Problem Set 9

Shen-Han Chiu

April 5, 2024

13.1

```
library(dplyr)
library(dcData)

popular_names <-
  BabyNames %>%
  group_by(name, sex) %>%
  summarise(total_count = sum(count)) %>%
  ungroup() %>%
  arrange(desc(total_count)) %>%
  group_by(sex) %>%
  slice_max(order_by = total_count, n = 5) %>%
  ungroup()

print(popular_names)
```

```
## # A tibble: 10 x 3
##    name      sex   total_count
##    <chr>     <chr>       <int>
##  1 Mary      F         4112464
##  2 Elizabeth F         1591439
##  3 Patricia  F         1570135
##  4 Jennifer  F         1461186
##  5 Linda     F         1450328
##  6 James     M         5091189
##  7 John      M         5073958
##  8 Robert    M         4789776
##  9 Michael   M         4293460
## 10 William   M         4038447
```

For Females: Mary has the most popular name with a total count of 4,112,464. Elizabeth has 1,591,439, Patricia has 1,570,135, Jennifer has 1,461,186, and Linda has 1,450,328. For Males: James has a total count of 5,091,189. Then John has 5,073,958, Robert has 4,789,776, Michael has 4,293,460, and William has 4,038,447.

13.2

1.

```
library(dplyr)
library(tidyr)
```

```r
annual_name_counts <-
  BabyNames %>%
  group_by(year, name) %>%
  summarise(count = sum(count), .groups = 'drop')

annual_name_ranks <-
  annual_name_counts %>%
  group_by(year) %>%
  mutate(rank = rank(-count)) %>%
  ungroup()

annual_name_ranks <-
  annual_name_ranks %>%
  mutate(ranking = if_else(rank <= 100, "Top_100", "Below"))

totals_by_year_and_rank <-
  annual_name_ranks %>%
  group_by(year, ranking) %>%
  summarise(total = sum(count), .groups = 'drop') %>%
  arrange(year, desc(ranking))

print(totals_by_year_and_rank)
```

```
## # A tibble: 268 x 3
##     year ranking  total
##    <int> <chr>    <int>
##  1  1880 Top_100 132817
##  2  1880 Below    68667
##  3  1881 Top_100 126360
##  4  1881 Below    66340
##  5  1882 Top_100 144279
##  6  1882 Below    77258
##  7  1883 Top_100 141032
##  8  1883 Below    75920
##  9  1884 Top_100 155498
## 10  1884 Below    87970
## # i 258 more rows
```

2.

```r
wider_data <-
  totals_by_year_and_rank %>%
  pivot_wider(names_from = ranking, values_from = total)

wider_data <-
  wider_data %>%
  mutate(Fraction_Top_100 = Top_100 / (Top_100 + Below))

print(wider_data)
```

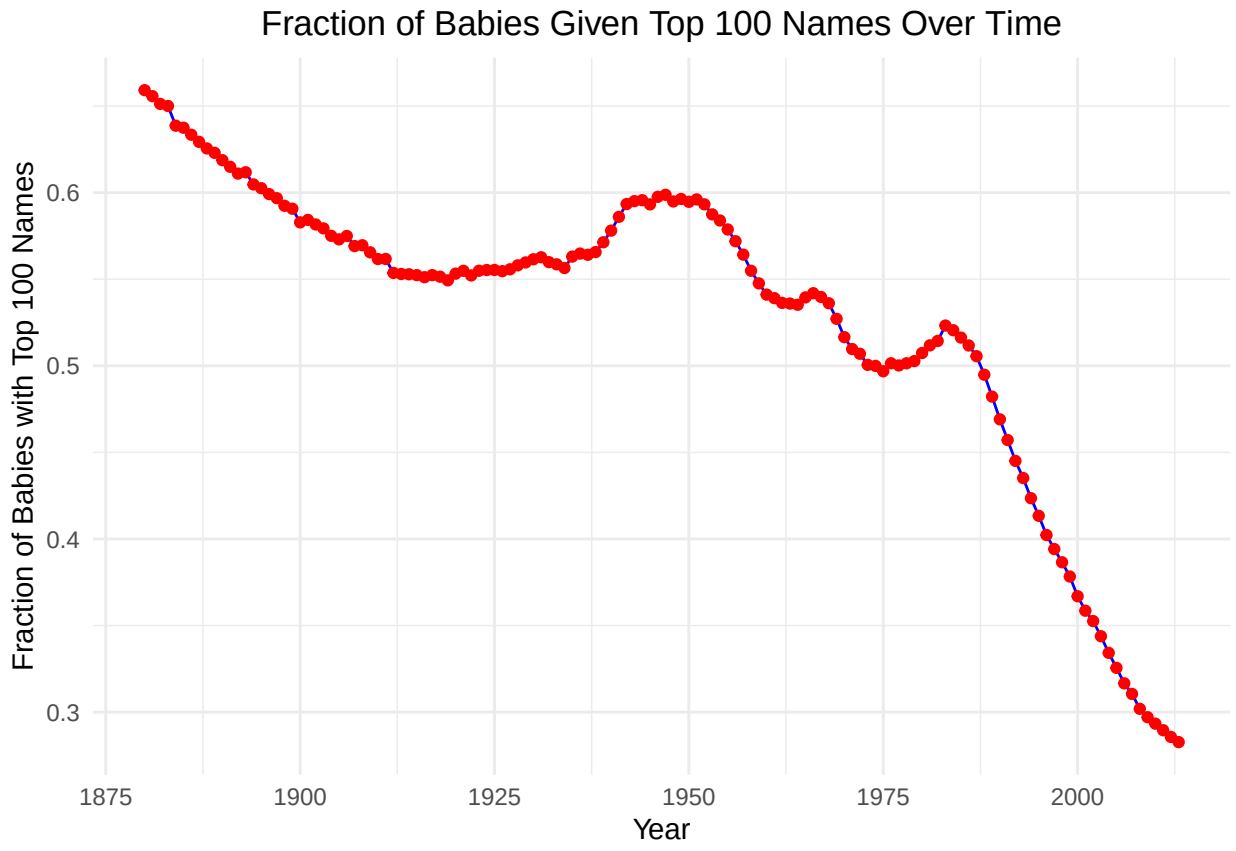
```
## # A tibble: 134 x 4
##     year Top_100  Below Fraction_Top_100
##    <int>   <int>  <int>            <dbl>
##  1  1880  132817  68667            0.659
##  2  1881  126360  66340            0.656
##  3  1882  144279  77258            0.651
##  4  1883  141032  75920            0.650
```

```
##  5  1884  155498  87970           0.639
##  6  1885  153544  87312           0.637
##  7  1886  161733  93587           0.633
##  8  1887  155699  91697           0.629
##  9  1888  187341 112140           0.626
## 10  1889  180016 108936           0.623
## # i 124 more rows
```

3.

```
library(ggplot2)

ggplot(wider_data, aes(x = year, y = Fraction_Top_100)) +
  geom_line(color = "blue") +
  geom_point(color = "red") +
  labs(title = "Fraction of Babies Given Top 100 Names Over Time",
       x = "Year",
       y = "Fraction of Babies with Top 100 Names") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```




```
# This template file is based off of a template created by Alex Hayes
# https://github.com/alexpghayes/rmarkdown_homework_template

# Setting Document Options
knitr::opts_chunk$set(
  echo = TRUE,
  warning = FALSE,
  message = FALSE,
```

```
  fig.align = "center"
)
library(dplyr)
library(dcData)

popular_names <-
  BabyNames %>%
  group_by(name, sex) %>%
  summarise(total_count = sum(count)) %>%
  ungroup() %>%
  arrange(desc(total_count)) %>%
  group_by(sex) %>%
  slice_max(order_by = total_count, n = 5) %>%
  ungroup()

print(popular_names)

library(dplyr)
library(tidyr)

annual_name_counts <-
  BabyNames %>%
  group_by(year, name) %>%
  summarise(count = sum(count), .groups = 'drop')

annual_name_ranks <-
  annual_name_counts %>%
  group_by(year) %>%
  mutate(rank = rank(-count)) %>%
  ungroup()

annual_name_ranks <-
  annual_name_ranks %>%
  mutate(ranking = if_else(rank <= 100, "Top_100", "Below"))

totals_by_year_and_rank <-
  annual_name_ranks %>%
  group_by(year, ranking) %>%
  summarise(total = sum(count), .groups = 'drop') %>%
  arrange(year, desc(ranking))

print(totals_by_year_and_rank)


wider_data <-
  totals_by_year_and_rank %>%
  pivot_wider(names_from = ranking, values_from = total)

wider_data <-
  wider_data %>%
  mutate(Fraction_Top_100 = Top_100 / (Top_100 + Below))

print(wider_data)

library(ggplot2)

ggplot(wider_data, aes(x = year, y = Fraction_Top_100)) +
  geom_line(color = "blue") +
  geom_point(color = "red") +
```

```
  labs(title = "Fraction of Babies Given Top 100 Names Over Time",
       x = "Year",
       y = "Fraction of Babies with Top 100 Names") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```