

Stat405

ddply case study

Hadley Wickham

1. Feedback & homework & project
2. Overall goal: dual-sex names vs. errors
3. Selecting smaller subset
4. Classification
5. Individual exploration

Feedback

I'll try and go slower when writing things on the board. Remind me!

Too much homework? Will try to reduce from now on. This week's homework is a bit different.

Homework

If you need more practice, all function drills, along with answers, are available on line.

Running behind on grading, sorry :(

Common mistakes

```
even <- function(x) {  
  is_even <- x %% 2 == 0  
  if (is_even) {  
    print("Even!")  
  } else {  
    print("Odd!")  
  }  
}
```

Problems

* does it work with vectors?

* can we easily define odd in terms of even?

```
even <- function(x) {  
  x %% 2 == 0  
}
```

```
even(1:10)
```

```
odd <- function(x) {  
  !even(x)  
}
```

```
# In general, always should return something useful  
# from functions, rather than printing or plotting
```

```
area <- function(r) {  
  a <- pi * r ^ 2  
  a  
}
```

Not necessary!

```
area <- function(r) {  
  pi * r ^ 2  
}
```

```
# Choose from a, b and c with equal probability
```

```
x <- runif(1)
if (x < 1/3) {
  "a"
} else (x < 2/3) {
  "b"
} else {
  "c"
}
```

```
# OR
sample(c("a", "b", "c"), 1)
```

Project

Still working on grading. Will have back to you by next Wednesday (no class on Monday).

Next project due Oct 30.

Basically same as last time, but working with baby names and you need to include an external data source.

Questions

For names that are used for both boys and girls, how has usage changed?

Can we use names that clearly have the incorrect sex to estimate error rates over time?

Getting started

```
options(stringsAsFactors = FALSE)
```

```
library(plyr)
```

```
library(ggplot2)
```

```
bnames <- read.csv("baby-names.csv")
```

First task

Identify a smaller subset of names that been in the top 1000 for both boys and girls. ~7000 names in total, we want to focus on ~100.

In real-life would probably use more, but starting with a subset for easier exploration is still a good idea.

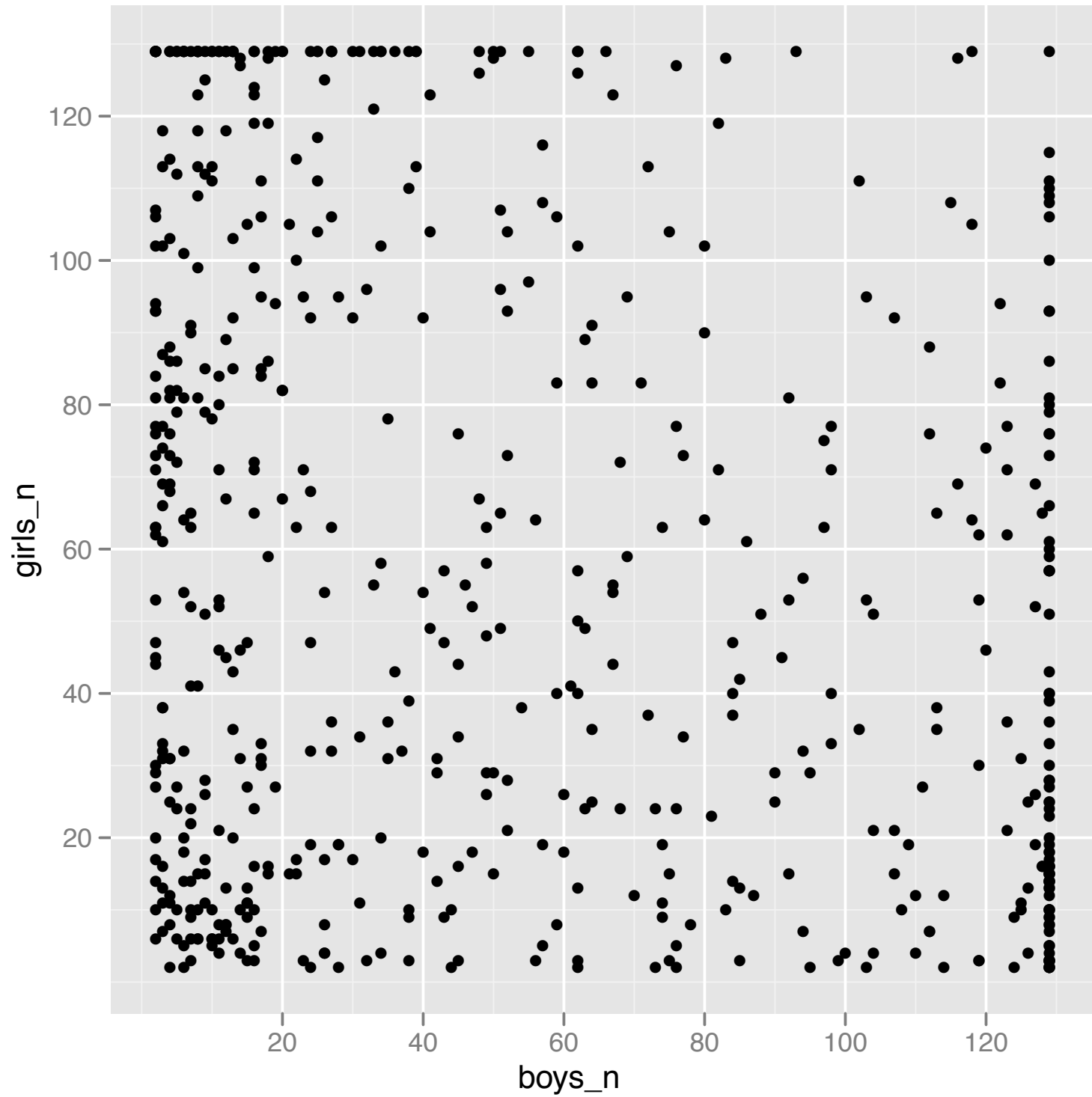
Take two minutes to brainstorm what variables we might to create to do this.

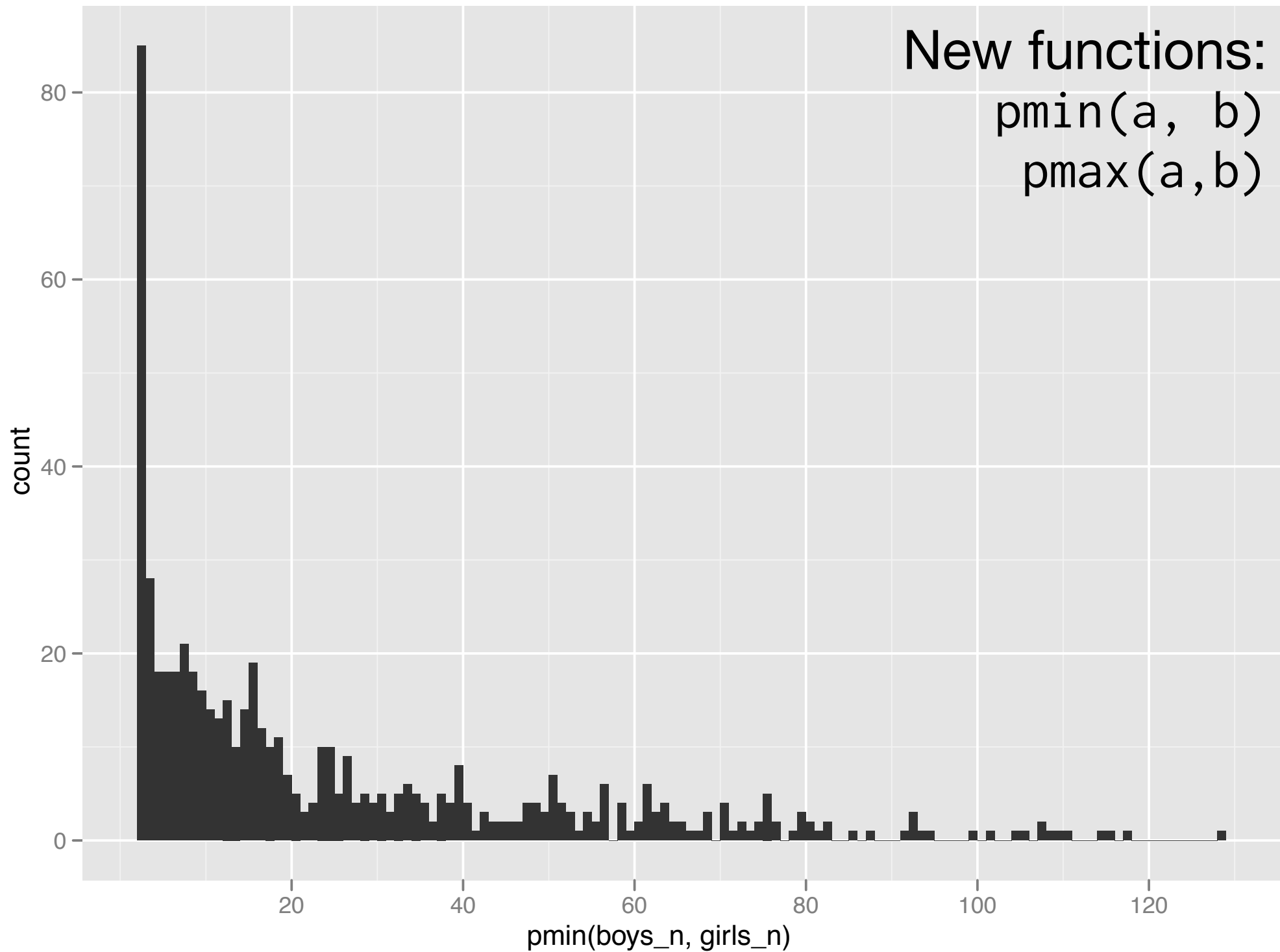
Your turn

Summarise each name with: the total proportion of boys, the total proportion of girls, the number of years the name was in the top 1000 as a girls name, the number of years the name was in the top 1000 as a boys name

Hint: Start with a single name and figure out how to solve the problem. **Hint:** Use summarise

```
times <- ddpoly(bnames, c("name"), summarise,  
  boys = sum(prop[sex == "boy"]),  
  boys_n = sum(sex == "boy"),  
  girls = sum(prop[sex == "girl"]),  
  girls_n = sum(sex == "girl"),  
  .progress = "text"  
)  
  
nrow(times)  
times <- subset(times, boys_n > 1 & girls_n > 1)
```





```
qplot(boys_n, girls_n, data = times)
```

```
qplot(pmin(boys_n, girls_n), data = times,  
      binwidth = 1)
```

```
times$both <- with(times, boys_n > 10 & girls_n > 10)
```

```
# Still a few too many names. Lets focus on names  
# that have managed a certain level of popularity.
```

```
qplot(pmin(boys, girls), data = subset(times, both),  
      binwidth = 0.01)
```

```
qplot(pmax(boys, girls), data = subset(times, both),  
      binwidth = 0.1)
```

```
qplot(boys + girls, data = subset(times, both),  
      binwidth = 0.1)
```

```
# Now save our selections
```

```
both_sexes <- subset(times, both &  
  boys + girls > 0.4)
```

```
selected_names <- both_sexes$name
```

```
selected <- subset(bnames, name %in% selected_names)  
nrow(selected) / nrow(bnames)
```

Yearly summaries

Next problem is to classify which names are dual-sex, and which are errors.

To do that, we'll need to calculate yearly summaries for each of those names, and use our knowledge of names to come up with a good classification criterion.

Your turn

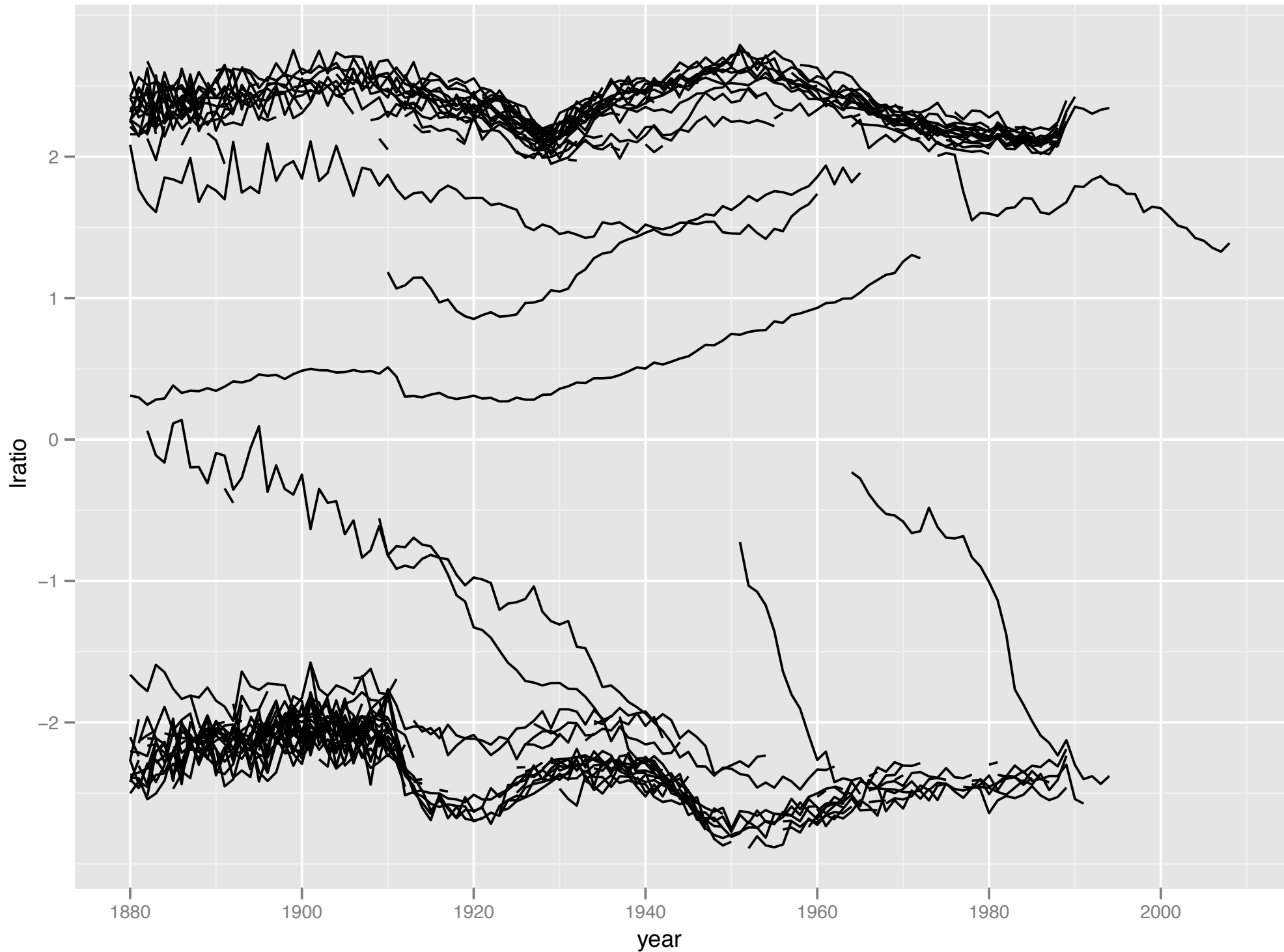
For each name, in each year, figure out the total number of boys and girls.

Think of ways to summarise the difference between the number of boys and girls, and start visualising the data.

```
bysex <- ddply(selected, c("name", "year"),
summarise,
  boys = sum(prop[sex == "boy"]),
  girls = sum(prop[sex == "girl"]),
  .progress = "text"
)
```

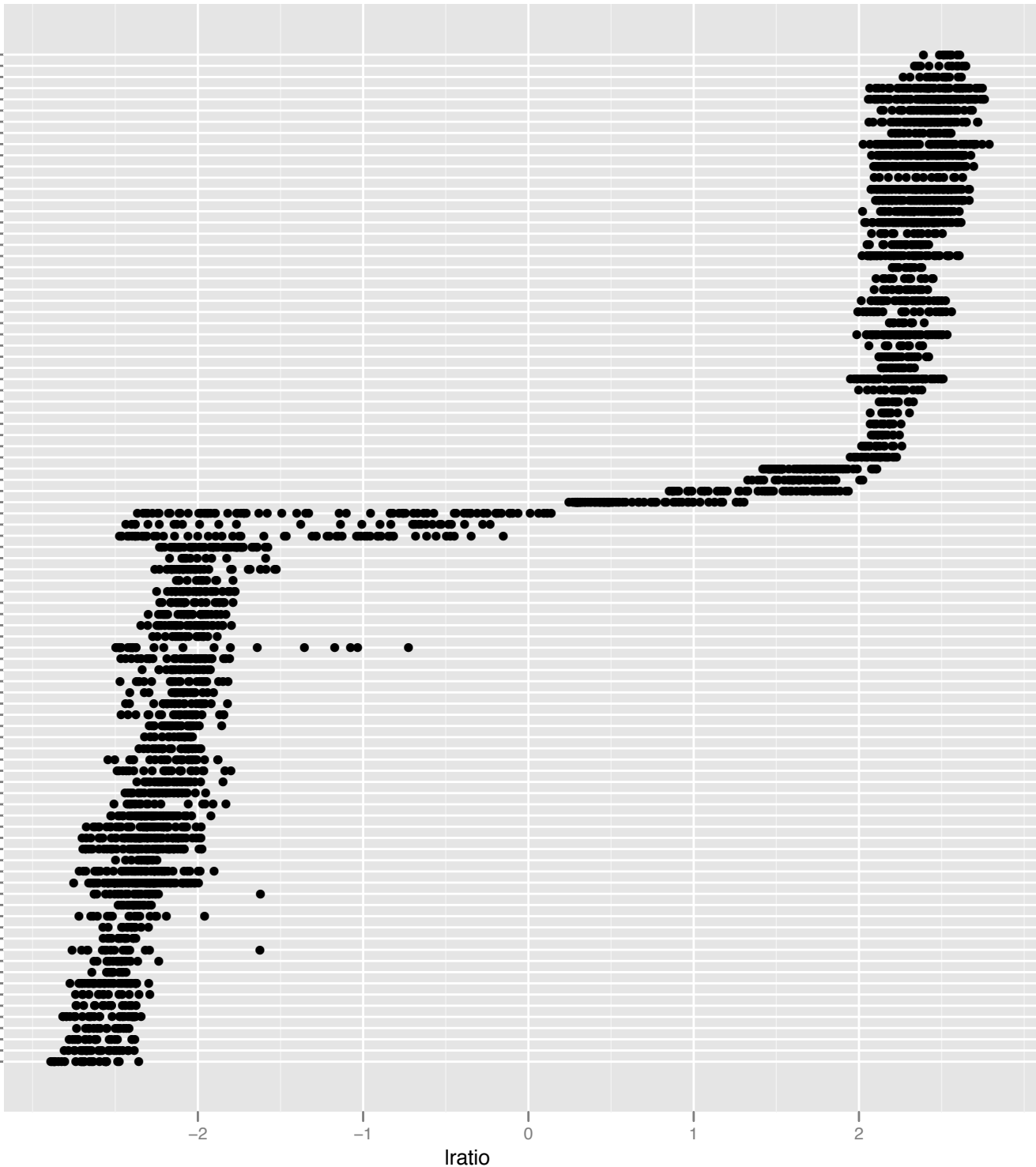
```
# It's useful to have a symmetric means of comparing
# the relative abundance of boys and girls - the log
# ratio is good for this.
```

```
bysex$lratio <- log10(bysex$boys / bysex$girls)
bysex$lratio[!is.finite(bysex$lratio)] <- NA
```



reorder(name, lratio, na.rm = T)

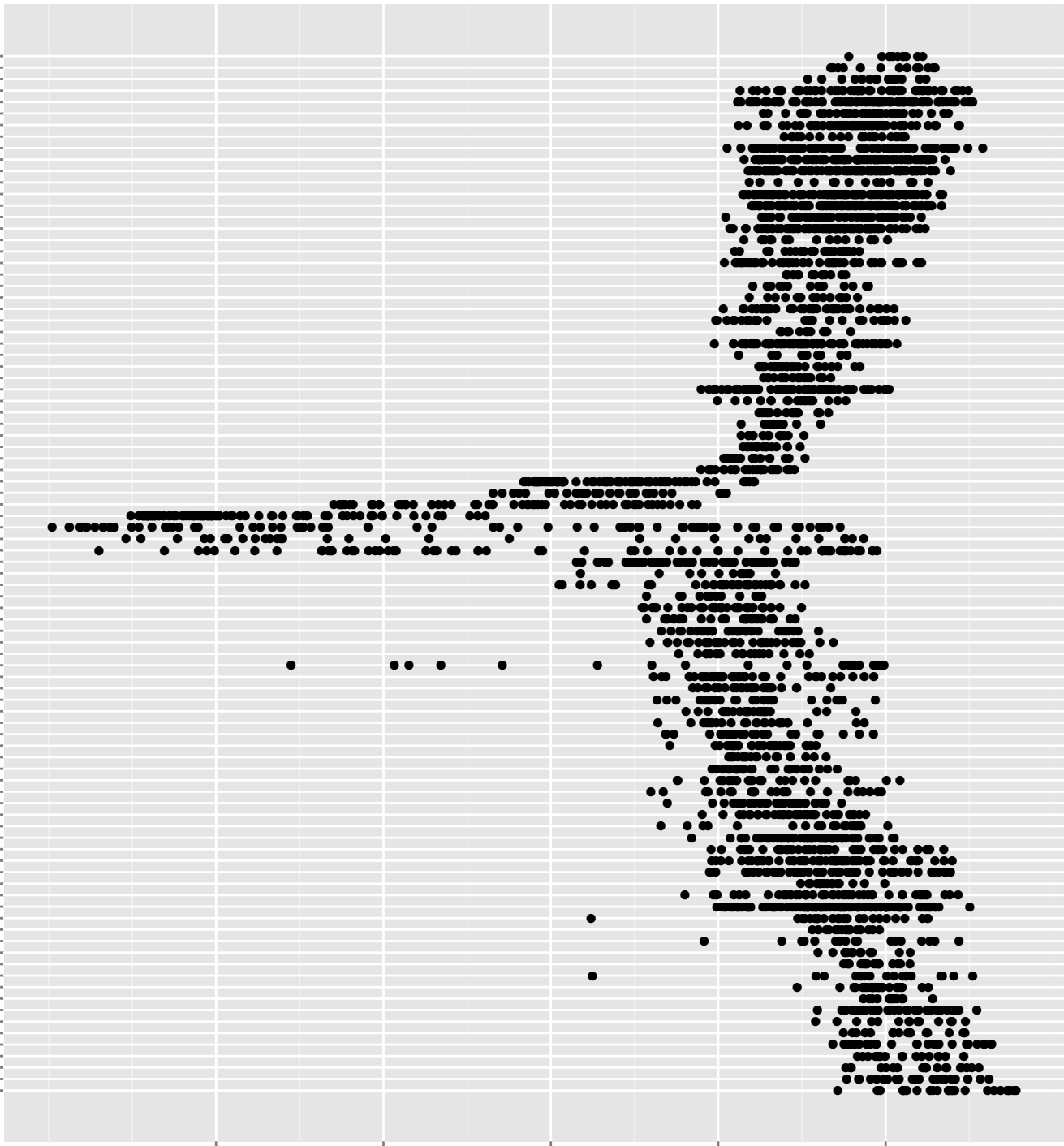
- Ronald
- Mark
- Larry
- Richard
- William
- Edward
- Thomas
- Donald
- David
- John
- Robert
- Harry
- James
- Joseph
- Frank
- Charles
- Albert
- Paul
- Michael
- Brian
- Kenneth
- Harold
- Walter
- Arthur
- Matthew
- George
- Kevin
- Christopher
- Jack
- Henry
- Fred
- Jason
- Joshua
- Eric
- Daniel
- Anthony
- Louis
- Joe
- Ryan
- Walter
- Shirley
- Shirley
- Ashley
- Carol
- Frances
- Julia
- Doris
- Irene
- Louise
- Rose
- Florence
- Ethel
- Edith
- Kimberly
- Annie
- Edna
- Minnie
- Grace
- Gara
- Bertha
- Lillian
- Martha
- Marie
- Emma
- Mildred
- Alice
- Anna
- Sarah
- Elizabeth
- Ruth
- Margaret
- Helen
- Virginia
- Dorothy
- Mary
- Betty
- Michelle
- Sharon
- Jessica
- Melissa
- Nancy
- Jennifer
- Amanda
- Patricia
- Donna
- Sandra
- Barbara
- Lisa
- Karen
- Linda
- Susan



lratio

reorder(name, lratio, na.rm = T)

Ronald
Mark
Larry
Richard
William
Edward
Thomas
Donald
David
John
Robert
Harry
James
Joseph
Frank
Charles
Albert
Paul
Michael
Brian
Kenneth
Walter
Arthur
Matthew
George
Kevin
Christopher
Jack
Henry
Fred
Jason
Joshua
Eric
Daniel
Anthony
Louis
Joe
Ryan
Willy
Shirley
Ashley
Carol
Frances
Julia
Doris
Louise
Rose
Florence
Ethel
Edith
Kimberly
Annie
Edna
Minnie
Grace
Gara
Bertha
Lillian
Martha
Marie
Emma
Mildred
Anna
Sarah
Elizabeth
Ruth
Margaret
Helen
Virginia
Dorothy
Mary
Betty
Michelle
Sharon
Jessica
Melissa
Nancy
Jennifer
Amanda
Patricia
Donna
Sandra
Barbara
Lisa
Karen
Linda
Susan



abs(lratio)

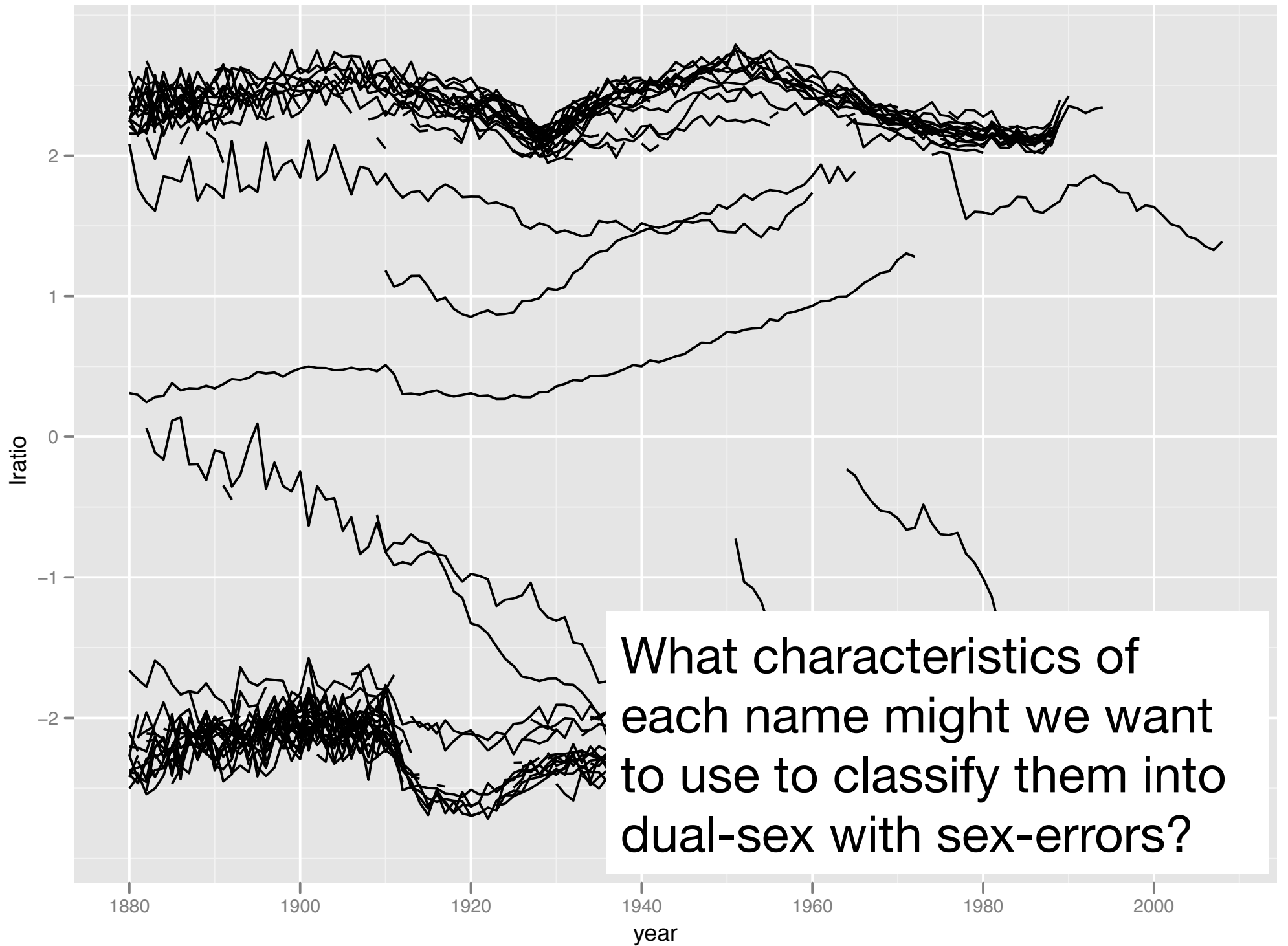
```
theme_set(theme_grey(10))
```

```
qplot(year, lratio, data = bysex, group = name,  
      geom = "line")
```

```
qplot(lratio, reorder(name, lratio, na.rm = T),  
      data = bysex)
```

```
qplot(abs(lratio), reorder(name, lratio, na.rm = T),  
      data = bysex)
```

```
qplot(abs(lratio), reorder(name, lratio, na.rm = T),  
      data = bysex) +  
  geom_point(data = both_sexes, colour = "red")
```



Your turn

Compute the mean and range of I_{ratio} for each name.

Plot and come up with cutoffs that you think separate the two groups.

```
rng <- ddply(bysex, "name", summarise,  
  diff = diff(range(lratio, na.rm = T)),  
  mean = mean(lratio, na.rm = T)  
)  
  
qplot(diff, abs(mean), data = rng)  
qplot(diff, abs(mean), data = rng,  
  colour = abs(mean) < 1.75 | diff > 0.9)  
  
shared_names <- subset(rng, abs(mean) < 1.75 |  
  diff > 0.9)$name  
  
qplot(abs(lratio), reorder(name, lratio, na.rm=T),  
  data = subset(bysex, name %in% shared_names))  
qplot(year, lratio, geom = "line", group = name,  
  data = subset(bysex, name %in% shared_names))
```

Next time

Now that we've separated the two groups, we'll explore each in more detail.