Activity 14- A First QMD File

Ian Wang

2025-11-07

Armed Forces Data

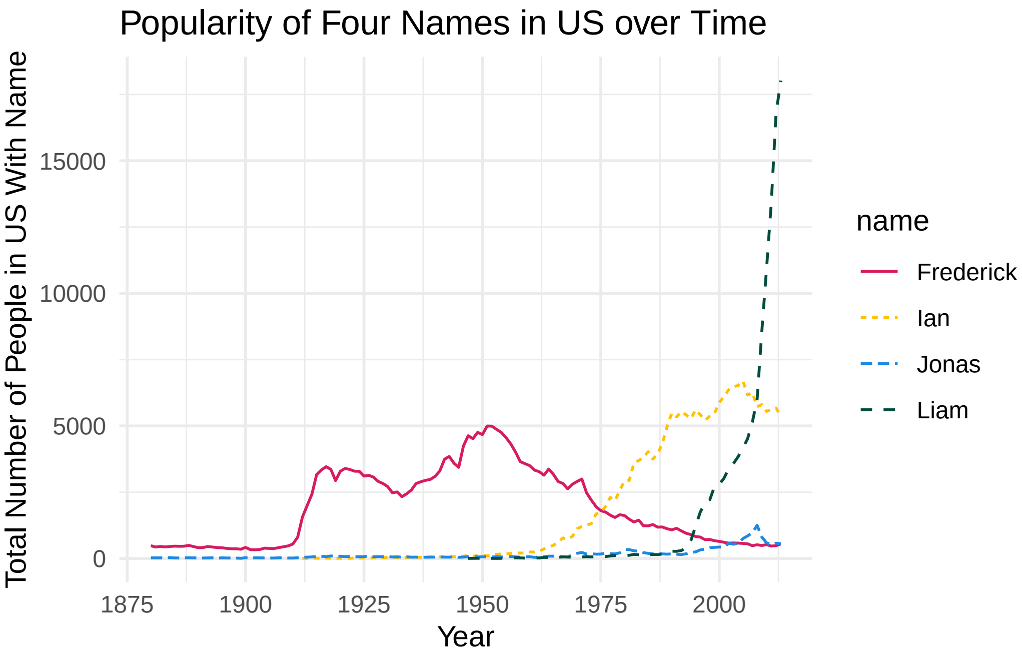
Table 1: Ranks of US Army Officers by Sex

Rank/Sex	Female	Male	Total
Brigadier General	18 (0.02%)	100 (0.13%)	118 (0.15%)
Captain	6,053 (7.88%)	20,986 (27.30%)	27,039 (35.18%)
Colonel	452 (0.59%)	3,161 (4.11%)	3,613 (4.70%)
First Lieutenant	3,006 (3.91%)	9,550 (12.42%)	12,556 (16.34%)
General	0 (0.00%)	11 (0.01%)	11 (0.01%)
Lieutenant Colonel	1,531 (1.99%)	6,939 (9.03%)	8,470 (11.02%)
Lieutenant General	5 (0.01%)	46 (0.06%)	51 (0.07%)
Major	3,044 (3.96%)	12,350 (16.07%)	15,394 (20.03%)
Major General	8 (0.01%)	80 (0.10%)	88 (0.11%)
Second Lieutenant	2,400 (3.12%)	7,122 (9.27%)	9,522 (12.39%)
Total	16,517 (21.49%)	60,345 (78.51%)	76,862 (100.00%)

Table 1 is a frequency table showing the totals and relative frequencies for male and female officers in the U.S. Army, sorted by rank. Examining Table 1 shows that there are always more males than females in every rank. However, the relative frequencies of people in each rank are similar when examining only one gender at a time. For example, the rank with the highest amount of people is Captain and the rank with the lowest is General for both genders. This indicates that rank and sex are independent of each other, as sex does not seem to have an effect on rank as shown by the relative frequencies.

Popularity of Four Baby Names Over Time

Figure 1: A time series plot showing the change in popularity for four names: Frederick, Ian, Jonas, and Liam, over a time frame from 1880 to 2013.



I chose the names Frederick and Liam because they are the names of two of my friends. I chose the name Jonas because that is my roommate's name. I chose the name Ian because it is my name. Naming kids Frederick peaked in popularity around 1915 and 1950. However, it was never that popular of a name, as even at its peak, the name hovered right under 5000 babies. Ian as a name only started becoming used around 1955, slowly and steadily increasing to around 6000 in 2005 before declining. Jonas was never a popular name, with its peak being only a few hundred children at around 2010. Liam was not a common name until it started being used around 1990, quickly increasing at a high rate to peak at over 20,000 children being named Liam in 2013.

Box Problem

Figure 2: A line plot showing the maximum box volume for every cutout side length from 0 to 20 inches on a paper measuring 36 inches x 48 inches

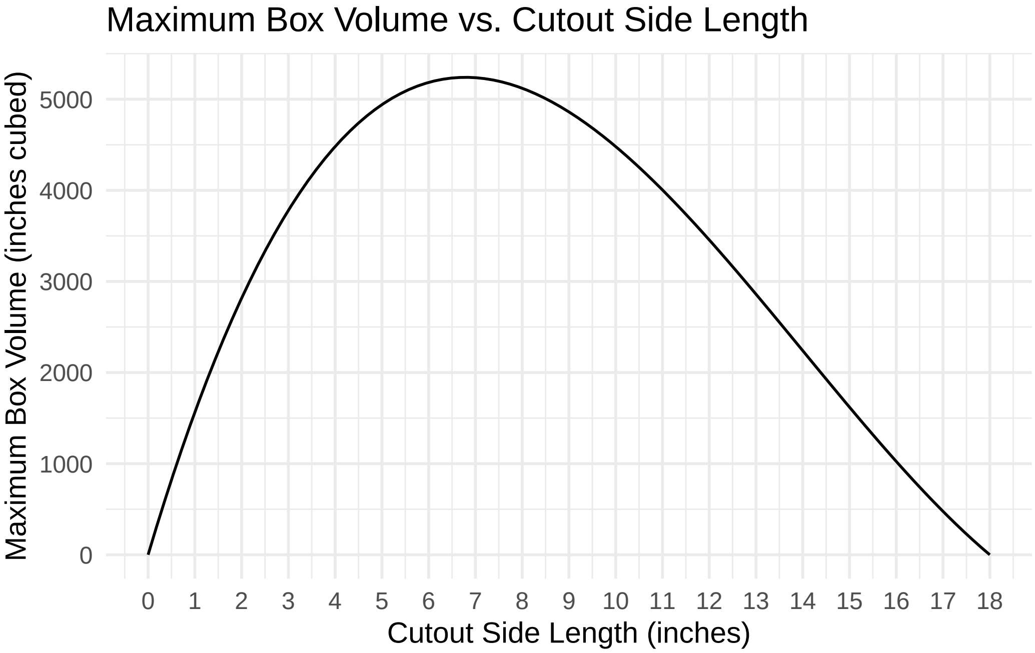


Figure 2 is a line plot showing the maximum box volume for cutout side lengths from 0 to 20 using a 36" x 48" piece of paper. The distribution begins at a cutout side length of 0 inches and ends at a cutout side length of 18 inches as those measurements result in a maximum box volume of 0 inches cubed. The maximum box volume is around 5250 inches cubed, with a cutout side length of around 6.75 inches needed to obtain it.

Self Reflection

This course has given me much more than just lessons in R. Firstly, I've made connections that will last me the rest of my life through this course. I go out to eat with my friends from this class at least every other week, and we all do work and study sessions together at least twice a week. Secondly, I have learned how to establish a professional relationship with my professor and take advantage of the resources provided to me. Ever since the assignment in PSU 16 was assigned, I've done my best to show up to office hours whether I need help or not, and I frequently use those office hours in conjunction with asking my classmates for help whenever I am stuck. Lastly, I have developed programming skills in a language that I will have to use for the rest of my time at Penn State and in the workforce. Not only has my code improved tremendously since I started (mainly in the form of adding comments and making it more readable), but my planning skills have also improved. My plans are much more understandable to me since I started this course, and I have found planning to be such a boon in my academics, even outside the context of STAT 184. In conclusion, this course has developed me as a programmer, student, friend, and more, and I am so grateful that I was able to take it this semester.

Code Appendix

```
# Data Wrangling & Creating 2 Data Visualizations: One for groups, one for individuals

# load necessary packages
# install.packages('googlesheets4')
library(googlesheets4)
library(dplyr)
library(tidyverse)

gs4_deauth() # read google sheet into R
usafRaw <- read_sheet(
  ss = 'https://docs.google.com/spreadsheets/d/19xQnI1cBh6Jkw7eP8YQuuicMlVDF7Gr-nXCb5qbwb_E/edi
)
#View(usafRaw) # check the raw data to see what needs to be changed (view, str)
#str(usafRaw)

# create case = group of soldiers df
usafCleanGroups <- usafRaw %>% # begin pipe
  rename(
    Pay_Grade = `Active-Duty Personnel by Service Branch, Sex, and Pay Grade`,
    Army_Male = `...2`,
    Army_Female = ...3, # rename columns (rename)
    junk1 = ...4,
    Navy_Male = ...5,
    Navy_Female = ...6,
    junk2 = ...7,
    MC_Male = ...8,
    MC_Female = ...9,
    junk3 = ...10,
    AF_Male = ...11,
    AF_Female = ...12,
    junk4 = ...13,
    SF_Male = ...14,
    SF_Female = ...15,
    junk5 = ...16,
    junk6 = ...17,
    junk7 = ...18,
    junk8 = ...19
  ) %>%
  select(-junk1) %>% # get rid of junk columns (select)
  select(-junk2) %>%
  select(-junk3) %>%
  select(-junk4) %>%
  select(-junk5) %>%
  select(-junk6) %>%
  select(-junk7) %>%
```

```r
  select(-junk8) %>%
  slice(-c(1, 2, 12, 18, 29, 30, 31)) %>% # get rid of junk rows (slice)
  mutate( # makes values numeric
    Army_Male = as.numeric(Army_Male),
    Army_Female = as.numeric(Army_Female),
    Navy_Male = as.numeric(Navy_Male),
    Navy_Female = as.numeric(Navy_Female),
    MC_Male = as.numeric(MC_Male),
    MC_Female = as.numeric(MC_Female),
    AF_Male = as.numeric(AF_Male),
    AF_Female = as.numeric(AF_Female),
    SF_Male = as.numeric(SF_Male),
    SF_Female = as.numeric(SF_Female)
  ) %>%
  mutate(
    SF_Male = case_match( # turns NAs into 0s
      .x = SF_Male,
      NA ~ 0,
      .default = SF_Male
    )
  ) %>%
  mutate(
    SF_Female = case_match( # turns NAs into 0s
      .x = SF_Female,
      NA ~ 0,
      .default = SF_Female
    )
  ) %>%
  pivot_longer( # pivot branch columns to rows (pivot_longer)
    cols = Army_Male:SF_Female,
    names_to = "Branch",
    values_to = "Count"
  ) %>%
  separate_wider_delim( # separate out branch and sex (separate_wider_delim)
    cols = Branch,
    delim = "_",
    names = c("Branch", "Sex")
  )
#View(usafCleanGroups) # check for errors (View)

# create data frame for case = individual
library(rvest) #load rvest
ranksRaw <- read_html("https://neilhatfield.github.io/Stat184_PayGradeRanks.html") %>%
  html_elements(css = "table") %>% # read in the ranks from github (read_html)
  html_table()
ranksRawUsable <- ranksRaw[[1]]
# View(ranksRawUsable) check the table to see what needs to be changed (View, str)
# str(ranksRawUsable)
ranksRawUsable = ranksRawUsable[,-c(1,8)] # remove junk columns
ranksRawUsable = ranksRawUsable[-c(1),] # remove junk rows
colnames(ranksRawUsable) <- c("Pay_Grade", "Army", "Navy", "MC", "AF", "SF") # rename columns
ranksClean <- ranksRawUsable %>%
  pivot_longer( # pivot branch columns to rows (pivot_longer)
    cols = Army:SF,
    names_to = "Branch",
    values_to = "Rank"
  )
# View(ranksClean) # check for errors (View)

# create case = individual soldier df
usafCleanSoldiers <- left_join( # join the ranks and group data (left_join)
  x = usafCleanGroups,
  y = ranksClean,
  by = join_by(Branch == Branch, Pay_Grade == Pay_Grade)
) %>%
  uncount(
    weights = Count
  )

#View(usafCleanSoldiers) # check for errors (View)



# Creating Frequency Tables

library(knitr) # load in necessary packages
library(kableExtra)
library(janitor)

usafCleanSoldiersArmy <- usafCleanSoldiers %>%
  filter(Branch == "Army") %>% # filter for army
  filter( # filter for ranks in the "O" pay grades
    Pay_Grade == "O1" |
    Pay_Grade == "O2" |
    Pay_Grade == "O3" |
    Pay_Grade == "O4" |
    Pay_Grade == "O5" |
    Pay_Grade == "O6" |
    Pay_Grade == "O7" |
    Pay_Grade == "O8" |
    Pay_Grade == "O9" |
    Pay_Grade == "O10"
  ) %>%
  tabyl(Rank, Sex) %>% # creates table with rank as rows and sex as columns
  adorn_totals(where = c("row", "col")) %>% # appends a totals row
  adorn_percentages(denominator = "all") %>% # adds relative frequencies
  adorn_pct_formatting(digits = 2) %>% # sets relative frequencies to two decimal places
  adorn_title( # adds title
    placement = "combined",
    row_name = "Rank",
    col_name = "Sex"
  )

formatNs <- attr(usafCleanSoldiersArmy, "core") %>%
  adorn_totals(where = c("row", "col")) %>% # appends a totals row
  mutate(
    across(where(is.numeric), format, big.mark = ",") # adds commas into numbers into table
  )

usafCleanSolArmyFreqTab <- usafCleanSoldiersArmy %>%
  adorn_ns(position = "front", ns = formatNs) # adds back numbers in front of the relative fre

usafCleanSolArmyFreqTab %>%
  kable( # formatting table
    caption = "Ranks of US Army Officers by Sex", # adds captions
    booktabs = TRUE,
    align = c("l", rep("c", 6))
  ) %>%
  kableExtra::kable_styling( # styling table
    bootstrap_options = c("striped", "condensed"), # adding stripes and condensing table
    font_size = 10 # setting font size
  )



# Creating the Baby Names Popularity Time Series

# setting up data
library(dcData) # load packages
library(dplyr)
library(tidyverse)
library(ggplot2)
#View(BabyNames)

# Wrangle Data to be useful for the visualization
babynamesConcise <- BabyNames %>%
  group_by( # groups by name and year for filtering
    name, year
  ) %>%
  summarize( # adds together all names for both sexes
    count = sum(count)
  ) %>%
  filter( # filters out wanted names
    name == "Jonas" |
    name == "Liam" |
    name == "Frederick" |
    name == "Ian"
  )

# creating data visualization
ggplot(babynamesConcise) +
  aes( # maps x y and color
    x = year,
    y = count,
    colour = name,
    linetype = name
  ) +
  geom_line( # makes line plot
  ) +
  labs( # adds title and edits y and x axes for readability
    title = "Popularity of Four Names in US over Time",
    x = xlab("Year"),
    y = ylab("Total Number of People in US With Name"),
    alt = "Time series plot showing the change in popularity for four names over time."
  ) +
  scale_colour_manual(values = c( # picks colors for the line
    "Frederick" = "#D81B60",
    "Jonas" = "#1E88E5",
    "Ian" = "#FFC107",
    "Liam" = "#004D40")
  ) +
  theme_minimal() # selects the theme



# Maximum Box Volume
library(ggplot2) # loading packages

box_volume <- function(lengthPaper = 36, widthPaper = 48, x){ # creating function
  lengthBox <- lengthPaper - (x * 2)
  widthBox <- widthPaper - (x * 2)
  heightBox <- x
  volumeBox <- lengthBox * widthBox * heightBox # volume with cutout side length subtracted
  return(volumeBox)
}

cutoutSideLength <- data.frame(x = seq(from = 0, to = 18, by = 1)) # creating data frame to be
```

```
ggplot( # mapping
  data = cutoutSideLength,
  mapping = aes(
    x = x
  )
) +
  stat_function(
    geom = "line", # choosing plot type (line plot)
    fun = box_volume, # using the previously made function
    args = list(
      lengthPaper = 36,
      widthPaper = 48
    )
  ) +
  labs(
    title = "Maximum Box Volume vs. Cutout Side Length", # relevant title
    x = xlab("Cutout Side Length (inches)"), # labeling x axis
    y = ylab("Maximum Box Volume (inches cubed)"), # labeling y axis
    alt = "Line plot showing change in maximum box volume by cutout side length." # alt text
  ) +
  theme_minimal( # adding theme
  ) +
  scale_x_continuous( # making x axis count by 1 from 0 to 18
    breaks = seq(
      from = 0,
      to = 18,
      by = 1
    )
  )
```