CPSC 340:
Machine Learning and Data Mining

Data Exploration

Bonus slides

This lecture roughly follow:
http://www-users.cs.umn.edu/~kumar/dmbook/dmslides/chap2_data.pdf

A Simple Setting: Coupon Collecting

- Assume we have a categorical variable with 50 possible values:
  - {Alabama, Alaska, Arizona, Arkansas,…}.
- Assume each category has probability of 1/50 of being chosen:
  - How many examples do we need to see before we expect to see them all?
- Expected value is ~225.
- Coupon collector problem: O(n log n) in general.
  - Gotta Catch'em all!
- Obvious sanity check, is need more samples than categories:
  - Situation is worse if they don't have equal probabilities.
  - Typically want to see categories more than once to learn anything.

Distances and Similarities

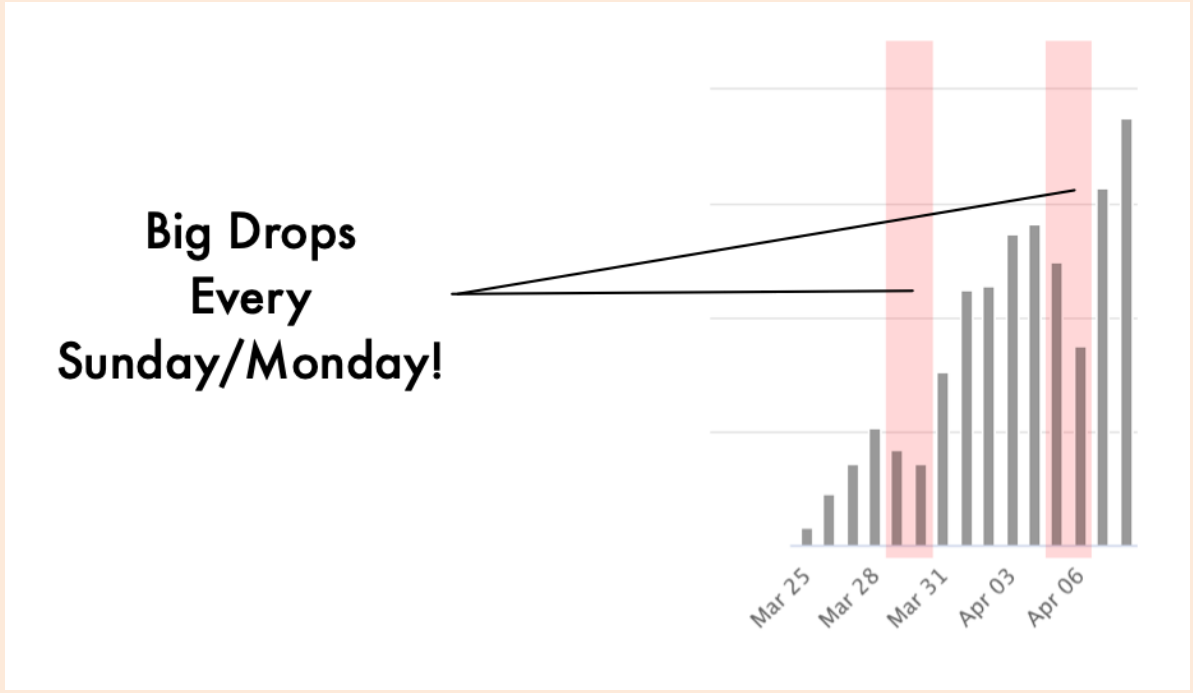
- There are also summary statistics between features ‘x’ and ‘y’.
  - Rank correlation:
    - Does one increase/decrease as the other increases?
      - Not necessarily in a linear way.

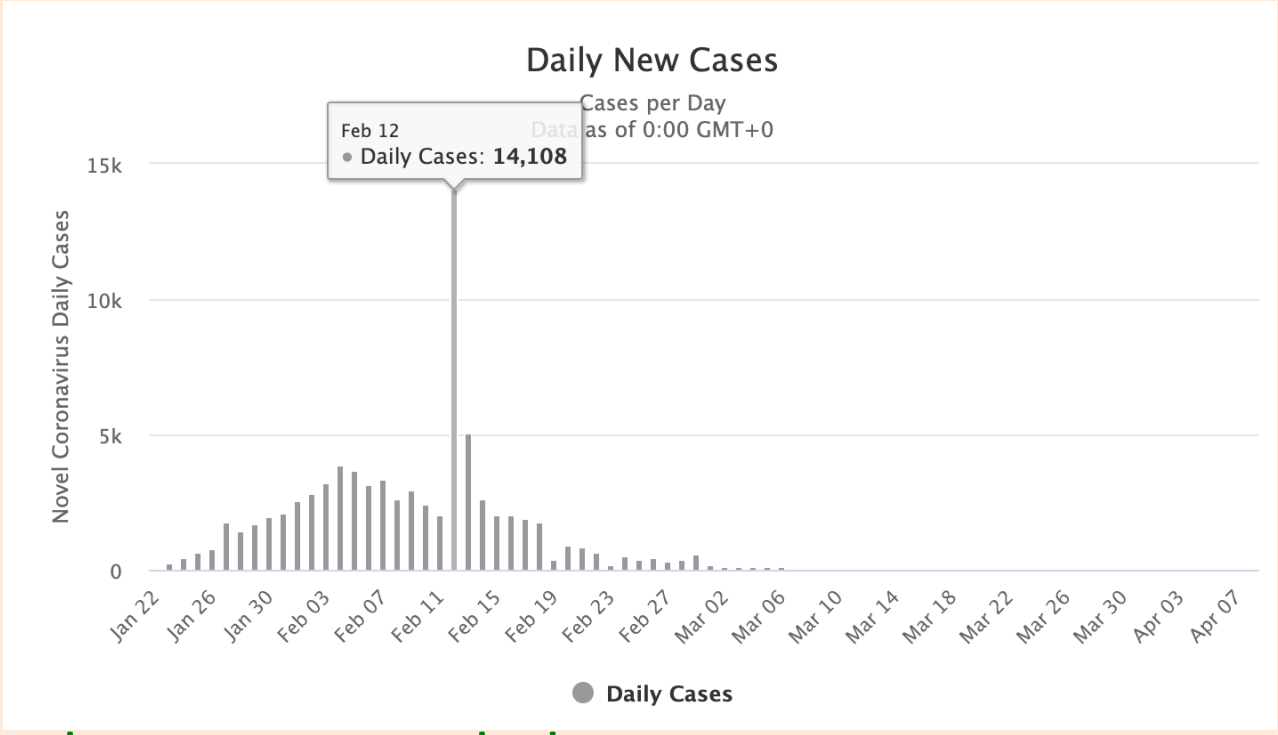
x	y
0	0
0	0
1	0
0	1
0	1
1	1
0	0
0	1
0	1

- Distances/similarities between other types of data:
  - Jaccard coefficient (distance between sets):
    - (size of intersection of sets) / (size of union of sets)
  - Edit distance (distance between strings):
    - How many characters do we need to change to go from x to y?
    - Computed using dynamic programming (CPSC 320).

Histogram

- "Four Basic Data Science Lessons Illustrated by COVID-19 Data"
  - First two lessons come from just plotting:





  - These oddities had to do with how data was recorded.

“Why Not to Trust Plots”

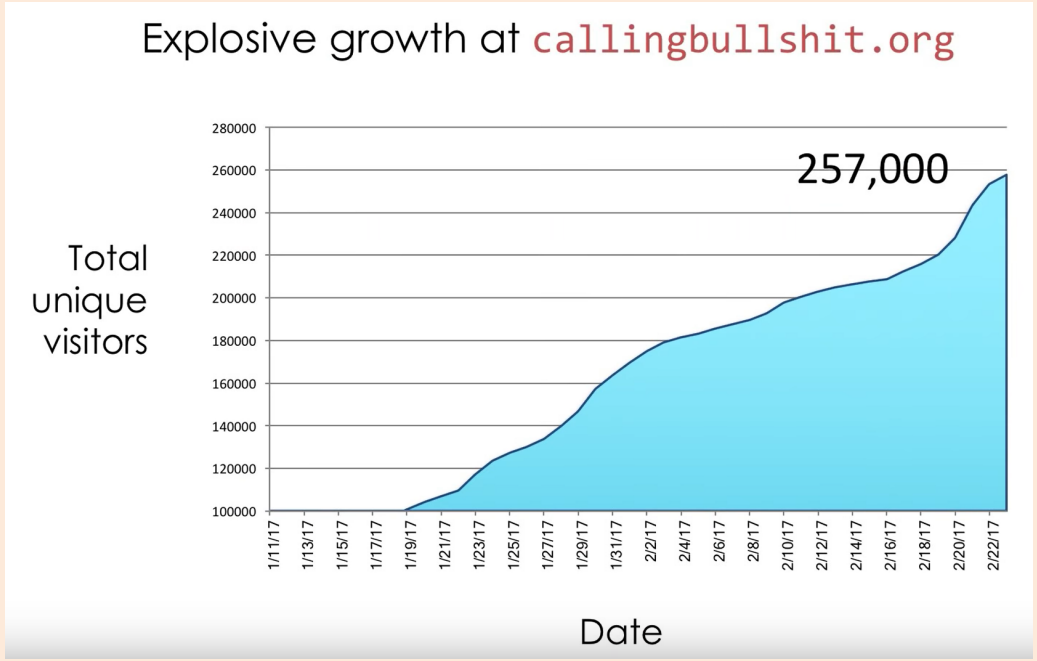
- We’ve seen how summary statistics can be mis-leading.
- Note that plots can also be mis-leading, or can be used to mis-lead.

- Next slide: first example from UW’s excellent course:
  - “Calling Bullshit in the Age of Big Data”.
    - A course on how to recognize when people are trying to mis-lead you with data.
  - I recommend watching all the videos here:
    - https://www.youtube.com/watch?v=A2OtU5vlR0k&list=PLPnZfvKID1Sje5jWxt-4CSZD7bUI4gSPS

  - Recognizing BS not only useful for data analysis, but for daily life.

Mis-Leading Axes

- This plot seems to show amazing recent growth:

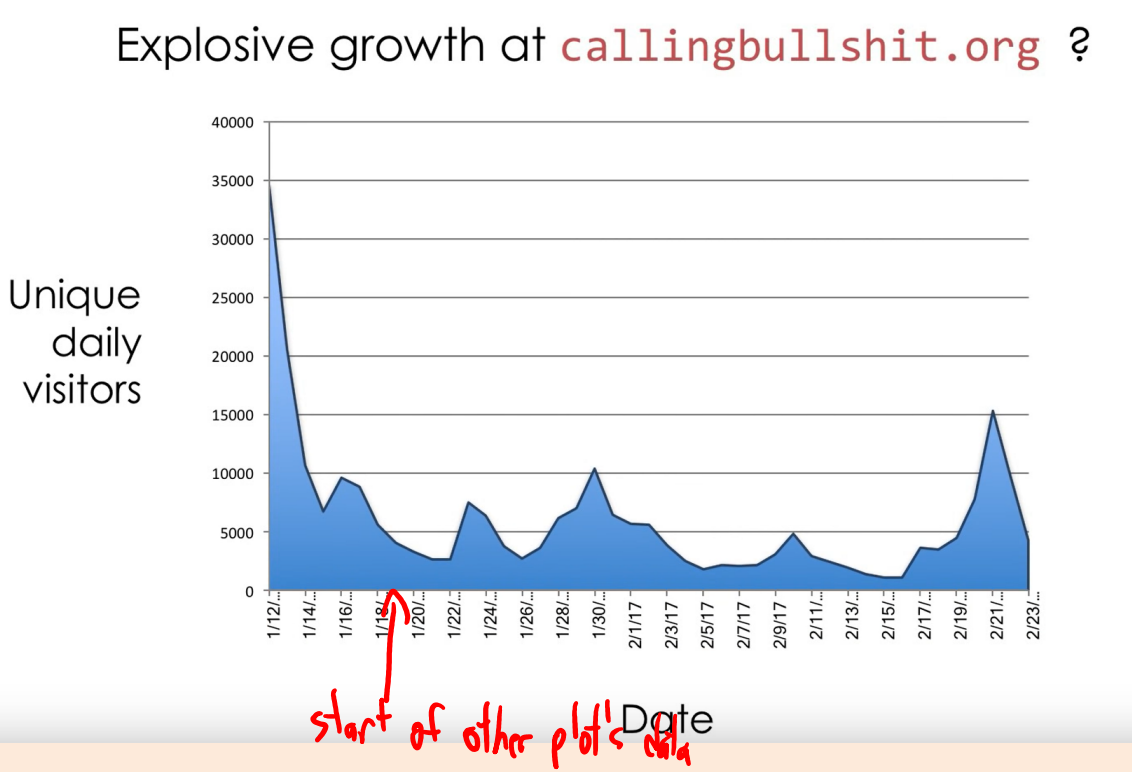


- But notice y-axis starts at 100,000 (so ~40% of growth was earlier).
- And it plots "total" users (which necessarily goes up).

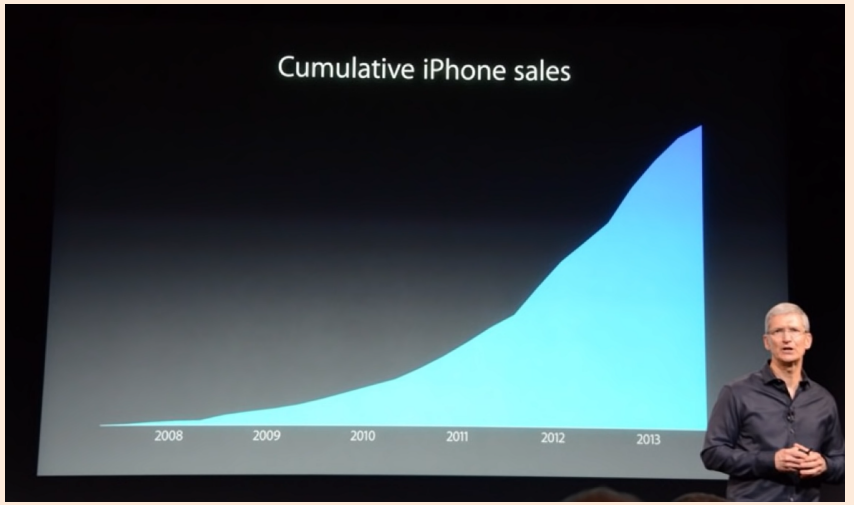
https://www.youtube.com/watch?v=q94VJ3KToK8

Mis-Leading Axes

- Plot of actual daily users (starting from 0) looks totally different:



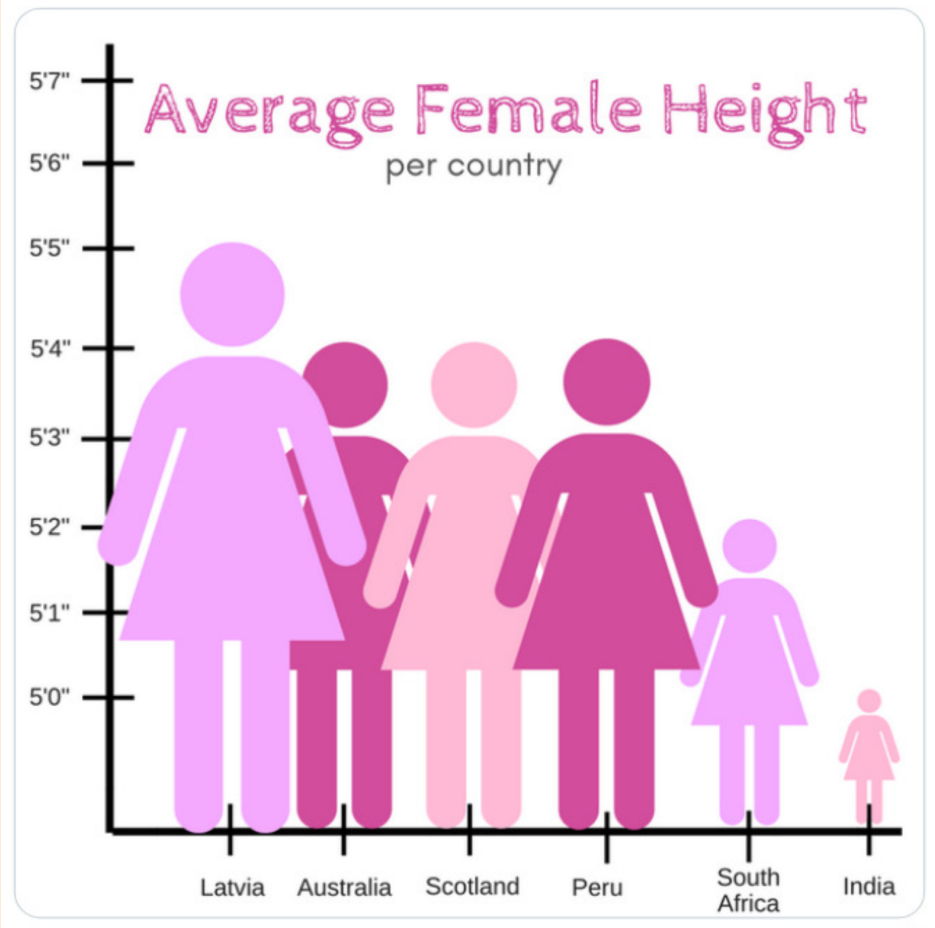
- People can mis-lead to push agendas/stories:



https://www.youtube.com/watch?v=q94VJ3KToK8

Mis-Leading Axes

- Watch out for the starting point of the axes too:



https://www.boredpanda.com/average-women-height-data-chart-latvian-indian-women

Mis-Leading Axes

- We see "lack of appropriate axes" ALL THE TIME in the news:
  - "British research revealed that patients taking ibuprofen to treat arthritis face a 24% increased risk of suffering a heart attack"
    - What is probability of heart attack if I you don't take it? Is that big or small?
    - Actual numbers: less than 1 in 1000 "extra" heart attacks vs. baseline frequency.
      - There is a risk, but "24%" is an exaggeration.
    - "Health-scare stories often arise because their authors simply don't understand numbers."
      - Or it could be that they do understand, but media wants to "sensationalize" mundane news.
  - Bonus slides: more "Calling Bullshit" course examples on "political" issues:
    - Global warming, vaccines, gun violence, taxes.

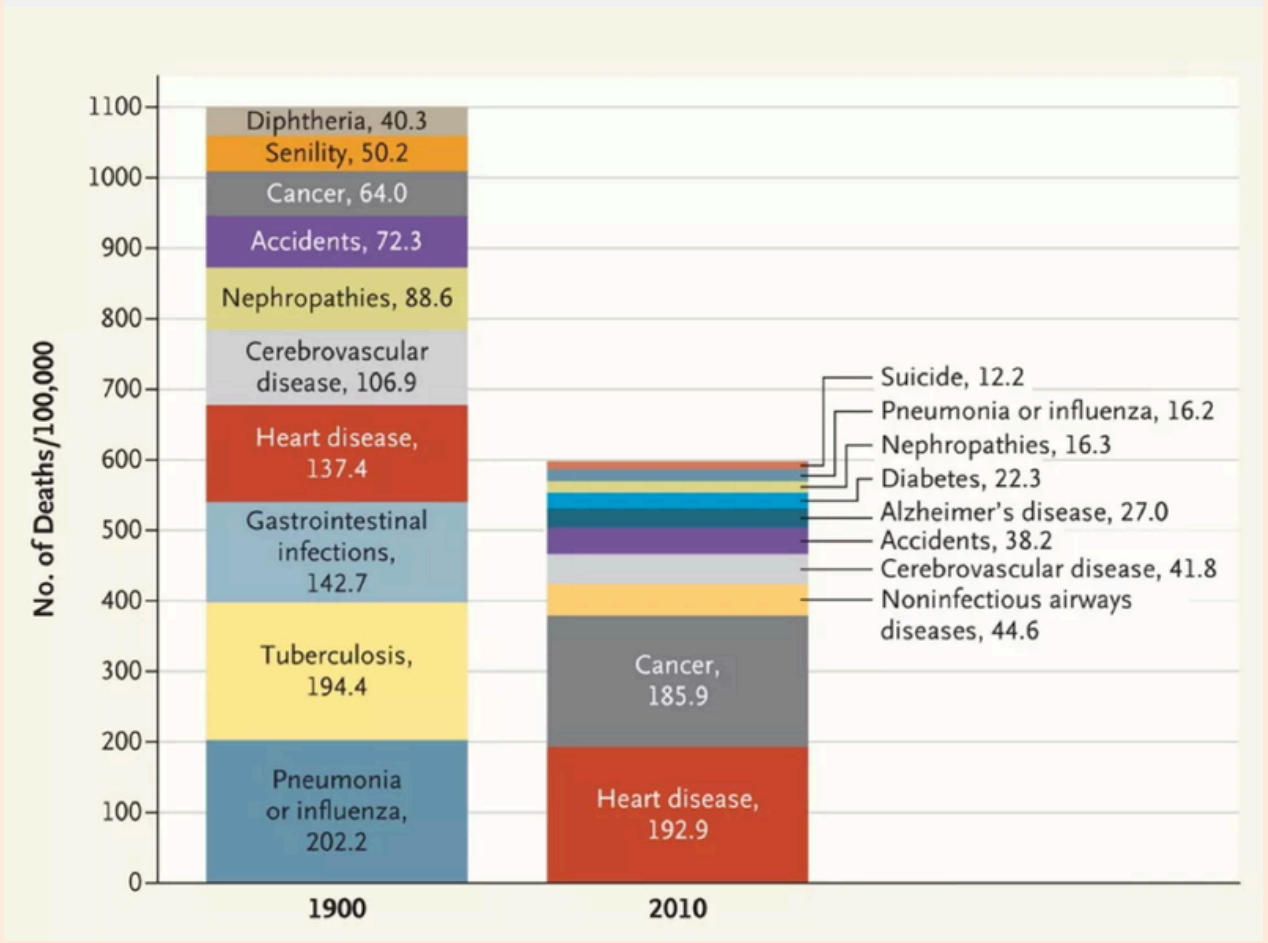
Data Cleaning and the Duke Cancer Scandal

- See the Duke cancer scandal:
  - http://www.nytimes.com/2011/07/08/health/research/08genes.html?_r=2&hp

- Basic sanity checks for data cleanliness show problems in these (and many other) studies:
  - E.g., flipped labels, off-by-one mistakes, switched columns etc.
  - https://arxiv.org/pdf/1010.1092.pdf

Histogram

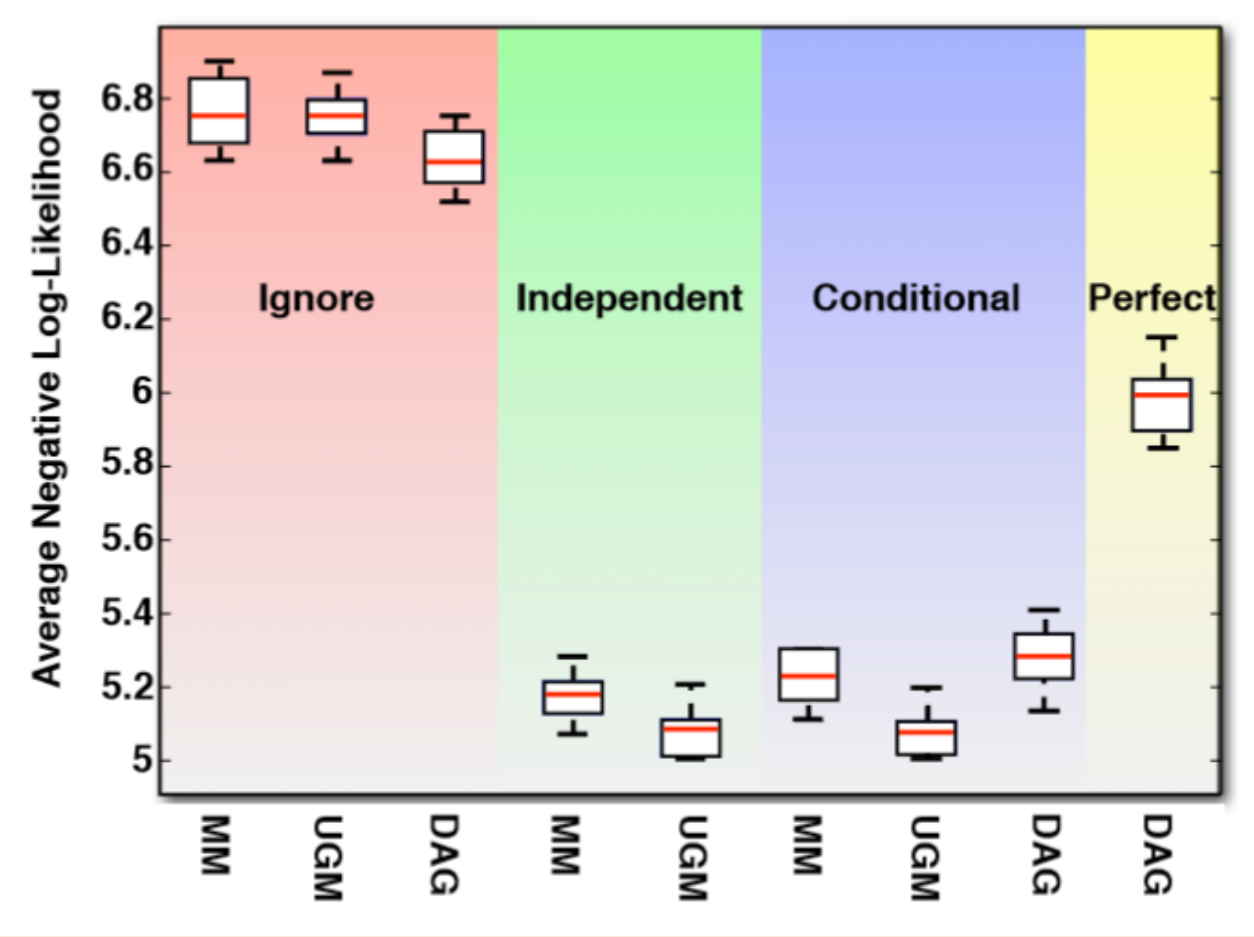
- Histogram with grouping:



http://www.vox.com/2016/5/10/11608064/americans-cause-of-death

Box Plots

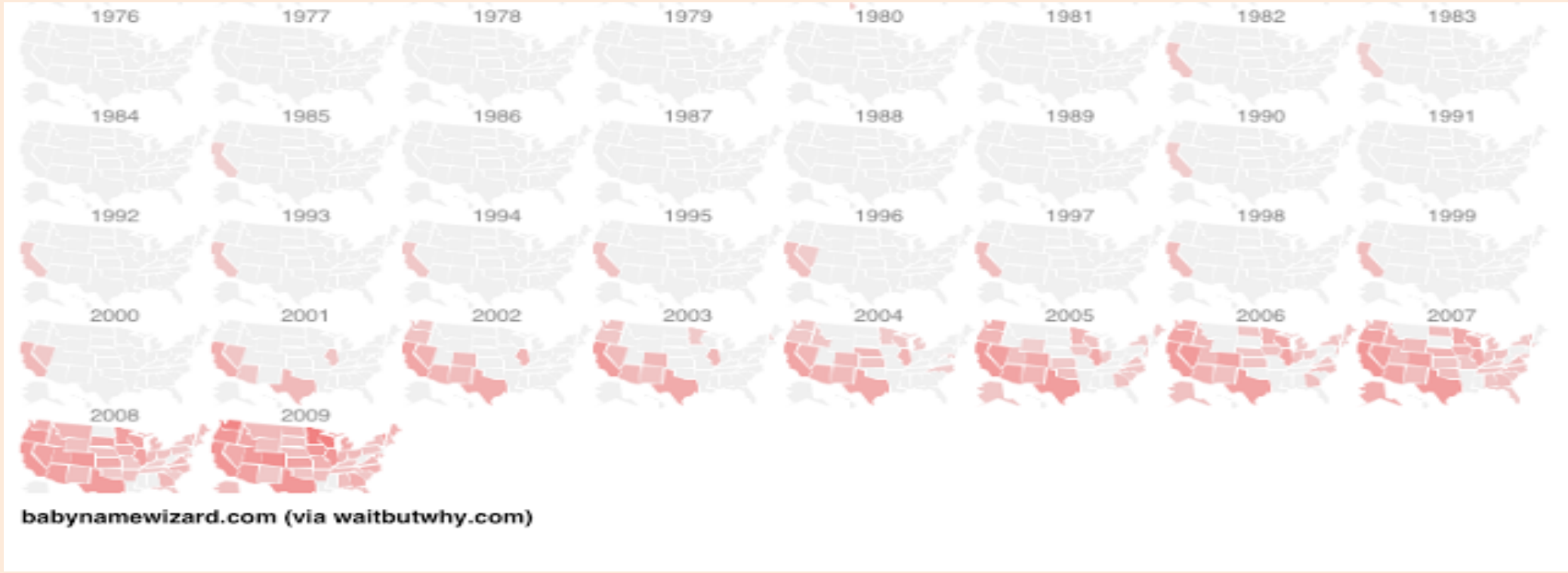
- Box plot with grouping:



Map Coloring

- Color/intensity can represent feature of region.

Popularity of naming baby “Evelyn” over time:



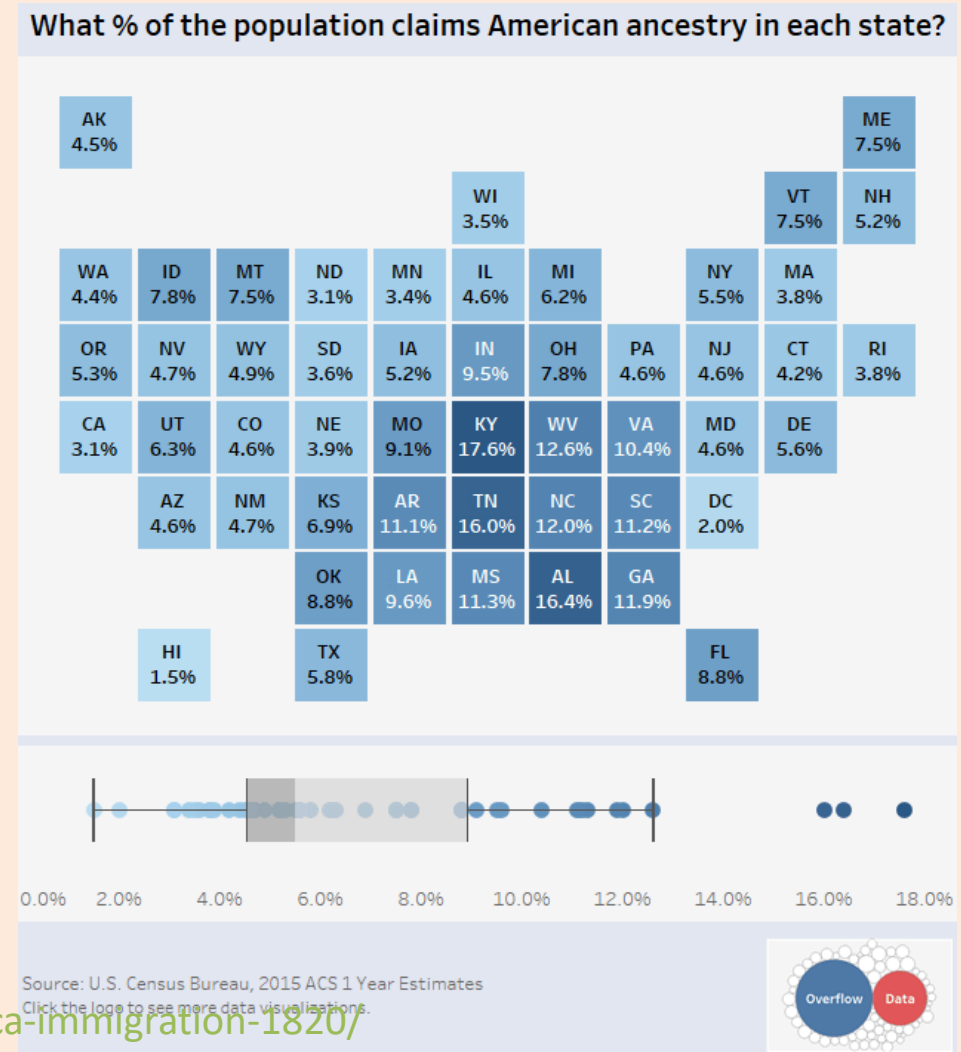
But not very good if some regions are very small.

[Canadian Income Mobility]

http://waitbutwhy.com/2013/12/how-to-name-baby.html

Map Coloring

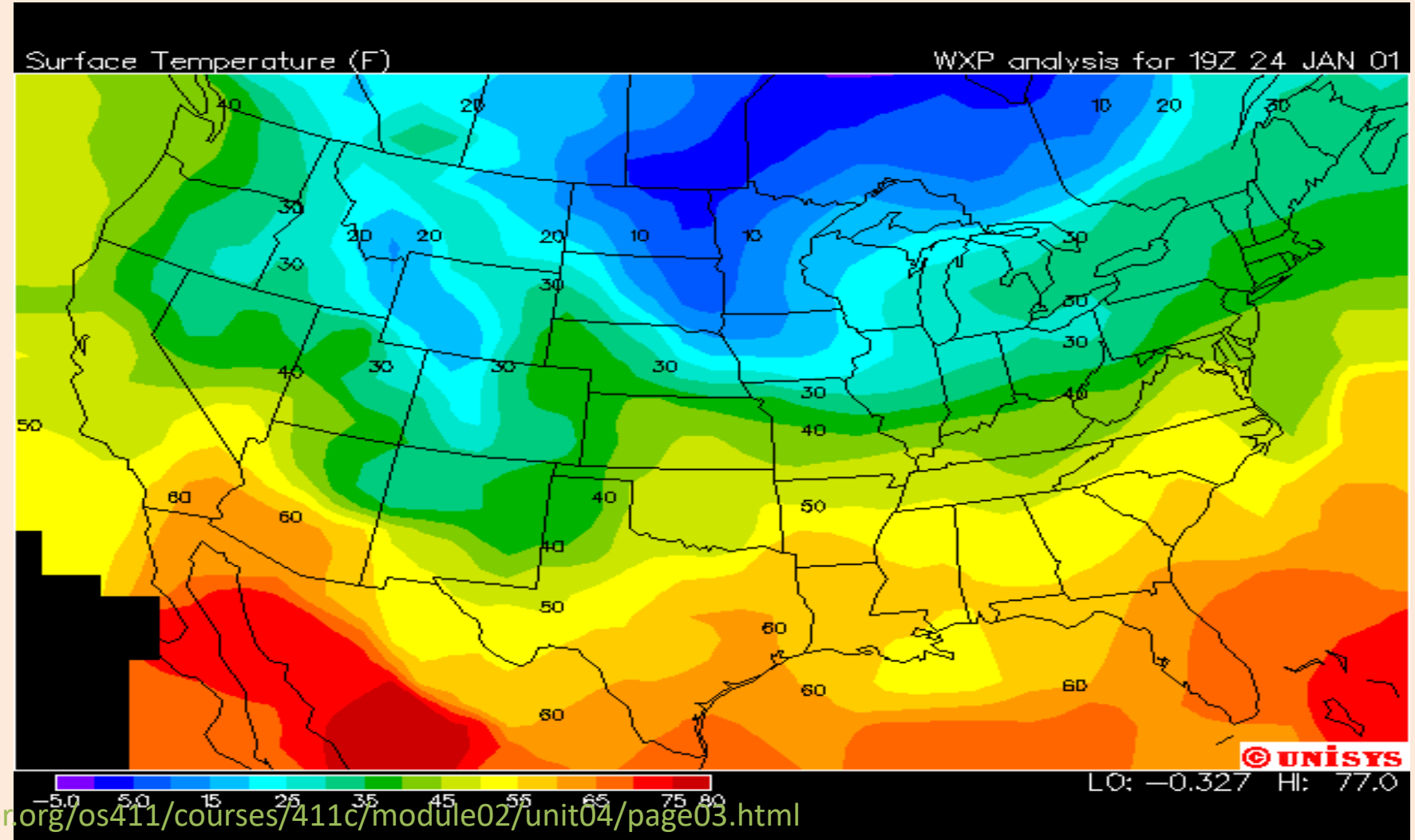
- Variation just uses fixed-size blocks and tries to arrange geographically:



http://www.visualcapitalist.com/america-immigration-1820/

Contour Plot

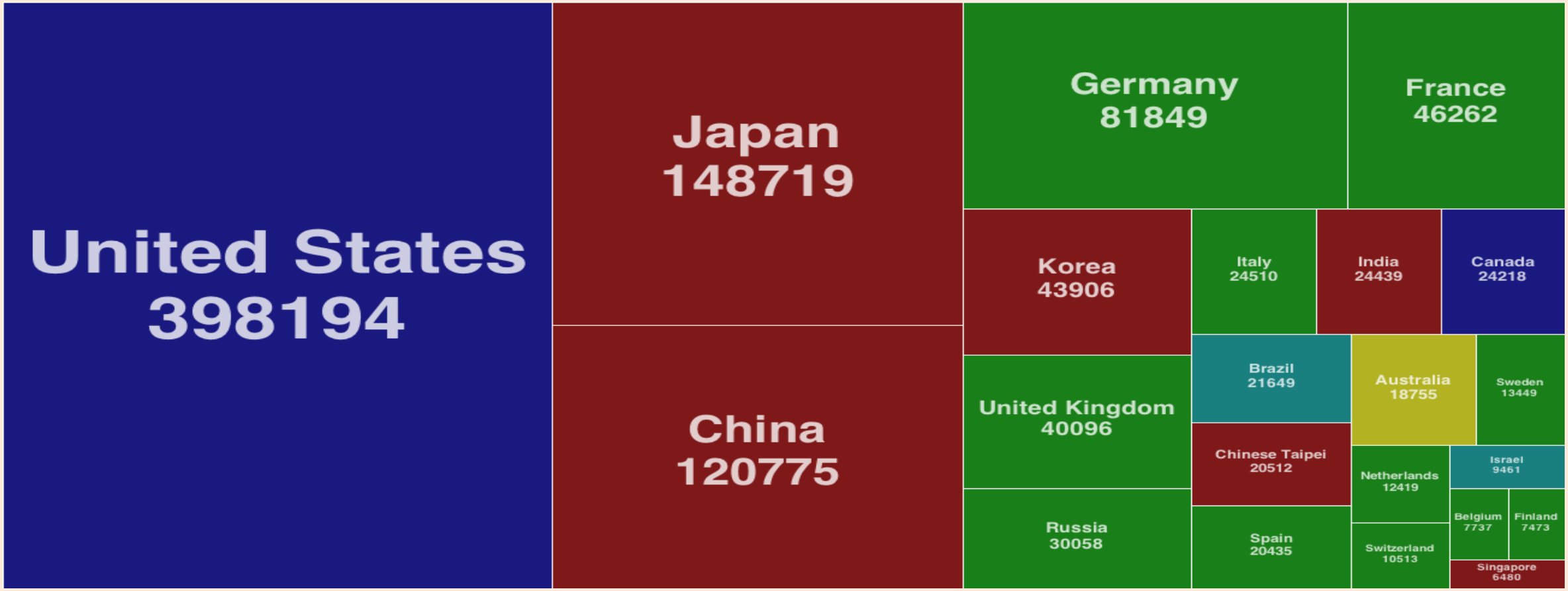
- Colour visualizes ‘z’ as we vary ‘x’ and ‘y’.



http://www.shodor.org/os411/courses/411c/module02/unit04/page03.html

Treemaps

- Area represents attribute value:



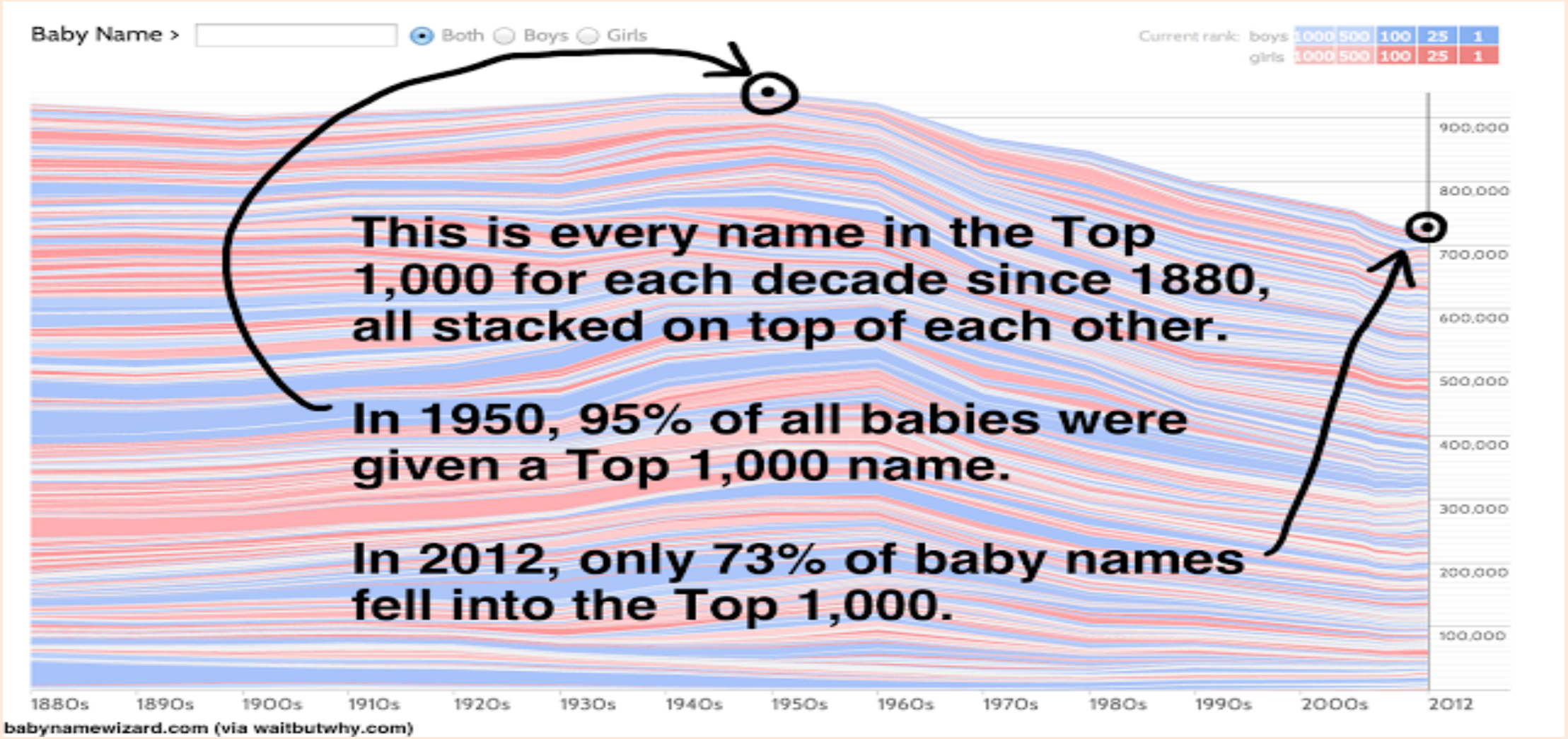
http://mjwest.org/vis/treemaps/

Cartogram

- Fancier version of treemaps:

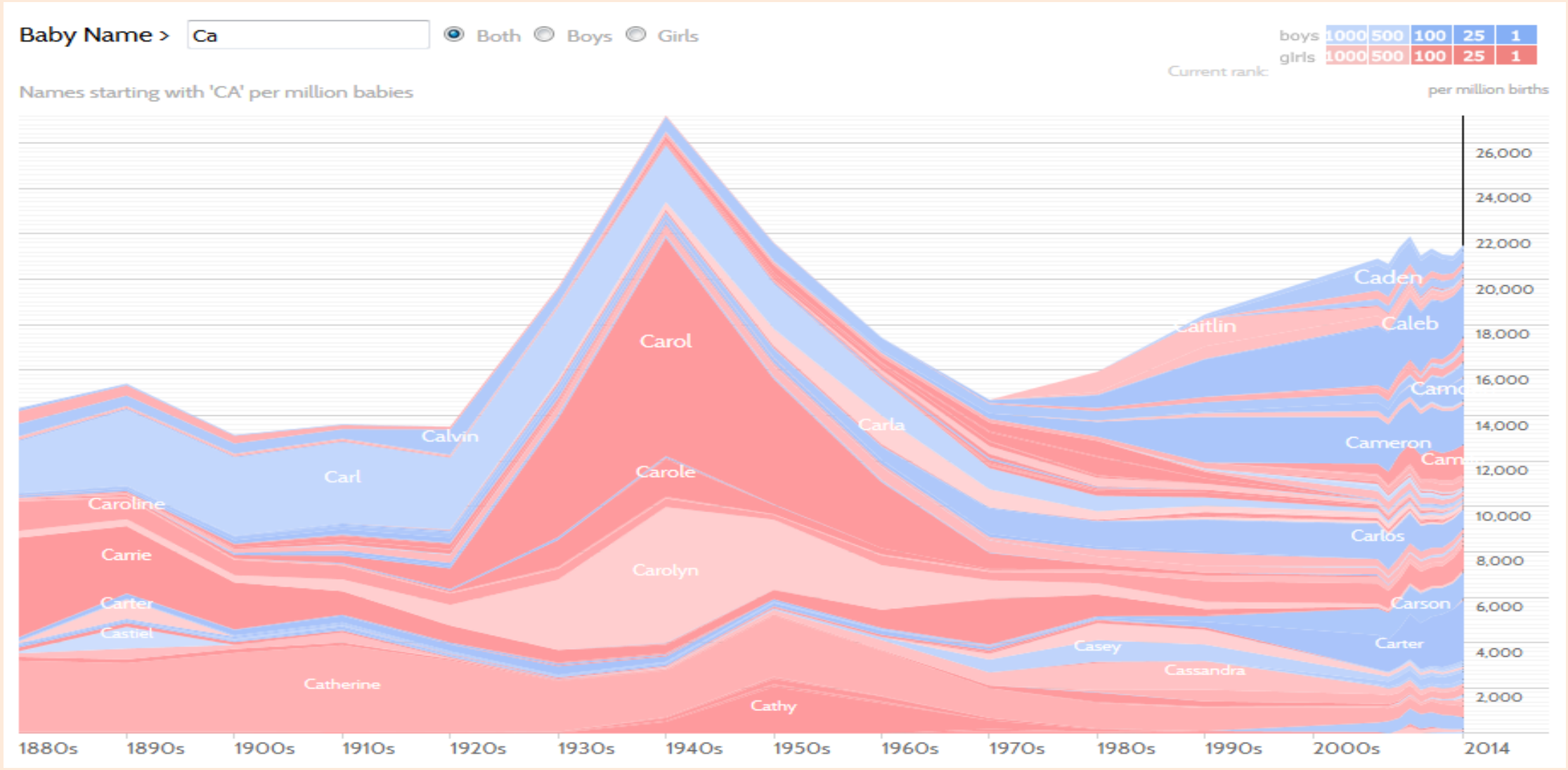
http://www-personal.umich.edu/~mejn/cartograms/

Stream Graph



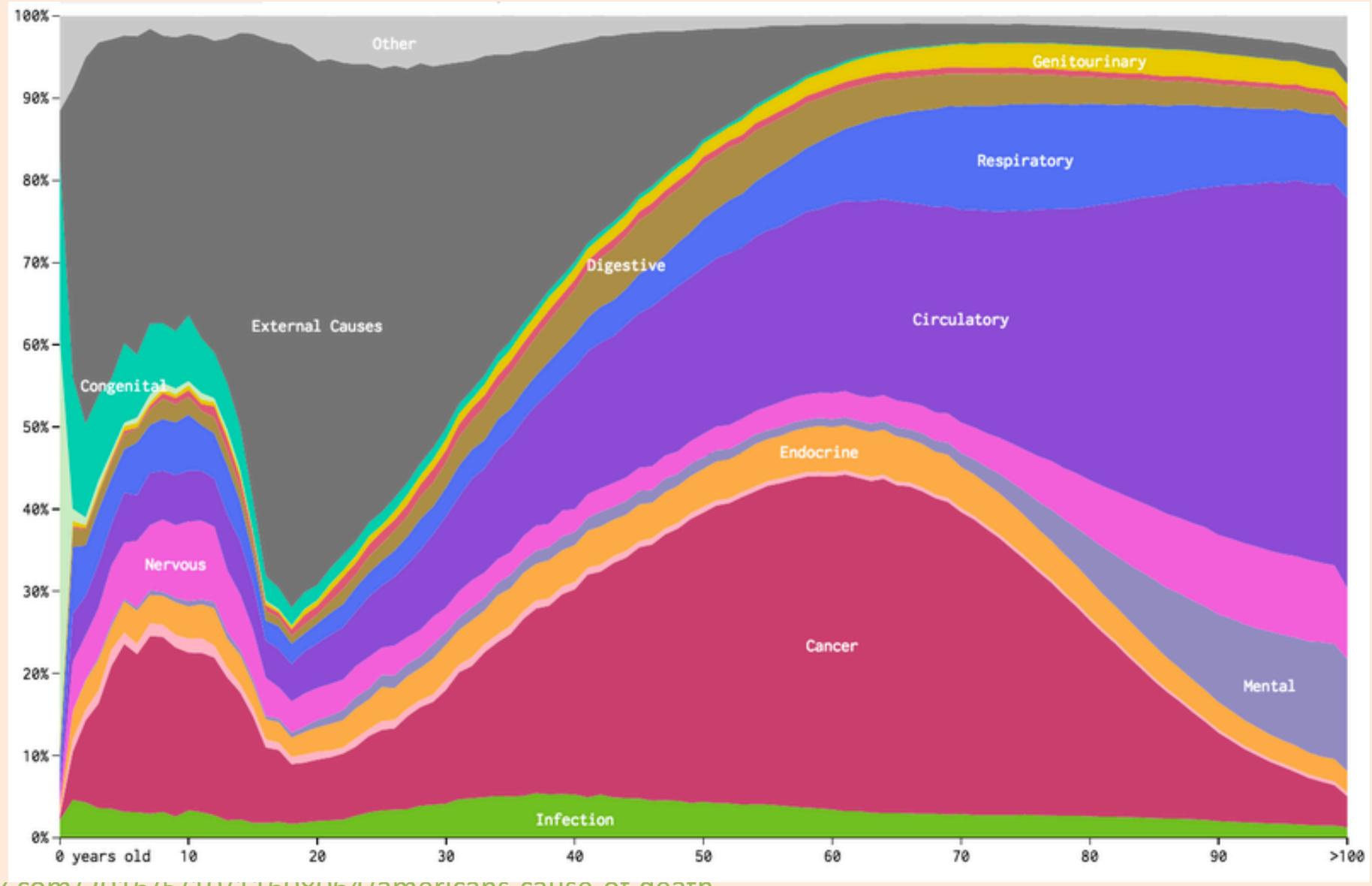
http://waitbutwhy.com/2013/12/how-to-name-baby.html

Stream Graph



http://www.babynamewizard.com/d3js-voyager/popup.html#prefix=ca&sw=both&exact=false

Stream Graph



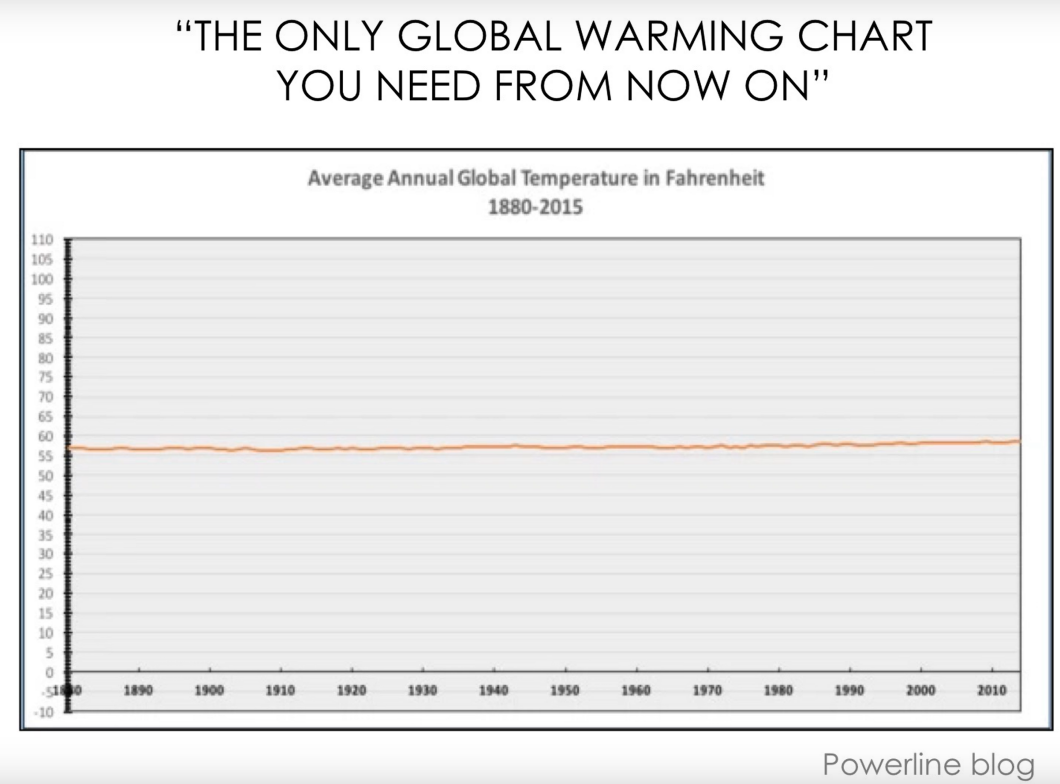
http://www.vox.com/2016/5/10/11608064/americans-cause-of-death

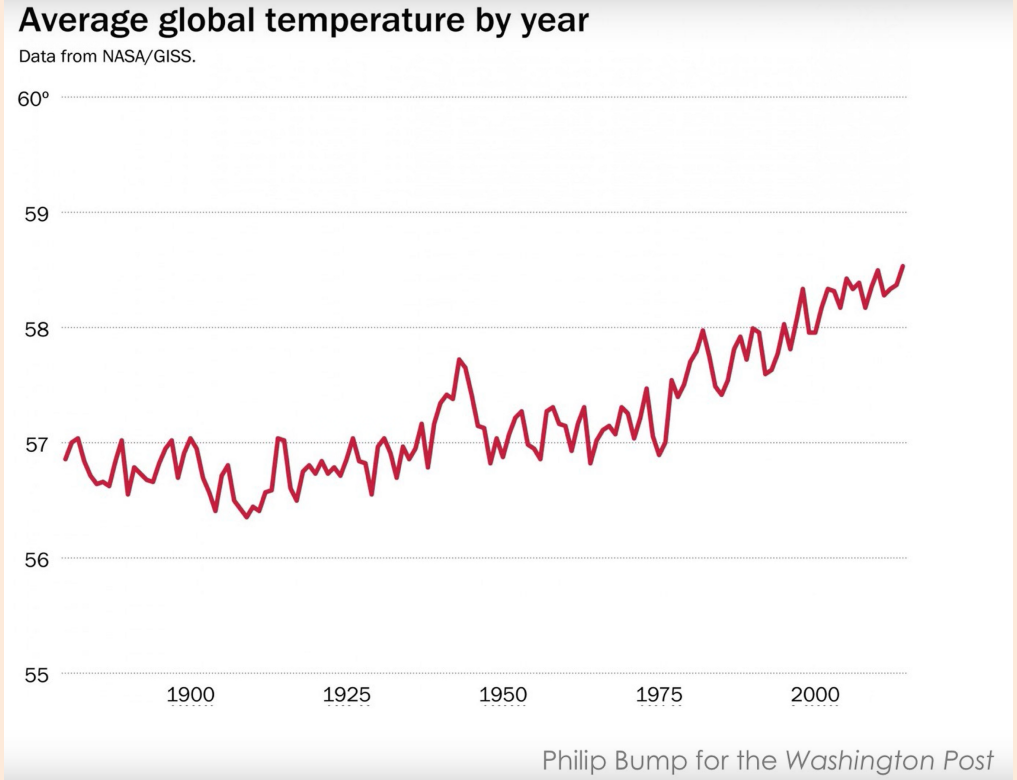
Videos and Interactive Visualizations

- For data recorded over time, videos can be useful:
  - Map colouring over time.

- There are also lots of neat interactive visualization methods:
  - Sale date for most expensive paintings.
  - Global map of wind, weather, and oceans.
  - Many examples here.

More Mis-Leading Axes from “Calling Bullshit”





https://www.youtube.com/watch?v=9pNWVMxaFuM

More Mis-Leading Axes from “Calling Bullshit”

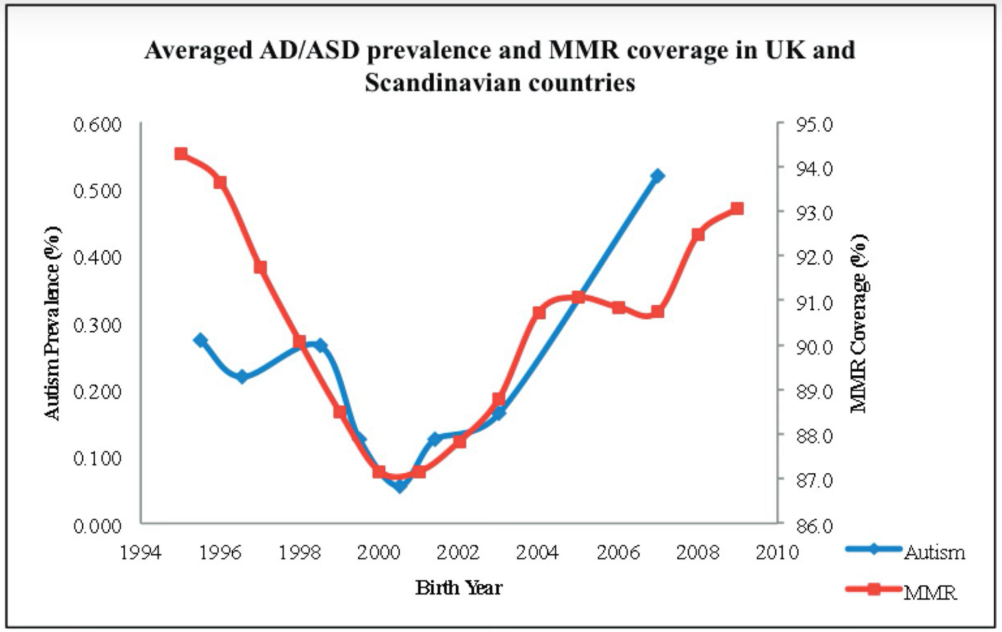
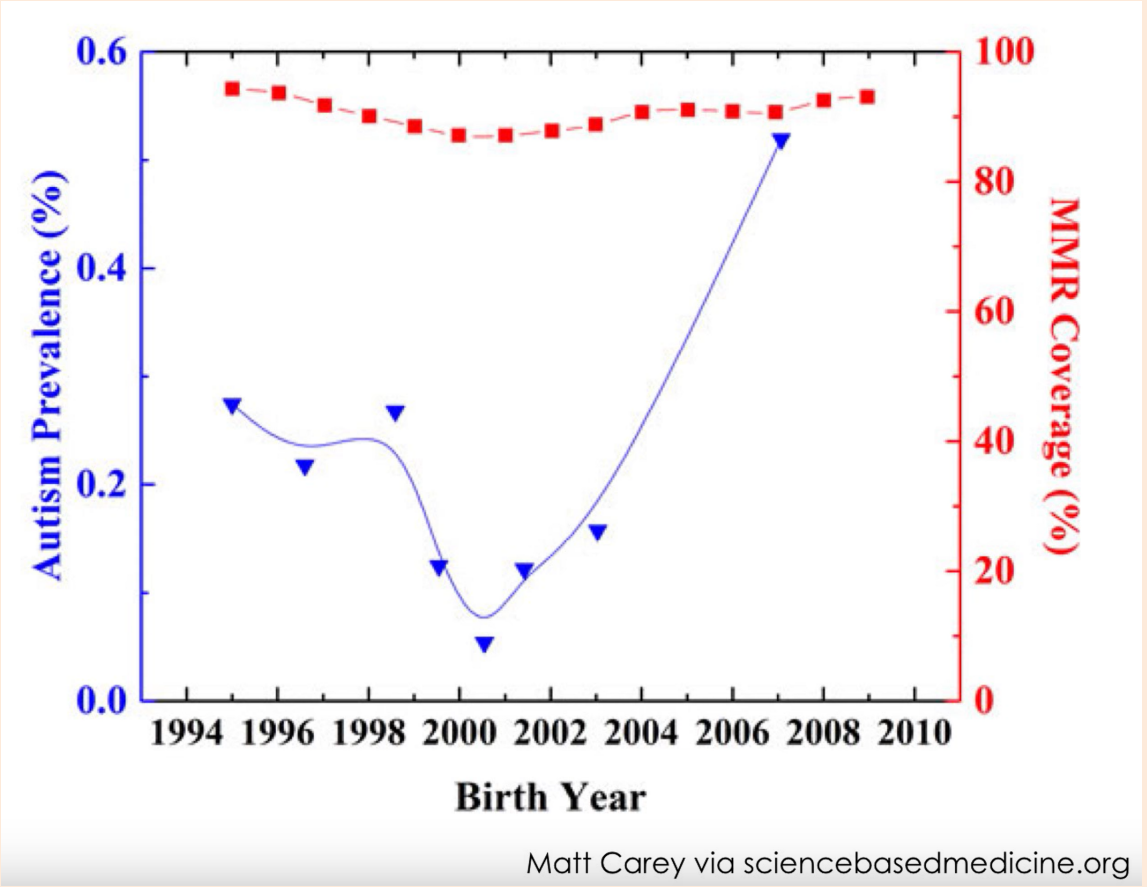


Figure 1-Averaged AD/ASD prevalence and MMR coverage in UK, Norway and Sweden. Both MMR and AD/ASD data are normalized to the maximum coverage/prevalence during the time period of this analysis.

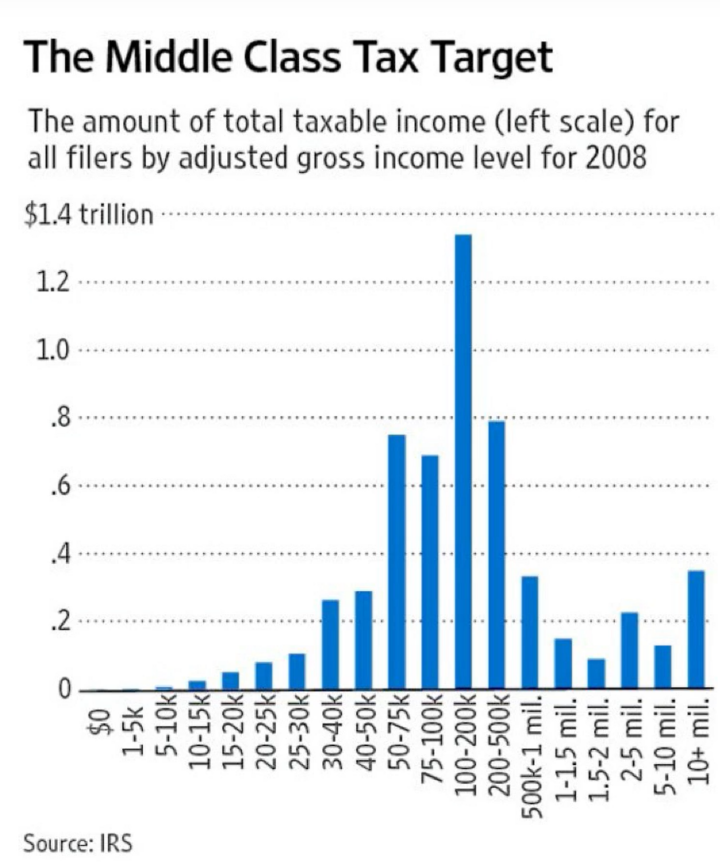
Diesher *et al.* 2015 *Issues in Law and Medicine*



Matt Carey via sciencebasedmedicine.org

https://www.youtube.com/watch?v=9pNWVMxaFuM

More Mis-Leading Axes from “Calling Bullshit”



“The rich, in short, aren't nearly rich enough to finance Mr. Obama's entitlement state ambitions—even before his health-care plan kicks in.

So who else is there to tax? Well, in 2008, there was about $5.65 trillion in total taxable income from all individual taxpayers, and most of that came from middle income earners. The nearby chart shows the distribution, and the big hump in the center is where Democrats are inevitably headed for the same reason that Willie Sutton robbed banks.”

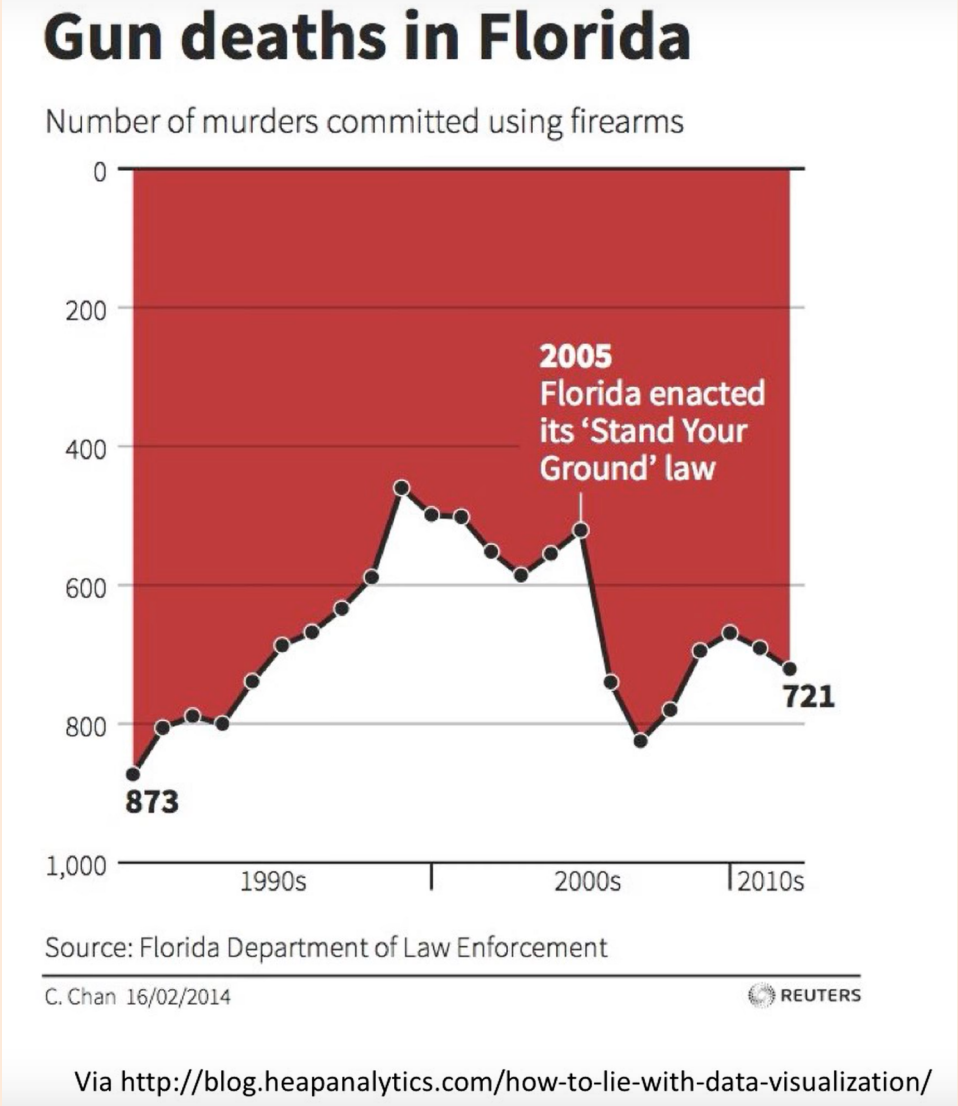
-The Wall Street Journal
April 17, 2011

- Look at the histogram bin widths.

https://www.youtube.com/watch?v=zAg1wsYfwsM

More Mis-Leading Axes from “Calling Bullshit”

- Axis is upside down.
- Looks like law makes murder go down, but number of murders go up!

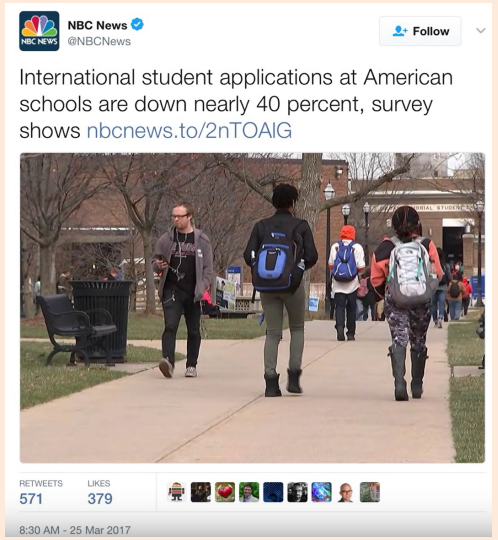


Via http://blog.heapanalytics.com/how-to-lie-with-data-visualization/

https://www.youtube.com/watch?v=9pNWVMxaFuM

More Mis-Leading Axes from “Calling Bullshit”

- Calling BS gives this as another example:



- Actual numbers don’t say much of anything:

Key findings of the survey include:

-- 39% of responding institutions reported a decline in international applications, 35% reported an increase, and 26% reported no change in applicant numbers.

- 39% vs. 35% (without sizes) doesn’t mean “down nearly 40 percent”.
  - Data can be used in mis-leading ways to “push agendas”.
  - Even by reputed sources.
  - Even if you agree with the message.

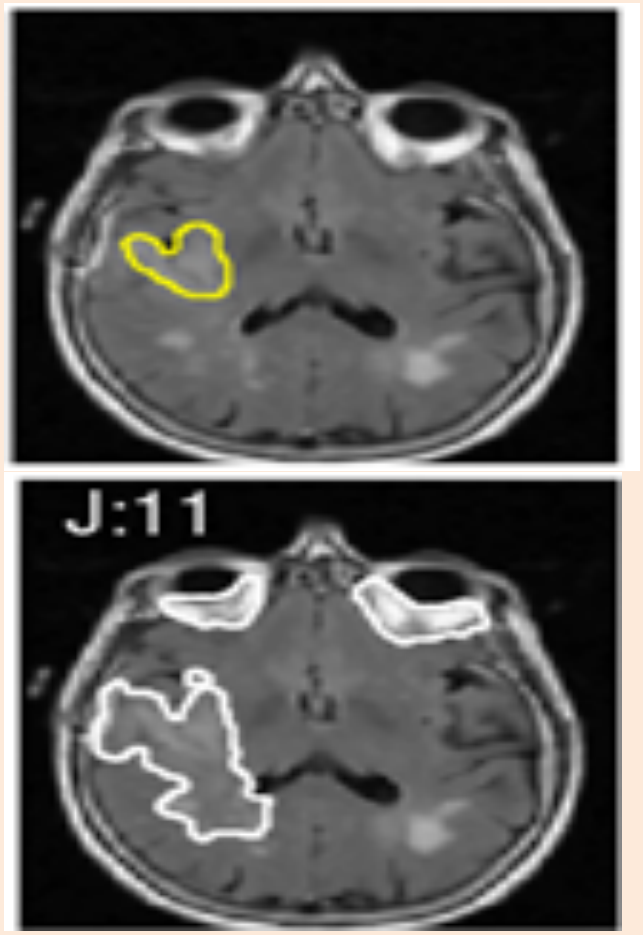
https://www.youtube.com/watch?v=-Mtmi8smpfo

Hamming Distance vs. Jaccard Coefficient

A	B
1	0
1	0
1	0
0	1
0	1
1	0
0	0
0	0
0	1

- These vectors agree in 2 positions.
  - Normalizing Hamming distance by vector length, similarity is 2/9.
- If we're really interested in predicting 1s, we could find set of 1s in both and compute Jaccard:
  - A -> {1,2,3,6}, B -> {4,5,9}
  - No intersection so Jaccard similarity is actually 0.

Hamming Distance vs. Jaccard Coefficient



- Let's say we want to find the tumour in an MR image.
- We have an expert label (top) and a prediction from our ML system (bottom).

- The normalized Hamming distance between the predictions at each pixel is 0.91. This sounds good, but since there are so many non-tumour pixels this is misleading.
- The ML system predicts a much bigger tumour so hasn't done well. The Jaccard coefficient between the two sets of tumour pixels is only 0.11 so reflects this.

Coupon Collecting

- Consider trying to collect 50 uniformly-distributed states, drawing at random.
- The probability of getting a new state if there ‘x’ states left: p=x/50.
- So expected number of samples before next “success” (getting a new state) is 50/x.

  (mean of geometric random variable with p=x/50)
- So the expected number of draws is the sum of 50/x for x=1:50.
- For ‘n’ states instead of 50, summing until you have all ‘n’ gives:

$$\sum_{i=1}^{n} \frac{n}{i} = n \sum_{i=1}^{n} \frac{1}{i} \leq n(1 + \log(n)) = O(n \log n)$$

Huge Datasets and Parallel/Distributed Computation

- Most sufficient statistics can be computed in linear time.
- For example, the mean of ‘n’ numbers is computed as:

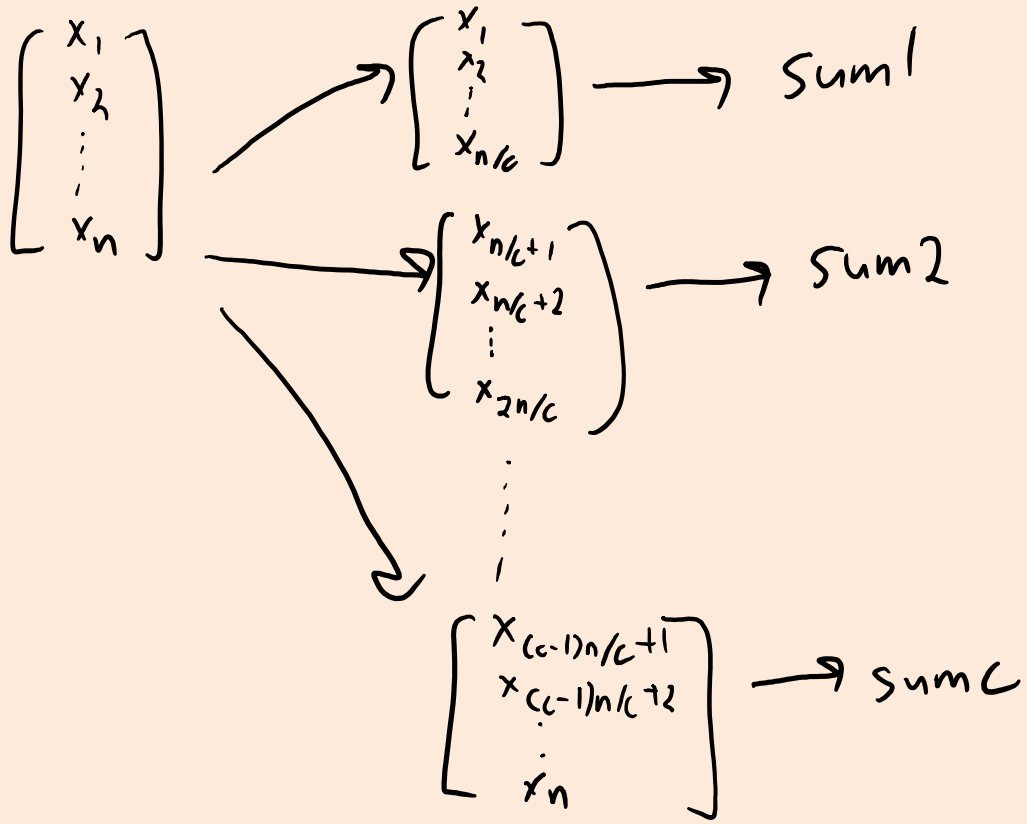
$$\text{mean}(x_1, x_2, x_3, \cdots, x_n) = \frac{x_1 + x_2 + x_3 + \cdots + x_n}{n}$$

- This costs O(n), which is great.

- But if ‘n’ is really big, we can go even faster with parallel computing…

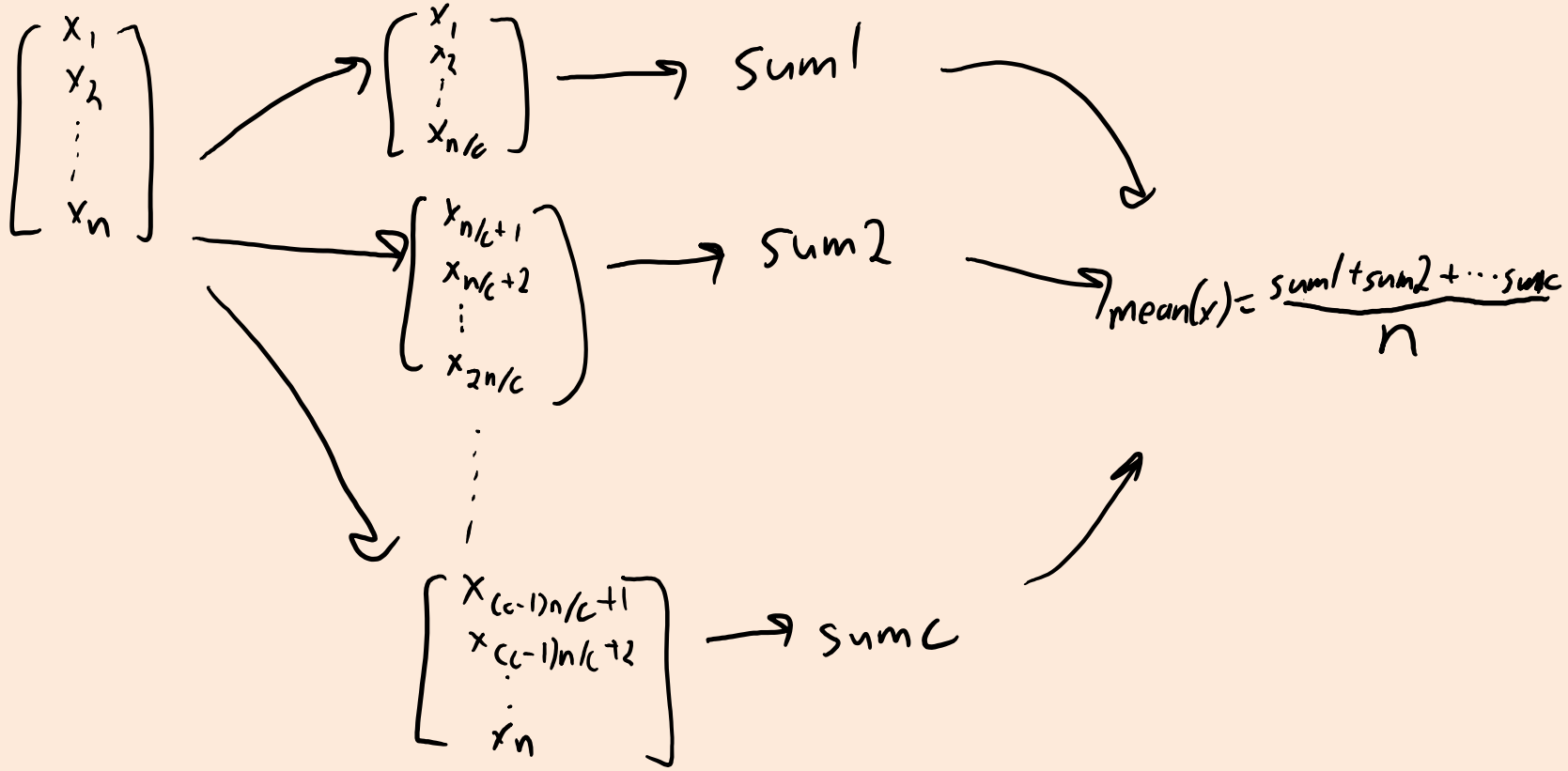
Huge Datasets and Parallel/Distributed Computation

- Computing the mean with multiple cores:
  - Each of the ‘c’ cores computes the sum of O(n/c) of the data:



Huge Datasets and Parallel/Distributed Computation

- Computing the mean with multiple cores:
  - Each of the ‘c’ cores computes the sum of O(n/c) of the data:
  - Add up the ‘c’ results from each core to get the mean.



Huge Datasets and Parallel/Distributed Computation

- Computing the mean with multiple cores:
  - Each of the ‘c’ cores computes the sum of O(n/c) of the data.
  - Add up the ‘c’ results from each core to get the mean.
  - Cost is only O(n/c + c), which can be much faster for large ‘n’.

- This assumes cores can access data in parallel (not always true).
- Can reduce cost to O(n/c) by having cores write to same register.
  - But need to “lock” the register and might effectively cost O(n).

Huge Datasets and Parallel/Distributed Computation

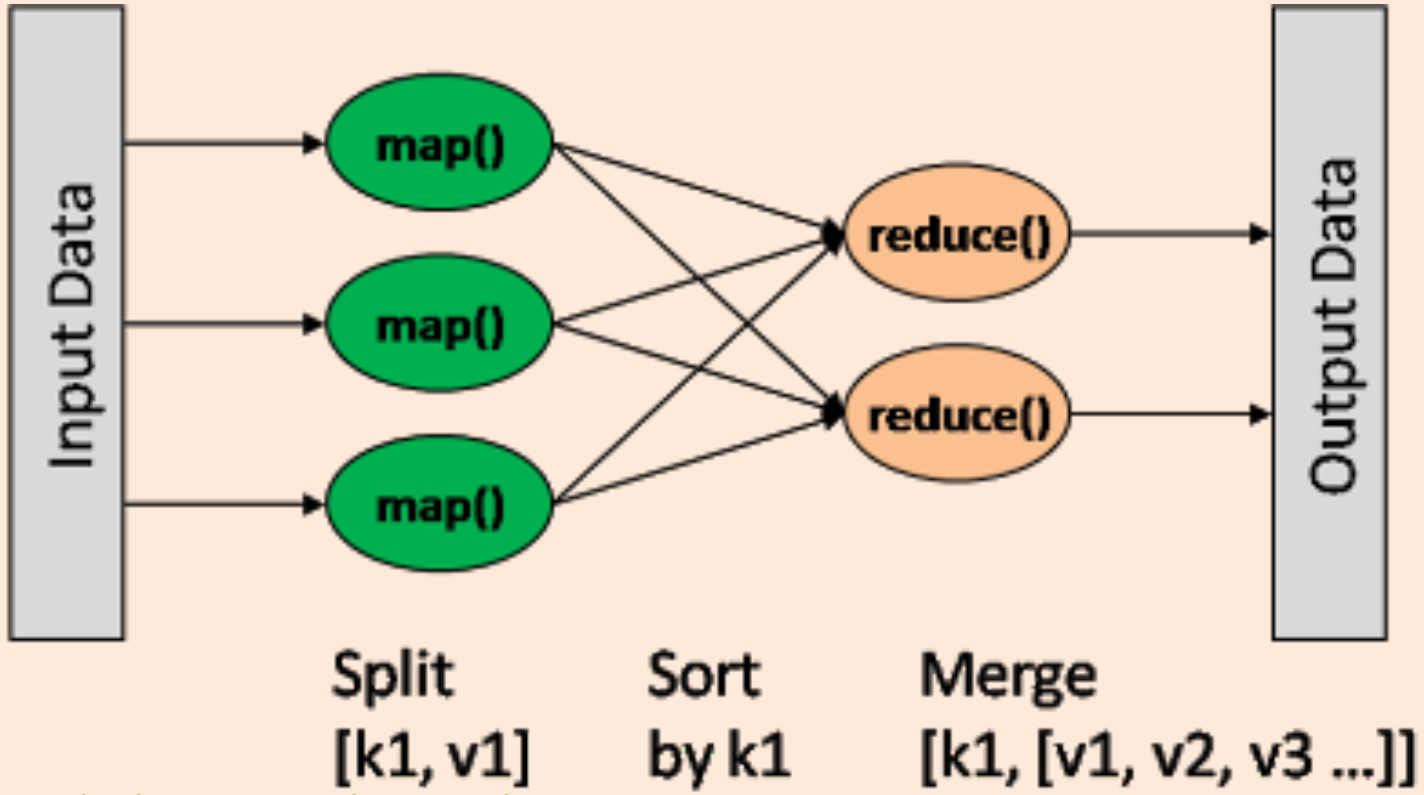
- Sometimes ‘n’ is so big that data can’t fit on one computer.

- In this case the data might be distributed across ‘c’ machines:
  - Hopefully, each machine has O(n/c) of the data.

- We can solve the problem similar to the multi-core case:
  - “Map” step: each machine computes the sum of its data.
  - “Reduce” step: each machine communicates sum to a “master” computer, which adds them together and divides by ‘n’.

Huge Datasets and Parallel/Distributed Computation

- Many problems in DM and ML have this flavour:
  - "Map" computes an operation on the data on each machine (in parallel).
  - "Reduce" combines the results across machines.



https://dzone.com/articles/how-hadoop-mapreduce-works

Huge Datasets and Parallel/Distributed Computation

- Many problems in DM and ML have this flavour:
  - "Map" computes an operation on the data on each machine (in parallel).
  - "Reduce" combines the results across machines.
  - These are standard operations in parallel libraries like MPI.

- Can solve many problems almost 'c' times faster with 'c' computers.

- To make it up for the high cost communicating across machines:
  - Assumes that most of the computation is in the "map" step.
  - Often need to assume data is already on the computers at the start.

Huge Datasets and Parallel/Distributed Computation

- Another challenge with “Google-sized” datasets:
  - You may need so many computers to store the data, that it’s inevitable that some computers are going to fail.
- Solution to this is a distributed file system.

- Two popular examples are Google’s MapReduce and Hadoop DFS:
  - Store data with redundancy (same data is stored in many places).
    - And assume data isn’t changing too quickly.
  - Have a strategy for restarting “map” operations on computers that fail.
  - Allows fast calculation of more-fancy things than sufficient statistics:
    - Database queries and matrix multiplications.