Capital One Data Science Challenge - Baby Names

Tri Nguyen

```r
## Read-in data from zip files. Add header. For faster
## compiling time, the code to download and unzip files is not
## executed.
temp <- tempfile()
download.file("http://www.ssa.gov/oact/babynames/state/namesbystate.zip",
    temp)
list.files = unzip(temp, junkpaths = TRUE, exdir = tempdir())
list.files = list.files[which(grepl(".TXT", list.files))]
unlink(temp)
df = data.frame(do.call(rbind, lapply(list.files, function(x) fread(x,
    header = FALSE, sep = ","))))
df = setNames(df, c("state", "gender", "year", "name", "count"))
saveRDS(df, "baby.dat.RDS")
```

A) Descriptive Analysis

Q1. Please describe the format of the data files. Can you identify any limitations or distortions of the data?

The baby data consists of 51 text comma delimited files and 1 read-me file in PDF. The files are compressed into 1 zip file.

There are 5 variables in the data: 2-digit state code, sex, year of birth, name, number of occurrences. Only names with at least 5 occurrences are included for each year.

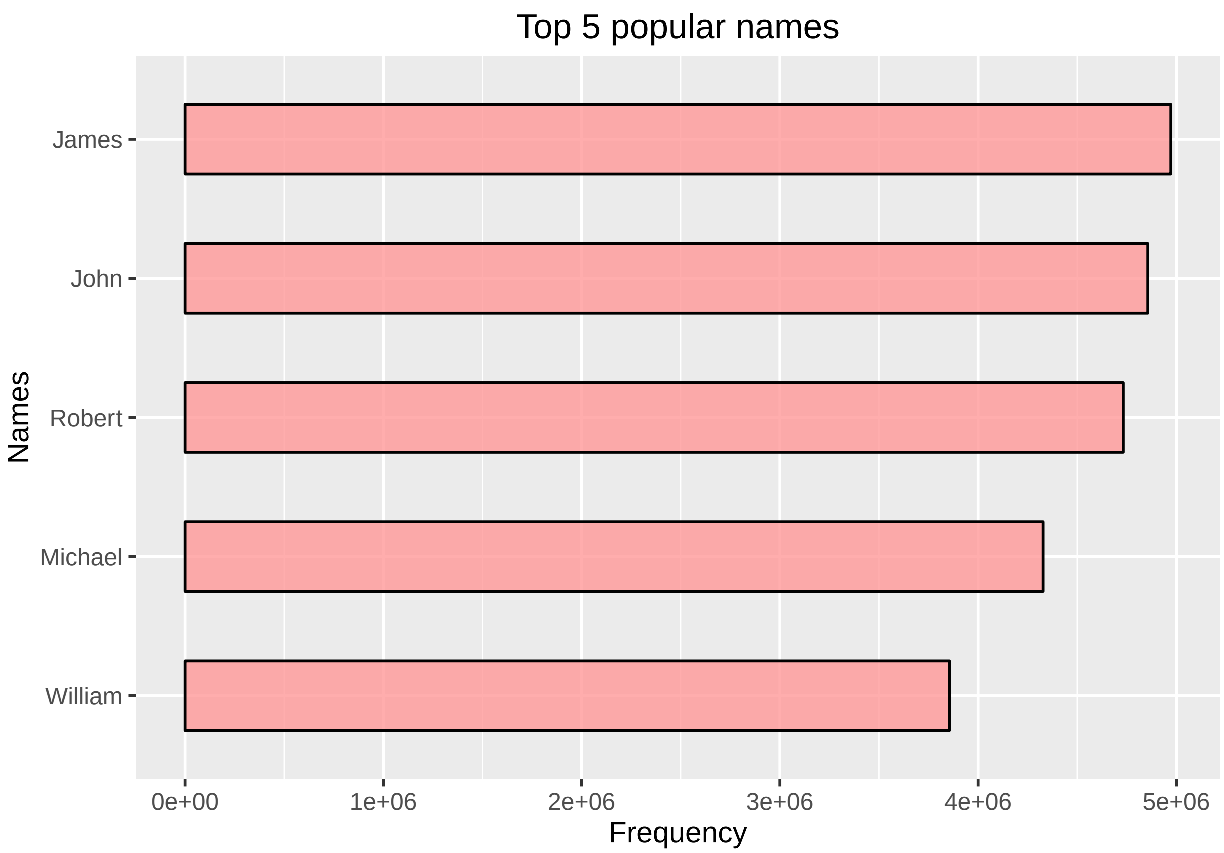
Some limitations of the data:

- The data includes records from 1910. In 1910, the U.S only had 46 states. Thus, we might not be able to perform time series or longitudinal analysis between all states on the full dataset.
- There is no information related to the state population at each year such as state total population or proportion of occurrences on state population. This might create some biases when studies between low and high population states are performed.

All the answers below are based on package "dplyr" and "magrittr" in order to achieve coincise and elegant solutions.

Q2. What is the most popular name of all time? (Of either gender.)

We can simply define that a name is "most popular" if it has the largest raw counts in the data set. By that definition, James is the most popular name of all time with 4,972,245 occurrences since 1910.



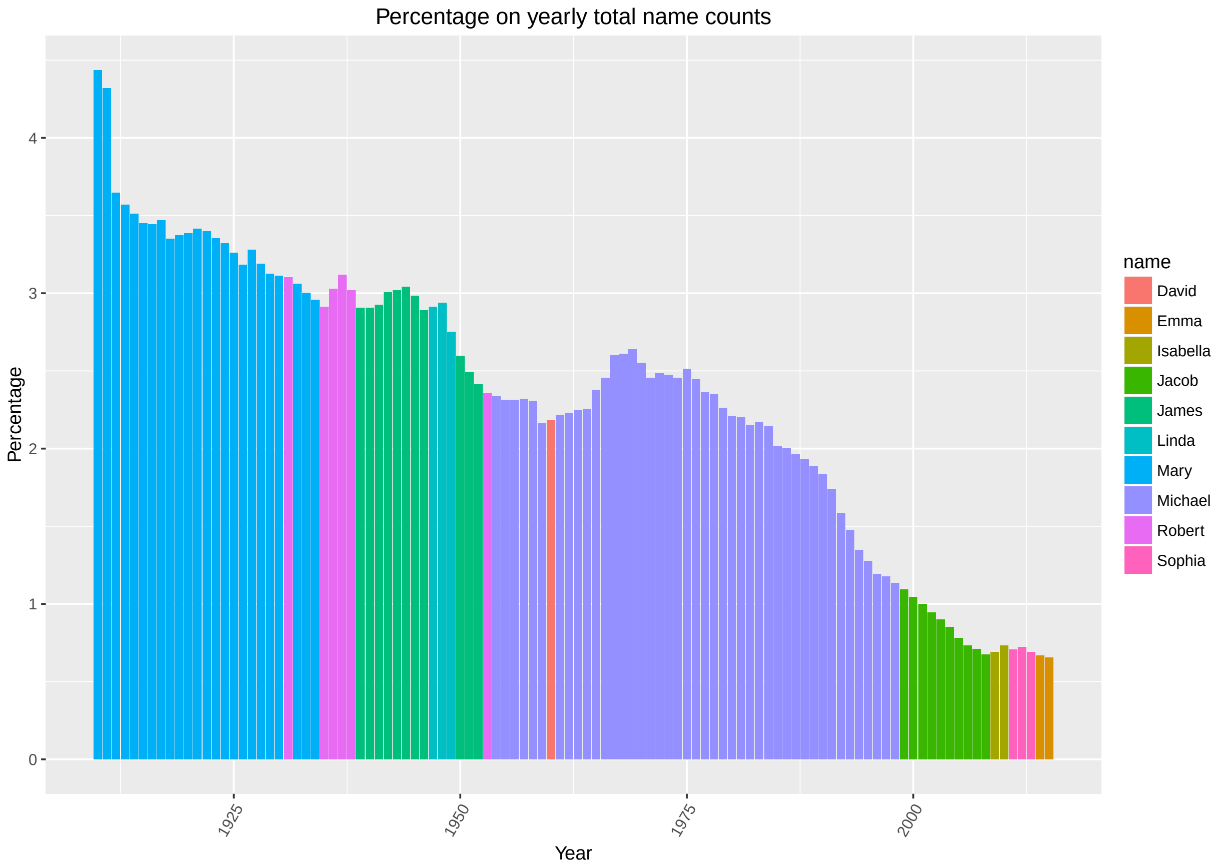
We can also use the graph below to determine most popular name by other criteria.

- Instead of defining most popular name as a name with largest raw counts, we can think that a name is most popular if it had the highest yearly proportion of all names since 1910. By this criteria, Mary is the best candidate because it consisted of more than 4% of total name counts in 1910 (highest percentage since 1910).
- A name can also be called most popular if it had the highest yearly proportion of all name for highest number of years since 1910. By this criteria, Michael is the most popular name, with Mary comes in second.

We see that the name James, which came first if we base our selection on raw counts, were most popular from late 1930s to early 1950s.

Another interesting fact is the portion of most popular name each year against the total count has been decreasing. In 1910, Mary was the most popular name with 4.4% of total count, while in 2015, Emma was most popular name with 0.65% of total count. This might be explained due to immigrants bringing more unique names over time, or people nowadays tend to be more open and creative when giving name to their children while people in the past tended to stick with common/traditional names.

```
popular.alt = df %>% group_by(name, year) %>% summarize(tot.count = sum(count)) %>%
    group_by(year) %>% summarize(ratio = 100 * max(tot.count)/sum(tot.count),
    name = name[which.max(tot.count)])
ggplot(popular.alt, aes(x = year, y = ratio, fill = name)) +
    geom_bar(stat = "identity") + theme(axis.text.x = element_text(angle = 60,
    hjust = 1)) + labs(x = "Year", y = "Percentage", title = "Percentage on yearly total name counts")
```



Q3. What is the most gender ambiguous name in 2013? 1945?

A name is considered gender ambiguous if it is used approximately equally for male and female.

If we have to pick one for 2013, Milan seems to be the most reasonable choice as the most gender ambiguous name. But, it might be better to look at the detail analysis.

A simple approach is to look at names that have ratio closest to 1 between male and female, ignoring the name's total counts. However, this method might run into some criticism:

1) If the total count is low, a name will more likely have a ratio between Male vs. Female close to 1 (or equal to 1). For example, a random name ABC can have 1 male count and 1 female count, which gives a ratio of 1 even if ABC is not a gender ambiguous name.

2) If the total count is low, people might not hear those names before and have no idea if it is a male or female name; thus, those names should not be categorized as gender ambiguous.

Year 2013:

A more conservative approach is to show names with ratio close to 1 and arranged by their total counts. From the graph below, the names above the red dashed line seem to be gender ambiguous. Note that Charlie and Dakota have lower ratio than other names, but their total counts are much higher. Also note that there are 5 names with ratio equal to 1 (Arlin, Cree, Devine, Nikita, Sonam), but the graph only shows 2 due to overlapping. Overall, Milan seems to have reasonable total counts while still maintains high ratio (94.9%).

```
ambiguous.2013 = df %>% filter(year == 2013) %>% group_by(name,
    gender) %>% summarize(gender.count = sum(count)) %>% filter(n() >
    1) %>% summarize(ratio = min(gender.count[gender == "F"]/gender.count[gender ==
    "M"], gender.count[gender == "M"]/gender.count[gender ==
    "F"]), tot.count = sum(gender.count)) %>% filter(ratio >
    0.5 & tot.count < 4000)

top.ratio.2013 = ambiguous.2013 %>% arrange(desc(ratio)) %>%
    mutate(ratio = round(ratio, 2))
top.count.ratio.2013 = ambiguous.2013 %>% arrange(desc(tot.count,
    ratio)) %>% mutate(ratio = round(ratio, 2))

## Print tables
kable(top.ratio.2013[1:5, ], caption = "Top ratio 2013")
```

Table 1: Top ratio 2013

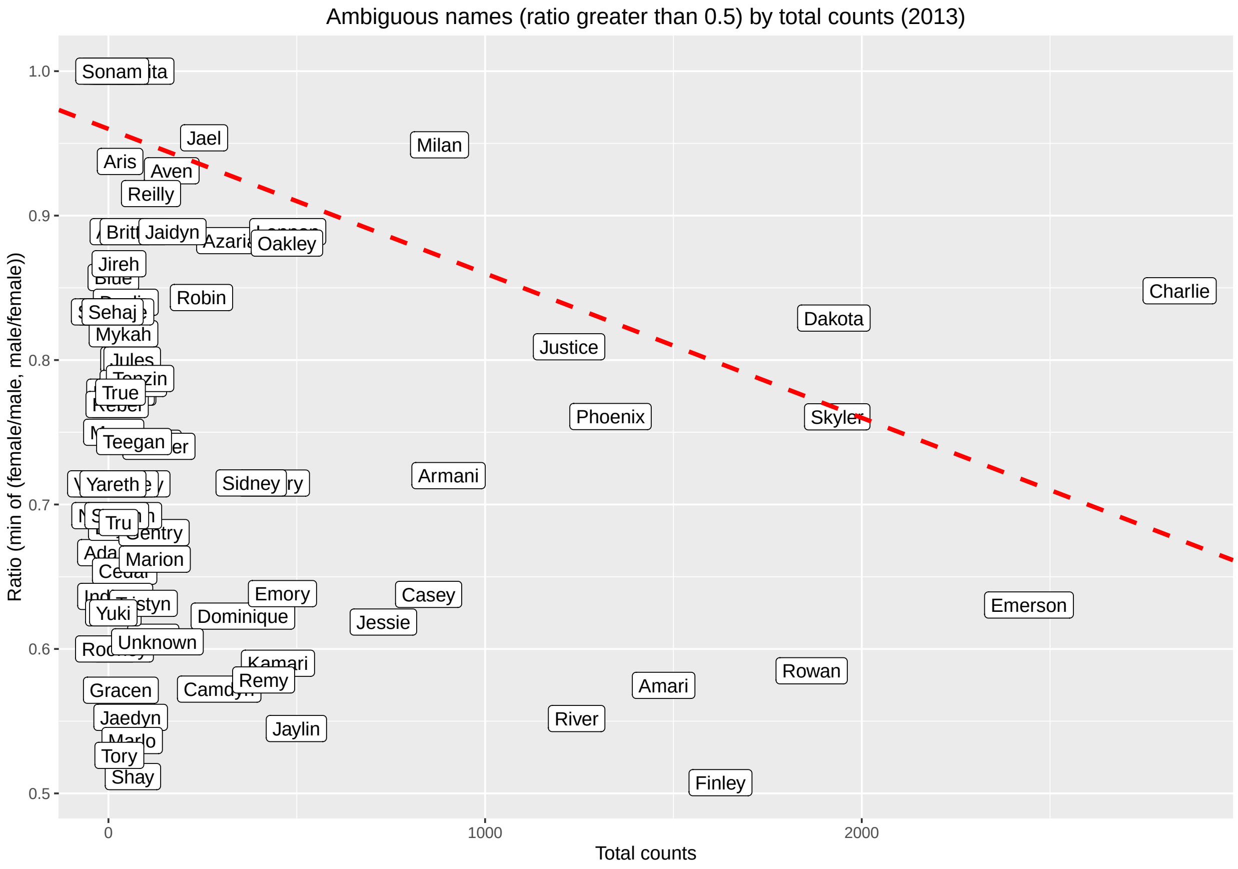
name	ratio	tot.count
Arlin	1	10
Cree	1	22
Devine	1	20
Nikita	1	94
Sonam	1	10

```
kable(top.count.ratio.2013[1:5, ], caption = "Top count and ratio 2013")
```

Table 2: Top count and ratio 2013

name	ratio	tot.count
Charlie	0.85	2844
Emerson	0.63	2444
Skyler	0.76	1935
Dakota	0.83	1926
Rowan	0.58	1867

```
## Plotting
ggplot(ambiguous.2013, aes(x = tot.count, y = ratio, label = name)) +
    geom_label() + geom_abline(intercept = 0.96, slope = -1e-04,
    colour = "red", linetype = "dashed", size = 1.2) + labs(x = "Total counts",
    y = "Ratio (min of (female/male, male/female))", title = "Ambiguous names (ratio greater than 0.5)
    cex = 1.5)
```



Year 1945:

Using the same approach, we come up with the tables and the graph below. The most potential gender ambiguous names are Maxie, Artie, Lavern, Frankie, Jessie, Jackie, Leslie. In all of them, Leslie might be a reasonable choice as the most gender ambiguous name.

```
ambiguous.1945 = df %>% filter(year == 1945) %>% group_by(name,
    gender) %>% summarize(gender.count = sum(count)) %>% filter(n() >
    1) %>% summarize(ratio = min(gender.count[gender == "F"]/gender.count[gender ==
    "M"], gender.count[gender == "M"]/gender.count[gender ==
    "F"]), tot.count = sum(gender.count)) %>% filter(ratio >
    0.5)

top.ratio.1945 = ambiguous.1945 %>% arrange(desc(ratio)) %>%
    mutate(ratio = round(ratio, 2))

top.count.ratio.1945 = ambiguous.1945 %>% arrange(desc(tot.count,
    ratio)) %>% mutate(ratio = round(ratio, 2))

## Print tables
kable(top.ratio.1945[1:5, ], caption = "Top ratio 1945")
```

Table 3: Top ratio 1945

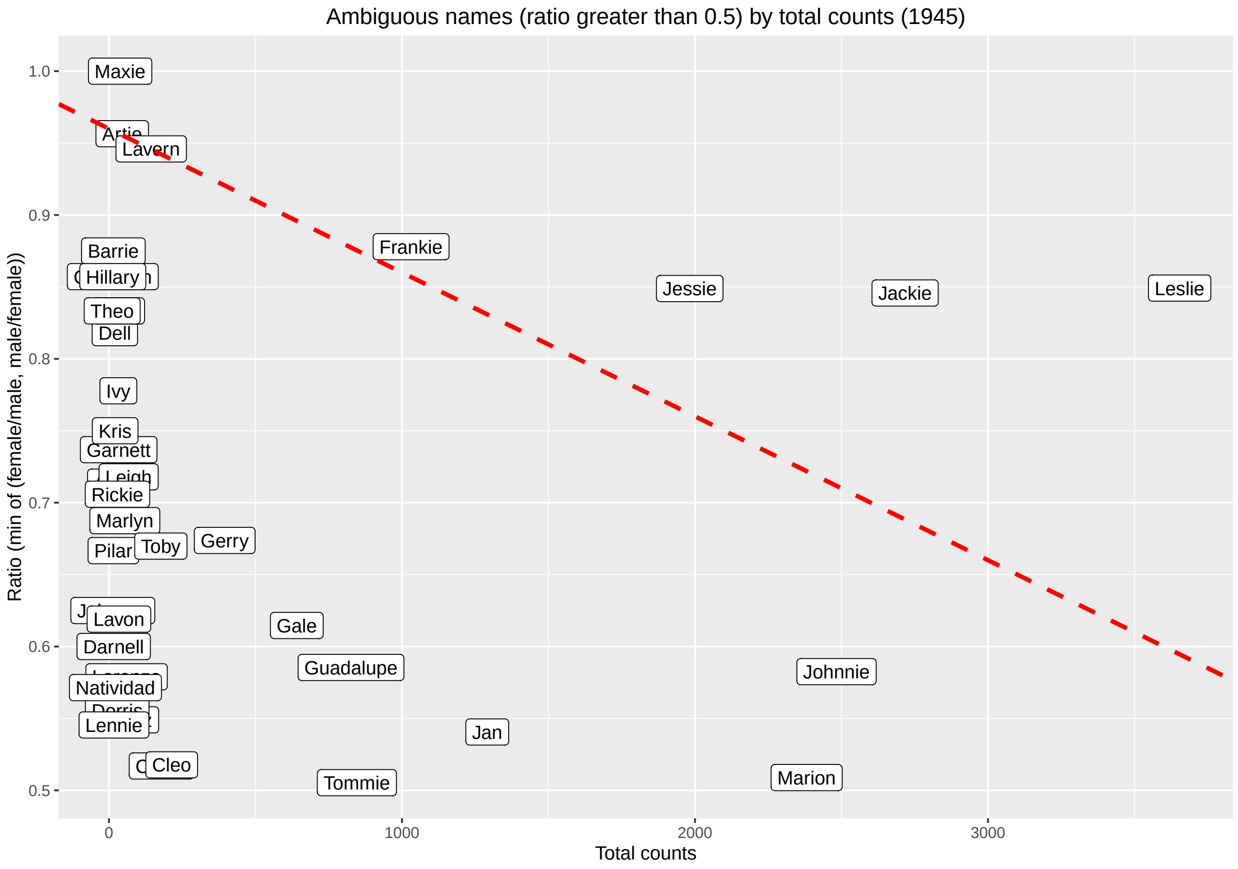
name	ratio	tot.count
Maxie	1.00	38
Artie	0.96	45
Lavern	0.95	144
Frankie	0.88	1031
Barrie	0.88	15

```r
kable(top.count.ratio.1945[1:5, ], caption = "Top count and ratio 1945")
```

Table 4: Top count and ratio 1945

name	ratio	tot.count
Leslie	0.85	3654
Jackie	0.85	2717
Johnnie	0.58	2483
Marion	0.51	2381
Jessie	0.85	1982

```r
## Plotting
ggplot(ambiguous.1945, aes(x = tot.count, y = ratio, label = name)) +
    geom_label() + geom_abline(intercept = 0.96, slope = -1e-04,
    colour = "red", linetype = "dashed", size = 1.2) + labs(x = "Total counts",
    y = "Ratio (min of (female/male, male/female))", title = "Ambiguous names (ratio greater than 0.5) 
    cex = 1.5)
```



Q4. Of the names represented in the data, find the name that has had the largest percentage increase in popularity since 1980. Largest decrease?

```
df.perc = df %>% filter(year == 2015 | year == 1980) %>% group_by(name,
    year) %>% summarize(tot.count = sum(count)) %>% filter(n() >
    1) %>% summarize(perc.change = (tot.count[year == 2015] -
    tot.count[year == 1980])/tot.count[year == 1980]) %>% mutate(perc.change = 100 *
    round(perc.change, 4))
```

Name with largest percentage increase: Aria

```
largest.incr = df.perc %>% arrange(desc(perc.change))
kable(largest.incr[1:5, ], caption = "Names with largest percentage increase since 1980")
```

Table 5: Names with largest percentage increase since 1980

name	perc.change
Aria	127440

name	perc.change
Colton	125260
Skylar	110980
Mateo	99240
Mila	94100

Name with largest percentage decrease: Jill

```
largest.decr = df.perc %>% arrange(perc.change)
kable(largest.decr[1:5, ], caption = "Names with largest percentage decrease since 1980")
```

Table 6: Names with largest percentage decrease since 1980

name	perc.change
Jill	-99.85
Misty	-99.75
Jodi	-99.70
Brandy	-99.61
Kristy	-99.56

Q5. Can you identify names that may have had an even larger increase or decrease in popularity?

As mentioned at the beginning, due to excluding names that have under 5 counts, it is possible that there are names that may have had an even larger increase or decrease in popularity compared to the names shown in question 4.

In question 4, we have Aria as name with largest percentage increase 127440%. In other words, the count of Aria has increased 1274.4 times since 1980. Thus, we need to look for names that could possibly increase more than 1274.4 times. Specifically, we look for names that have more than 1275 counts in 2015 but were not on record in 1980, assuming that these names have 1 count in 1980.

Similarly, Jill was the name with largest percentage decrease 99.85%. We look at names on record in 1980 but not on record in 2015, assuming that they only had 1 count in 2015. Thus, names that may have had larger decrease are names that have more than 715 counts in 1980. ($\frac{x-1}{x} > .9985 \;\; => \;\; x > \frac{1}{1-.9985} = 714.3$)

Based on the result, there are 97 names that may have had a larger increase in popularity and 27 names that may have had a larger decrease in popularity.

```
names.2015 = df %>% filter(year == 2015) %>% select(name) %>%
    unique
names.1980 = df %>% filter(year == 1980) %>% select(name) %>%
    unique
names.only.2015 = unlist(setdiff(names.2015, names.1980))
names.only.1980 = unlist(setdiff(names.1980, names.2015))
potential.larger.incr = df %>% filter(year == 2015 & (name %in%
    names.only.2015)) %>% group_by(name) %>% summarize(tot.count = sum(count)) %>%
    filter(tot.count >= 1275) %>% arrange(desc(tot.count))
kable(potential.larger.incr[1:5, ], caption = "Top 5 names that may have had a larger increase in popula
```

Table 7: Top 5 names that may have had a larger increase in popularity

name	tot.count
Aiden	13396
Jayden	11874
Harper	10546
Madison	10079
Jaxon	8011

```
potential.larger.decr = df %>% filter(year == 1980 & (name %in%
    names.only.1980)) %>% group_by(name) %>% summarize(tot.count = sum(count)) %>%
    filter(tot.count >= 715) %>% arrange(desc(tot.count))
kable(potential.larger.decr[1:5, ], caption = "Top 5 names that may have had a larger decrease in popula
```

Table 8: Top 5 names that may have had a larger decrease in popularity

name	tot.count
Tonya	3073
Beth	2844
Kristi	2521
Latoya	2480
Tasha	2256

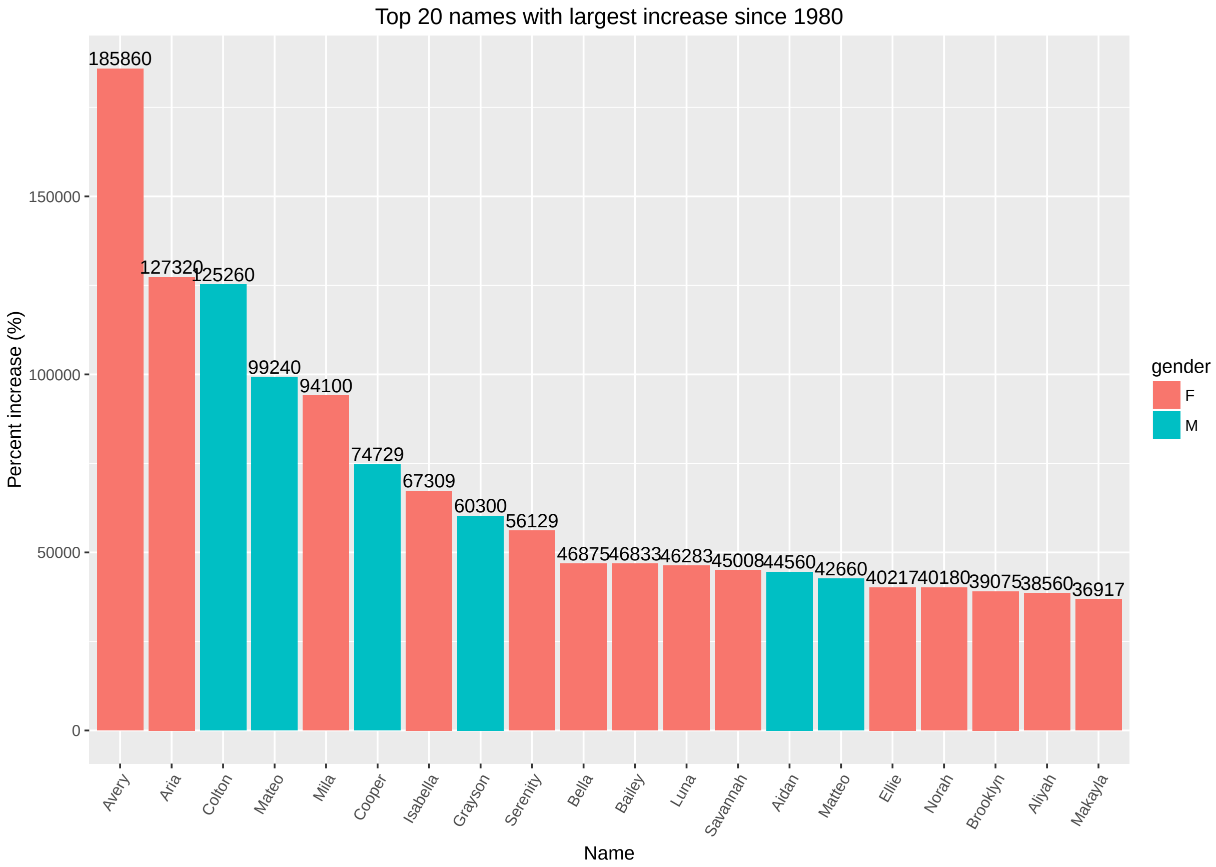
B) Onward to Insight!

Extending question 4: Differentiate between male vs. female name (eg. treat Milan for male and for female as two different names)

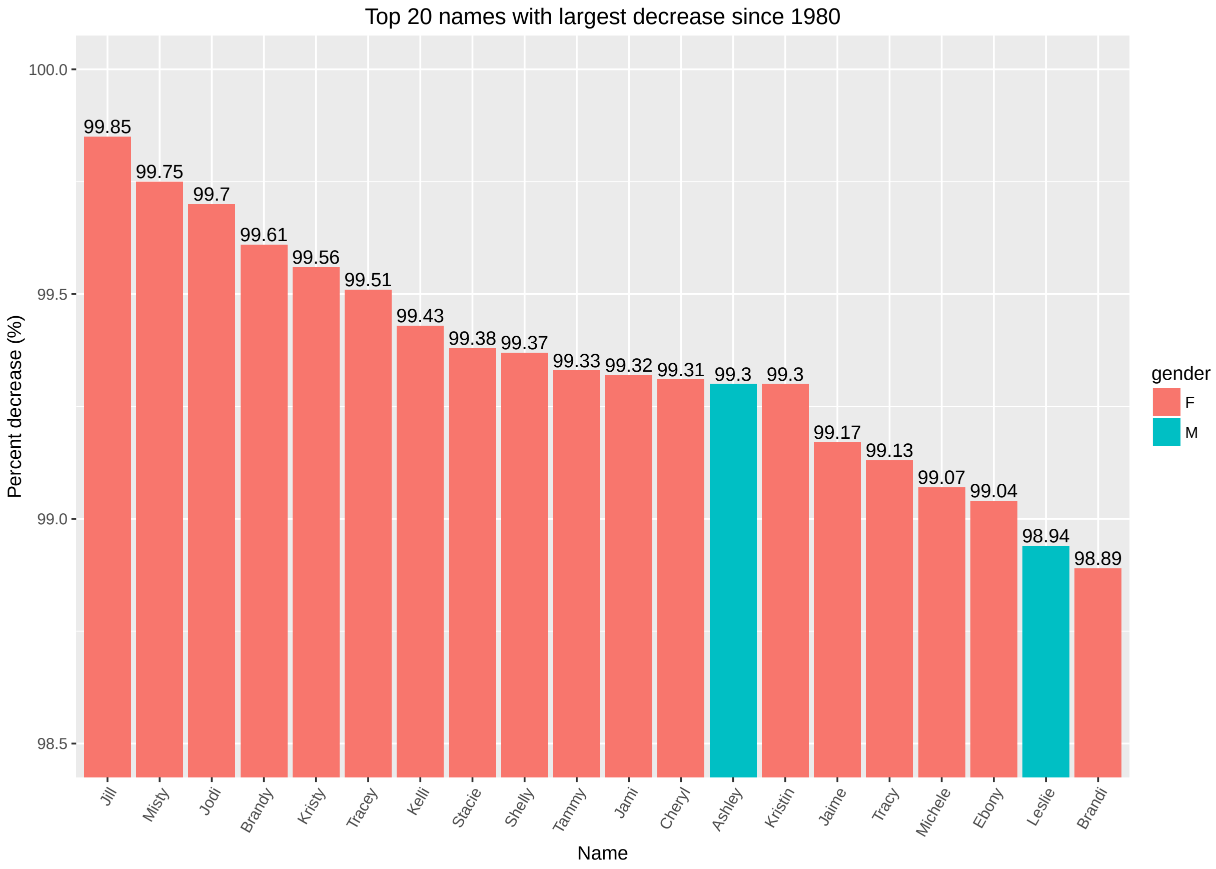
Looking at the percent change of names since 1980 while distinguishing between male vs. female name, an interesting fact is that female names was major in the top 20 largest increase, and completely dominant in the top 20 largest decrease.

```
df.perc.gender = df %>% filter(year == 2015 | year == 1980) %>%
    group_by(name, gender) %>% summarize(count.1980 = sum(count[year ==
    1980]), count.2015 = sum(count[year == 2015])) %>% filter(count.1980 >
    0 & count.2015 > 0) %>% mutate(perc.change = 100 * round((count.2015 -
    count.1980)/count.1980, 4))

## Plotting for top 20 names with largest increase
largest.incr.by.gender = df.perc.gender %>% arrange(desc(perc.change))
ggplot(largest.incr.by.gender[1:20, ], aes(x = reorder(name,
    -perc.change), y = perc.change, fill = gender)) + geom_bar(stat = "identity") +
    geom_text(aes(label = round(perc.change, 0)), vjust = -0.25) +
    theme(axis.text.x = element_text(angle = 60, hjust = 1)) +
    labs(x = "Name", y = "Percent increase (%)", title = "Top 20 names with largest increase since 1980
```



```
## Plotting for top 20 names with largest decrease
largest.decr.by.gender = df.perc.gender %>% arrange(perc.change)
ggplot(largest.decr.by.gender[1:20, ], aes(x = reorder(name,
    perc.change), y = abs(perc.change), fill = gender)) + geom_bar(stat = "identity") +
    geom_text(aes(label = abs(perc.change)), vjust = -0.25) +
    coord_cartesian(ylim = c(98.5, 100)) + theme(axis.text.x = element_text(angle = 60,
    hjust = 1)) + labs(x = "Name", y = "Percent decrease (%)",
    title = "Top 20 names with largest decrease since 1980")
```



Web scrapping to collect U.S population historical data.

After collecting data on U.S population from 1910 to 2015, I merged it with the baby names data aggregated by total count for each year.

The graph below shows the yearly changes in population and name counts over time. The time series line of "name counts" is expected to be always above "population" because it only takes into account birth but not death, while $population\ change = number\ of\ birth - number\ of\ death$. However, we should also keep in mind that "name counts" excludes names with less than 5 counts, which might be a large number of birth.

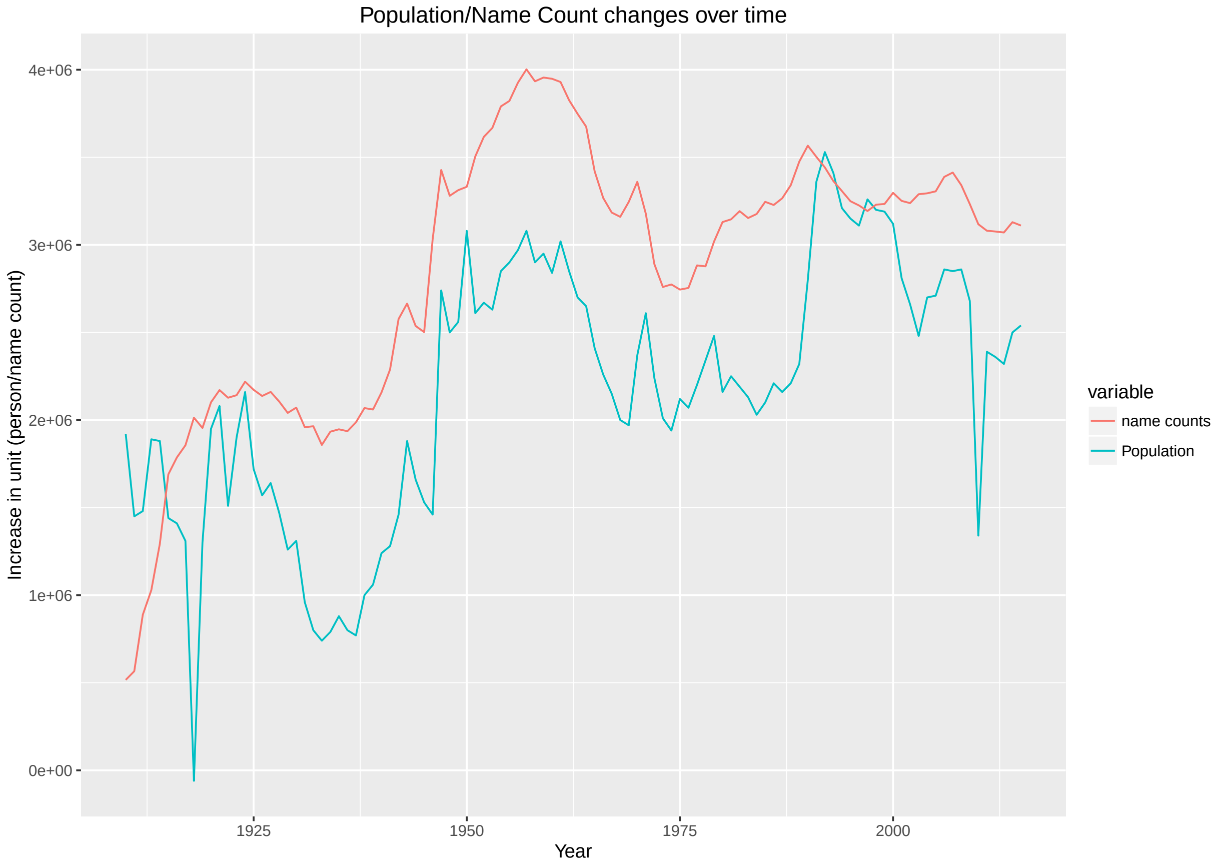
Surprisingly, name counts line is below population line for a few year at the beginning. This might be due to the constraint of data collection back then or not everyone registered their children with Social Security.

In 1918, population change was negative (-6000), which means that death outnumbered birth. This is due to one of the deadliest flu pandemic which killed around 500,000 people in the U.S.

The two lines get closer to each other over time. This might be explained by the fact that since 1960, U.S population growth has been the result of immigration and increase life expectancy with new technology.

```
url = "http://www.multpl.com/united-states-population/table"
web = read_html(url)
web.table = html_nodes(web, "table")
pop.table = html_table(web.table)[[1]]
names(pop.table) = c("year", "population")
pop.table$year = as.integer(sub("(.)*, ", "", pop.table$year))
pop.table$population = as.numeric(sub(" million", "", pop.table$population))
```

```
## Keep only data from 1910 - 2015
pop.table %<>% filter(year >= 1909 & year <= 2015) %>% arrange(year) %>%
    mutate(grow = 10^6 * (population - lag(population, default = NA))) %>%
    filter(year >= 1910)
names.count = df %>% group_by(year) %>% summarize(count = sum(count))
pop.merge = merge(pop.table, names.count, by = "year")
ggplot(pop.merge, aes(x = year, y = value, color = variable)) +
    geom_line(aes(y = grow, col = "Population")) + geom_line(aes(y = count,
    col = "name counts")) + labs(x = "Year", y = "Increase in unit (person/name count)",
    title = "Population/Name Count changes over time")
```



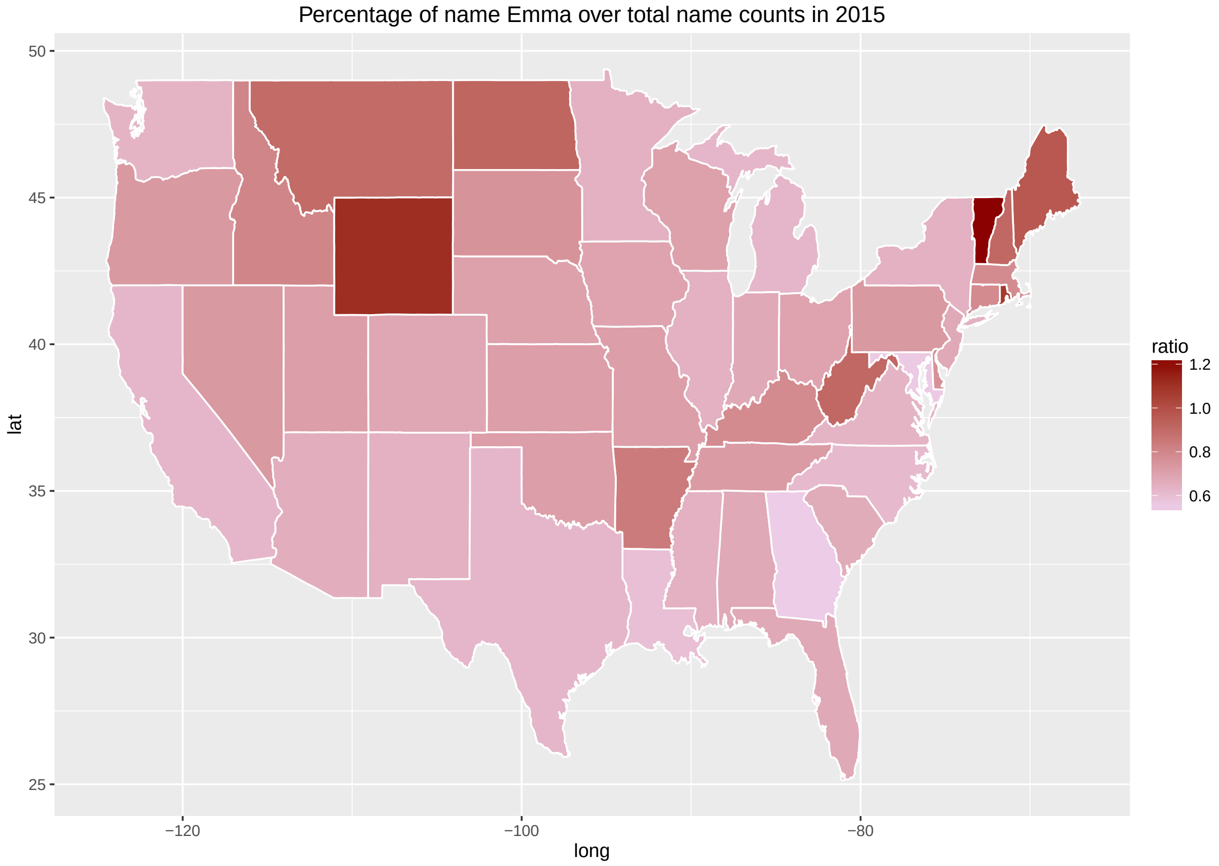
Fun with geospatial visualization

Most “trendy” state

From question 2, we see that Emma has been the most popular name for the last 2 years (2014,2015). Lets see which states have the highest ratio of newborn named Emma over total name counts of the last two years. Ratio is a good metric to look at instead of raw counts to avoid bias toward states with large population.

Based on the graph, Vermont is the most “trendy” state with highest ratio of Emma name. Wyoming, Rhode Island and Maine come after that.

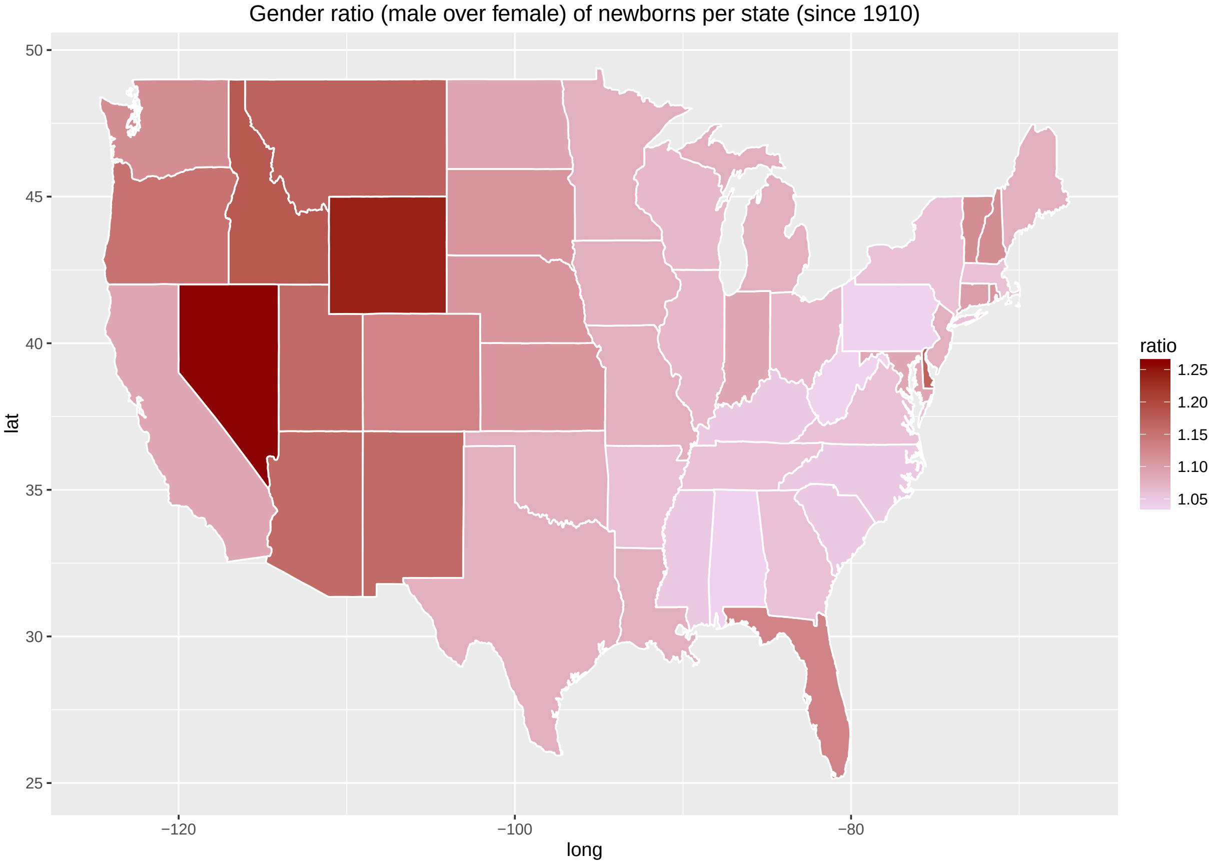
```
map.dat = map_data("state")
df$region = ifelse(df$state == "DC", "district of columbia",
    tolower(state.name[match(df$state, state.abb)]))
emma.per.state = df %>% filter(year == 2015 | year == 2014) %>%
    group_by(region) %>% summarize(emma.count = sum(count[name ==
    "Emma"]), tot.count = sum(count)) %>% mutate(ratio = round(100 *
    emma.count/tot.count, 2))
merge.dat = merge(emma.per.state, map.dat, by = "region")
ggplot() + geom_polygon(data = merge.dat, aes(x = long, y = lat,
    group = group, fill = ratio), colour = "white") + scale_fill_continuous(low = "thistle2",
    high = "darkred", guide = "colorbar") + labs(title = "Percentage of name Emma over total name count
```



Gender ratio of newborns per state

Alaska (not on the map below) is the state with greatest gender imbalance of newborns (1.3 male per female). We can also see on the graph that most of states on the west have imbalance of newborns gender, while states on the east coast are closer to 1:1 ratio.

```
gender.ratio = df %>% group_by(region) %>% summarize(ratio = round(sum(count[gender ==
    "M"])/sum(count[gender == "F"]), 2))
merge.dat2 = merge(gender.ratio, map.dat, by = "region")
ggplot() + geom_polygon(data = merge.dat2, aes(x = long, y = lat,
    group = group, fill = ratio), colour = "white") + scale_fill_continuous(low = "thistle2",
    high = "darkred", guide = "colorbar") + labs(title = "Gender ratio (male over female) of newborns p
```


Gender ratio (male over female) of newborns per state (since 1910)
50
45
40
35
30
25
lat
−120
−100
−80
long
ratio
1.25
1.20
1.15
1.10
1.05