CSE 564
Visualization & Visual Analytics

Applications and Basic Tasks

Klaus Mueller

Computer Science Department
Stony Brook University

Course Website

http://www.cs.stonybrook.edu/~mueller/teaching/cse564/

Everything you need is there:

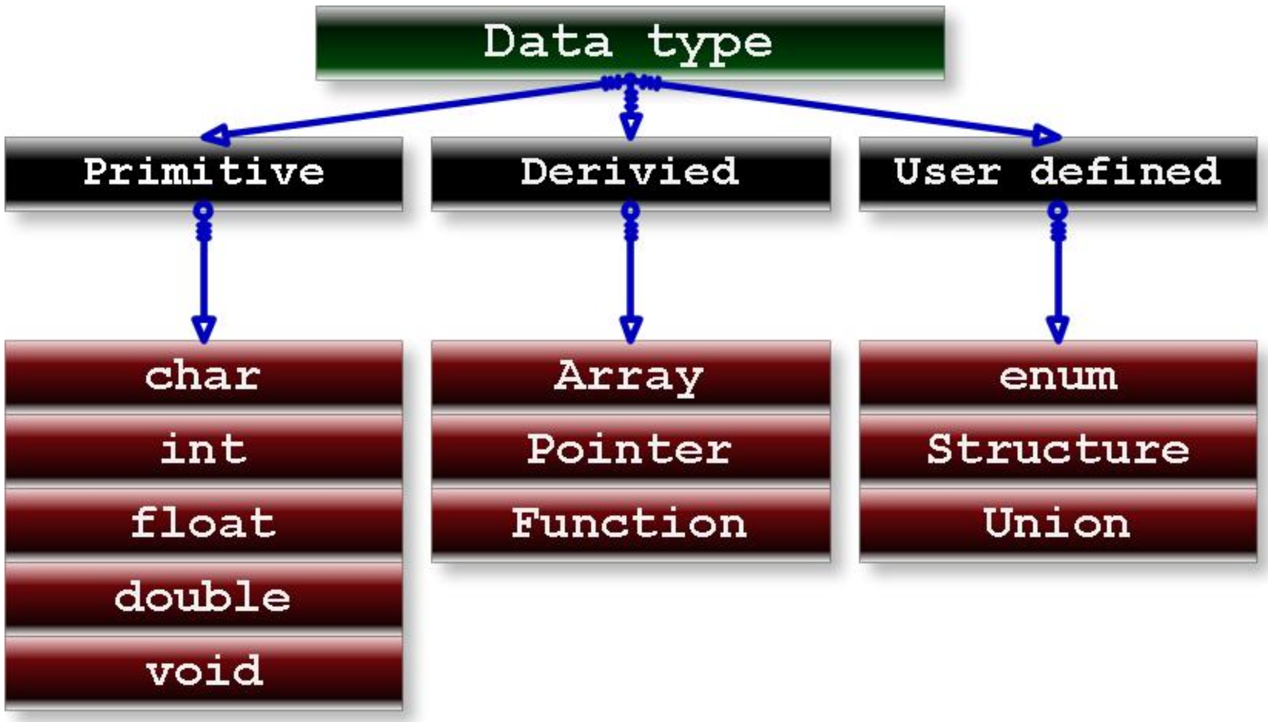
- syllabus
- course notes (slides) posted shortly after the lecture
- lab assignments
- course policy

There will also be (soon to be announced)

- a server for lab assignments
- piazza for online support

Lecture	Topic	Projects
1	Intro, schedule, and logistics	
2	Applications of visual analytics, basic tasks, data types	
3	Introduction to D3, basic vis techniques for non-spatial data	Project #1 out
4	Visual perception and cognition	
5	Visual design and aesthetics	
6	Data types, notion of similarity and distance	
7	Data preparation and reduction	Project #1 due
8	Introduction to R, statistics foundations	Project #2 out
9	Data mining techniques: clusters, text, patterns, classifiers	
10	Data mining techniques: clusters, text, patterns, classifiers	
11	Computer graphics and volume rendering	
12	Techniques to visualize spatial (3D) data	Project #2 due
13	Scientific and medical visualization	Project #3 out
14	Scientific and medical visualization	
15	Midterm #1	
16	High-dimensional data, dimensionality reduction	Project #3 due
17	Big data: data reduction, summarization	
18	Correlation and causal modeling	
19	Principles of interaction	
20	Visual analytics and the visual sense making process	Final project proposal due
21	Evaluation and user studies	
22	Visualization of time-varying and time-series data	
23	Visualization of streaming data	
24	Visualization of graph data	Final Project preliminary report due
25	Visualization of text data	
26	Midterm #2	
27	Data journalism	
	Final project presentations	Final Project slides and final report due

Data Types Every CS Person Knows



Data Types in Visual Analytics

Numeric

Categorical

Text

Time series

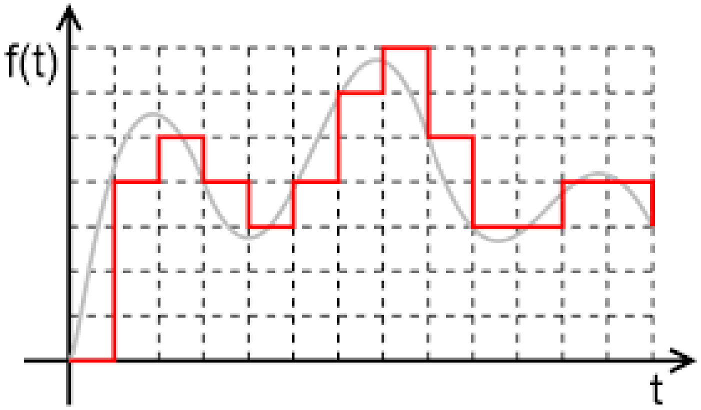
Graphs and networks

Hierarchies

Variables in Statistics

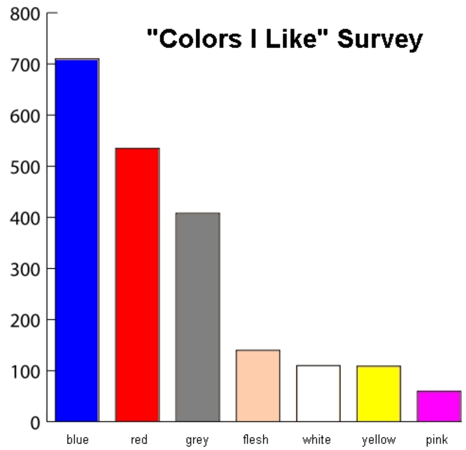
Numeric variables

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



Categorical variables

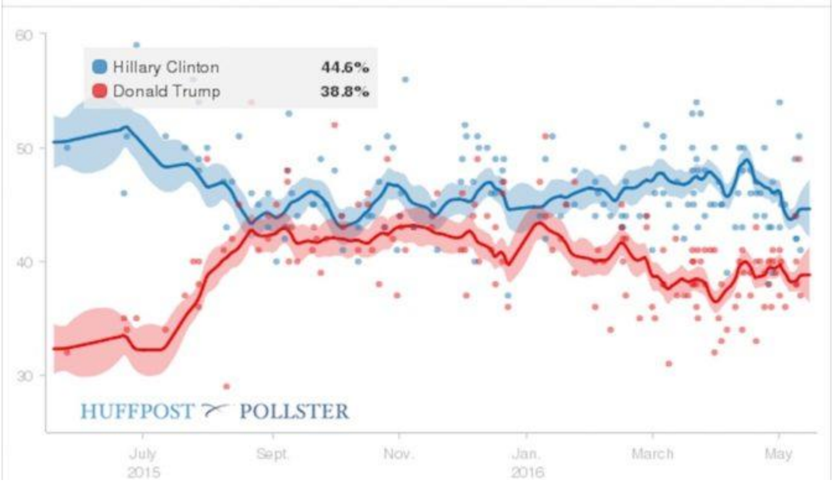
- describe a **quality** or characteristic
- like: 'what type' or 'which category’



Numeric Variables

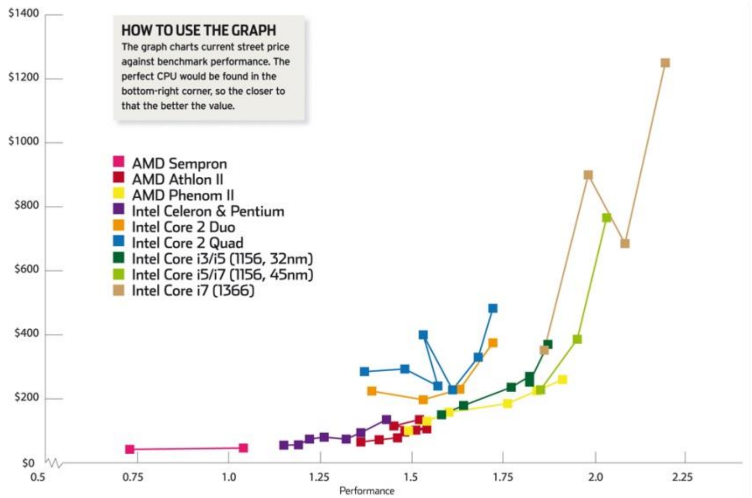
Most often the x-axis is 'time'

- 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

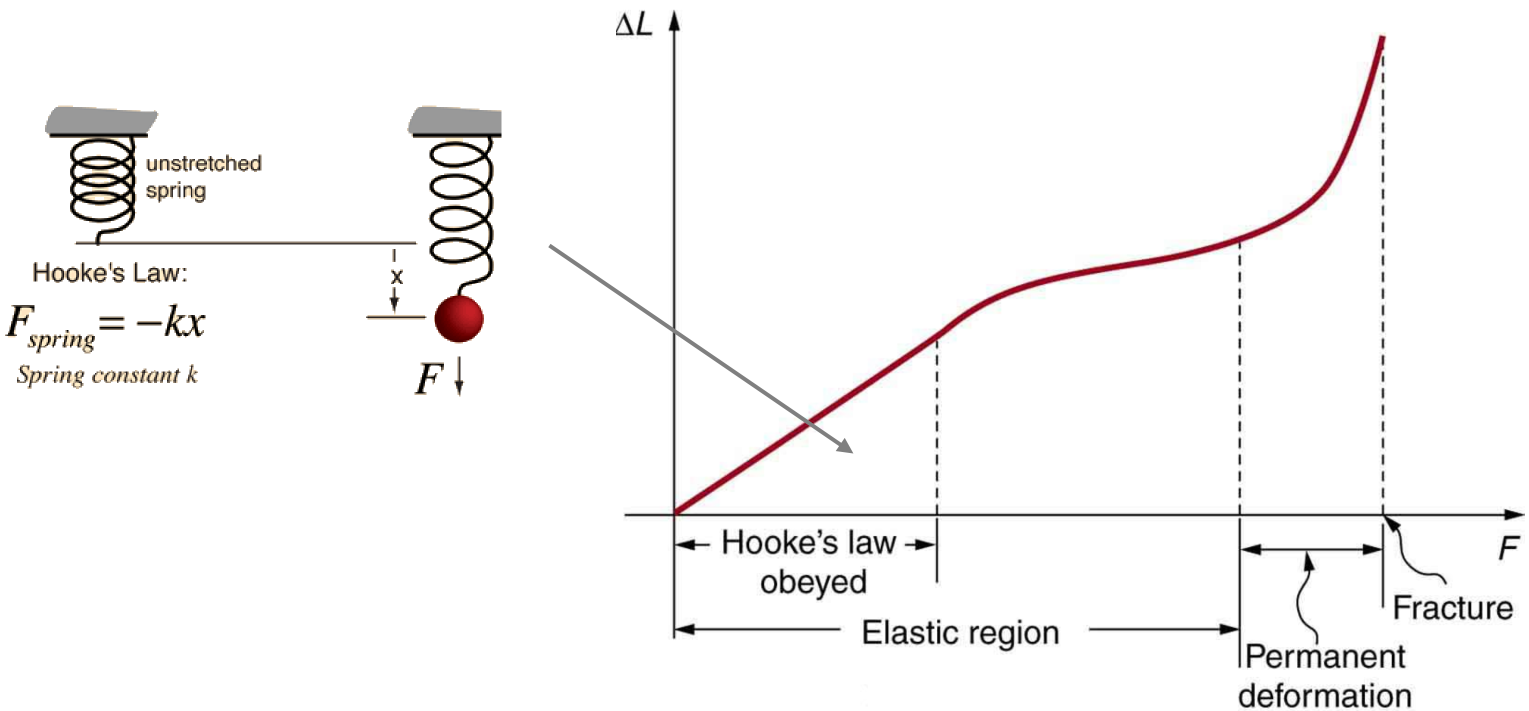
- engineers, statisticians, etc. will be receptive to this idea
- can you think of an example?



Numeric Variables

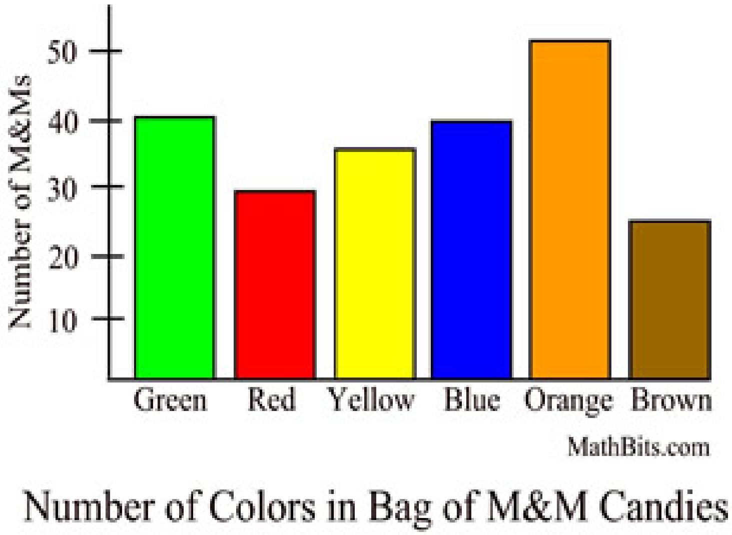
Another plot where 'time' is not the x-axis

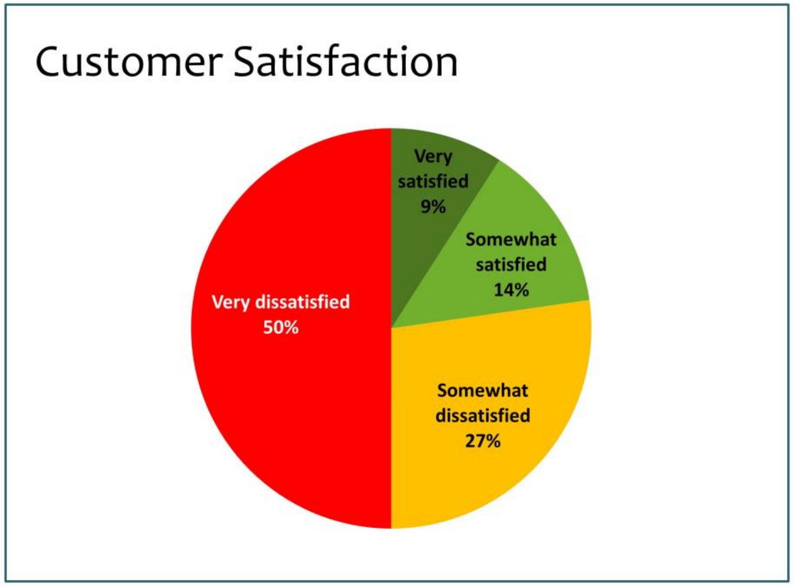
- from the engineering / physics domain
- in some sense, it tells a story



Categorical Variables

Usually plotted as bar charts or pie charts





?? ??

nominal

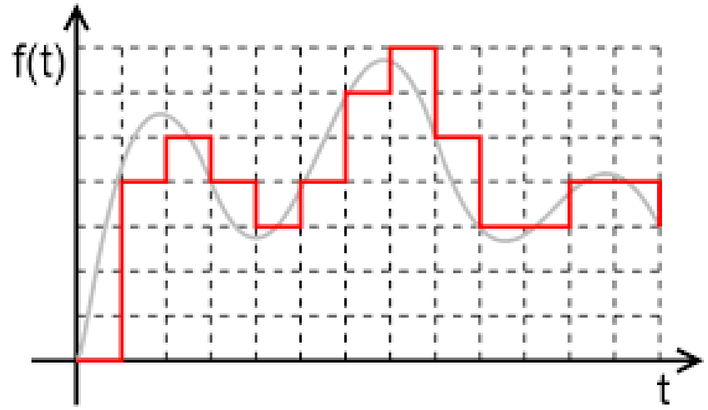
ordinal

but of course you can plot either of them in either of these two representations

Variables in Statistics

Numeric 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

Numbers are Good

But not everything is expressed in numbers

- images
- video
- text
- web logs
- …

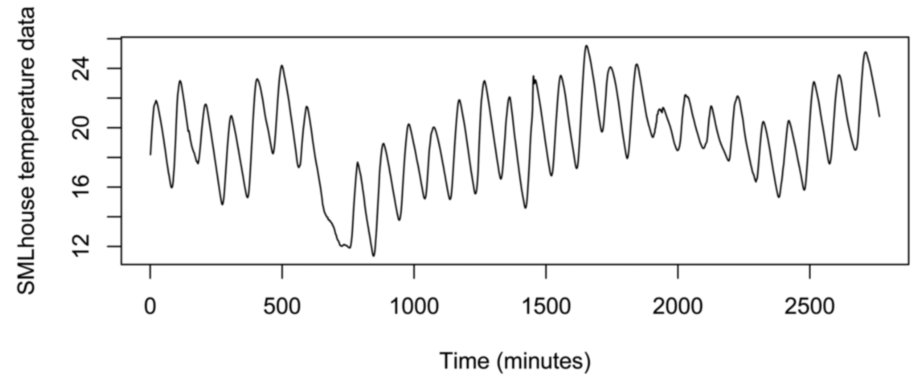
Do feature analysis to turn these abstract things into numbers

- 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

Sensor Data

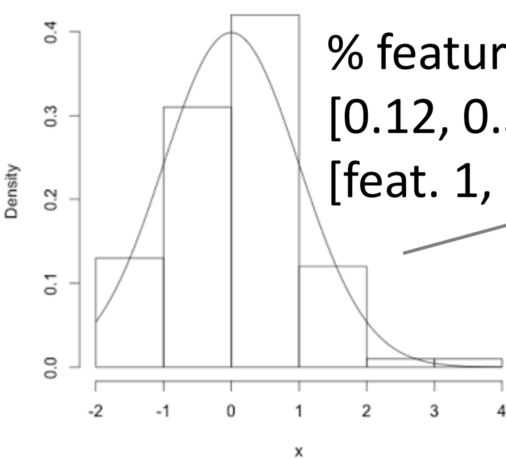
Characteristics

- often large scale
- time series

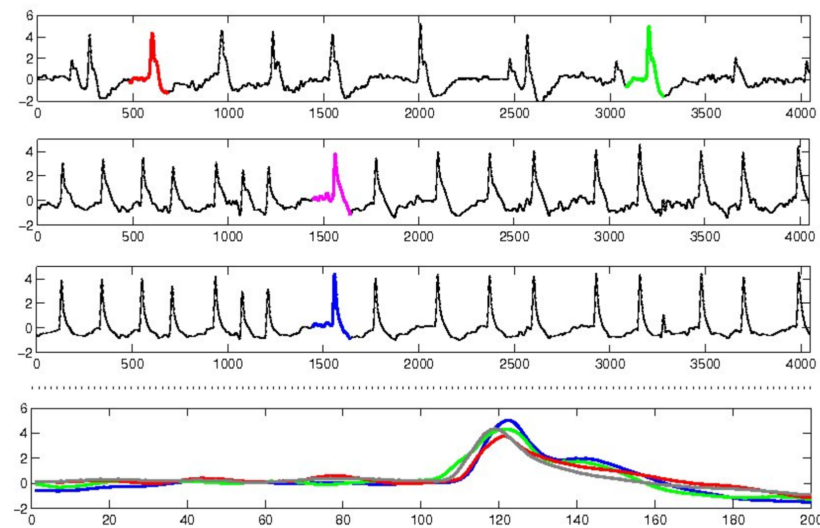


Feature Analysis

- example: Motif discovery
- encode into 5D data vector



% features discovered in stream
[0.12, 0.3, 0.41, 0.12, 0.05]
[feat. 1, feat. 2, .., feat. 5]

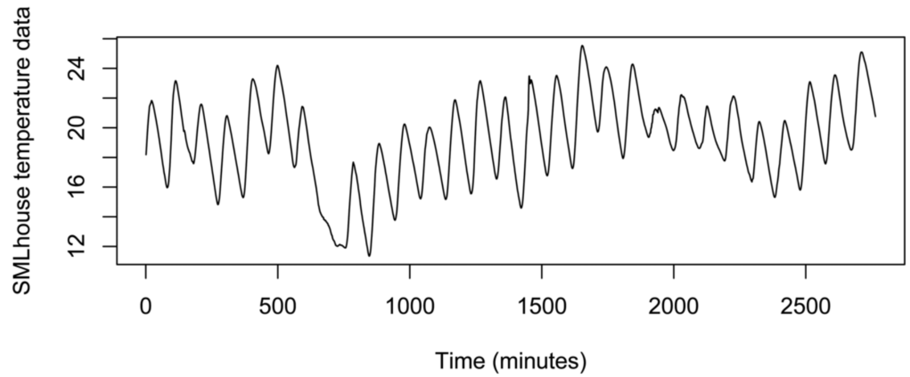


Motif discovery

Sensor Data

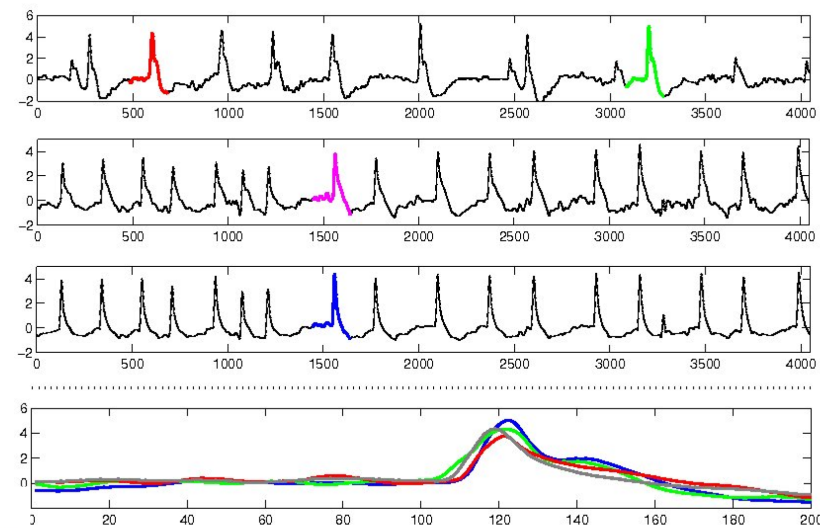
Characteristics

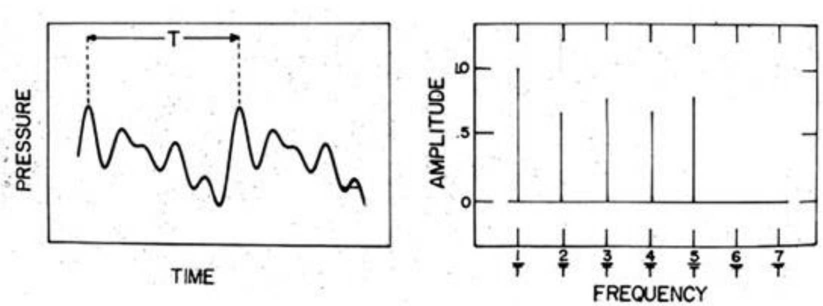
- often large scale
- time series



Feature Analysis

- Fourier transform (FT, FFT)
- Wavelet transform (WT, FWT)





Fourier transform

Image Data

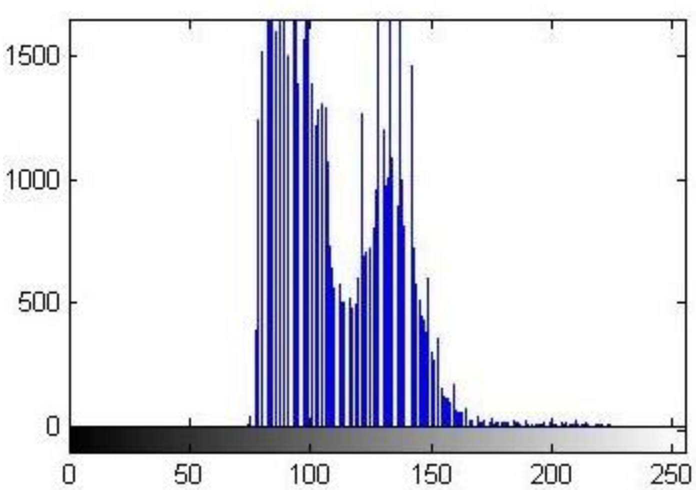
Characteristics

- array of pixels

histograms

Feature Analysis

- value histograms
- encode into a 256-D vector



[0, 0, 0, …., 10, …, 1200, …..]

Image Data

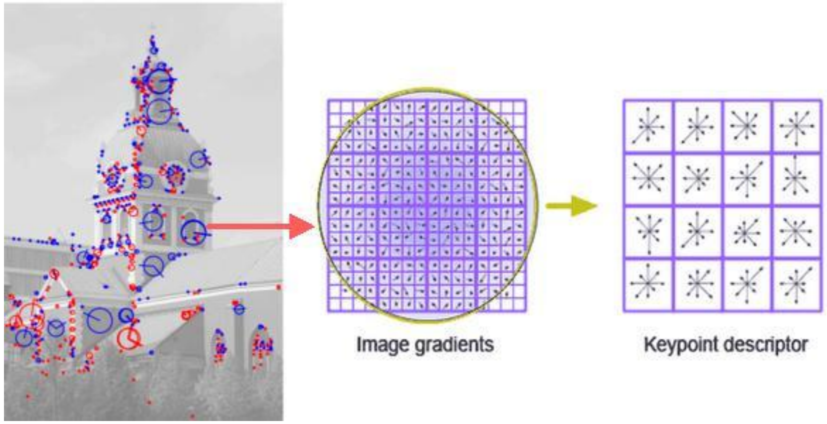
Characteristics

- array of pixels

Feature Analysis

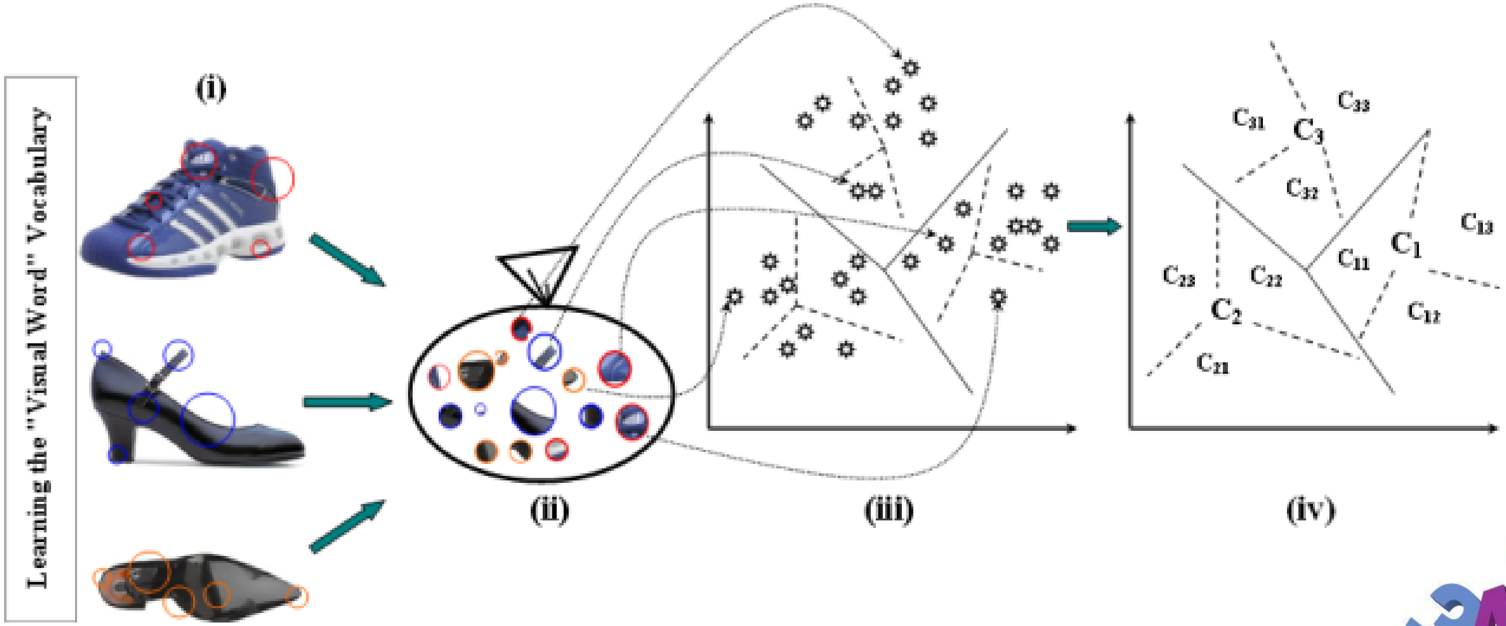
- value histograms
- gradient histograms
- FFT, FWT
- Scale Invariant Feature Transform (SIFT)
- Bag of Features (BoF)
- visual words

histograms



SIFT

Bag of Features (BoF)



Bag of Features (BoF)

1. Obtain the set of bags of features
   - (i) Select a large set of images
   - (ii) Extract the SIFT feature points of all the images in the set and obtain the SIFT descriptor for each feature point extracted from each image
   - (iii) Cluster the set of feature descriptors for the amount of bags we defined and train the bags with clustered feature descriptors
   - (iv) Obtain the visual vocabulary

2. Obtain the BoF descriptor for a given image/video frame
   - (v) Extract SIFT feature points of the given image
   - (vi) Obtain SIFT descriptor for each feature point
   - (vii) Match the feature descriptors with the vocabulary we created in the first step
   - (viii) Build the histogram

More information

Video data

Characteristics

- essentially a time series of images

Feature Analysis

- many of the above techniques apply albeit extension is non-trivial

Text Data

Characteristics

- often raw and unstructured

Feature analysis

- first step is to remove stop words and stem the data
- perform **named-entity recognition** to gain atomic elements
  - identify names, locations, actions, numeric quantities, relations
  - understand the structure of the sentence and complex events
- example:
  - Jim bought 300 shares of Acme Corp. in 2006.
  - [Jim]$_{\text{Person}}$ bought [300 shares] $_{\text{Quantity}}$ of [Acme Corp.]$_{\text{Organiz.}}$ in [2006]$_{\text{Time}}$
- distinguish between
  - application of grammar rules (old style, need experienced linguists)
  - statistical models (Google etc., need big data to build)

Text to Numeric Data

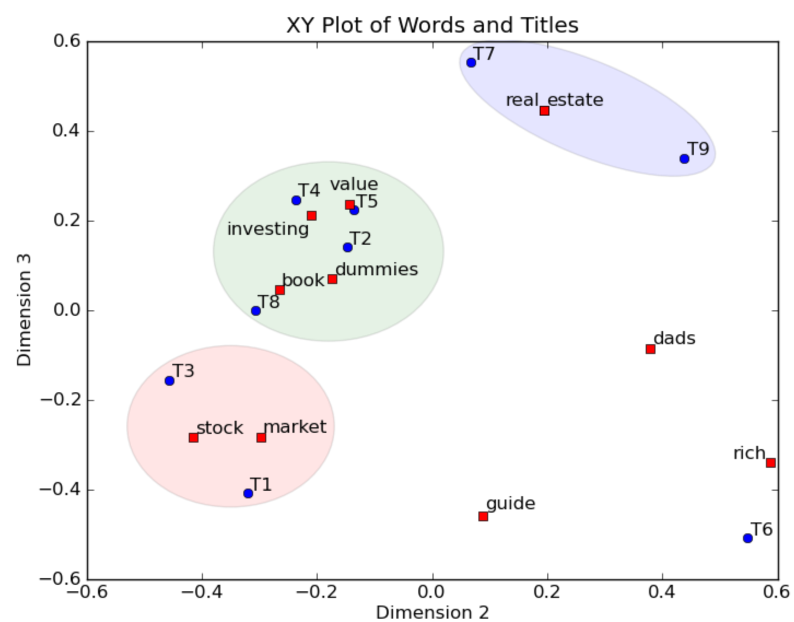
Create a term-document matrix

- 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 →

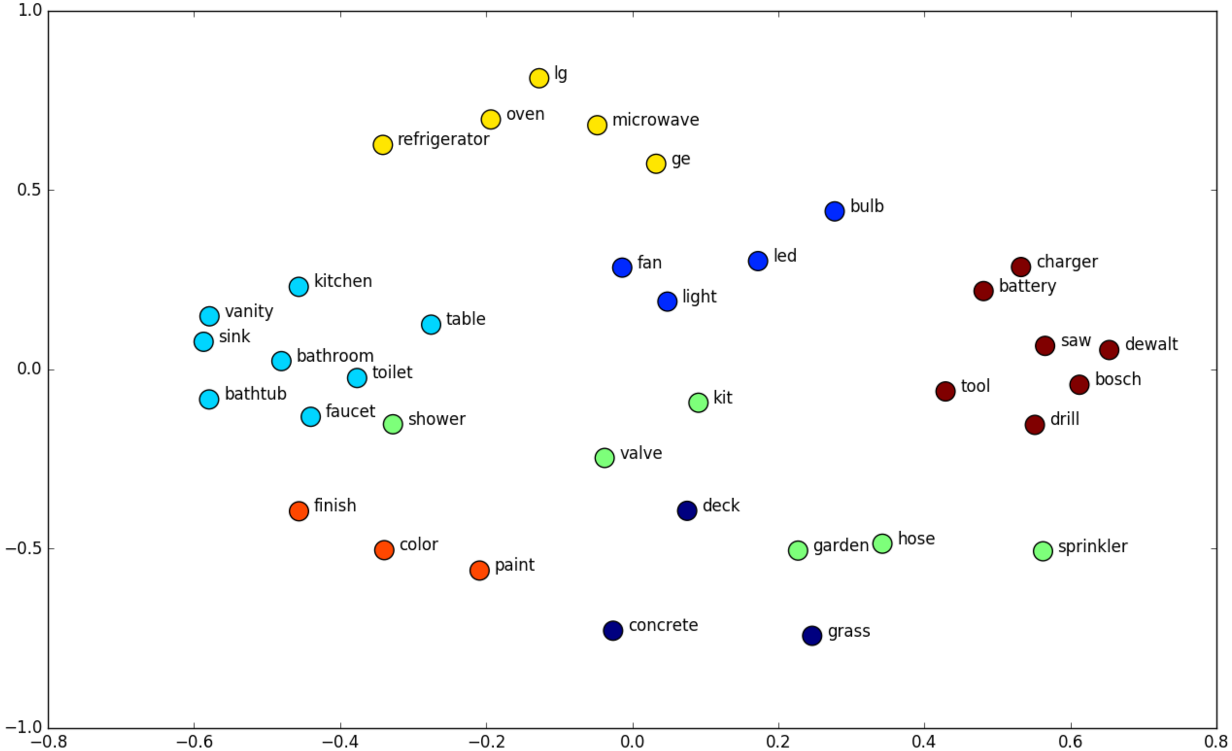


Word/document cluster

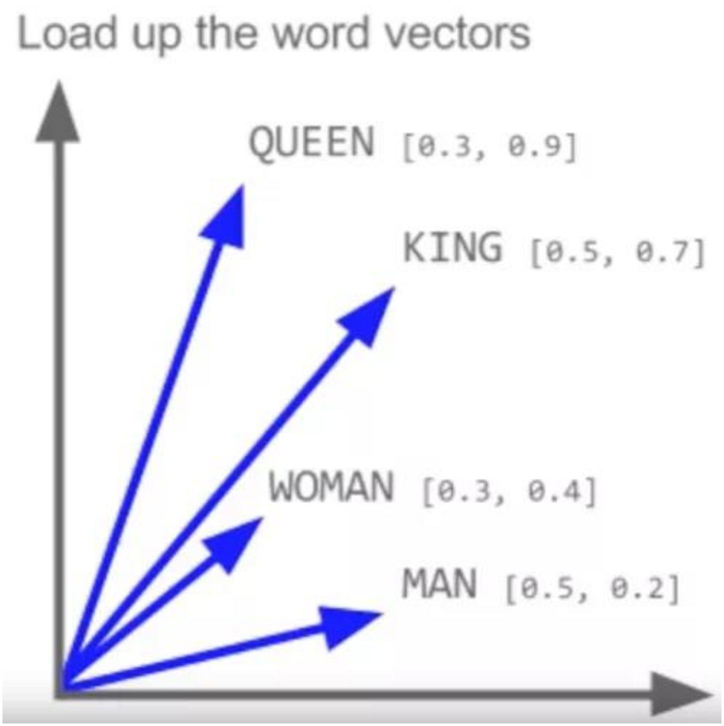
Word Embedding

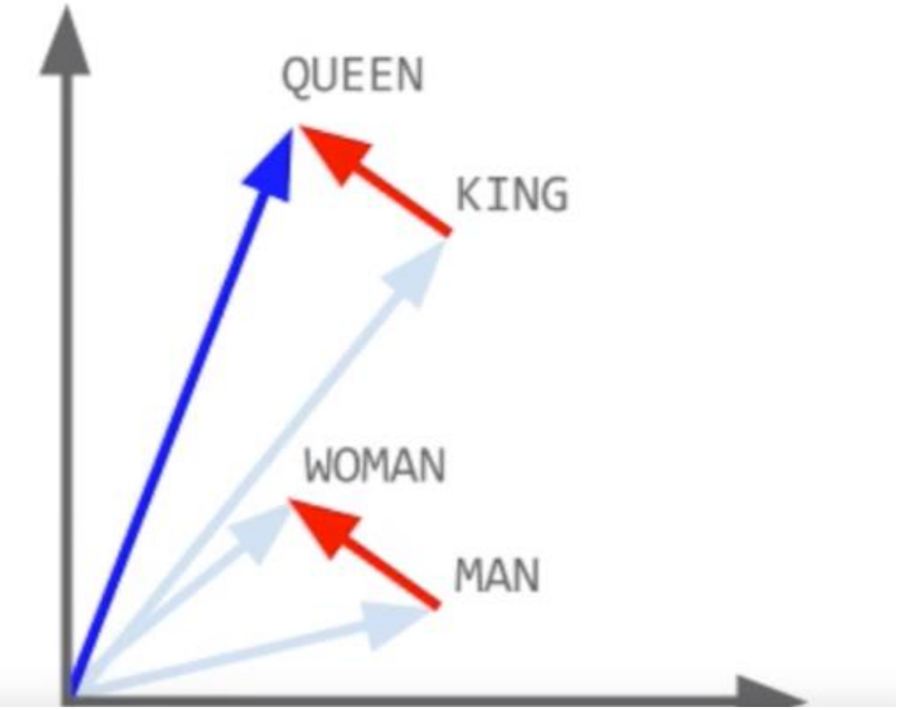
Train a shallow neural network (NN) on a corpus of text

- 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/

OTHER DATA

Weblogs

- typically represented as text strings in a pre-specified format
- this makes it easy to convert them into multidimensional representation of categorical and numeric attributes

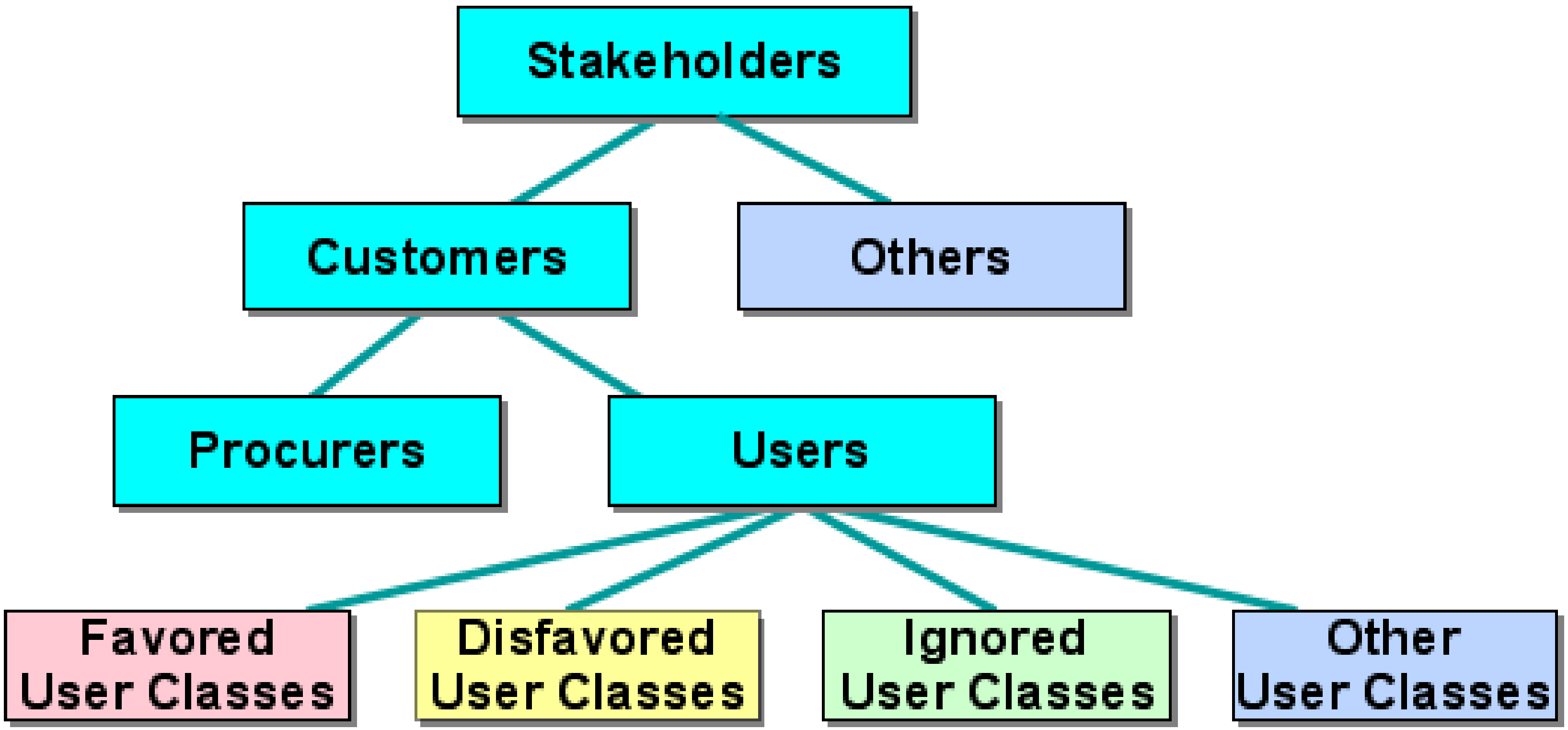
Network traffic

- characteristics of the network packets are used to analyze intrusions or other interesting activity
- a variety of features may be extracted from these packets
  - the number of bytes transferred
  - the network protocol used
  - IP ports used

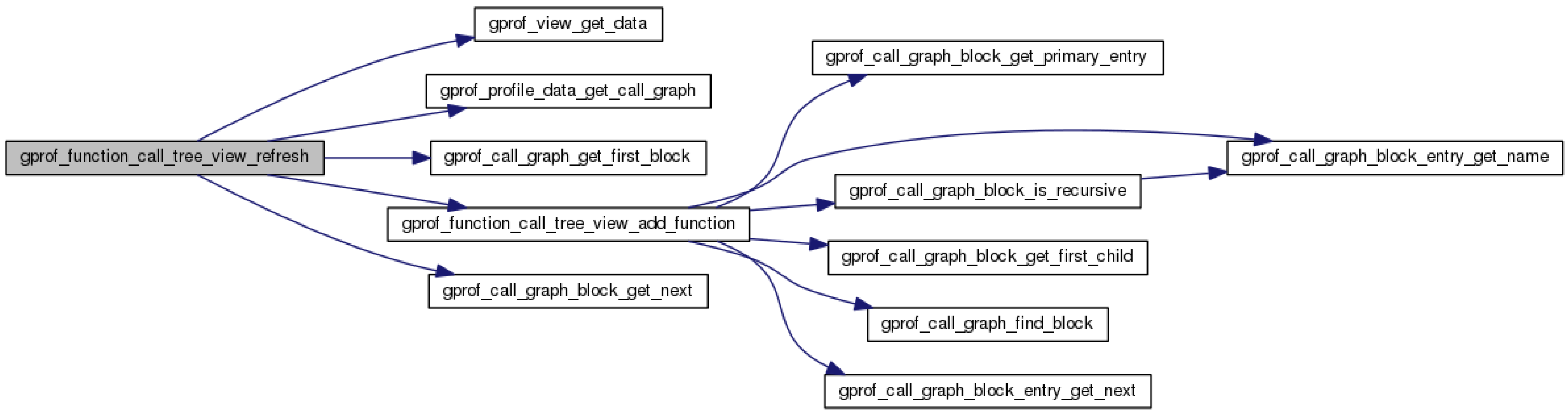
Let's Look at Some Essential Graphical Representations

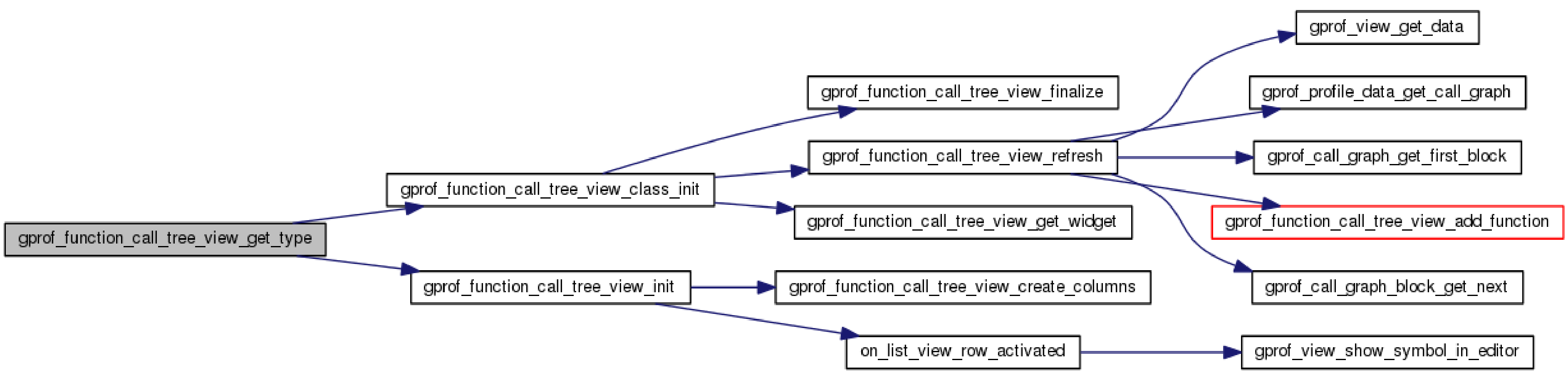
And Do Some Advertising for D3

Stakeholder Hierarchy

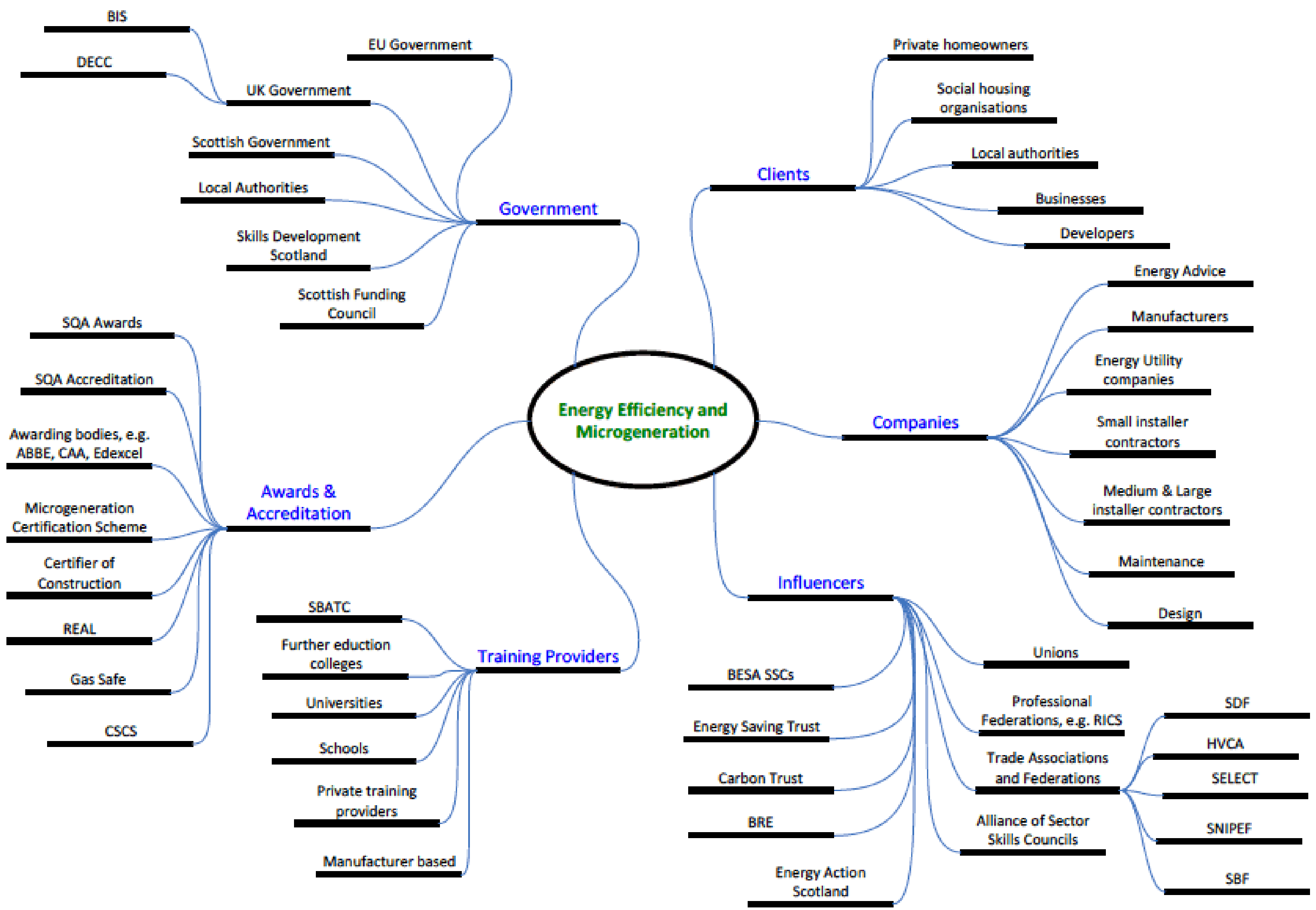


Function Call Tree





More Complex Stakeholder Hierarchy



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

http://bl.ocks.org/kerryrodden/7090426

Bubble Charts

No hierarchy information, just quantities

http://bl.ocks.org/mbostock/4063269

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

http://bl.ocks.org/mbostock/4062006

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

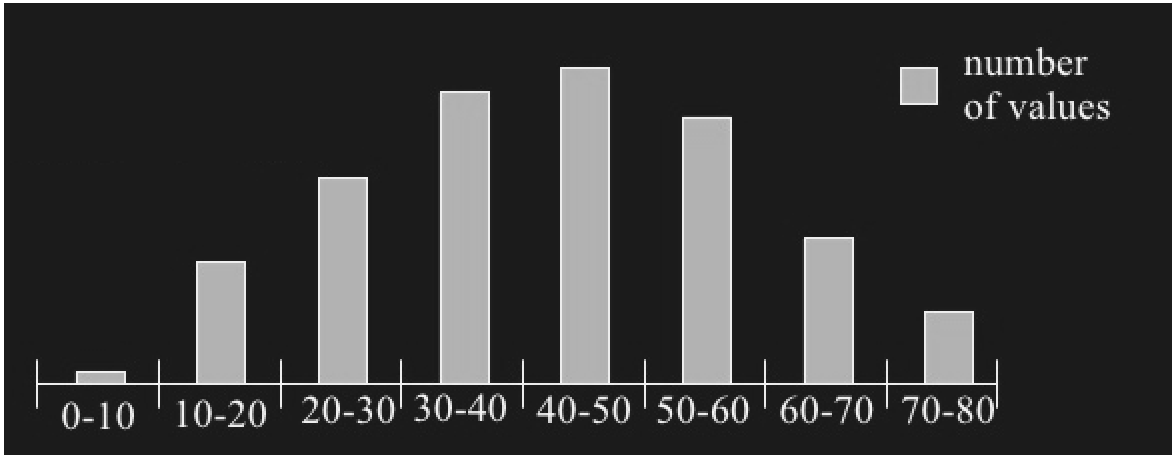
http://bl.ocks.org/mbostock/4060366

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

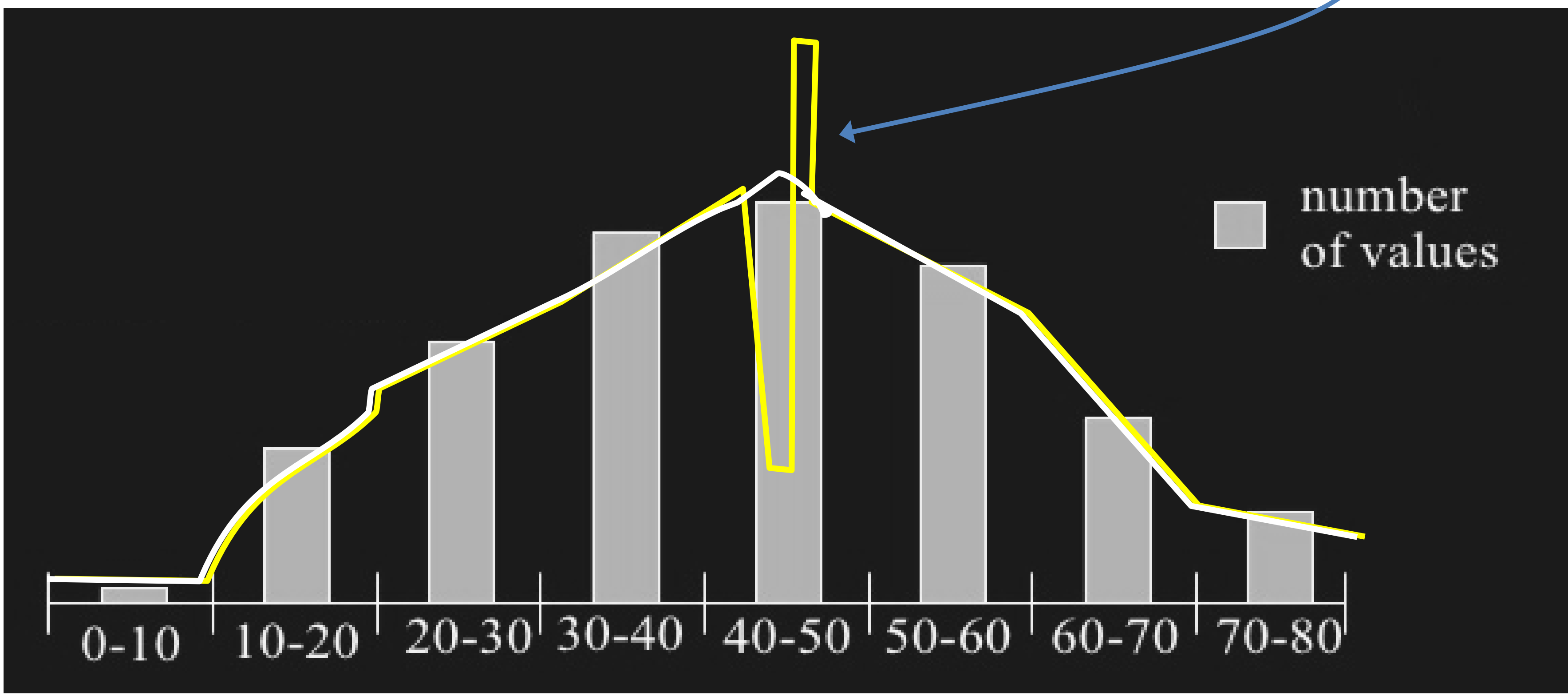


- what is lost here?

Problem With Equi-Width Histogram

Age ranges of customers could be unevenly distributed within a bin

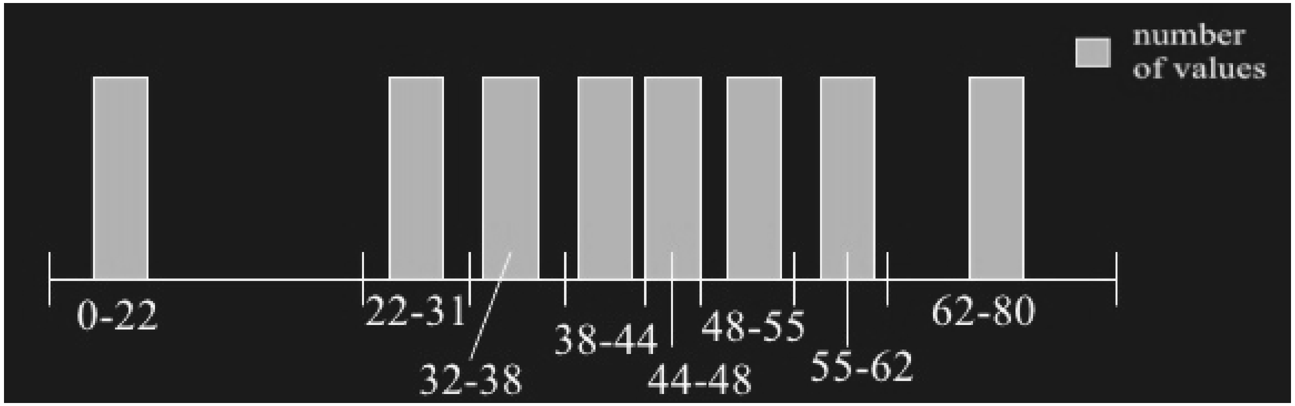
- this could be an interesting anomaly



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:

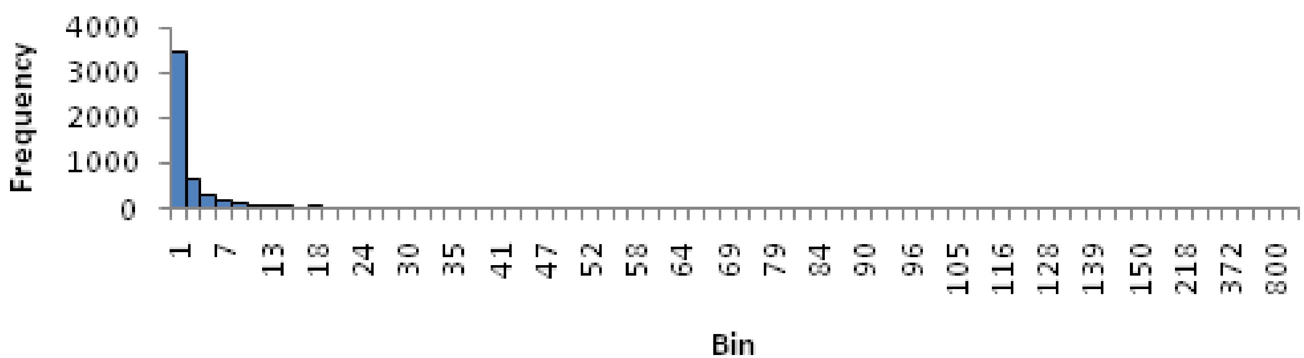


- what is the disadvantage here?
- extra storage needed: must store the start/end value for each bin

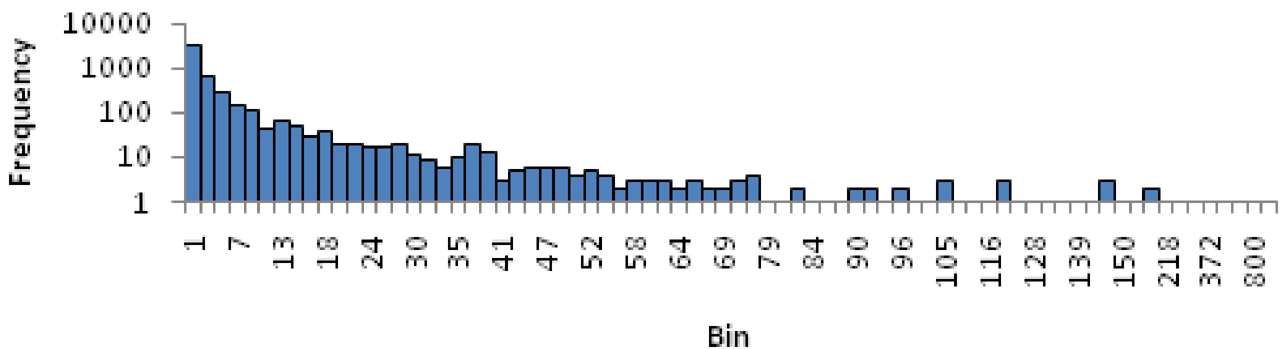
Numeric to Categorical Data: Discretization (3)

Solution 3:

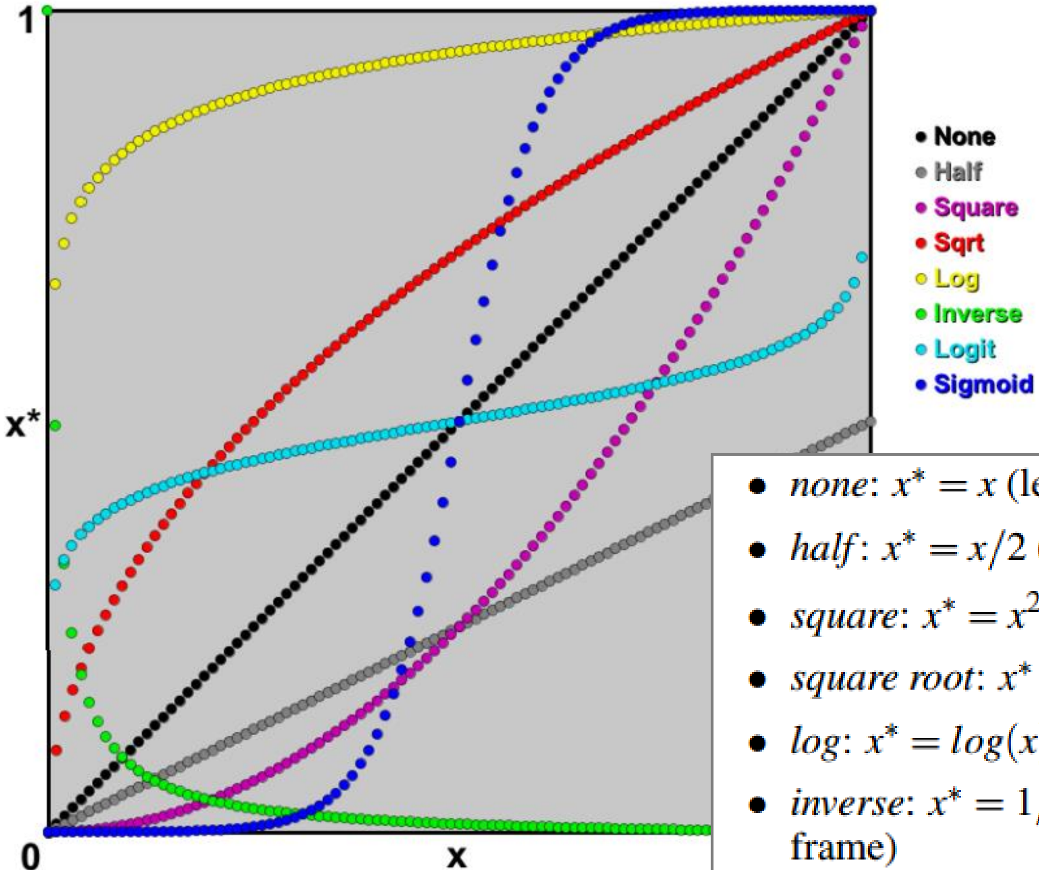
- what if all the bars have seemingly height
- or are dominated by one large peak



- switch to log scaling of the y-value



Other Transformations



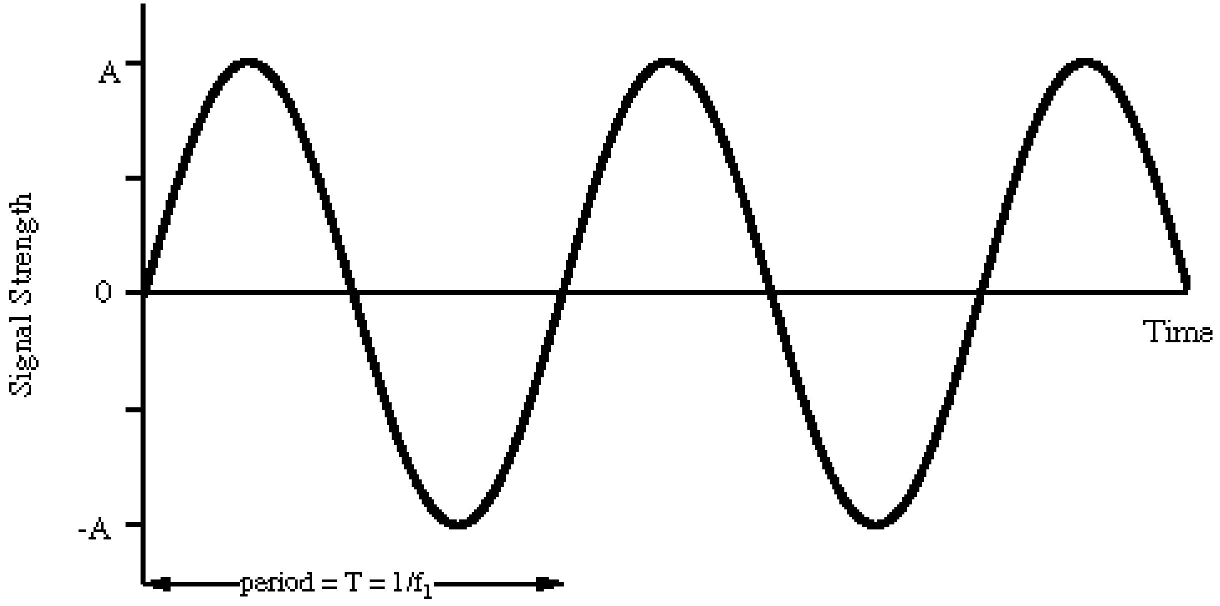
- *none*: $x^* = x$ (leaves points unchanged)
- *half*: $x^* = x/2$ (squeezes all points together)
- *square*: $x^* = x^2$ (pulls points toward left of frame)
- *square root*: $x^* = \sqrt{x}$ (mildly pulls points toward right of frame)
- *log*: $x^* = log(x)$ (strongly pulls points toward right of frame)
- *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

Continuous to Discrete

Why discrete?

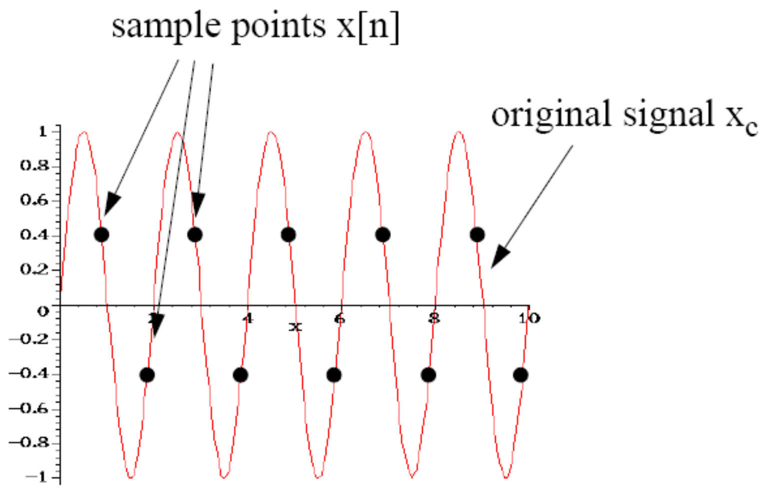
- because we can't store continuous data
- we can only store samples of the continuous data
- how many samples do we need?
- also keep this in mind for data reduction



Continuous to Discrete

Why discrete?

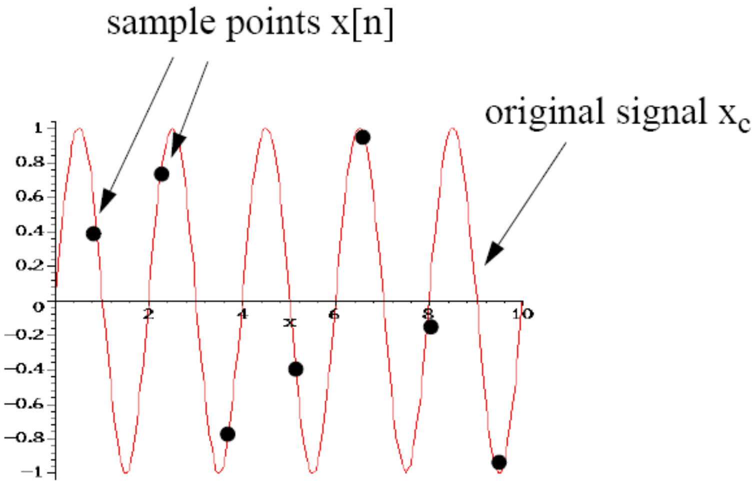
- because we can't store continuous data
- we can only store samples of the continuous data
- how many samples do we need?



Continuous to Discrete

Why discrete?

- because we can't store continuous data
- we can only store samples of the continuous data
- how many samples do we need?



Continuous to Discrete

Why discrete?

- because we can't store continuous data
- we can only store samples of the continuous data
- how many samples do we need?

We need a certain number of samples to represent a continuous phenomenon

- twice as many samples as the highest frequency in the signal
- called the *Nyquist frequency*
- else we get *aliasing*

Practical Implications

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

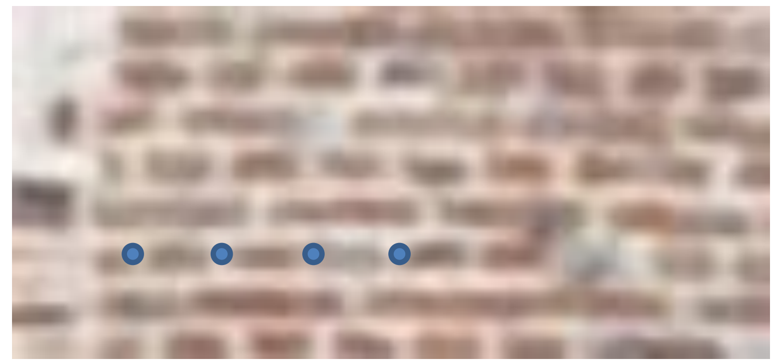
This is aliasing

Practical Implications

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*

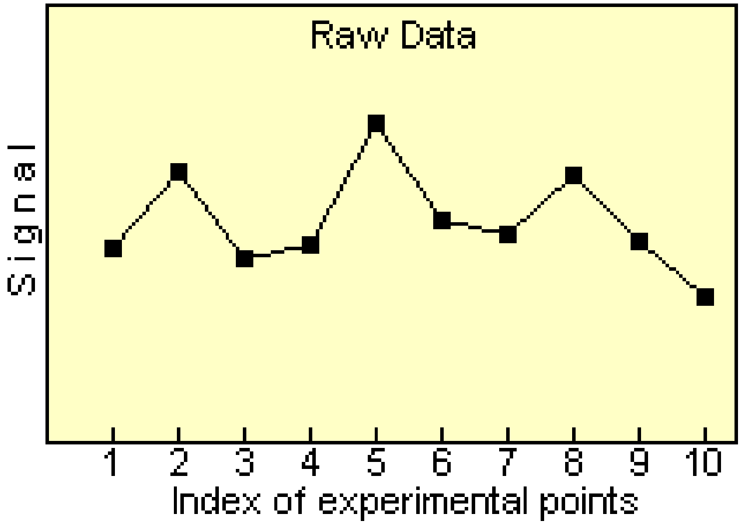
Anti-Aliasing Via Smoothing

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



Anti-Aliasing Via Smoothing: Tradeoffs

looks sharper, but has “jaggies” a bit blurred, but no more jaggies

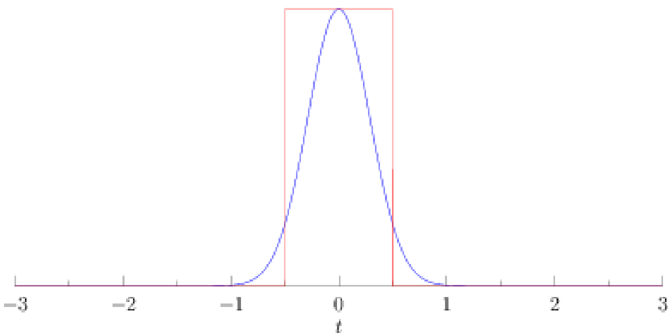
Filters

What is the filter we just used called?

- it's called a *box filter*

There are other filters

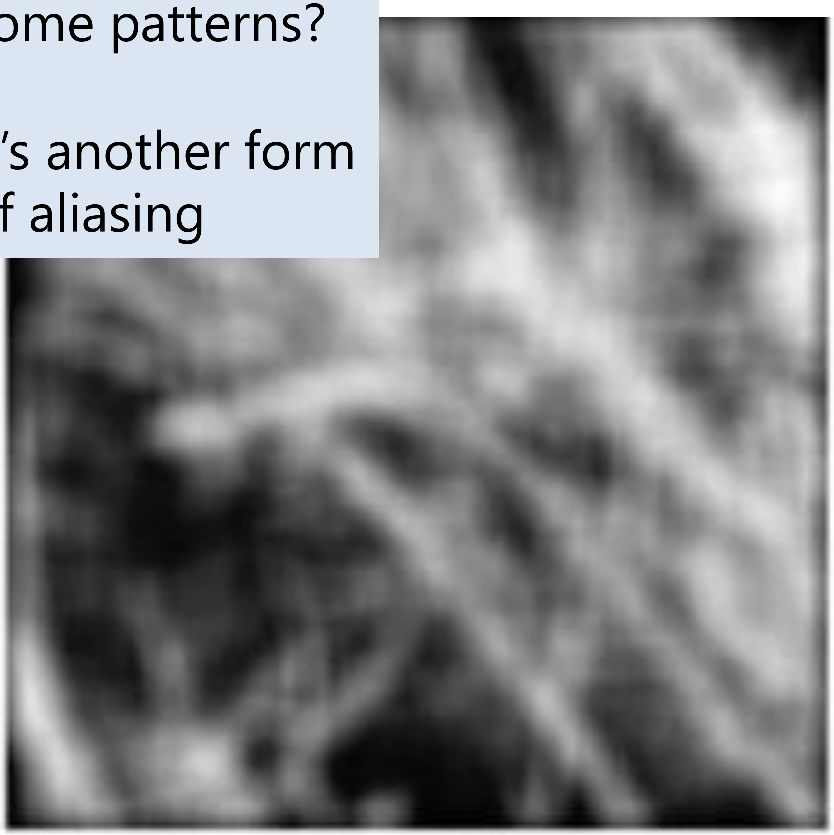
- for example, Gaussian filter
- yields a smoother result
- box filtering is simplest

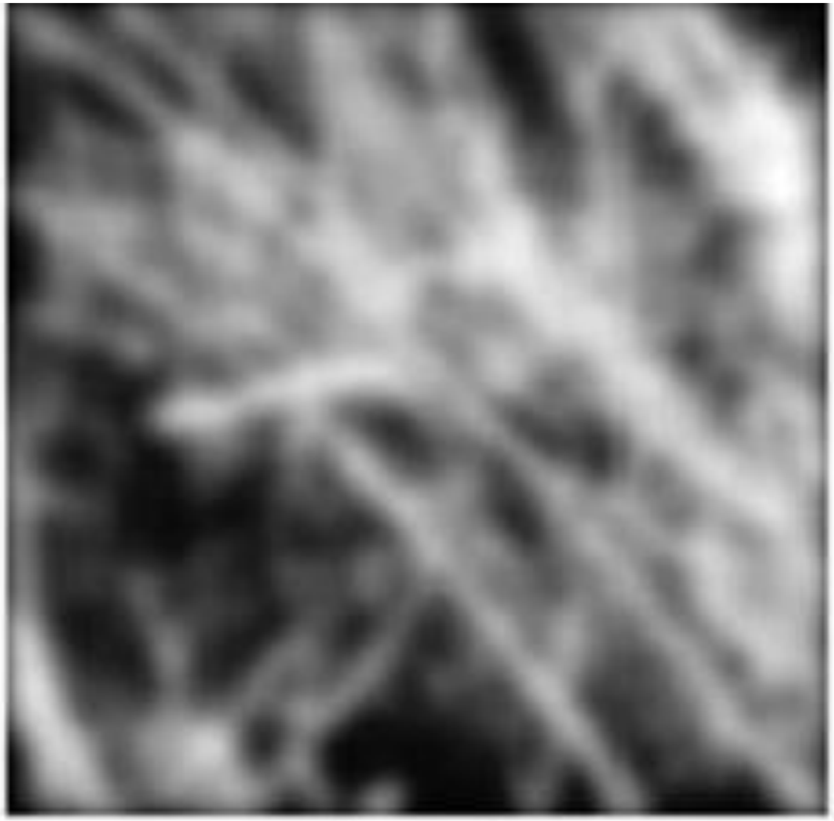


Box Filter vs. Gaussian Filter

Can you see some patterns?

It's another form of aliasing





2D box

2D Gaussian

The Solution

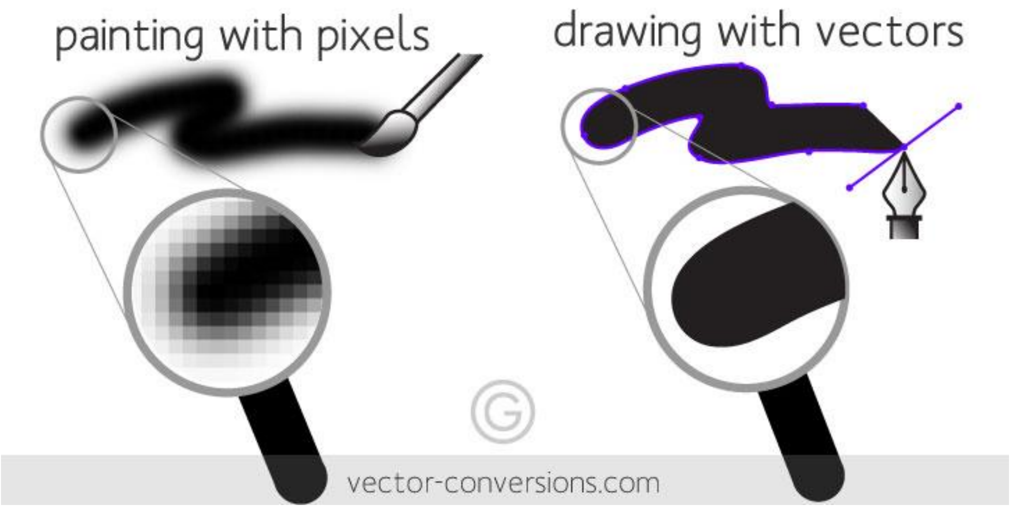
What's the underlying problem?

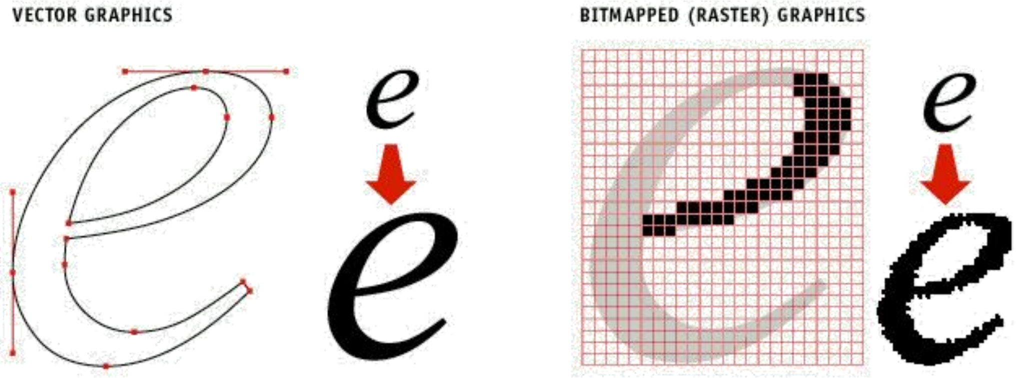
- detail can't be refined upon zoom
- can just be replicated or blurred

The solution…

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

Scalable Vector Graphics (SVG)





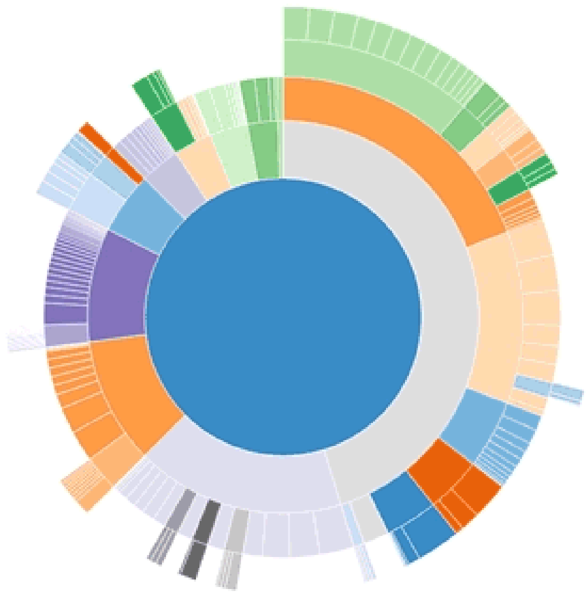
Photographs and Images in SVG

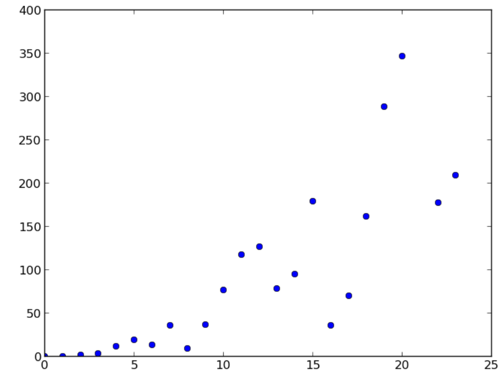
Vector graphics tends to have an “cartoonish” look

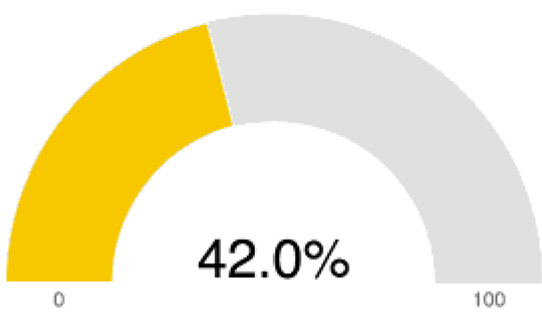
raster graphics vector graphics

Photographs and Images in SVG

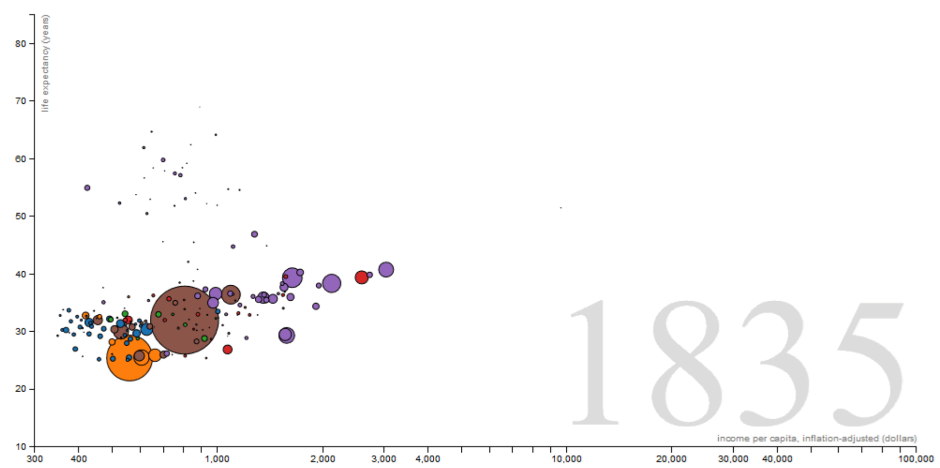
D3 Uses SVG





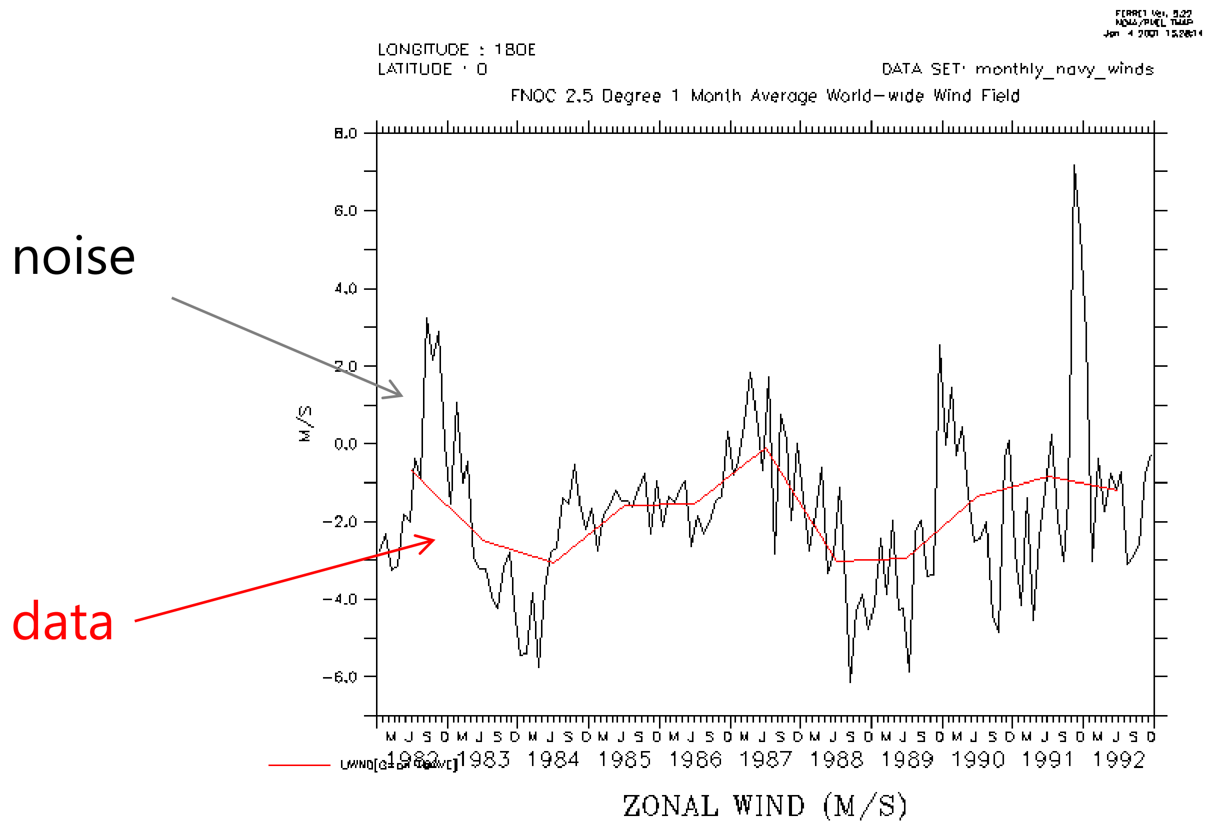


The Wealth & Health of Nations



Smoothing for De-Noising

Filtering/smoothing also eliminates noise in the data



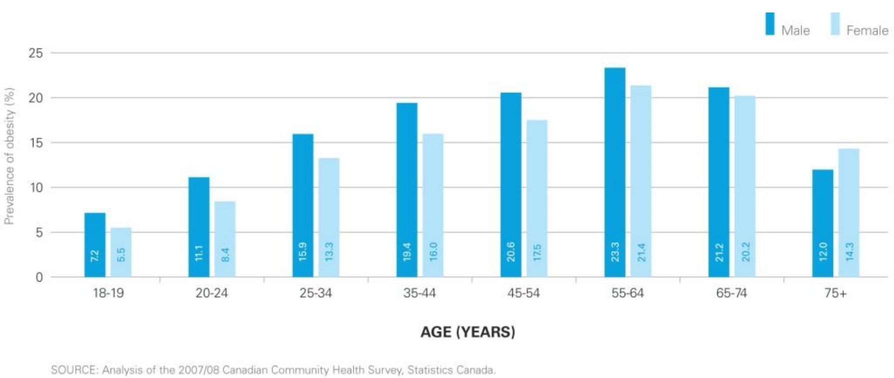
Back to Bar Charts

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

- the bins do the smoothing

Example:

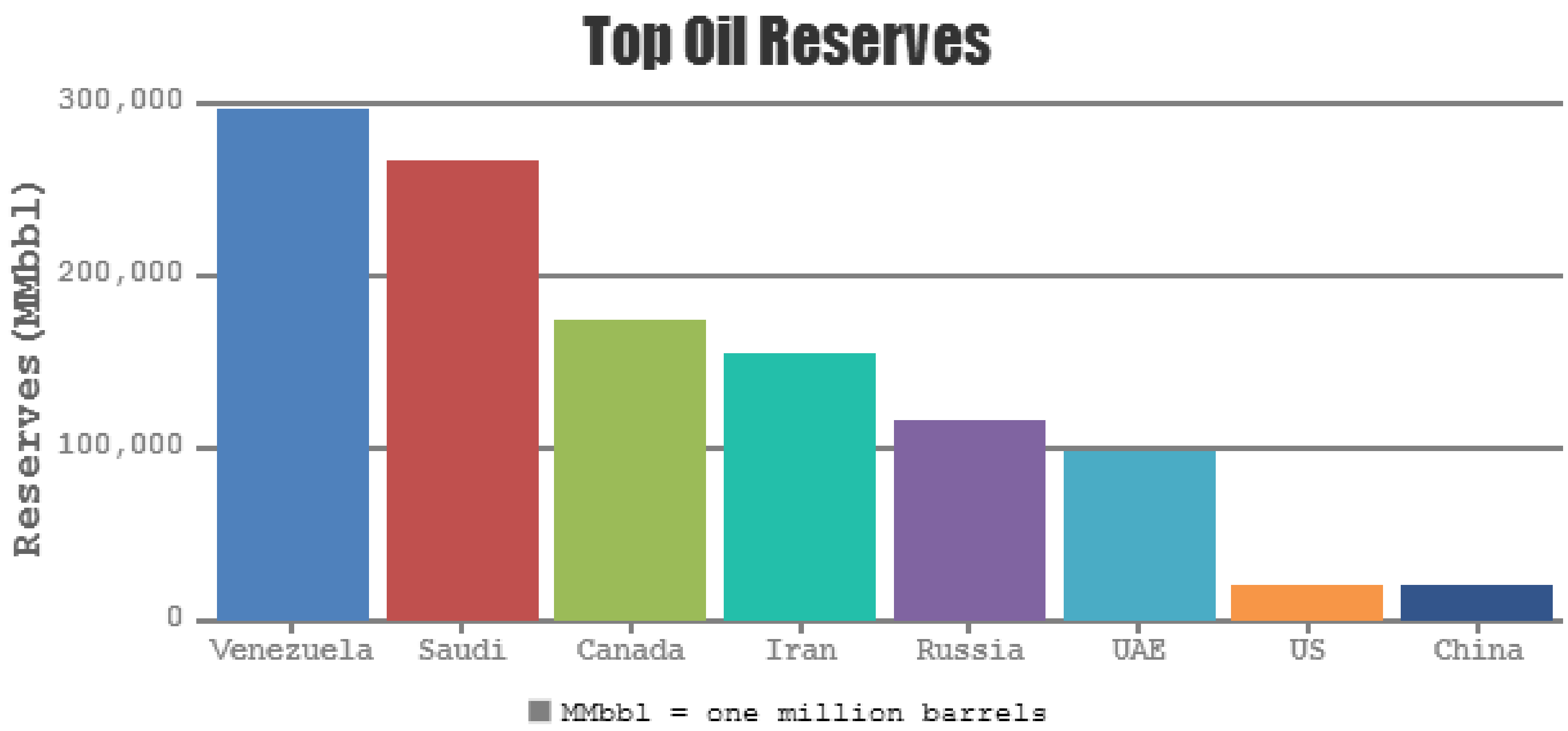
- obesity over age (group)



Bar Charts

Of course, bar charts can also hold categorical data

- smoothing by semantic grouping
- for example, Europe vs. {France, Spain, Italy, Germany, …}



Bar Charts in D3

http://bl.ocks.org/mbostock/3885304

Working with bar charts will be your job for Lab 1

- 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

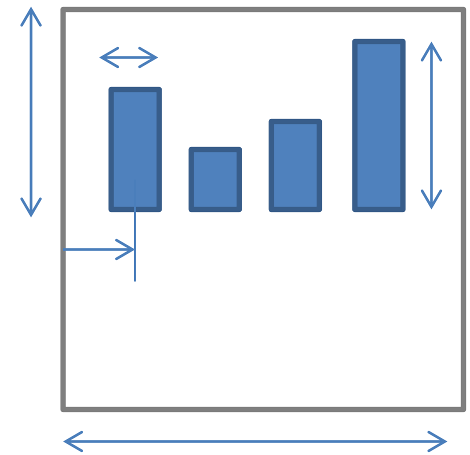
$$bin\ size = \frac{\max(data) - \min(data)}{number\ of\ bins}$$

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