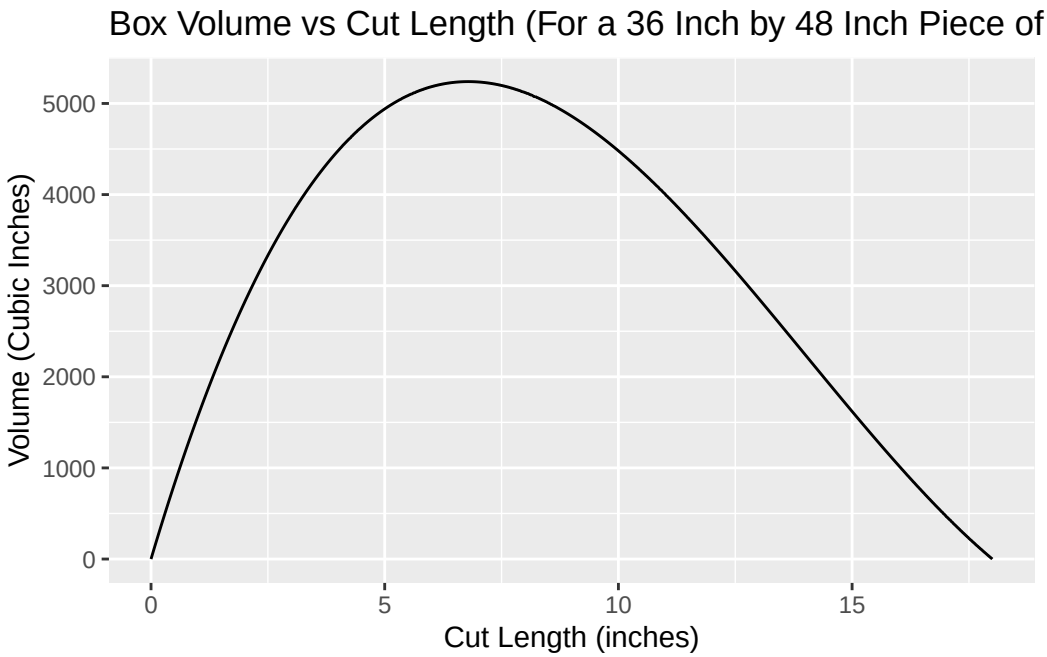

Figure 3: Box Volume by Cut Length (Max Volume = 5239.819, Optimal Cut Length = 6.789)

4 Self Reflection

Throughout this course I learned how to program in R, specifically how to tidy data (via data wrangling), and how to create effective, appealing plots and tables.

Going through the Diamond activity and Armed Forces Activity multiple times across several assignments has not only given me practice, but forced me to confront and learn from my mistakes.

The data visualization readings helped me understand what makes a graph or table effective, and the accompanying activities (10, 12, and 13) enabled me to solidify this knowledge through practice.

5 Code Appendix

5.1 Armed Forces Data

```
# Create the Armed Forces Data Frame
```

```r
library('dplyr')
library('tidyr')
library('readr')
library('tidyverse')
library('rvest')
library(knitr)
library(kableExtra)
library(janitor)
library(googlesheets4)

# access via url
gs4_deauth()
url = "https://docs.google.com/spreadsheets/d/19xQnI1cBh6Jkw7eP8YQuuicMlVDF7Gr-nXCb5qbwb_E/ed
# remove headers and read everything as a character to work with "N/A*"
data = read_sheet(url, col_names = FALSE, skip=3, col_types = "c")

group_tidy = data%>%
  # rename columns and remove trash
  rename(`Pay Grade` = "...1",
         Male_Army="...2",
         Female_Army="...3",
         trash1="...4",
         Male_Navy="...5",
         Female_Navy="...6",
         trash2="...7",
         `Male_Marine Corps`="...8",
         `Female_Marine Corps`="...9",
         trash3="...10",
         `Male_Air Force`="...11",
         `Female_Air Force`="...12",
         trash4="...13",
         `Male_Space Force`="...14",
         `Female_Space Force`="...15",
         trash5="...16",
         trash6="...17",
         trash7="...18",
         trash8="...19"
  ) %>%
  dplyr::select(-trash1, -trash2, -trash3, -trash4,
                -trash5, -trash6, -trash7, -trash8) %>%
  # remove footers and summary rows
  slice_head(n=26) %>%
```

```
  filter(`Pay Grade` != 'Total Enlisted'
         & `Pay Grade` != 'Total Warrant Officers') %>%
  # begin to tidy (rank, sex_branch, count)
  pivot_longer(
    cols = c(Male_Army, Female_Army, Male_Navy, Female_Navy,
             "Male_Marine Corps", "Female_Marine Corps", "Male_Air Force",
             "Female_Air Force", "Male_Space Force", "Female_Space Force"),
    names_to = 'sex_branch',
    values_to = 'count'
  ) %>%
  filter(count != 'N/A*') %>%
  # separate sex and branch
  separate_wider_delim(
    cols = sex_branch,
    delim = "_",
    names = c('Sex', 'Branch')
  )

# get rank data
rank_data <- read_html(
  x = "https://neilhatfield.github.io/Stat184_PayGradeRanks.html"
) %>%
  html_elements(css = "table") %>%
  html_table()

rank_df = rank_data[[1]]

# adds "Ranks" (will be header)
rank_df[1, 1] = "Rank Category"
# rename the headers to be the the first row then remove the first row
colnames(rank_df) = rank_df[1, ]
rank_df = rank_df[-1, ]

rank_tidy = rank_df%>%
  # keep pay grade and necessary branches
  select(1:7) %>%
  # removes the footer
  slice_head(n=24) %>%
  # each row (pay_grade, branch, rank)
  pivot_longer(
    cols = c("Army", "Navy", "Marine Corps", "Air Force", "Space Force"),
    names_to = 'Branch',
```

```
    values_to = 'Rank'
  )

# merging, joy by paygrade and branch
group_join_tidy = left_join(
  x = group_tidy,
  y = rank_tidy,
  by = join_by(`Pay Grade` == "Pay Grade", Branch == Branch)
) %>%
  # turns count into a numeric column
  mutate(
    count = readr::parse_number(count)
  )

# uncounts so that each case is a single solider
single_tidy = group_join_tidy %>%
  uncount(count)
```

```
# create the army sex by  rank table frequency table
army_table = single_tidy %>%
  # filter to show only rank of Enlisted Members
  filter(Branch == "Army" & `Rank Category` == "Enlisted Members") %>%
  # re-counts based on paygrade, rank, and sex
  count(`Pay Grade`, Rank, Sex) %>%
  # use sex as columns for better table dimensions
  pivot_wider(names_from = Sex, values_from = n, values_fill = 0) %>%
  adorn_percentages("col") %>% # show percentage of sex that make up each rank
  arrange(`Pay Grade`) %>% # arrange by pay grade
  adorn_totals("row") %>%
  adorn_pct_formatting(digits = 1) %>% # round percentages
  adorn_ns(position = "front") %>% # include the counts as well as percentags
  # make fancy
  kable() %>%
  kable_styling(bootstrap_options = c("striped", "condensed"),
                full_width = FALSE)

army_table
```

5.2 Baby Names

```
# creates the baby names dataframe

library("ggplot2")
library(dplyr)
library(tidyr)
library(knitr)
library(kableExtra)
library(janitor)
library(dcData)
data(BabyNames)

# will create line plot of the popularity of select names
names = c("Chloe", "Elizabeth", "Mackenzie", "Robert")
new_babynames = BabyNames %>%
  filter(name %in% names) %>% # filters for names
  group_by(name, year) %>%
  summarize(new_count = sum(count)) # combines counts ignores each gender

# divides count by 1000 for better readability
new_babynames$new_count = new_babynames$new_count / 1000
```

```
# creates line plot
ggplot(new_babynames, aes(x = year, y = new_count, color = name)) +
  geom_line(aes(linetype = name)) +
  labs(
    title = "Popularity of Select Baby Names (in Thousands) from 1880 to 2013",
    x = "Year", y = "Number of People with Name in Thousands")
```

5.3 Box Problem

```
library(ggplot2)
library(dplyr)

# function to get volume for given length, width and cut length
volume = function(length = 36, width=48, cut_length){
  return((length - 2*cut_length)*(width - 2*cut_length)*cut_length)
}
```

```r
# create the cut-length-table by getting volume at each interval
interval = seq(from = 0, to = 18, by = 0.001)
cut_length_table = data.frame(cut_length = interval,
                              vol = volume(cut_length = interval))

# (cut_length = 6.789, vol = 5239.819)
max_values = cut_length_table %>%
  filter(vol == max(cut_length_table$vol))

# plot the volume by cut-length
ggplot(data=cut_length_table, aes(x = cut_length, y = vol)) +
  geom_line() +
  labs(x = "Cut Length (inches)",
       y = "Volume (Cubic Inches)",
       title = "Box Volume vs Cut Length (For a 36 Inch by 48 Inch Piece of Paper)")
```