CSE 332
Introduction to Visualization

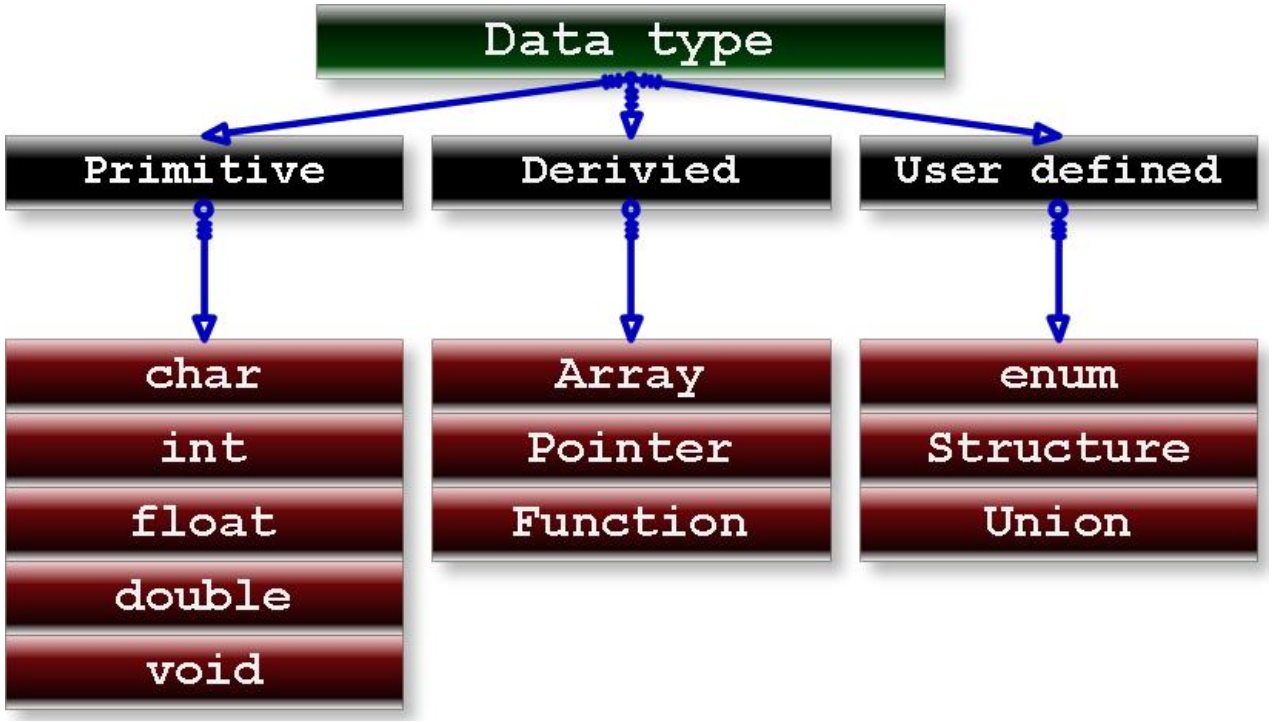
Data Types & basic Applications

Klaus Mueller

Computer Science Department
Stony Brook University

Lecture	Topic	Projects
1	Intro, schedule, and logistics	
2	Applications of visual analytics, data, and basic tasks	
3	Basic vis techniques for non-spatial data	Project 1 out
4	Data preparation and reduction	
5	Perception and cognition, visual design and aesthetics	
6	Foundations of statistics	
7	Introduction to D3	Project 2 out
8	Data types, notion of similarity and distance	
9	Data mining techniques: clusters, text, patterns, classifiers	
10	Data mining techniques: clusters, text, patterns, classifiers	
11	High-dimensional data, dimensionality reduction	
12	Computer graphics and volume rendering	Project 3 out
13	Techniques to visualize spatial (3D) data	
14	Scientific and medical visualization	
15	Scientific and medical visualization	
16	Non-photorealistic rendering	
17	Midterm	
18	Principles of interaction	Project 4 out
19	Visual analytics and the visual sense making process	
20	Correlation and causal modeling	
21	Big data: data reduction, summarization	
22	Visualization of graphs and hierarchies	
23	Visualization of text data	Project 5 out
24	Visualization of time-varying and time-series data	
25	Memorable visualizations, visual embellishments	
26	Evaluation and user studies	
27	Narrative visualization and storytelling	
28	Data journalism	

Data Types Every CS Person Knows



Data Types in Visual Analytics

Numerical

Categorical

Text

Time series

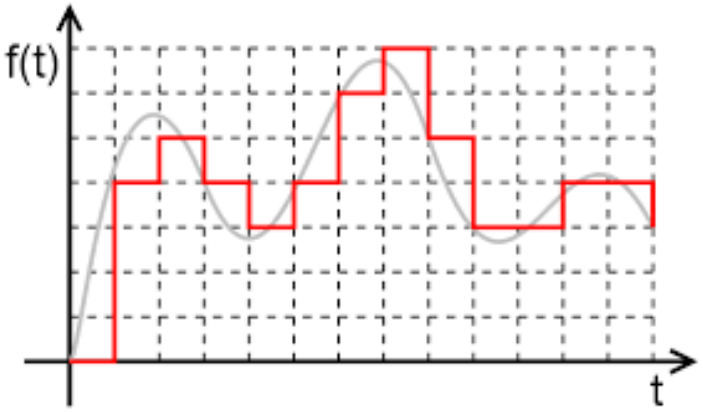
Graphs and networks

Hierarchies

Variables in Statistics

Numerical variables

- measure a **quantity** as a number
- like: ‘how many' or 'how much'
- can be continuous (grey curve)
- or discrete (red steps)



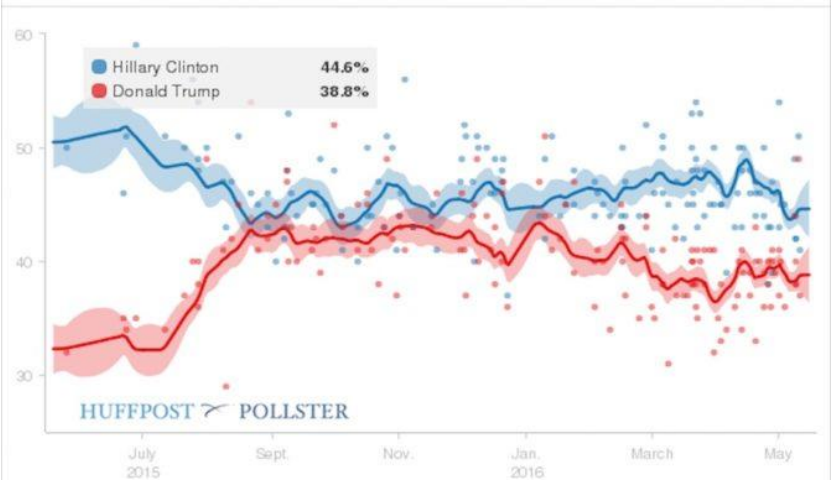
Categorical variables

- describe a **quality** or characteristic
- like: 'what type' or 'which category’
- can be ordinal = ordered, ranked (distances need not be equal)
  - clothing size, academic grades, levels of agreement
- or nominal = not organized into a logical sequence
  - gender, business type, eye color, brand

Numerical Variables

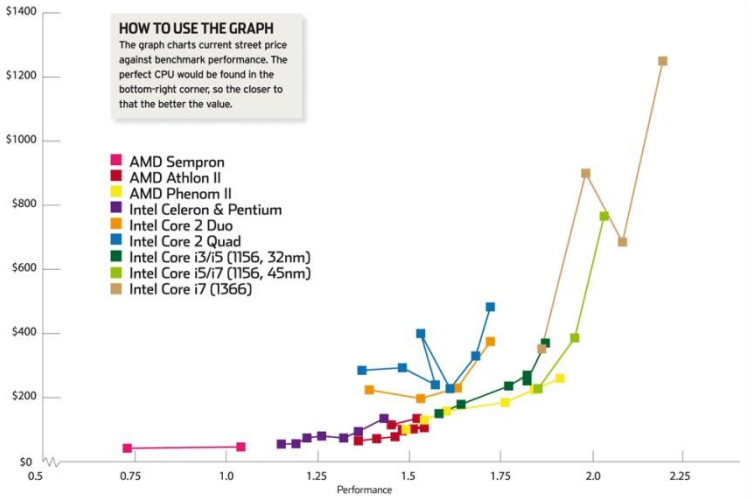
Most often the x-axis is ‘time’

- provides an intuitive & innate ordering of the data values
- the majority of people expect the x-axis to be ‘time’



But ‘time’ is not the only option

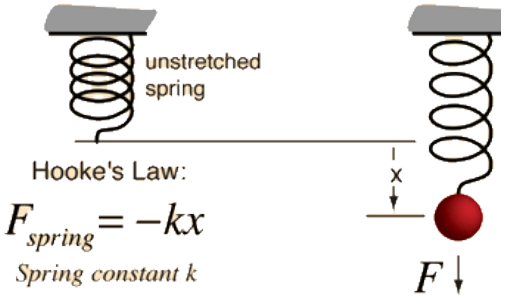
- engineers, statisticians, etc. will be receptive to this idea
- can you think of an example?

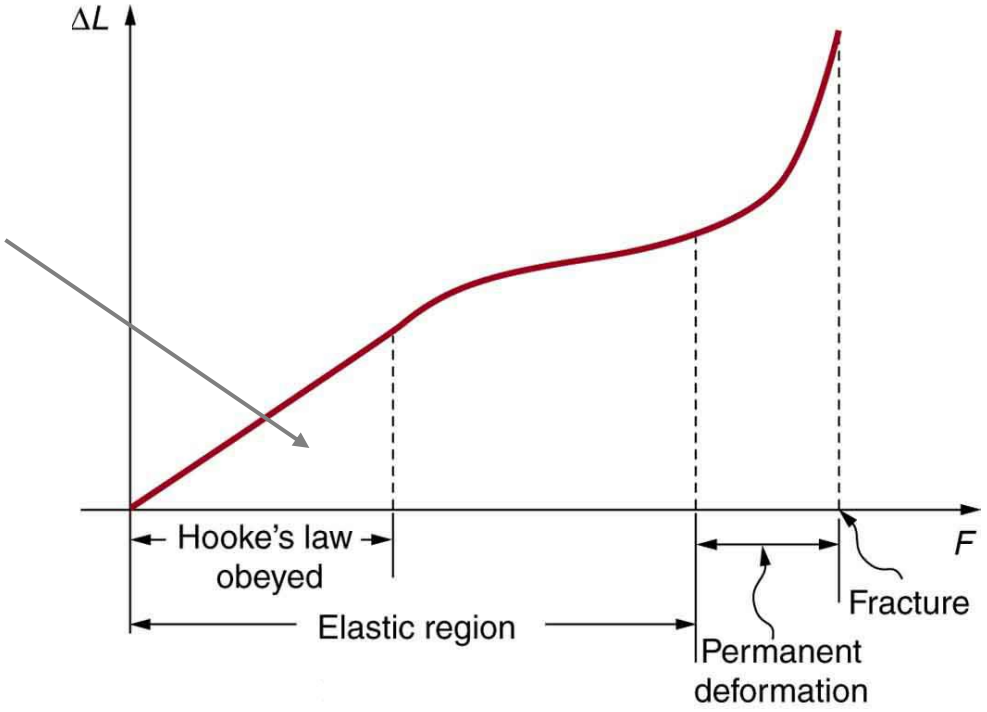


Numerical Variables

Another plot where ‘time’ is not the x-axis

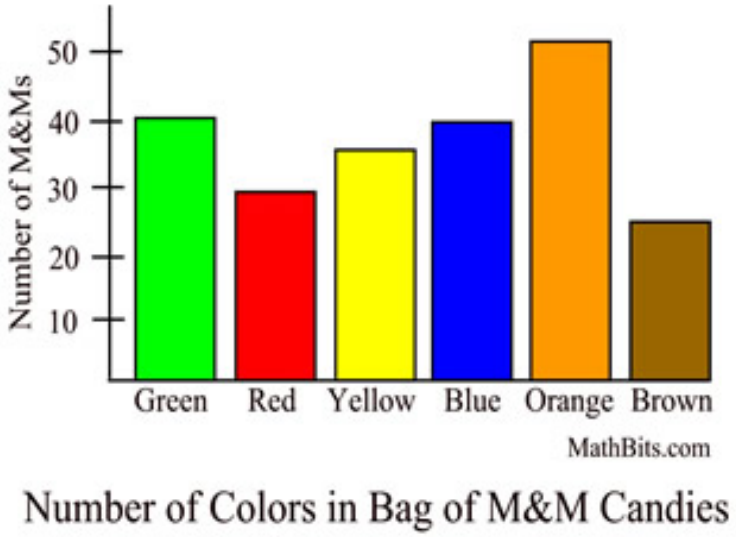
- from the engineering / physics domain
- in some sense, it tells a story



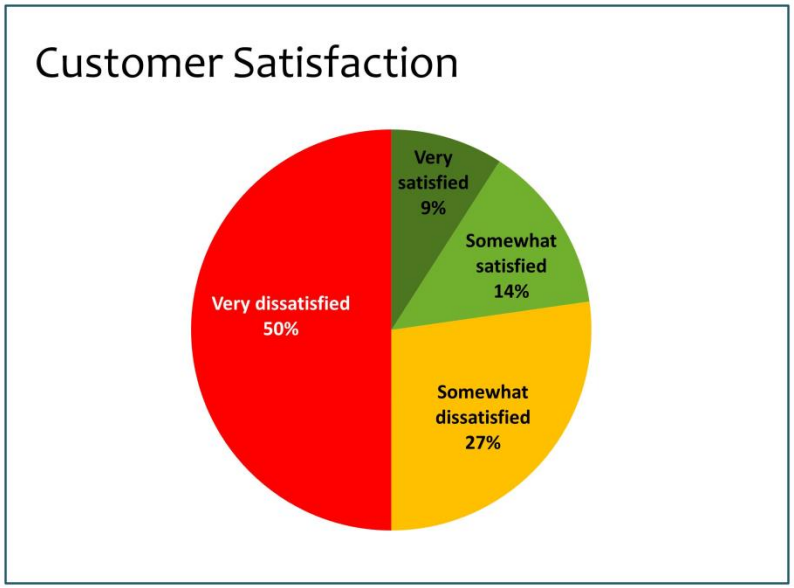


Categorical Variables

Usually plotted as bar charts or pie charts



??



??

nominal

ordinal

Numbers are Good

But not everything is expressed in numbers

- images
- video
- text
- web logs
- …

Need to do feature analysis to turn these abstract things into numbers

- then apply your analysis as usual
- but keep the reference to the original data so you can return to the native domain where the analysis problem originated

Image Data

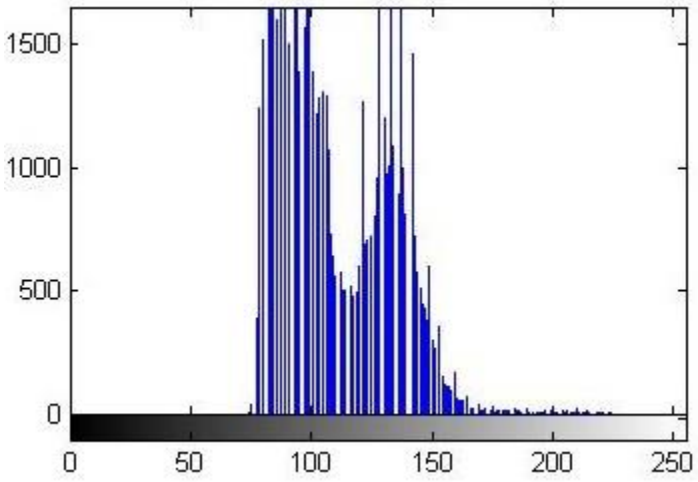
Characteristics

- array of pixels

Feature Analysis

- example: value histograms
- encode into a 256-D vector

histograms



[0, 0, 0, …., 10, …, 1200, …..]

Video data

Characteristics

- essentially a time series of images

Feature Analysis

- many of the image techniques apply but extension is non-trivial

Text Data

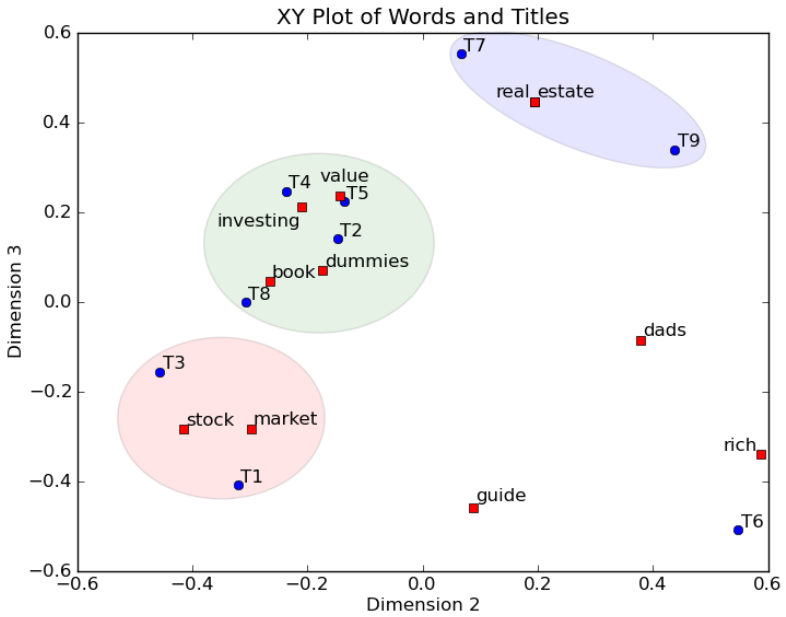
Create a term-document matrix

- turns text into a high-dimensional vector which can be compared
- use Latent Semantic Analysis (LSA) to derive a visualization

Index Words	Titles								
	T1	T2	T3	T4	T5	T6	T7	T8	T9
book			1	1					
dads						1			1
dummies		1						1	
estate							1		1
guide	1					1			
investing	1	1	1	1	1	1	1	1	1
market	1		1						
real							1		1
rich						2			1
stock	1		1					1	
value				1	1				

Term-Document Matrix

LSA

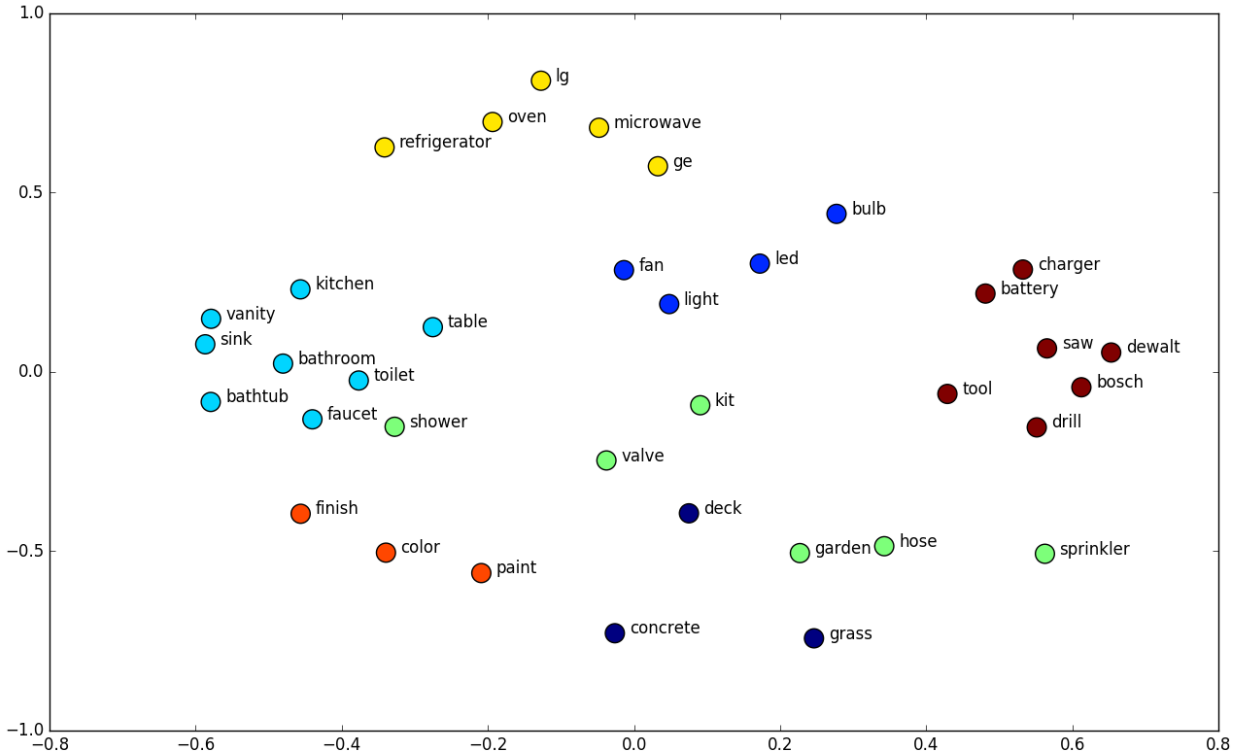


Word/document cluster

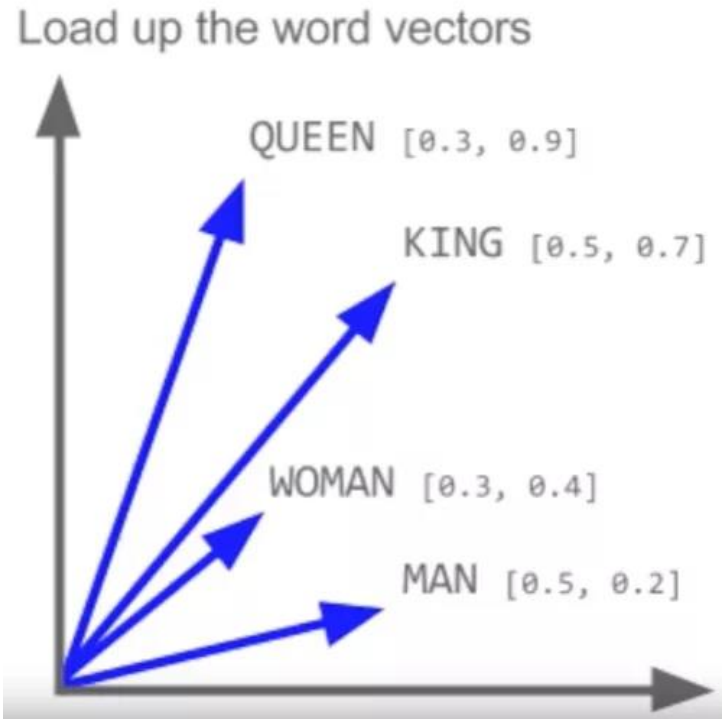
Word Embedding

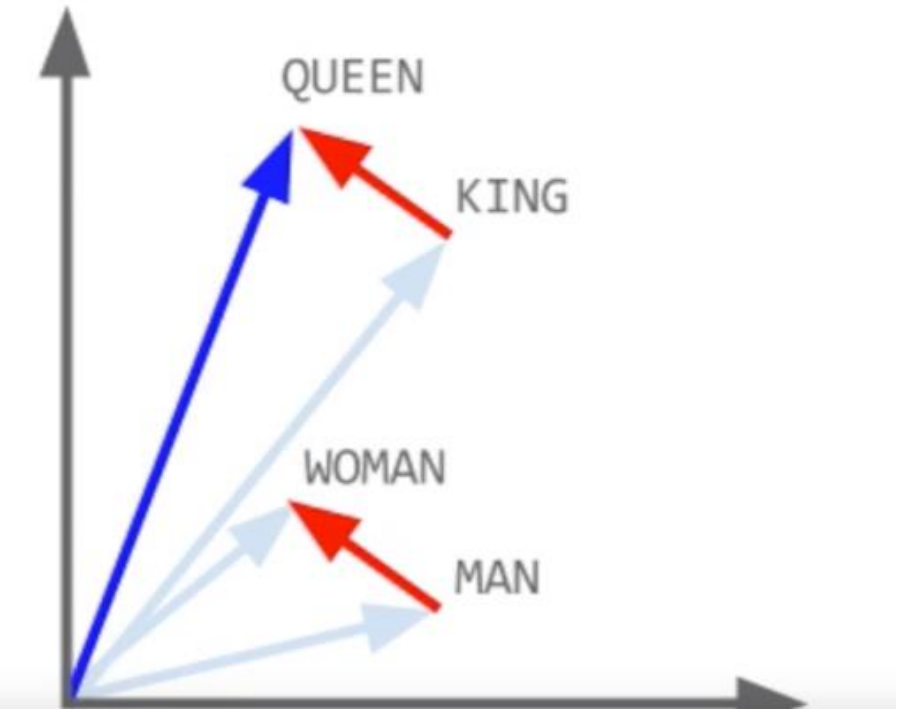
Train a shallow neural network (NN) on a corpus of text

- the NN weight vectors encode word similarity as a high-D vector
- use a 2D embedding technique to display



Word Embedding Algebra





gender = WOMAN – MAN
QUEEN = KING + gender

QUEEN = KING – MAN + WOMAN

Word Cloud

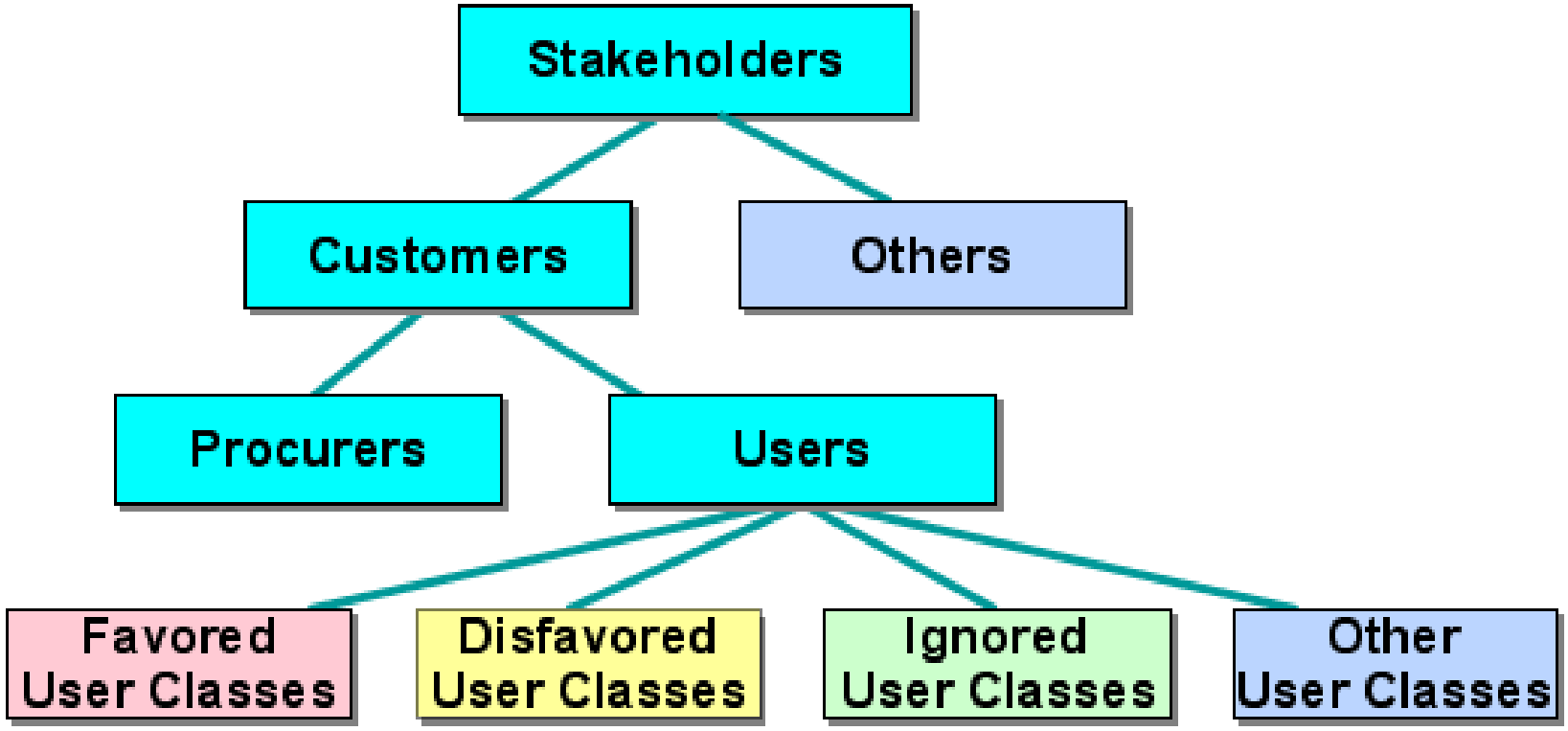
Maps the frequency of words in a corpus to size

https://www.jasondavies.com/wordcloud/

let's Look at Some Essential Graphical Representations

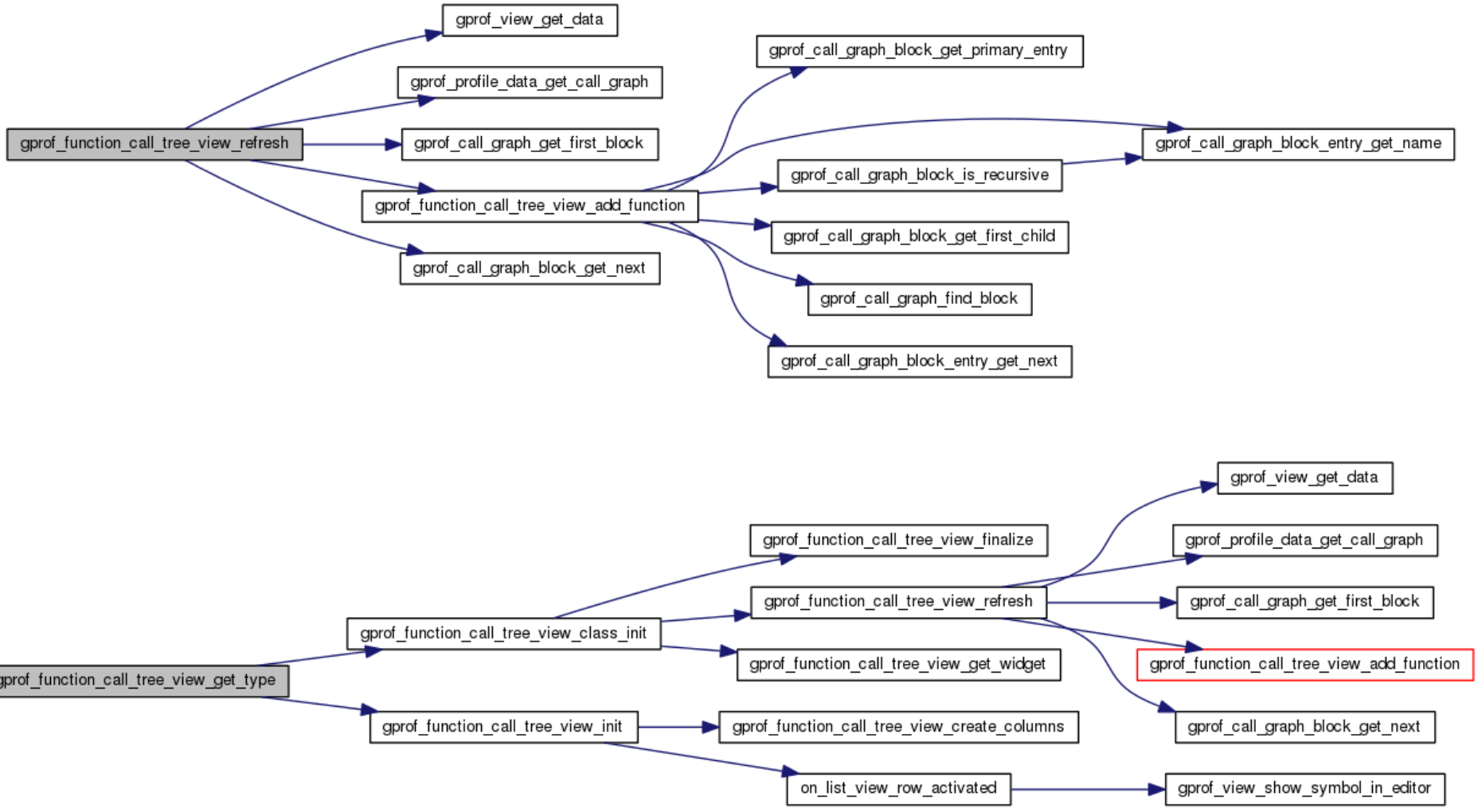
And Do Some Advertising for D3

Stakeholder Hierarchy



More Complex Stakeholder Hierarchy

Function Call Tree



Hierarchies

Questions you might have

- how large is each group of stakeholders (or function)?
  - tree with quantities
- what fraction is each group with respect to the entire group?
  - partition of unity
- how is information disseminated among the stakeholders (or functions)?
  - information flow
- how close (or distant) are the individual stakeholders (functions) in terms of some metric?
  - force directed layout

Invoke Nature

More scalable tree, and natural with some randomness

http://animateddata.co.uk/lab/d3-tree/

Collapsible Tree

A standard tree, but one that is scalable to large hierarchies

http://mbostock.github.io/d3/talk/20111018/tree.html

Zoomable Partition Layout

A tree that is scalable and has partial partition of unity

http://mbostock.github.io/d3/talk/20111018/partition.html

Sunburst

More space efficient since it's radial, has partial partition of unity

https://www.jasondavies.com/coffee-wheel/

https://www.data-to-viz.com/graph/sunburst.html

https://observablehq.com/@kerryrodden/sequences-sunburst

Bubble Charts

No hierarchy information, just quantities

https://observablehq.com/@d3/bubble-chart

Circle Packing

Quantities and containment, but not partition of unity

http://mbostock.github.io/d3/talk/20111116/pack-hierarchy.html

Treemap

Quantities, containment, and full partition of unity

http://mbostock.github.io/d3/talk/20111018/treemap.html

Chord Diagram

Relationships among group fractions, not necessarily a tree

https://observablehq.com/@d3/chord-diagram

Hierarchical Edge Bundling

Relationships of individual group members, also in terms of quantitative measures such as information flow

http://mbostock.github.io/d3/talk/20111116/bundle.html

Collapsible Force Layout

Relationships within organization members expressed as distance and proximity

http://mbostock.github.io/d3/talk/20111116/force-collapsible.html

Voronoi Tessellation

Shows the closest point on the plane for a given set of points… and a new point via interaction

http://bl.ocks.org/mbostock/4060366

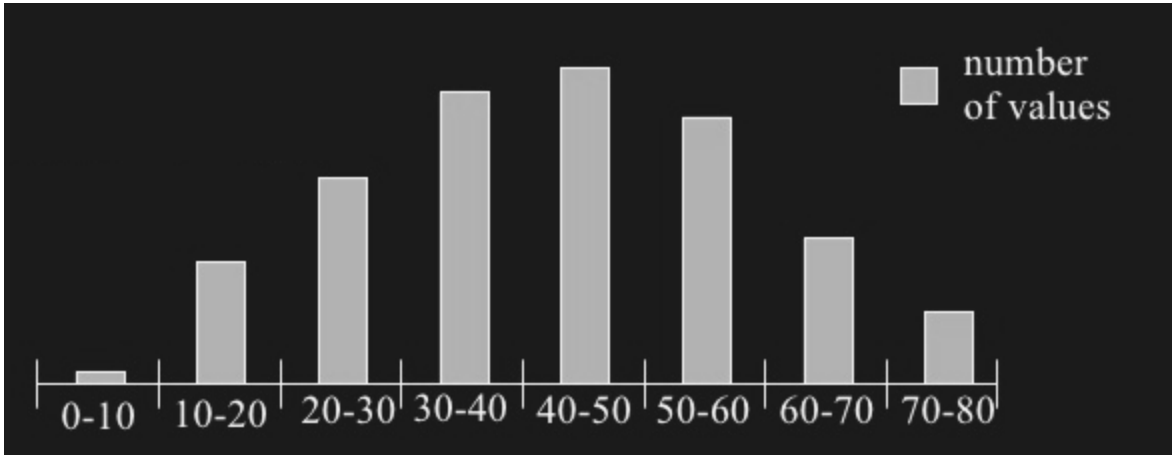
https://observablehq.com/@mbostock/voronoi-particles

Data Type Conversions and Transformation

Numeric to Categorical Data: Discretization (1)

Solution 1:

- divide the numeric attribute values into φ **equi-width** ranges
- each range/bucket has the same width
- example: customer age

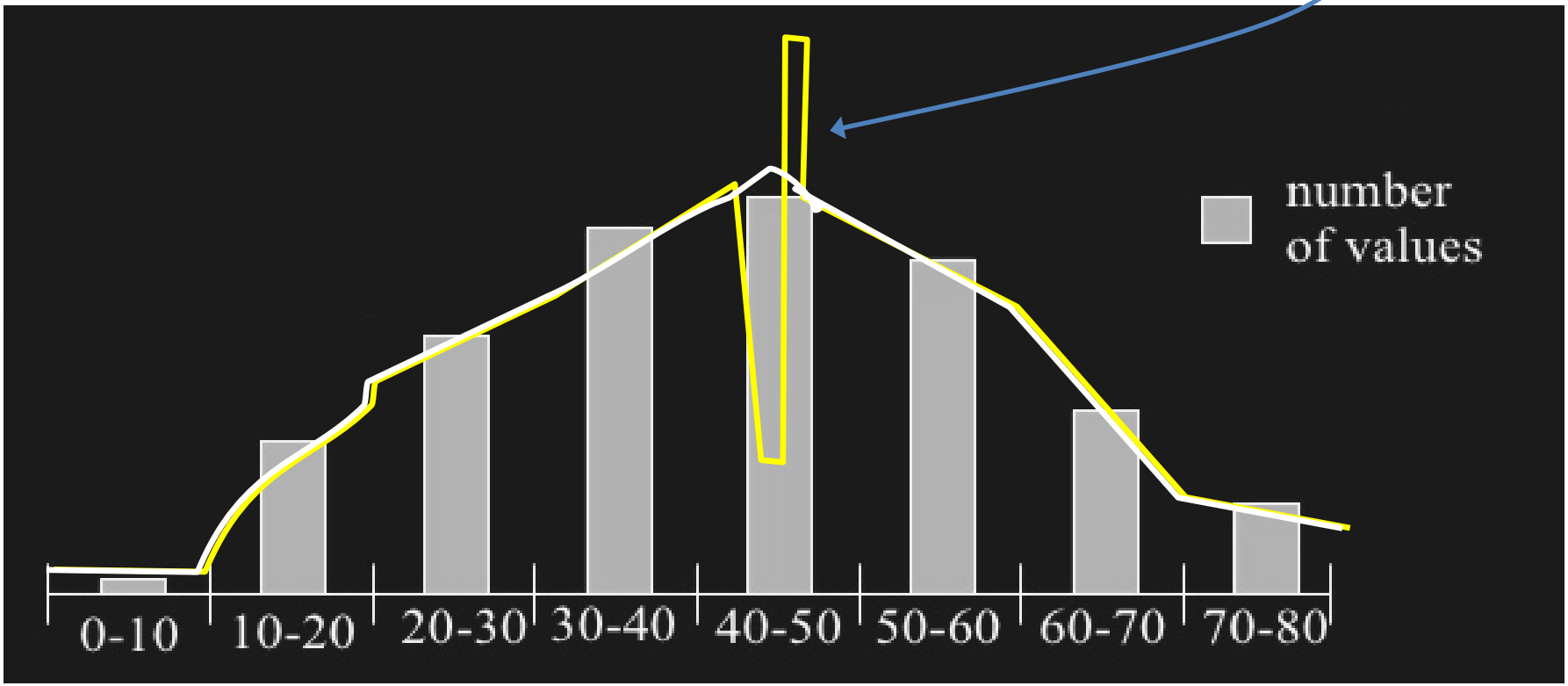


- what is lost here?

Problem With Equi-Width Histogram

Age ranges of customers could be unevenly distributed within a bin

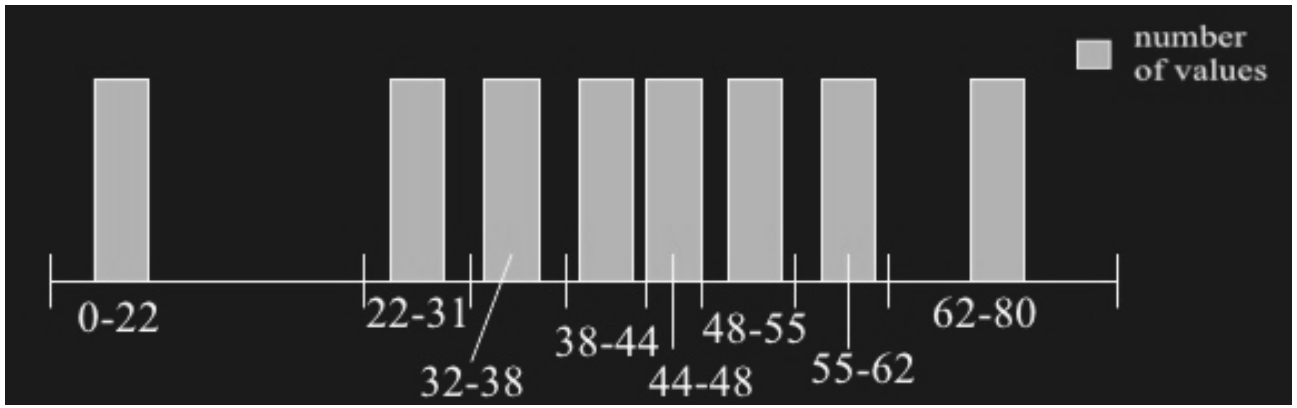
- this could be an interesting anomaly



Numeric to Categorical Data: Discretization (2)

Solution 2:

- divide the numeric attribute values into φ **equi-depth** ranges
- same number of samples in each bin
- (again) example: customer age:

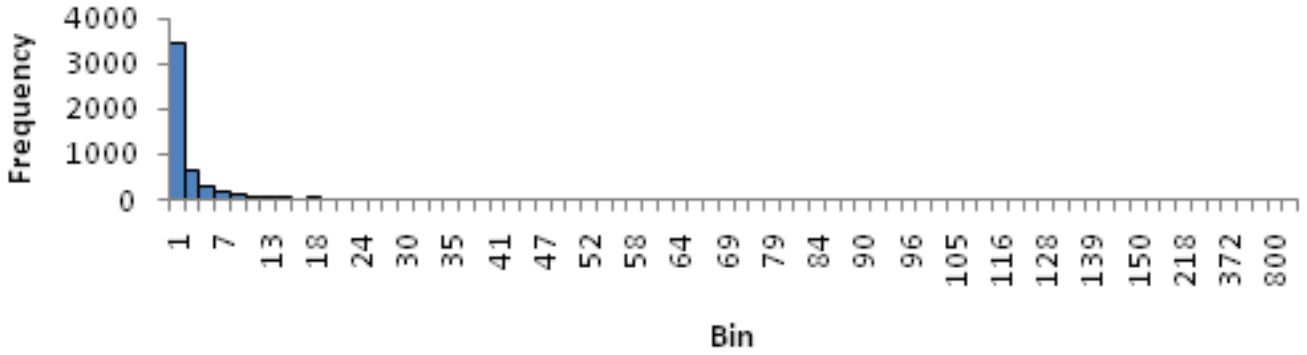


- what is the disadvantage here?
- extra storage needed: must store the start/end value for each bin

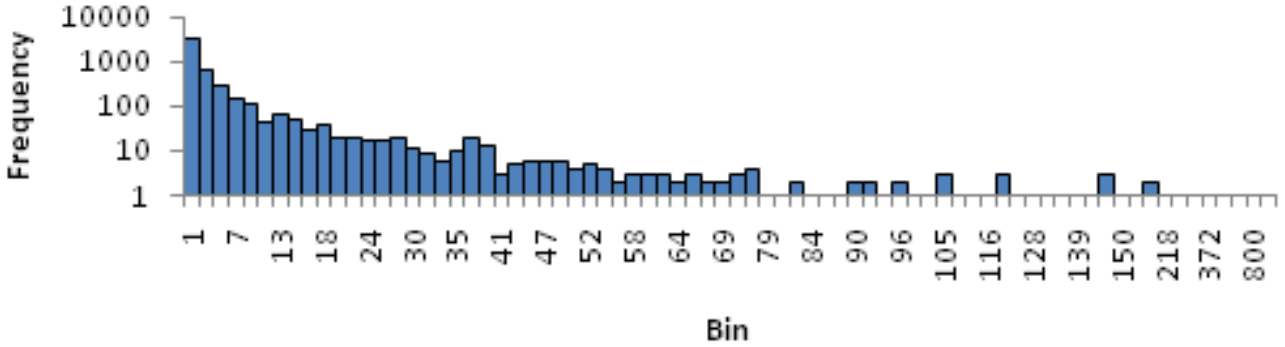
Numeric to Categorical Data: Discretization (3)

Solution 3:

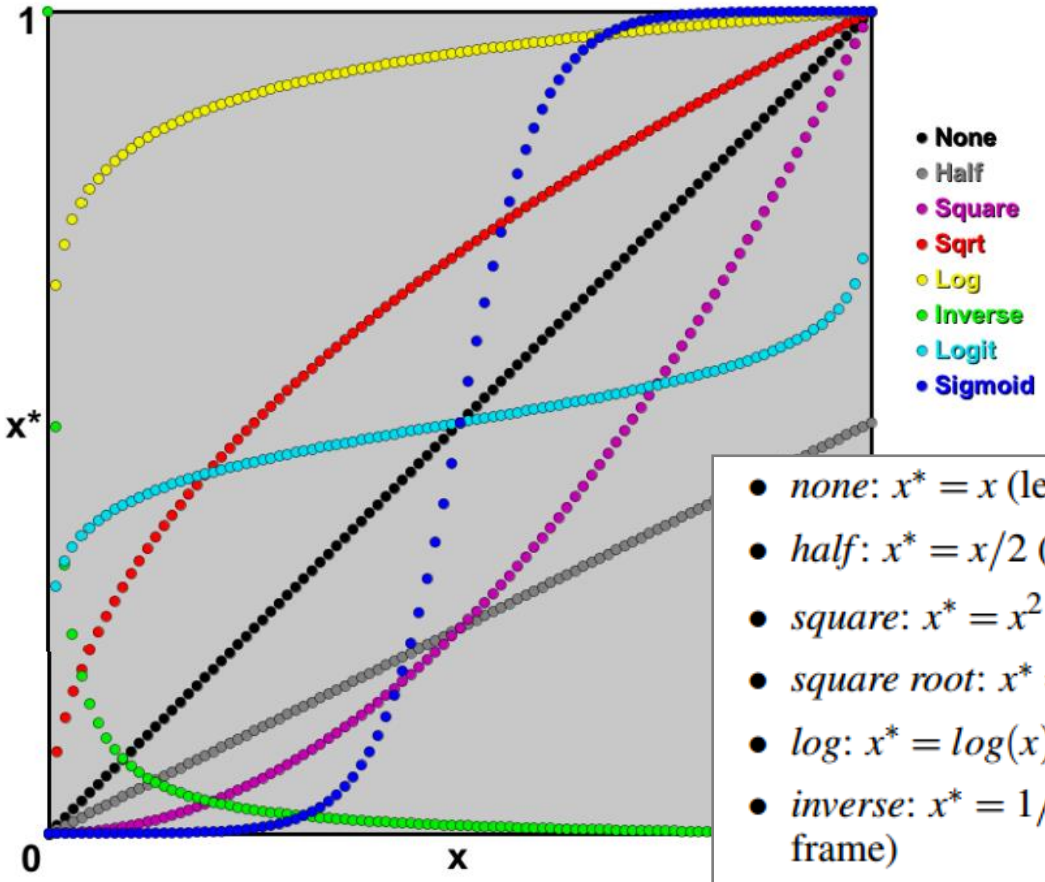
- what if all the bars have seemingly height
- or are dominated by one large peak



- switch to log scaling of the y-value



Other Transformations



- *none*: $x^* = x$ (leaves points unchanged)
- *half*: $x^* = x/2$ (squeezes all points together)
- *square*: $x^* = x^2$ (pulls points toward left of frame)
- *square root*: $x^* = \sqrt{x}$ (mildly pulls points toward right of frame)
- *log*: $x^* = log(x)$ (strongly pulls points toward right of frame)
- *inverse*: $x^* = 1/x$ (reverses scale and squeezes points into left of frame)
- *logit*: $x^* = (log(x/(1-x)) + 10)/20$ (squeezes points toward middle of frame)
- *sigmoid*: $x^* = 1/(1 + exp(-20x + 10))$ (expands points away from middle of frame)

Dang and Wilkinson,
"Transforming Scagnostics to Reveal Hidden Features", TVCG 2014

Data Representation

Data Representation

Ever tried to reduce the size of an image and you got this?

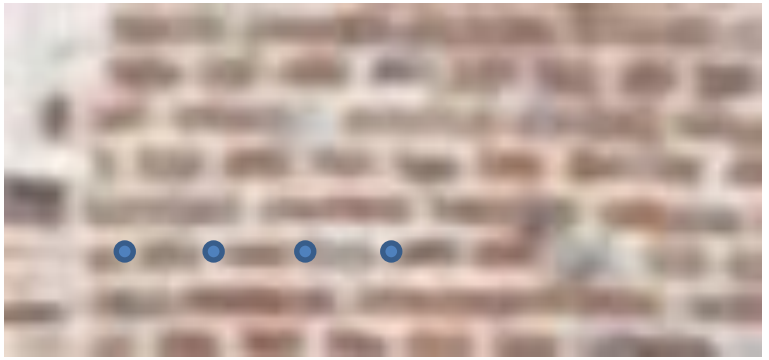
This is aliasing

Data Representation

But what you really wanted is this:

This is *anti-aliasing*

Why Is This Happening?



The smaller image resolution cannot represent the image detail captured at the higher resolution

- skipping this small detail leads to these undesired artifacts

What Is Anti-Aliasing

Procedure

- either sample at a higher rate
- or smooth the signal before sampling it
- the latter is called *filtering*

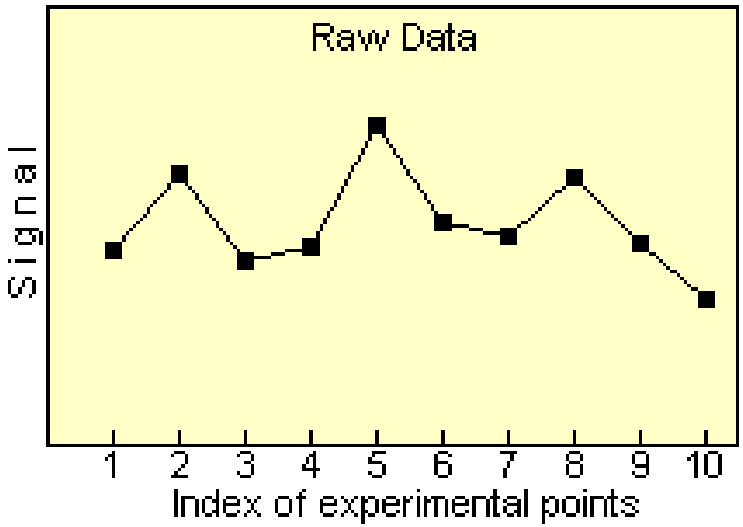
Anti-Aliasing Via Smoothing

Anti-Aliasing Via Smoothing

What is Smoothing?

Slide a window across the signal

- stop at each discrete sample point
- average the original data points that fall into the window
- store this average value at the sample point
- move the window to the next sample point
- repeat



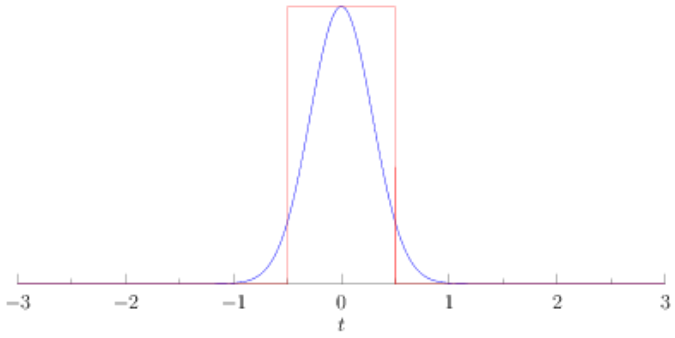
Filters

What is the filter we just used called?

- it's called a *box filter*

There are other filters

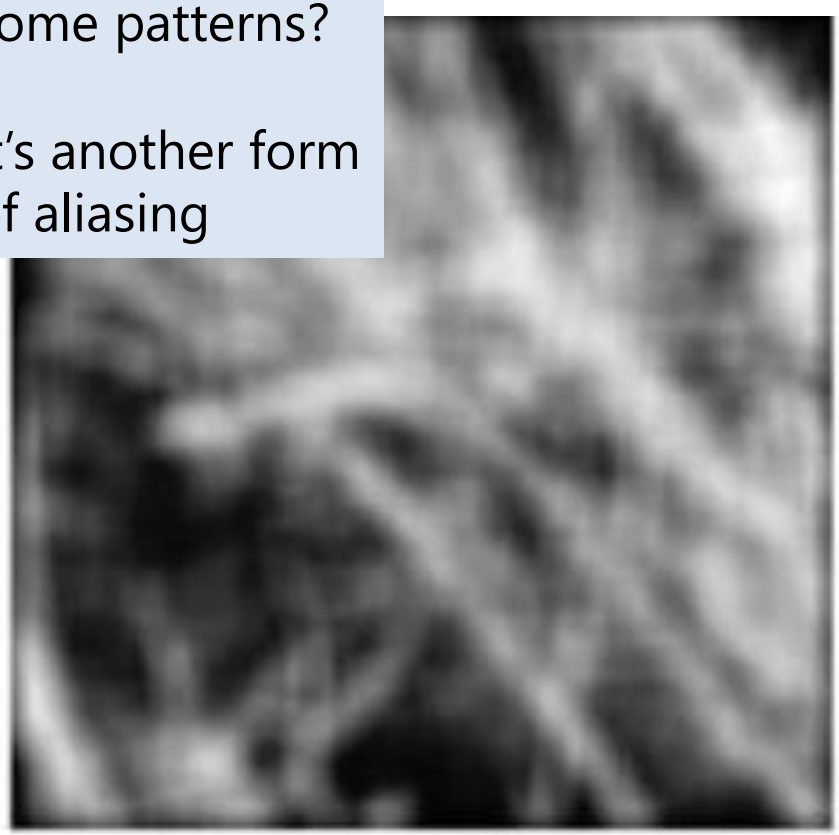
- for example, Gaussian filter
- yields a smoother result
- box filtering is simplest

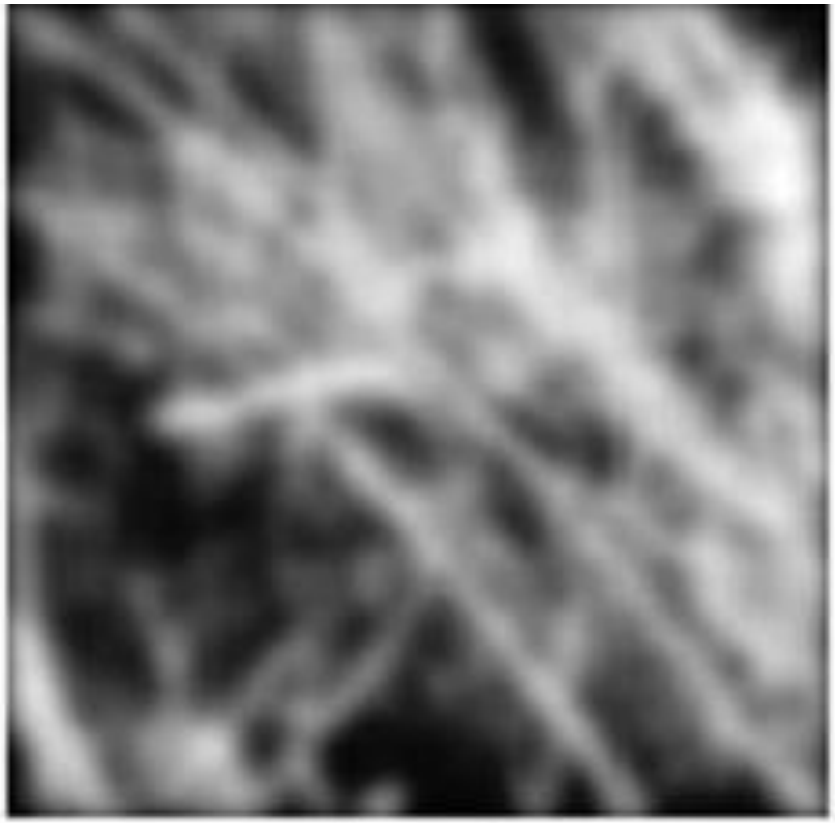


Box Filter vs. Gaussian Filter

Can you see some patterns?

It's another form of aliasing





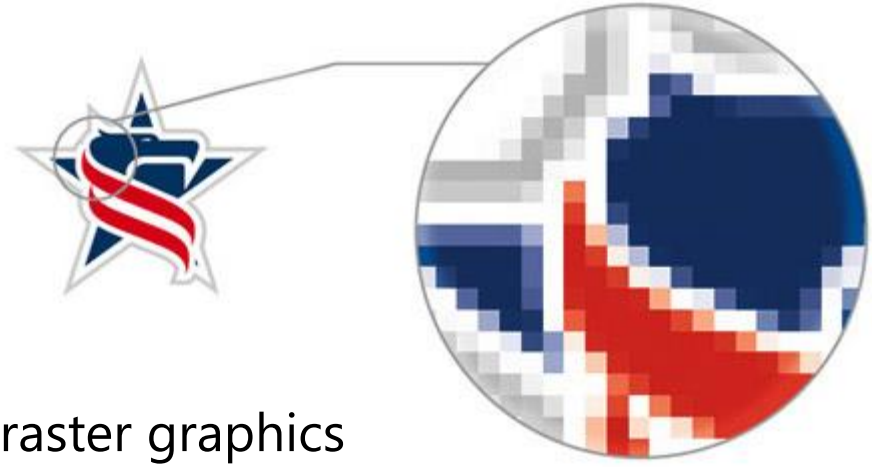
2D box

2D Gaussian

The Solution

What's the underlying problem?

- detail can't be refined upon zoom
- can just be replicated or blurred



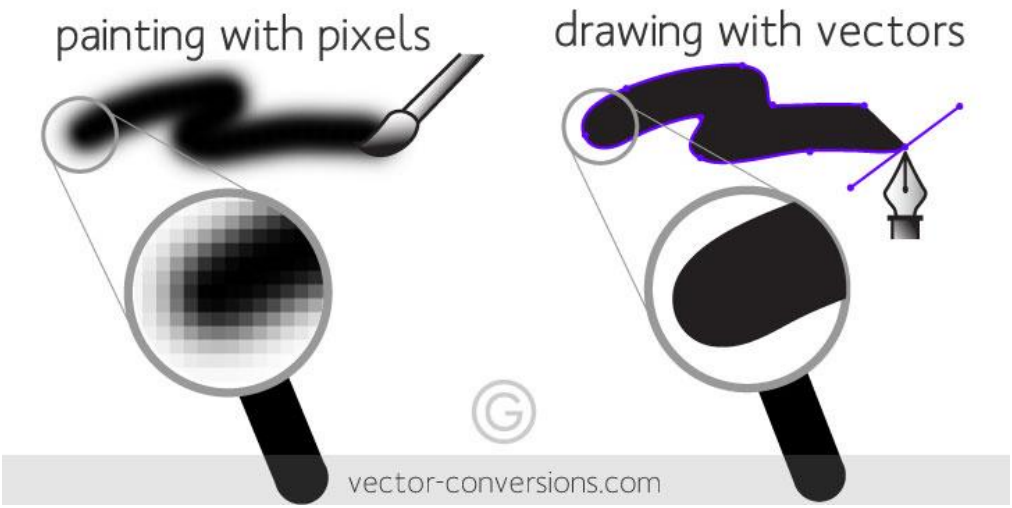
raster graphics

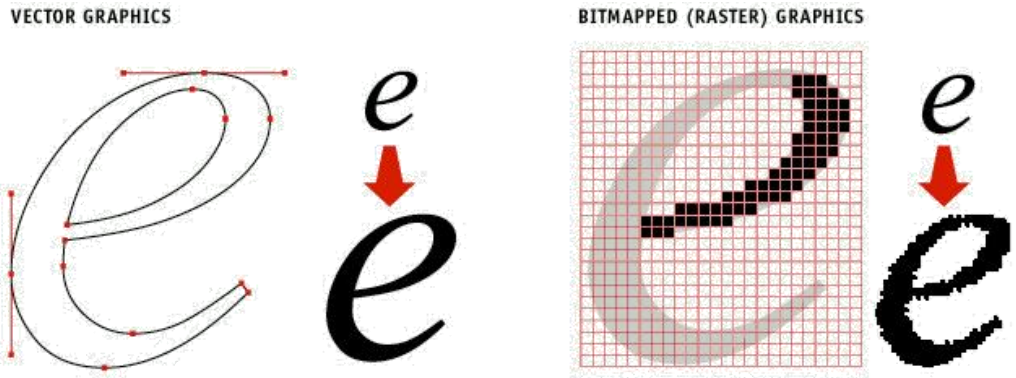
vector graphics

The solution…

- represent detail as a function that can be mathematically refined
- replace raster graphics by vector graphics

Scalable Vector Graphics (SVG)





Photographs and Images in SVG

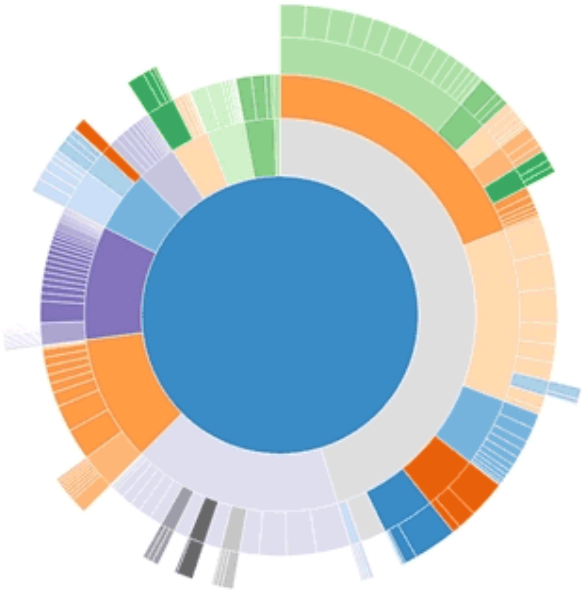
Vector graphics tends to have an “cartoonish” look

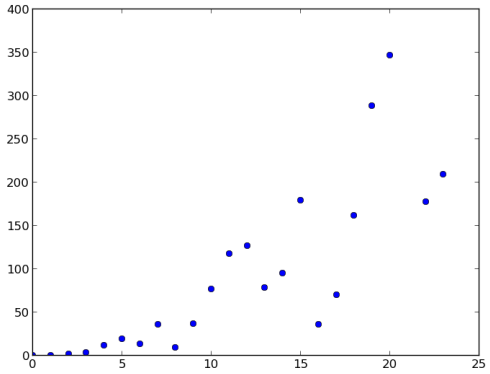
raster graphics

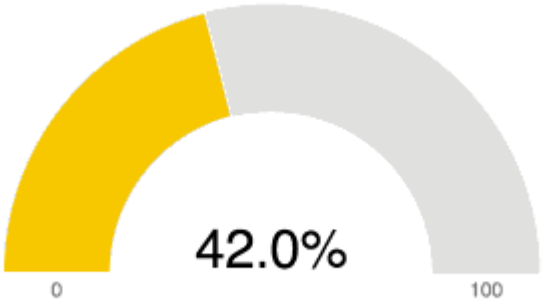
vector graphics

Photographs and Images in SVG

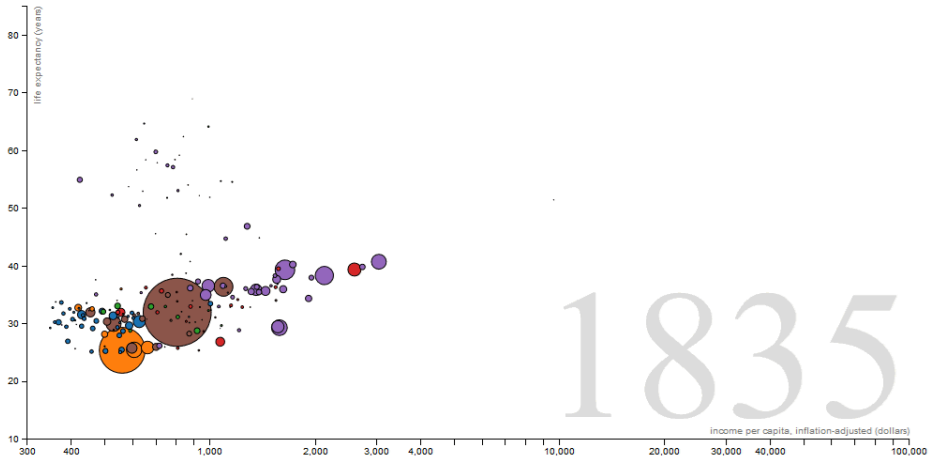
D3 Uses SVG





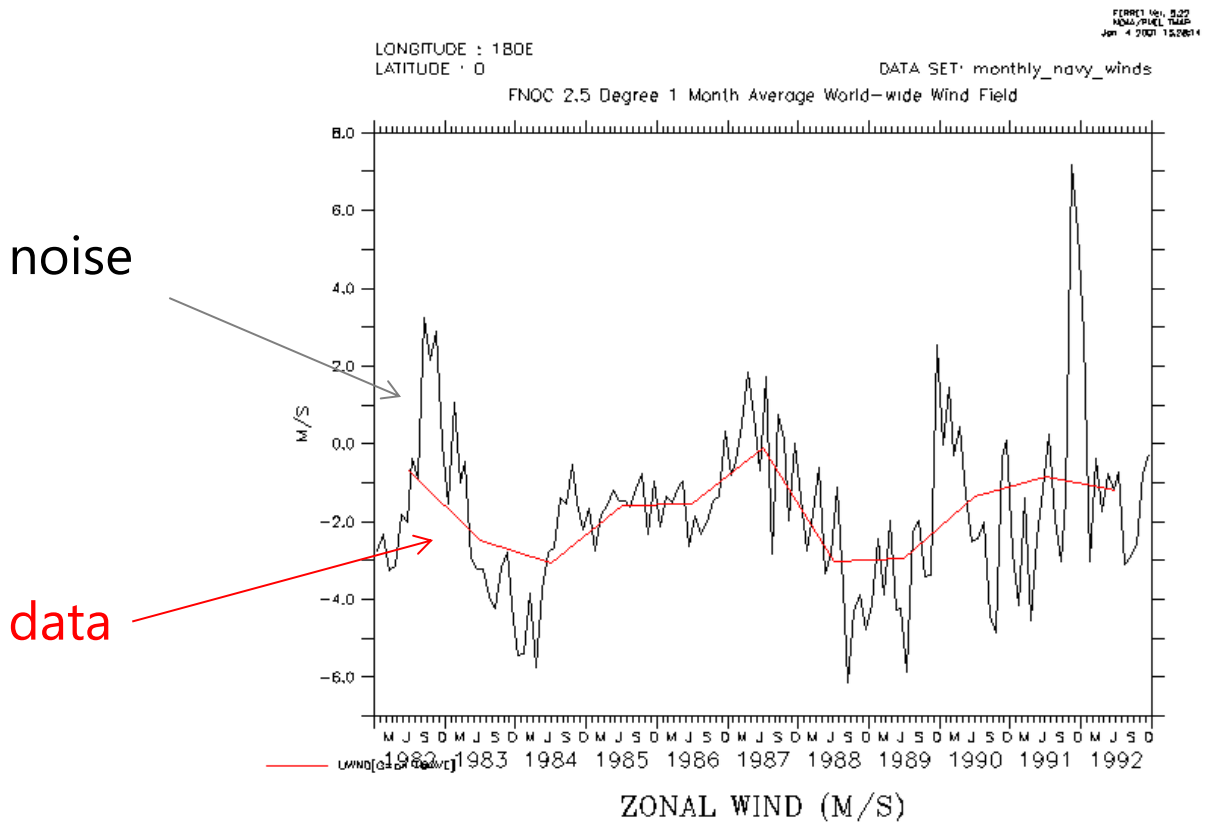


The Wealth & Health of Nations



De-Noising

Filtering also eliminates noise in the data



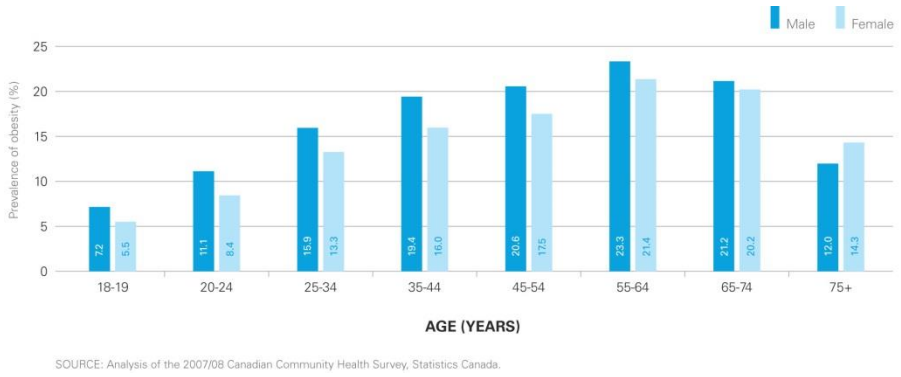
Back to Bar Charts

In some ways, bar charts reduce noise and uncertainties in the data

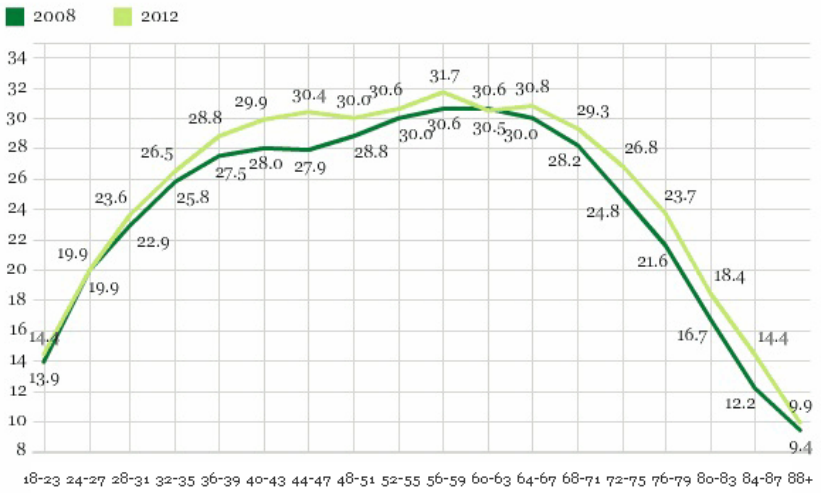
- the bins do the smoothing

Example:

- obesity over age (group)



SOURCE: Analysis of the 2007/08 Canadian Community Health Survey, Statistics Canada.

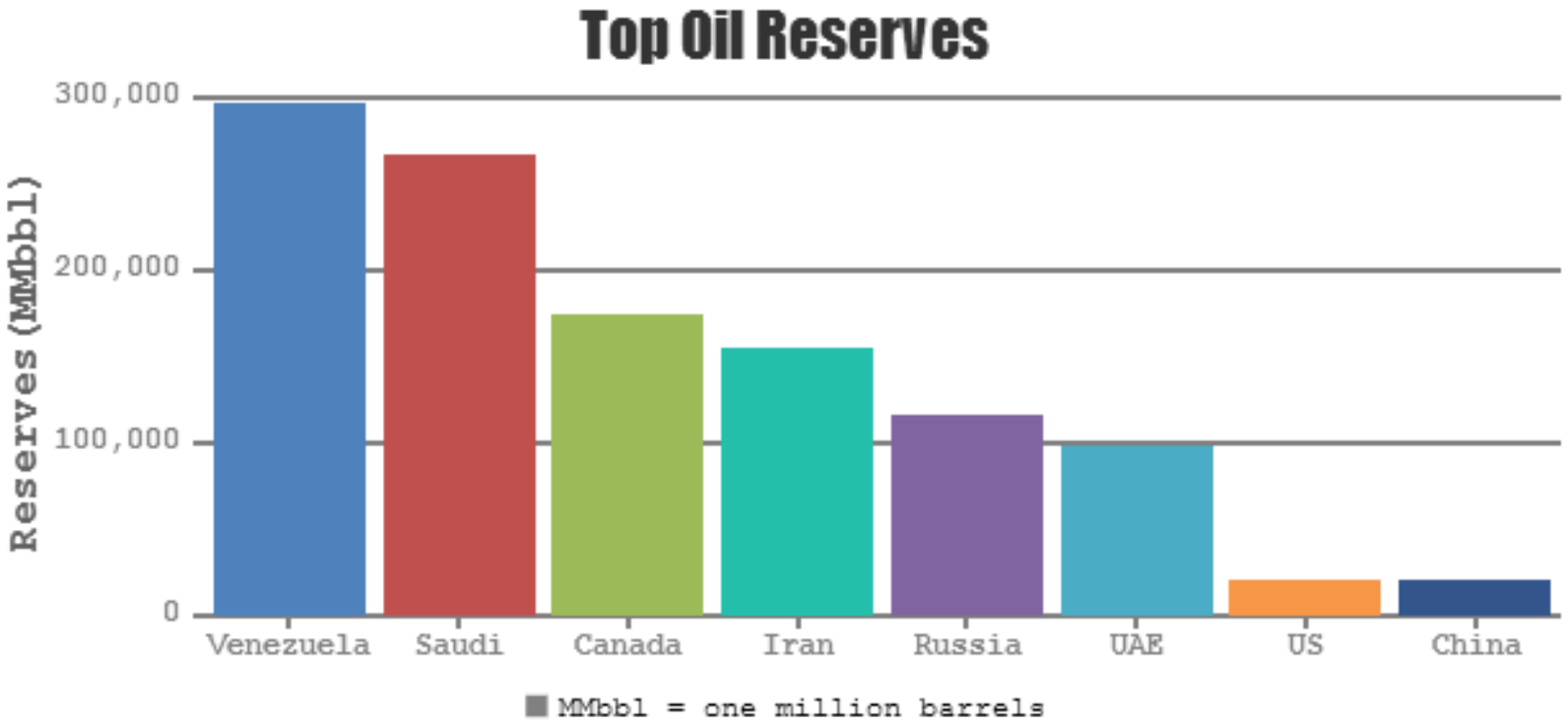


Gallup-Healthways Well-Being Index

GALLUP

Bar Charts

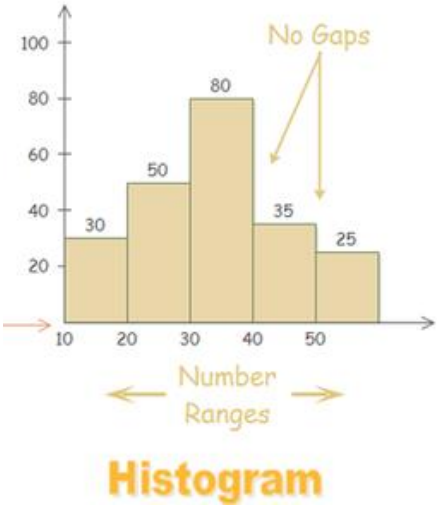
Of course, bar charts can also hold categorical data



Bar Charts vs. Histograms

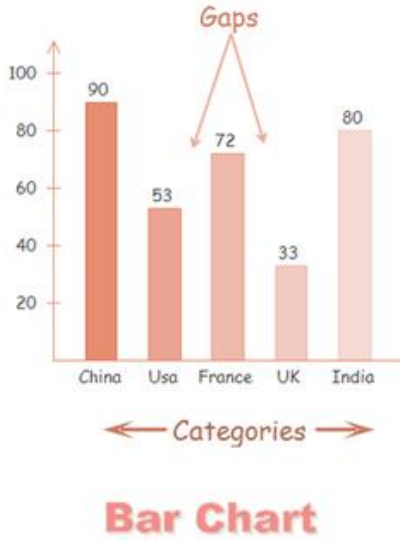
Histograms

- bars show the frequency of numerical data
- quantitative data
- elements are grouped together, so that they are considered as ranges
- bars cannot be reordered
- width of bars need not be the same



Bar charts

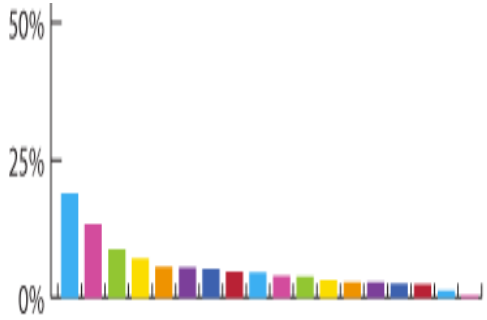
- uses bars to compare different categories of data
- comparison of discrete variables
- elements are taken as individual entities
- bars can be reordered
- width of bars need to be the same

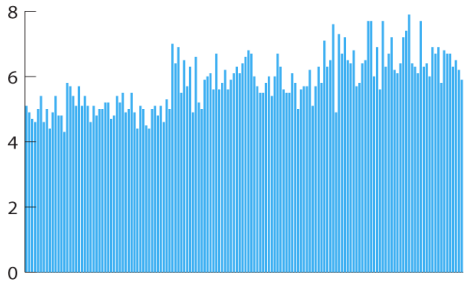


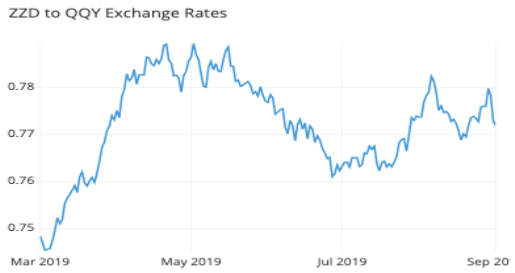
How many Bars in a Bar Chart

How many bars are too many (in a chart)

- if individual categories are the focus? 12 is a good rule
- if the overall trend is the important factor? 50 or even more
- eventually you can switch to a line chart







- sort bars by height and use ‘other’ to aggregate the bar chart tails into a single bar
- find a grouping that can semantically aggregate bars, for example aggregate countries into continents

more information

Bar Charts in D3

https://observablehq.com/@d3/bar-chart

Working with bar charts and histograms is the topic of Lab 2

- the next two slides offer some help with calculations

Bar Chart Calculations – Binning

Determine bin size

- min(data) is optional, can also use 0 or some reasonable value
- max(data) is optional, can also use some reasonable value

$$bin\ size = \frac{\max(data) - \min(data)}{number\ of\ bins}$$

Given a data value *val* increment (++) the bin value

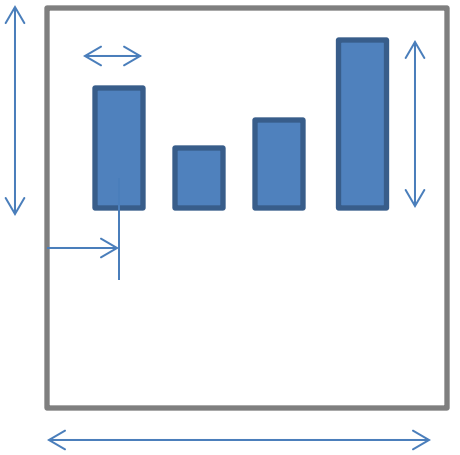
- but first initialize bin val array to 0

$$bin\ val\ array\left[\left\lfloor\frac{val - \min(data)}{bin\ size}\right\rfloor\right] ++$$

Bar Chart Calculations – Plotting

Determine bin size on the screen

$$bin\ size\ on\ screen = \frac{chart\ width}{number\ of\ bins}$$



Center of a bar for bin with index *bin index*

$$bar\ center\ on\ screen = (bin\ index \cdot bin\ size\ on\ screen) + 0.5$$

Height of the bar for a bin with index *bin index*

$$bar\ height(bin\ index) = bin\ val\ array(bin\ index) \cdot \frac{chart\ height}{\max(bin\ val\ array)}$$

Do not forget that the origin of a web page is the top left corner

D3, Vega, Vega-Lite

D3 – Data Driven Documents (we will use for this course)

- creates interactive webpages from data
- lots of creations are here

Vega (see here)

- higher-level visualization specification language on top of D3
- D3 is still more "expressive" and allows for more creative freedom

Vega-Lite (see here)

- a high-level grammar of interactive graphics
- built on top of Vega
- more concise & convenient form to author common visualizations
- supports data analytics (both data and visual transformations)
- better support for interactions

Tableau

Tableau is a leading commercial visual analytics platform

- founded in 2003 by a group of Stanford University researchers (Chris Stolte, Pat Hanrahan, and Christian Chabot)
- recently acquired by Salesforce
- goal was to make data more accessible through visualization
- key tech was VizQL – visualizes data by translating drag-and-drop actions into data queries through an intuitive interface

Example Tableau Dashboards

Account tracking

Quarterly results

Top accounts

Opportunity overview

Opportunity tracking

Marketing leads

D3 vs. Tableau

D3	Tableau
Open Source	Proprietary / Paid
Web Standards Focused	VizQL Language
Real-Time	Automated Updates but Not Real-Time
Expansive Viz Options	Limited Viz Choices*
Lots of Coding	Data to Viz in Seconds
Complex	Easy to Use
Limited Native Data Connections	Native Data Connections
Manual Calculations	Automated Calculations
Limited Data Manipulations	Strong Data Manipulations

source

Essentially, Tableau is great for expediently-developed in-house use
D3 is better for external use, real-time interactive web, and embedding into a product