GETTING TO KNOW YOUR DATA

Hari Sundaram

hs1@illinois.edu

http://sundaram.cs.illinois.edu

adapted from slides by Jiawei Han and Kevin Chang

thank you for responding to the survey

research

learning new ideas

big data

AI

Excited

Career

hot topic

Getting in

preparation
- coding skills
- different major
- Linear Algebra

workload
- Homework

difficulty
- getting an A
- MP's / Quiz

Concerned

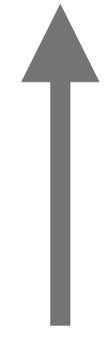
No concerns

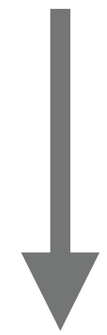
course relevance
- real world validity of knowledge
- obsolete topics

CS 205: data discoveries

CS 412: data mining

CS 446 / 498: ML / Applied ML





Apply by Jan 31st!

Eric & Wendy Schmidt

Data Science For Social Good

Summer Fellowship

THE UNIVERSITY OF CHICAGO

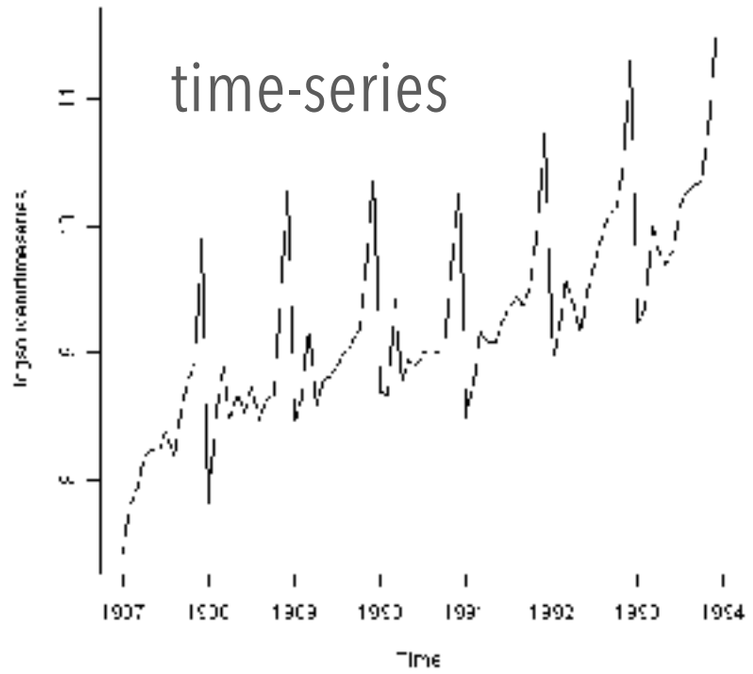
We're training data scientists to tackle problems that really matter.

Applications for Summer 2016 are now open. Click on the links below to learn more and apply.

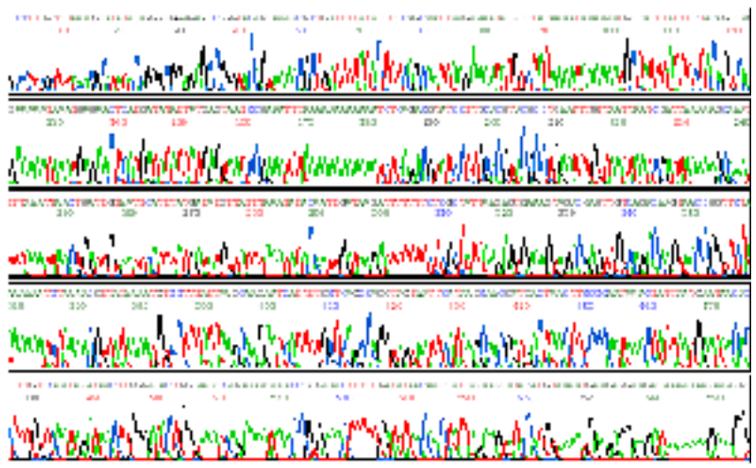
http://dssg.uchicago.edu/# http://www.datakind.org

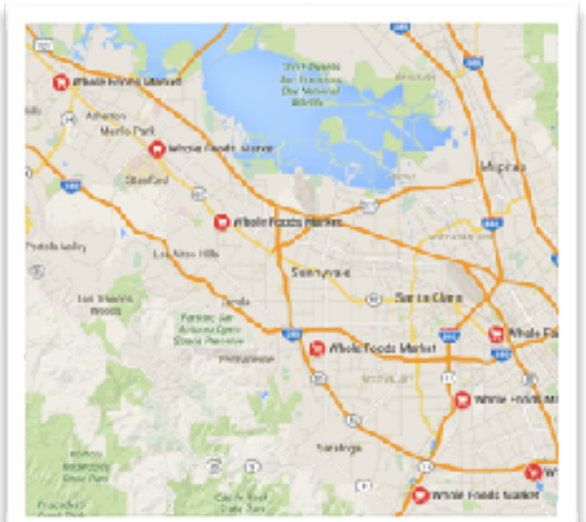
TYPES OF DATASETS

time-series



Genetic sequence data





Types of Datasets

Relational records

	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

Document data: text documents: term-frequency vector

Record

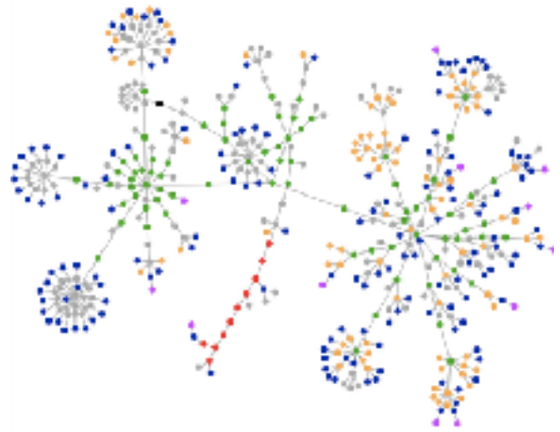
Transaction data

TID	*Items*
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

Data matrix, e.g., numerical matrix, crosstabs

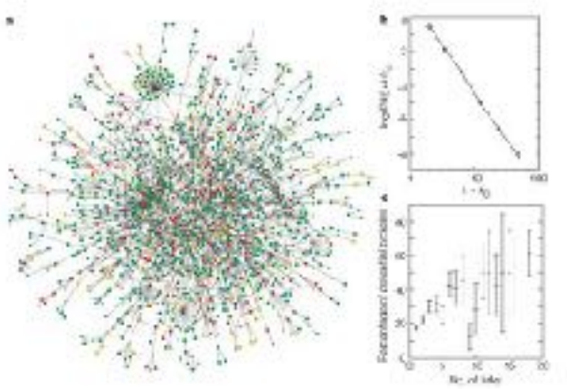
Types of Datasets

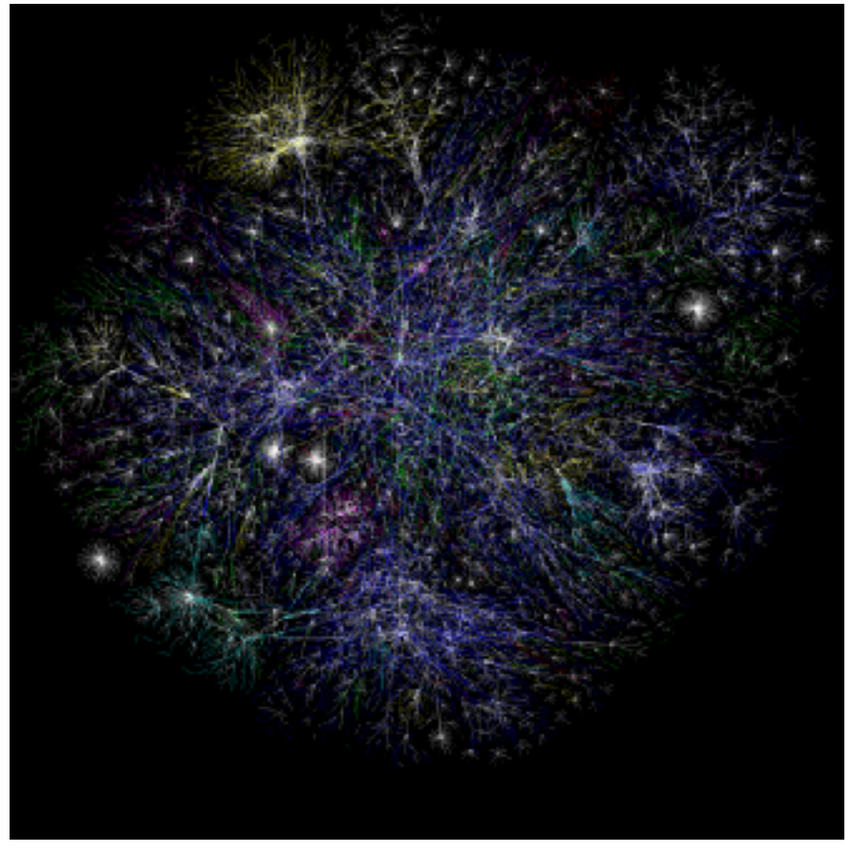
Social or information networks



Graphs and networks

Molecular Structures

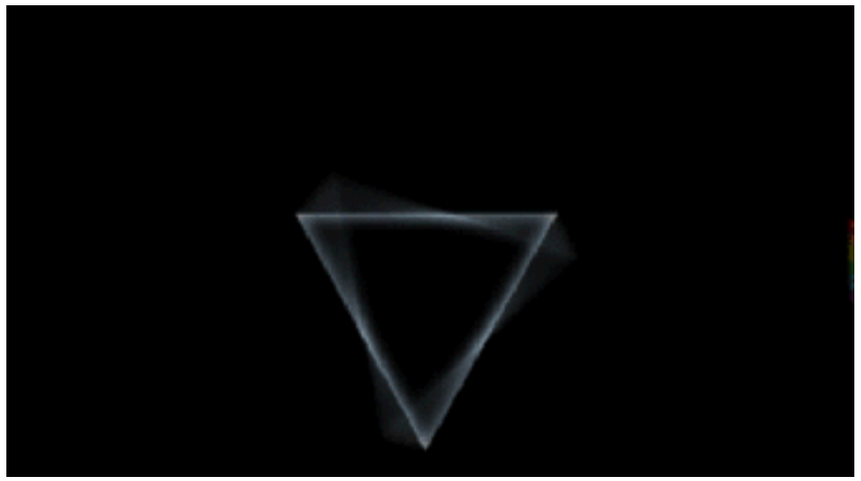




World Wide Web

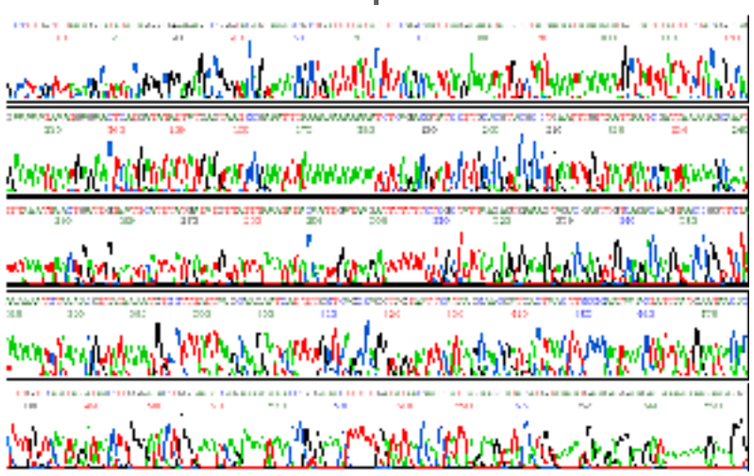
Types of Datasets

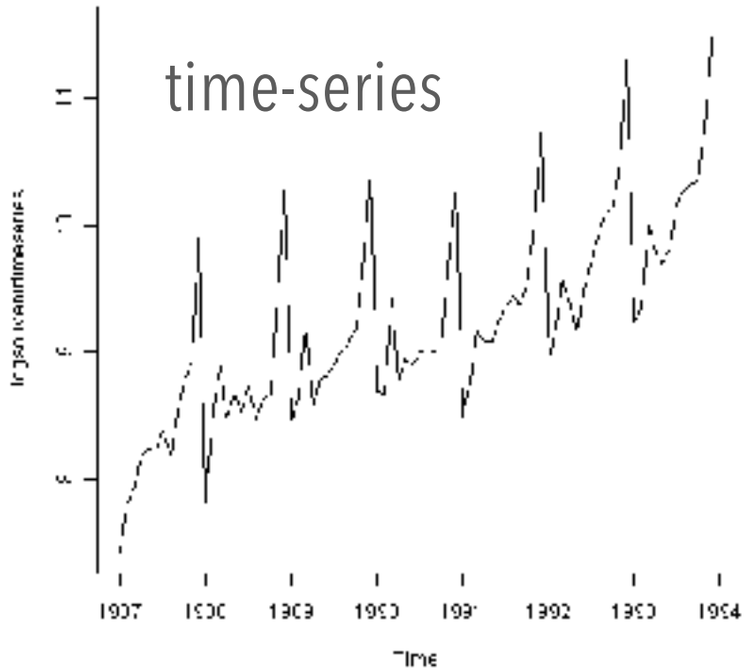
Video data: sequence of images



Ordered

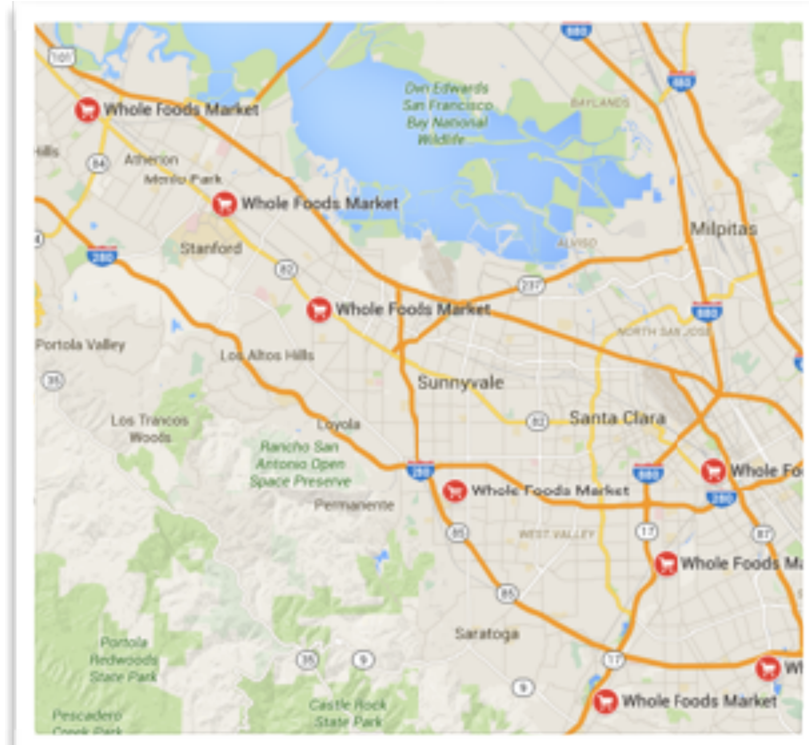
Genetic sequence data





Sequential Data: transaction sequences

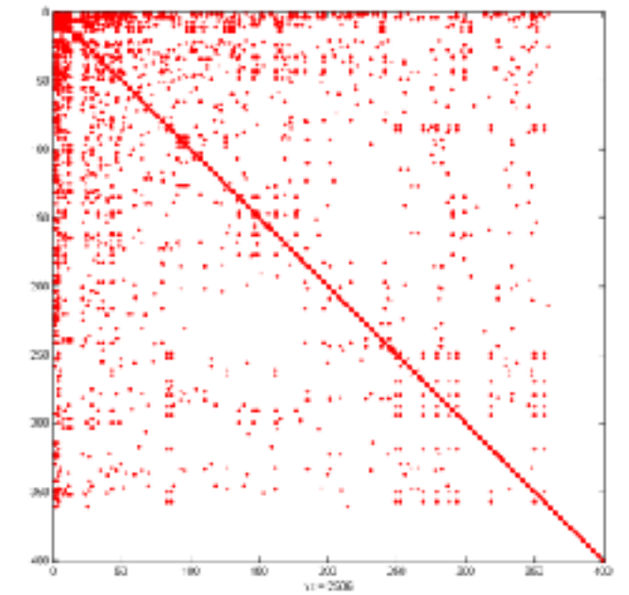
Types of Datasets

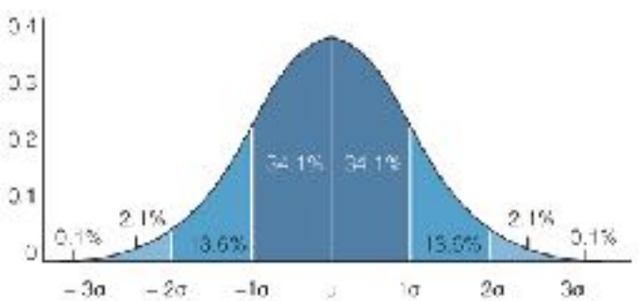


Spatial, image and multimedia

multiple sources

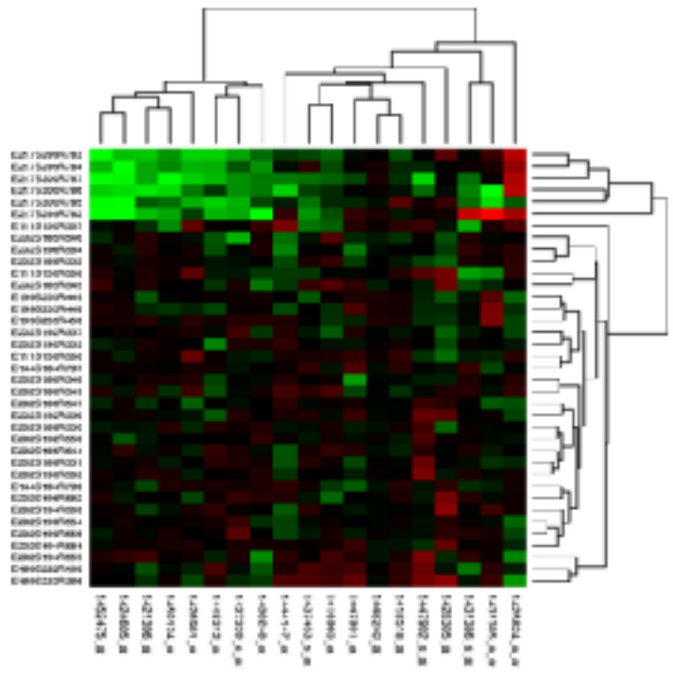
IMPORTANT CHARACTERISTICS OF STRUCTURED DATA



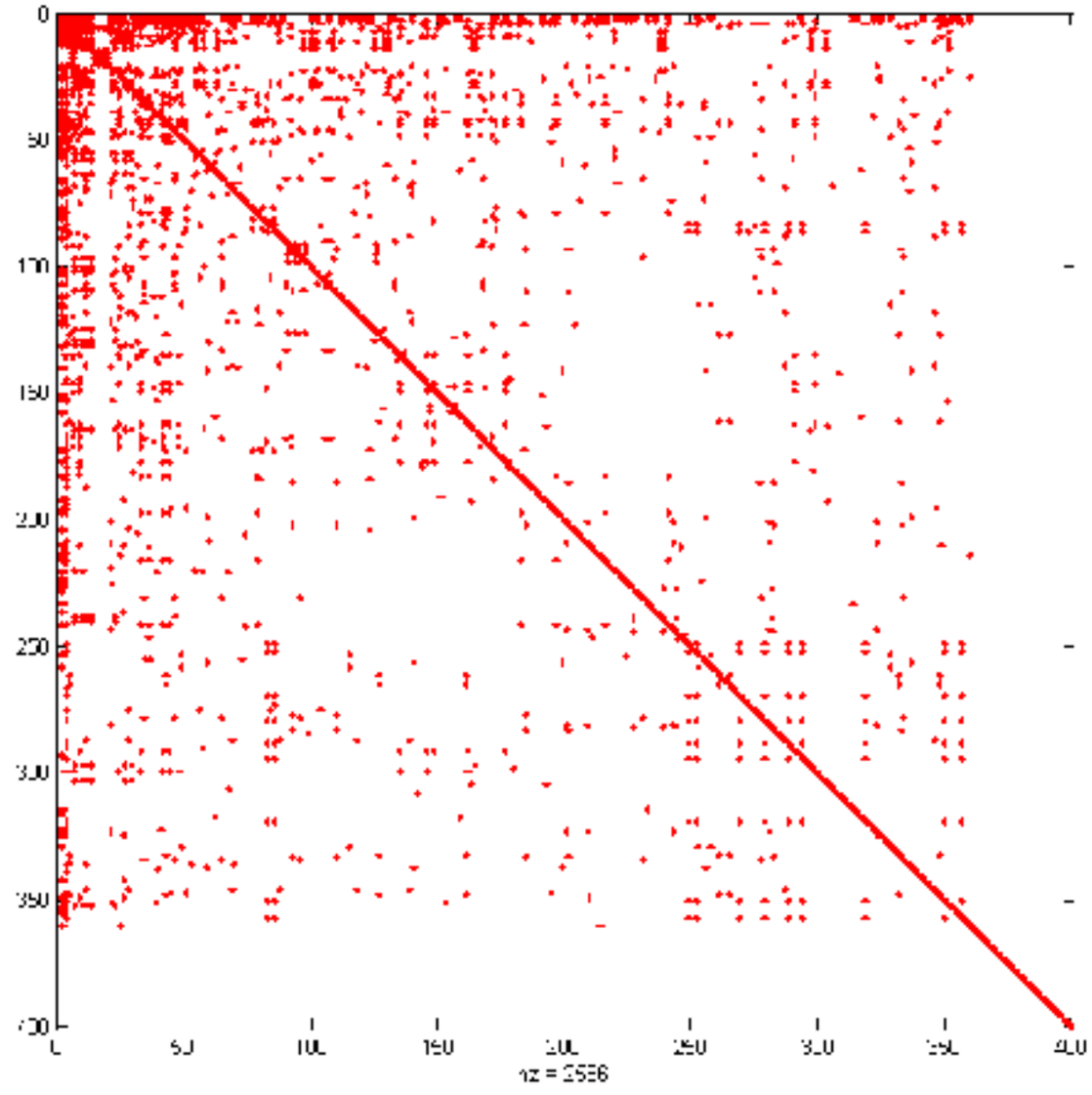


curse

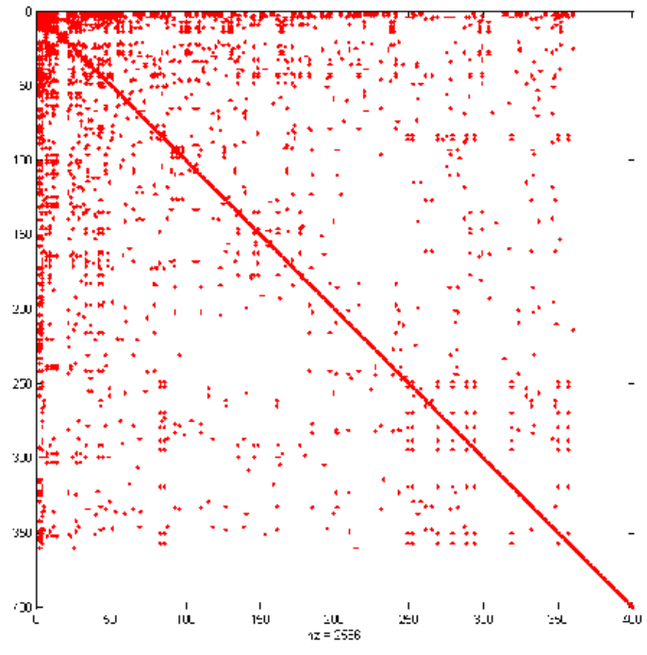
Dimensionality



Sparsity



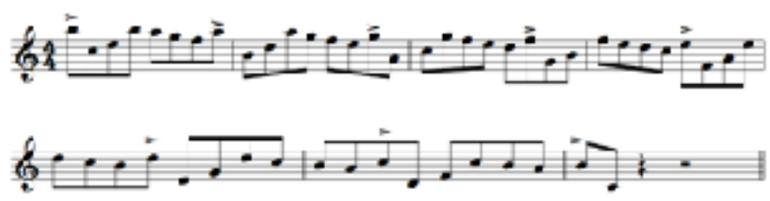
sparsity pattern of a co-occurrence matrix



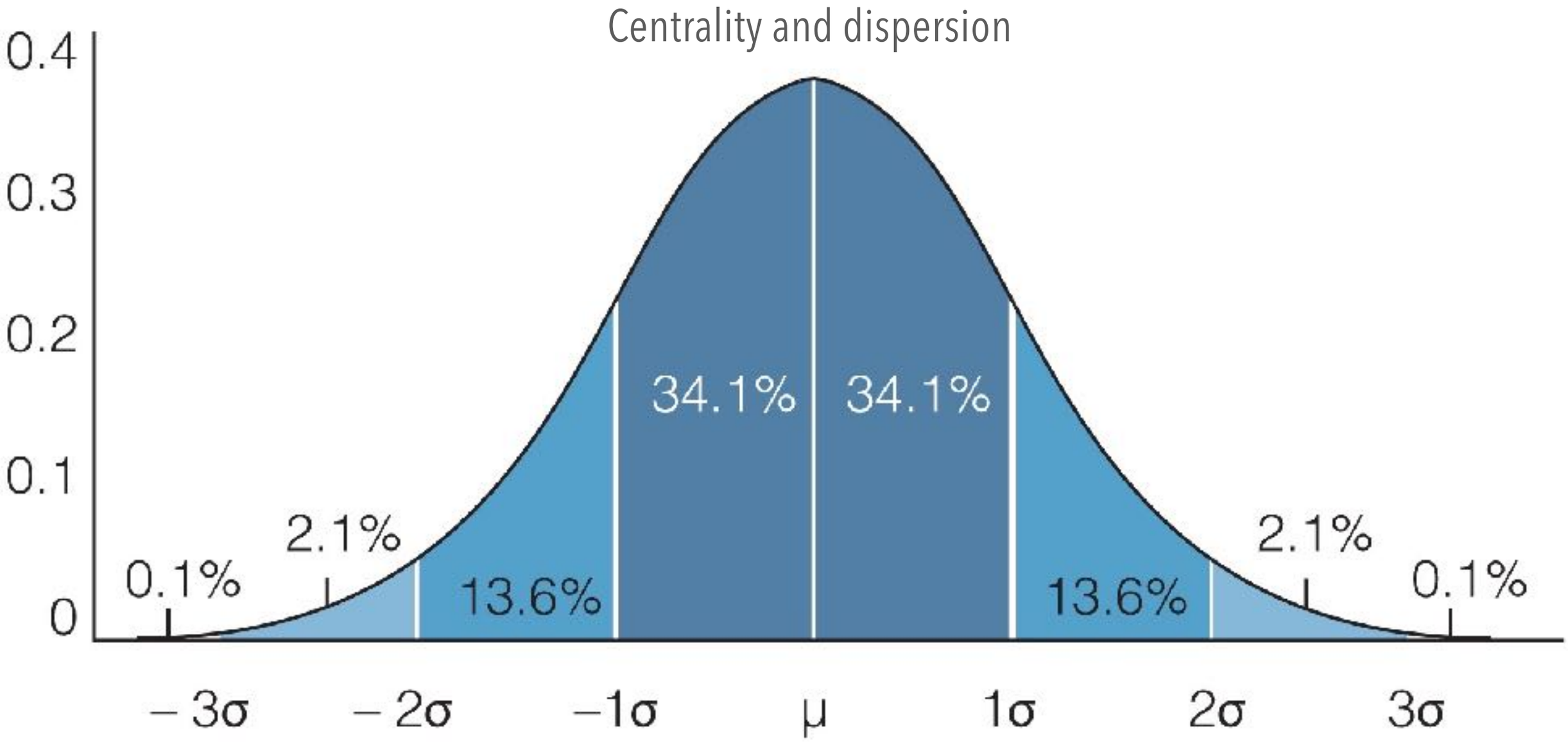

Why
do we
care?

pattern depends on scale

Resolution



Distribution



DATA

OBJECTS, ATTRIBUTES

a data object represents an entity

medical database:
patients, treatments

sales database:
customers, store items, sales

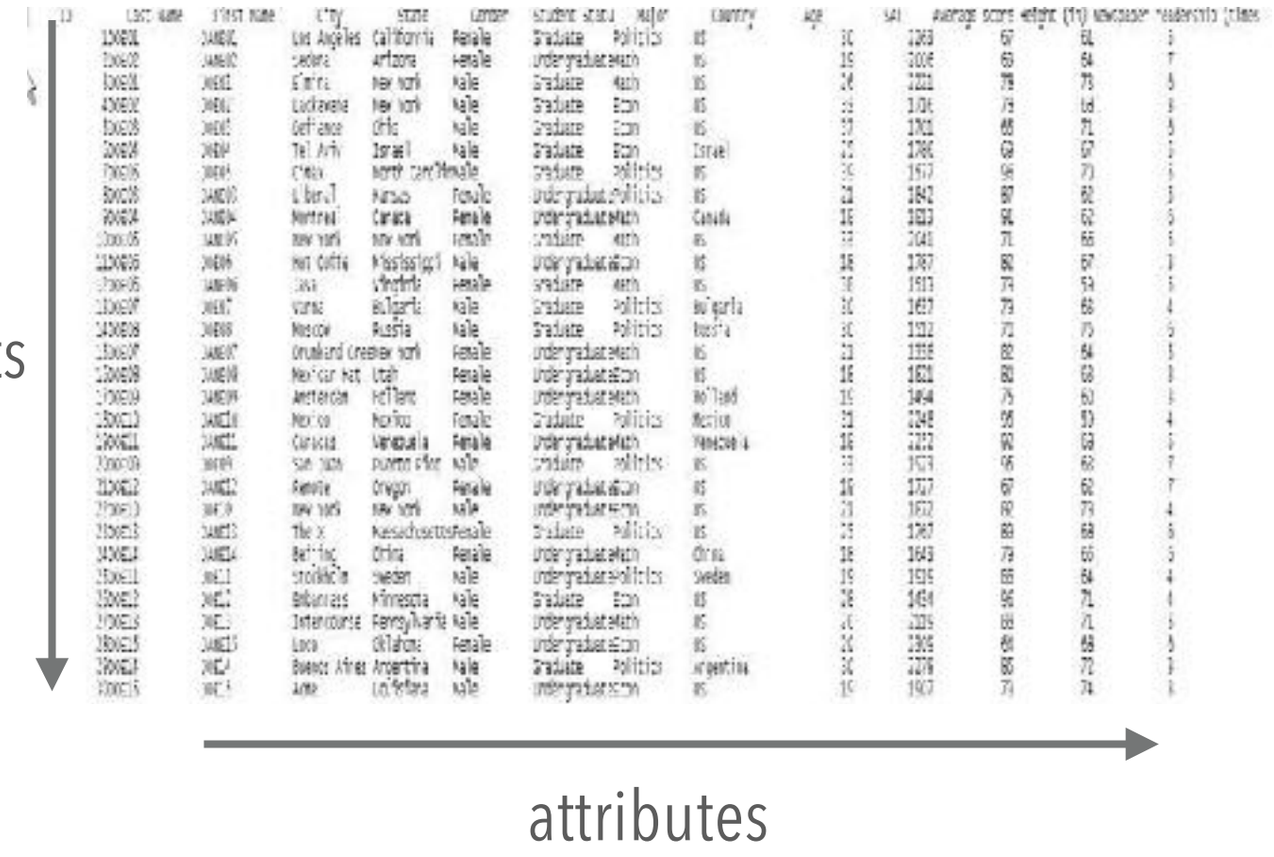
university database:
students, professors, courses

Also called samples , examples, instances, data points, objects, tuples.

Data objects are described by attributes.

datsets

data objects make up datasets



E.g., customer _ID, name, address

Attributes

(or dimensions, features, variables): a data field, representing a characteristic or feature of a data object.

Nominal (or categorical)

{green, red, blue}

Binary {true, false}

Types

Numeric

quantitative

Interval-scaled

Ratio-scaled

Ordinal

{small, medium, large}

marital status,
occupation, ID numbers,
zip codes

Nominal

(aka Categorical)

Hair color = {auburn, black,
blond, brown, grey, red,
white}

Exercise!

WHAT IS THE AVERAGE COLOR OF THIS IMAGE?

Asymmetric binary: outcomes not equally important. e.g., medical test (positive vs. negative) Convention: assign 1 to most important outcome (e.g., HIV positive)

Binary

Nominal attribute with only two states

Symmetric binary: both outcomes equally important

Size = {small, medium, large}, course grades, army rankings

Ordinal

Values have a meaningful order (ranking) but magnitude between successive values is not known.

Numeric

quantity: 7.8, 9, 101.89 etc.

Interval

Measured on a scale of equal-sized units

Values have order

e.g., temperature in C°or F°, calendar dates

No true zero-point

Inherent zero-point

Ratio

e.g., temperature in Kelvin, length, counts, monetary quantities

We can speak of values as being an order of magnitude larger than the unit of measurement (10 K° is twice as high as 5 K°).

Exercise!

It is 0°F in Chicago; **when will it be twice as hot?**

E.g., zip codes, profession, or the set of words in a collection of documents

Sometimes represented as integer variables

Discrete

Has only a finite or countably infinite set of values

Note: Binary attributes are a special case of discrete attributes

Has real numbers as attribute values

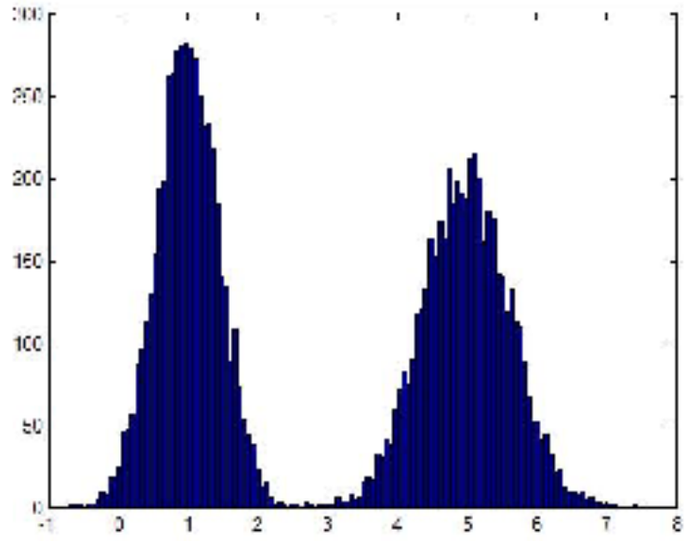
In practice, attribute values can only be measured and represented using a finite number of digits

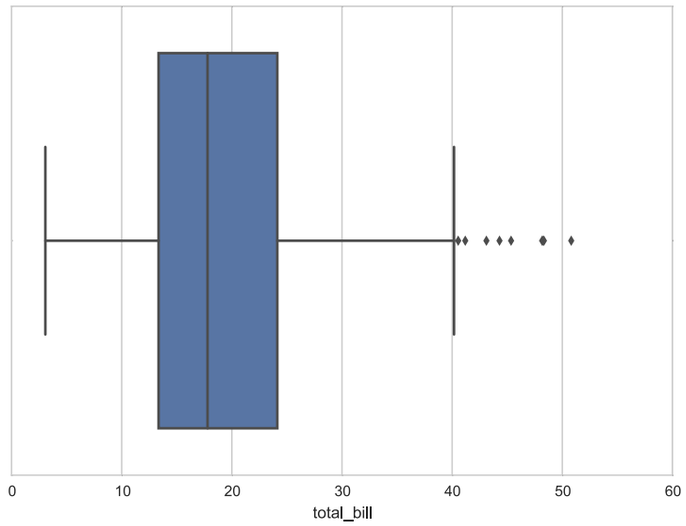
Continuous

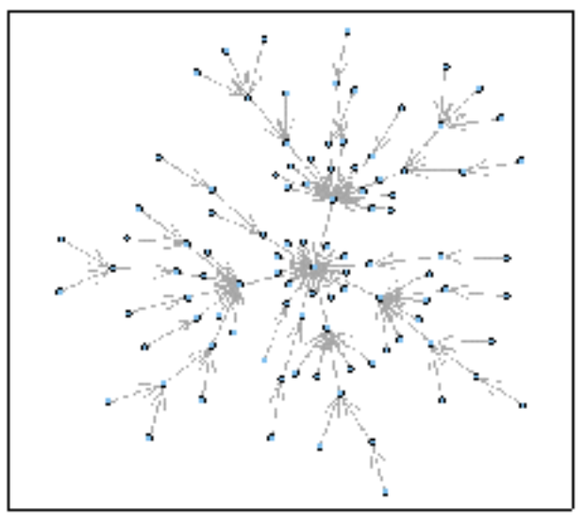
Continuous attributes are typically represented as floating-point variables

e.g., temperature, height, or weight;
weight=125.6 lb.

BASIC STATISTICAL DESCRIPTIONS OF DATA



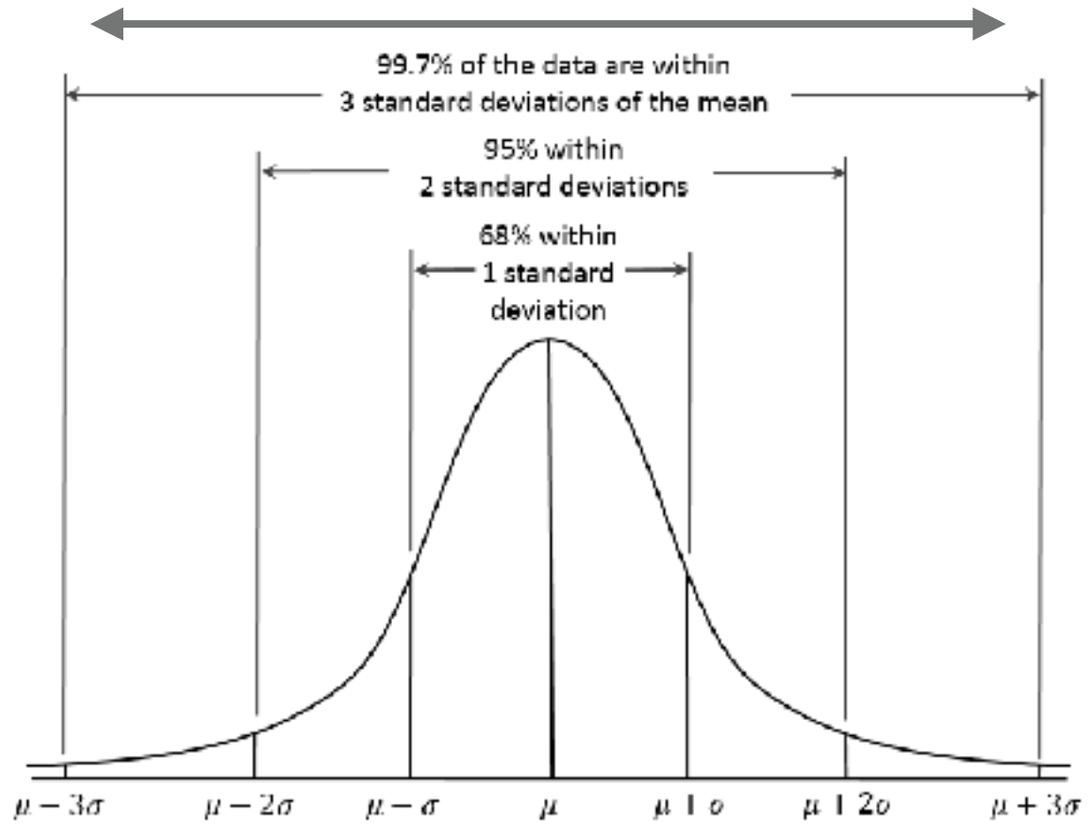




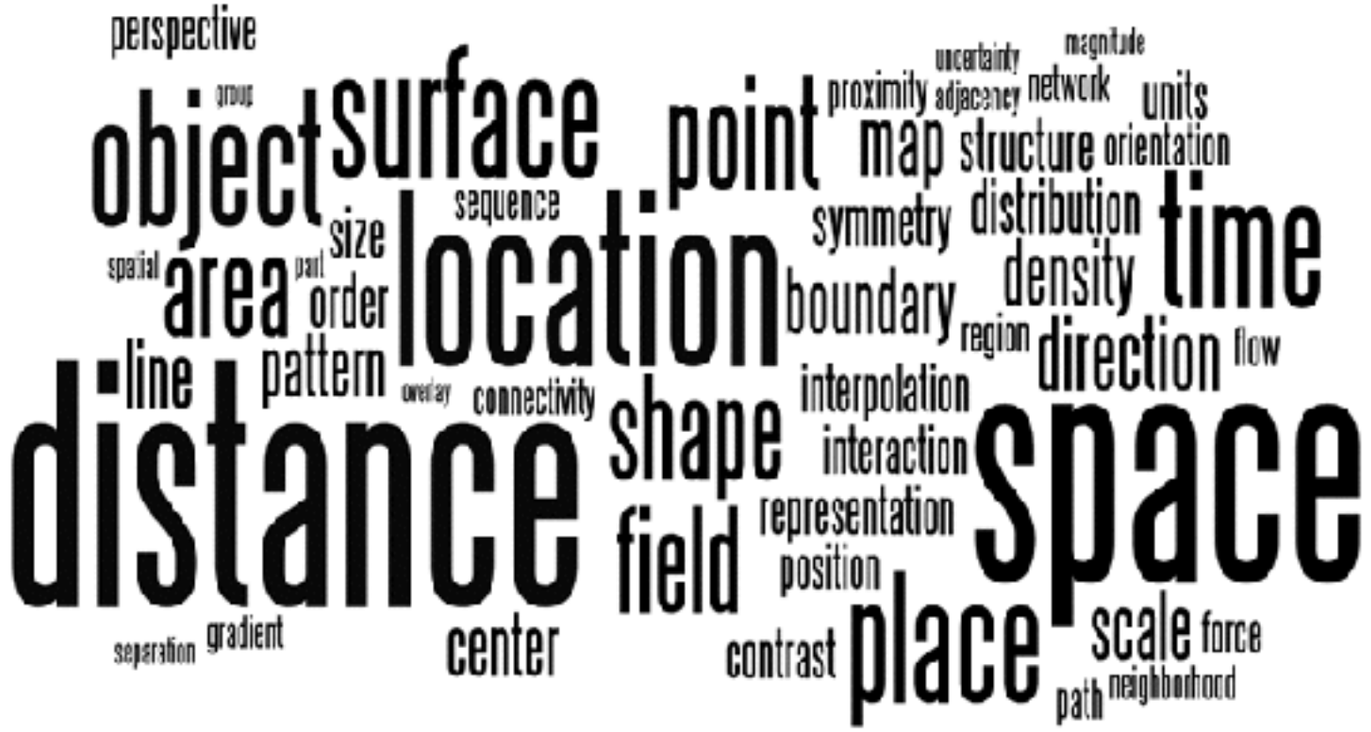
Motivation

To better understand the data.

Dispersion–variation, spread.



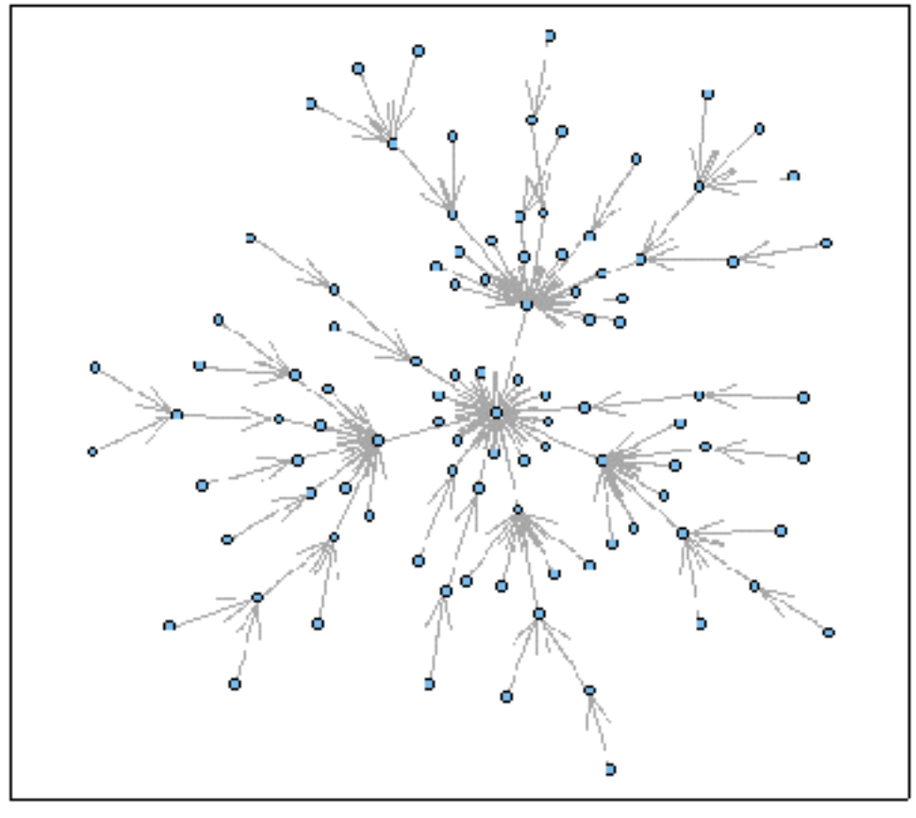
Central tendency–the center, the representative.



TF-IDF

Is that all?

non-numeric data



statistics from graphs

The mean

Trimmed mean: chopping extreme values

weighted sample mean

$$\bar{x} = \frac{\sum_i^n \omega_i x_i}{\sum_i^n \omega_i}$$

population mean

$$\mu = \frac{1}{N} \sum_i^N x_i$$

population size

different notation

sample mean

$$\bar{x} = \frac{1}{n} \sum_i^n x_i$$

sample size

The median

Middle value if odd number of values, or average of the middle two values otherwise

Estimated by interpolation (for grouped data):

age	*frequency*	
1–5	200	
6–15	450	
16–20	300	median is
21–50	1500 ←	to be
51–80	700	found here
81–110	44	

approximate median calculation

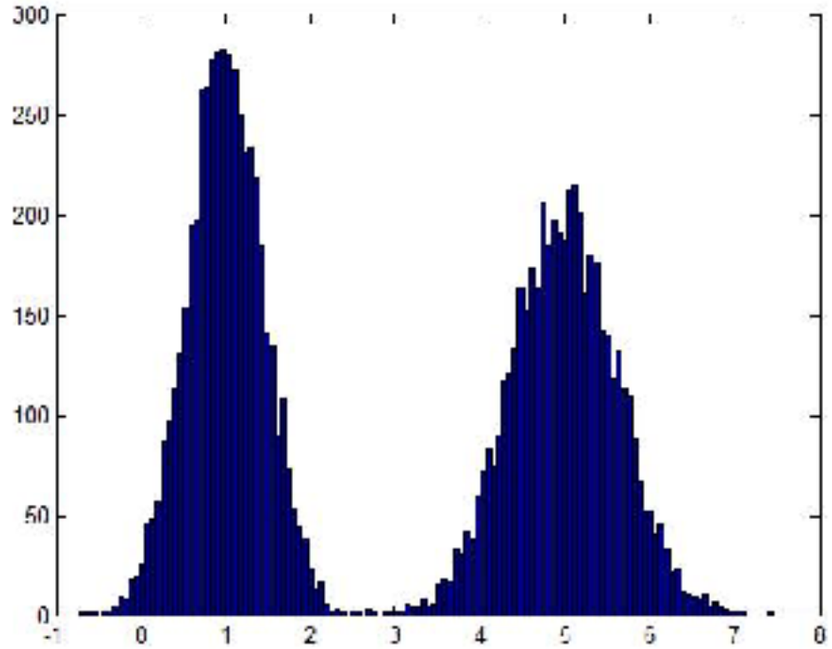
$$L_1 + \left(\frac{n/2 - \sum_l f_l}{f_{median}} \right) \times (L_2 - L_1)$$

L_1: lower interval limit

$\sum_l f_l$: sum before the median interval

$(L_2 - L_1)$: interval width

Unimodal,
bimodal, trimodal

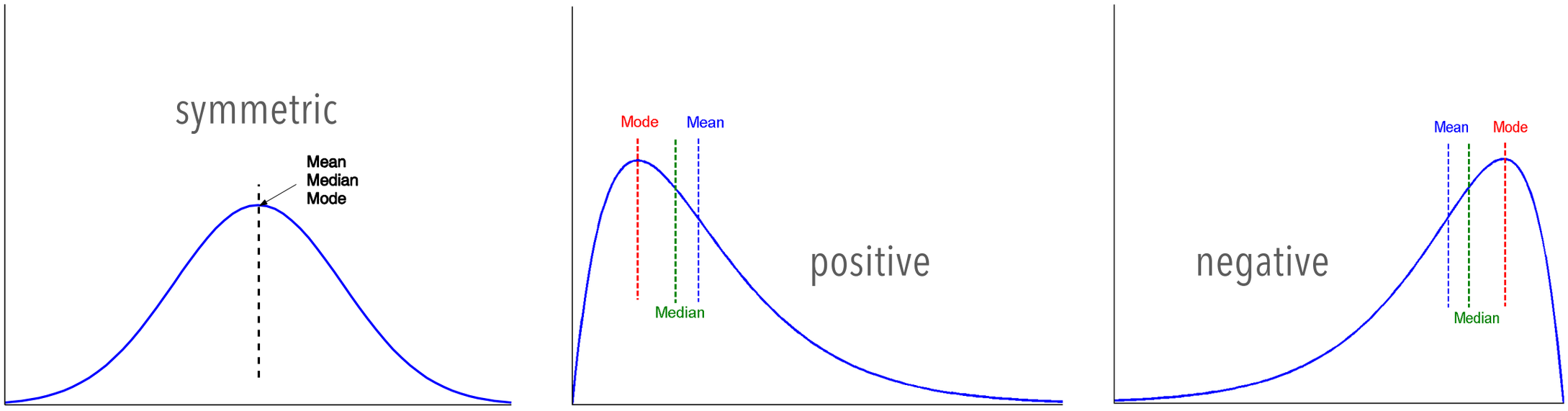


The mode

Value that occurs most frequently in the data

approximate formula

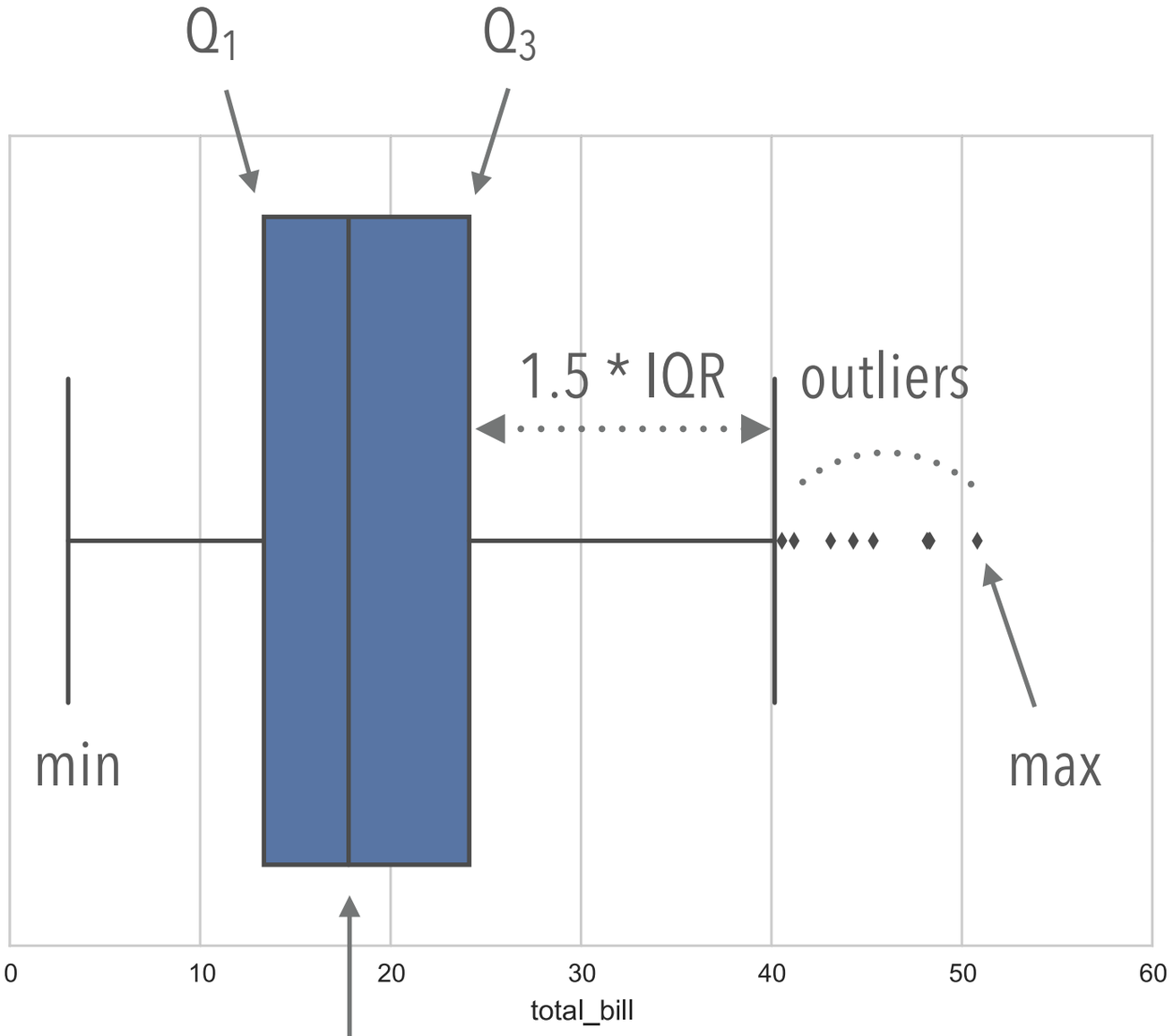
$$mean - mode = 3 \times (mean - median)$$



data skew

Median, mean and mode of symmetric, positively and negatively skewed data

QUARTILES, OUTLIERS AND BOX PLOTS




```
>>> import seaborn as sns
>>> sns.set_style("whitegrid")
>>> tips = sns.load_dataset("tips")
>>> ax = sns.boxplot(x=tips["total_bill"])
```

http://bit.ly/2BeUfFg

Quartiles: Q_1 (25th percentile), Q_3 (75th percentile)

Inter-quartile range: IQR = $Q_3 - Q_1$

Five number summary: min, Q_1, median, Q_3, max

Boxplot: ends of the box are the quartiles; median is marked; add whiskers, and plot outliers individually

Outlier: usually, a value higher/lower than 1.5 x IQR

sample variance

$$s^2 = \frac{1}{n-1}\sum_{i}^{n}(x_i - \bar{x})^2$$

Variance

$$\sigma^2 = \frac{1}{N}\sum_{i}^{N}(x_i - \mu)^2$$

population variance

that obvious formula is an underestimate

$$s^2 = \frac{1}{n}\sum_{i}^{n}(x_i - \bar{x})^2$$

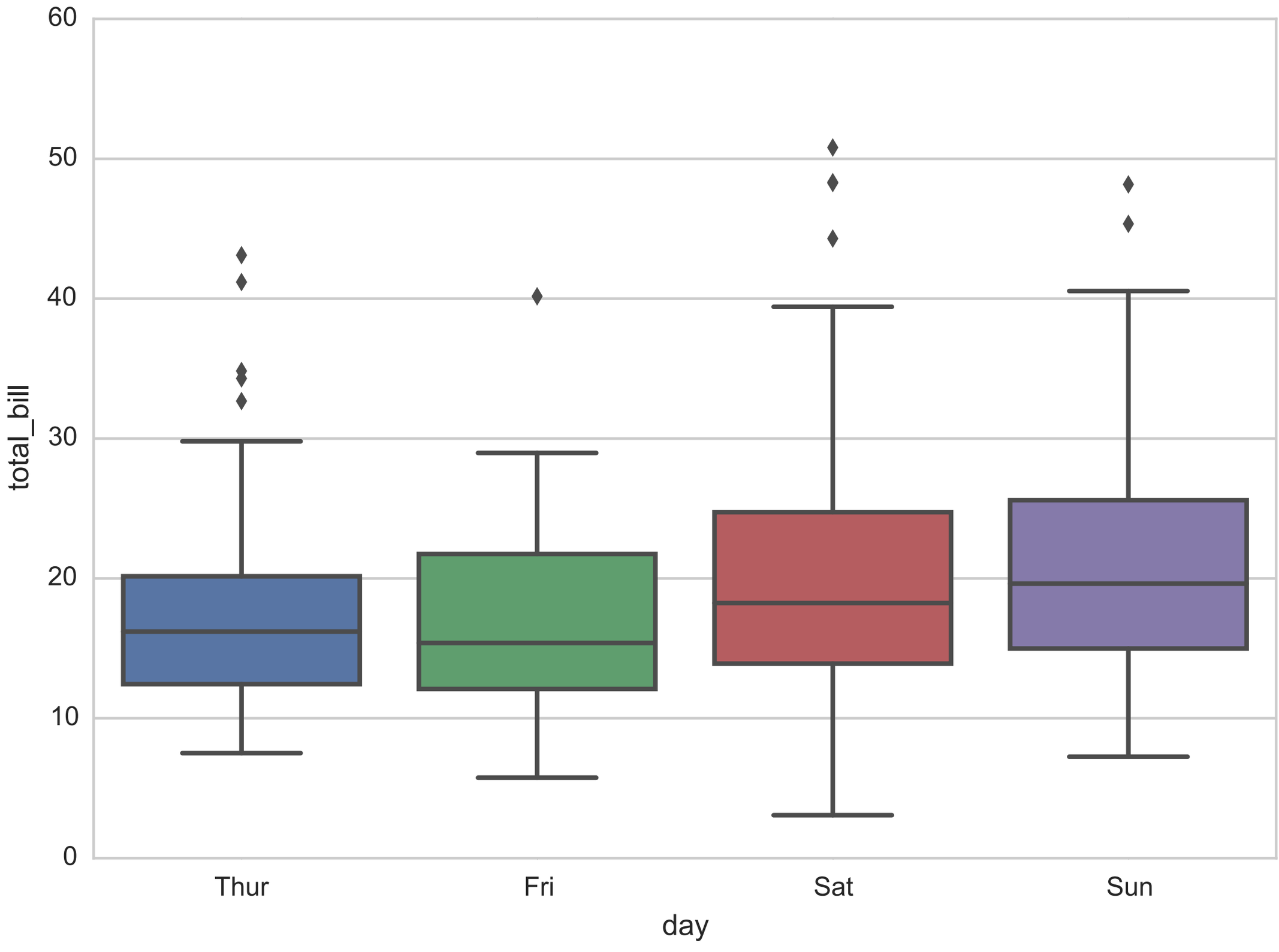
explaining the puzzle

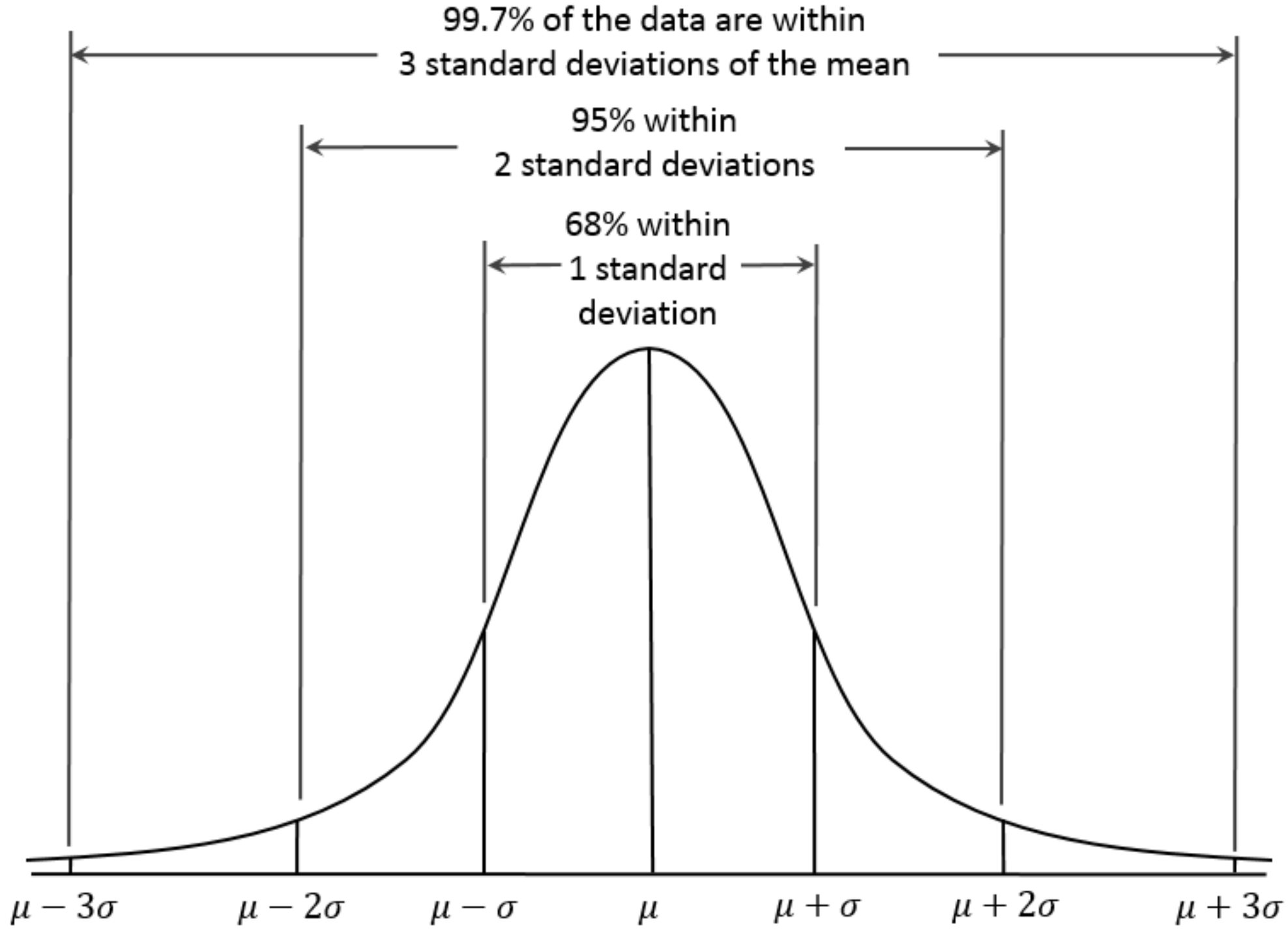
$$s^2 = \frac{1}{n}\left(\sum_i x_i^2 + \sum_i \left(\frac{\sum_j x_j}{n}\right)^2 - 2\sum_i x_i \left(\frac{\sum_j x_j}{n}\right)\right) \quad s^2 = \frac{1}{n}\left(\sum_i^n x_i^2 - \frac{\sum_i x_i^2 + 2\sum_{i\neq j} x_i x_j}{n}\right)$$

$$s^2 = \frac{n-1}{n}\sigma^2$$

$$E(x_i) = \mu$$

using $E(x_i^2) = \sigma^2 + \mu^2$

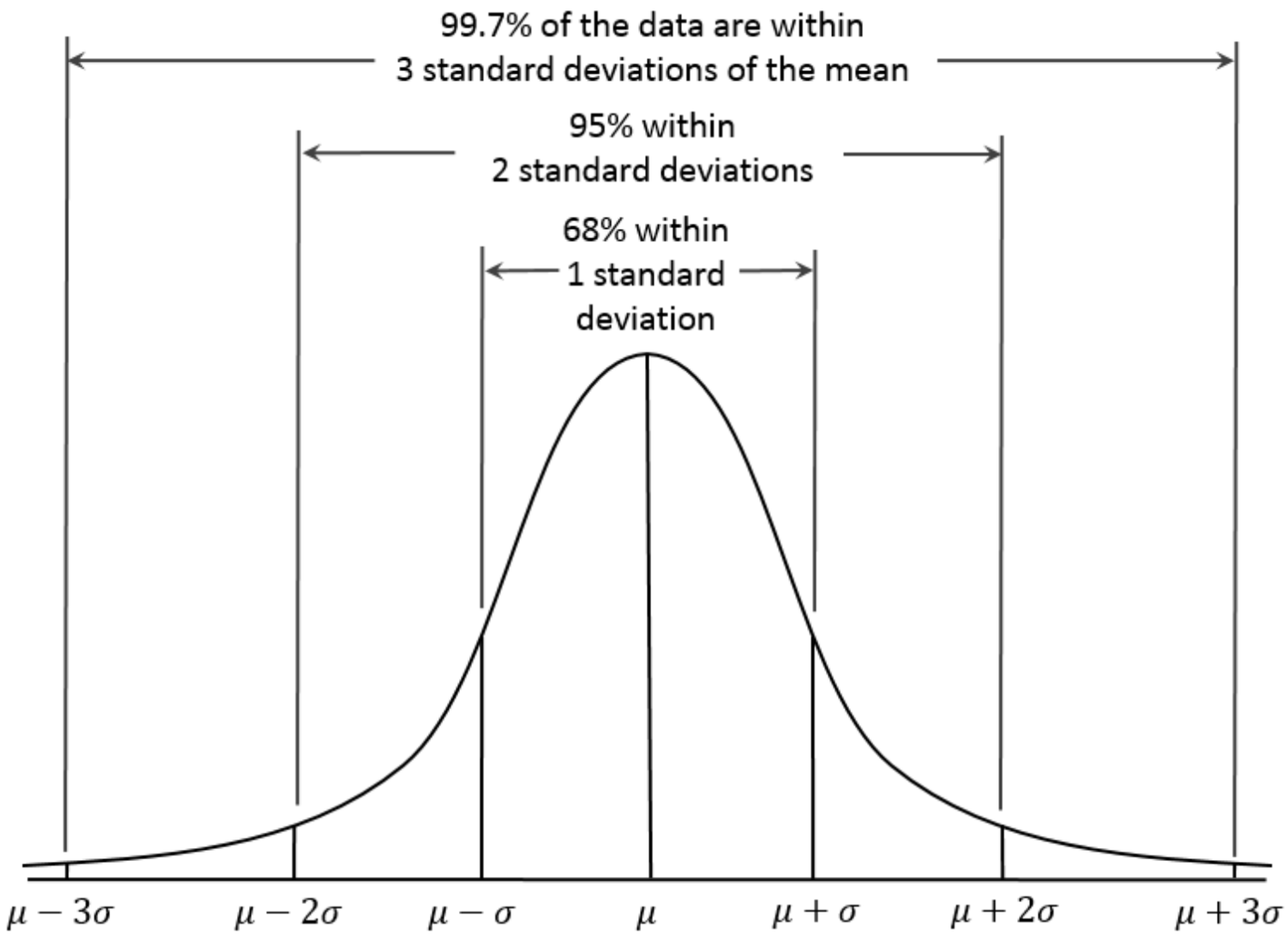

60
50
40
30
20
10
0
total_bill
Thur
Fri
Sat
Sun
day



the normal distribution

Why is the normal distribution so important?

Assume that we have n observations; n ≥30



$$\bar{x} = \frac{1}{n}\sum_i x_i$$

$$E(\bar{x}) = \mu$$

$$Var(\bar{x}) = \frac{\sigma^2}{n}$$

Central Limit Theorem

quantile plots

quantile-quantile plots

graphical displays of statistical properties

box plots

histograms

scatter

HISTOGRAMS


```
>>> import seaborn as sns, numpy as np
>>> sns.set(rc={"figure.figsize": (8, 4)}); np.random.seed(0)
>>> x = np.random.randn(100)
>>> ax = sns.distplot(x,kde=False)
```

http://stanford.io/1U7Uki3

Histogram: Graph display of tabulated frequencies, shown as bars

Differences between histograms and bar charts:

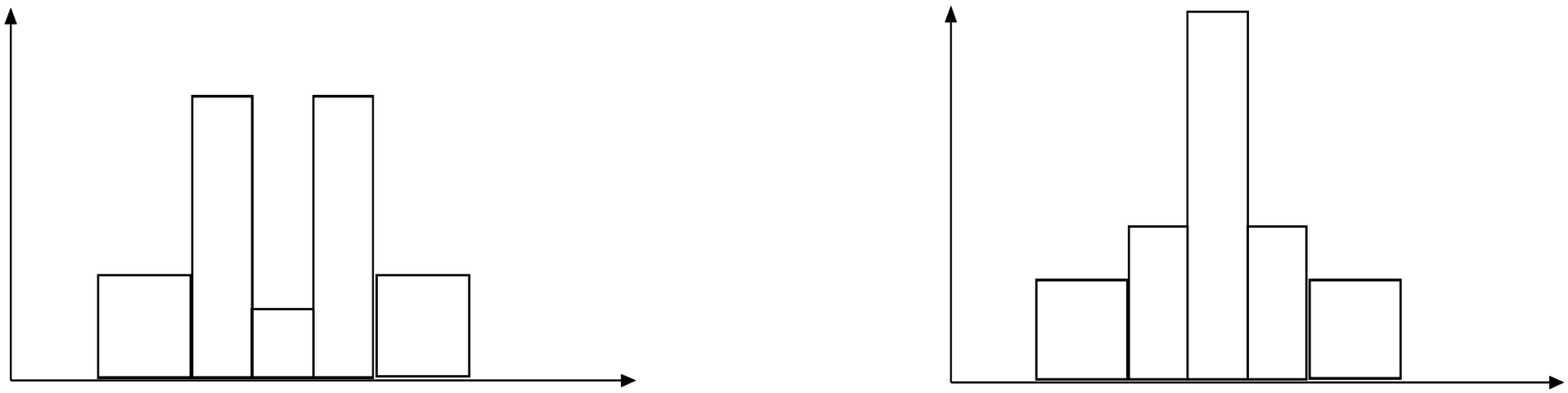
Histograms are used to show distributions of variables while bar charts are used to compare variables.

Histograms plot binned quantitative data while bar charts plot categorical data.

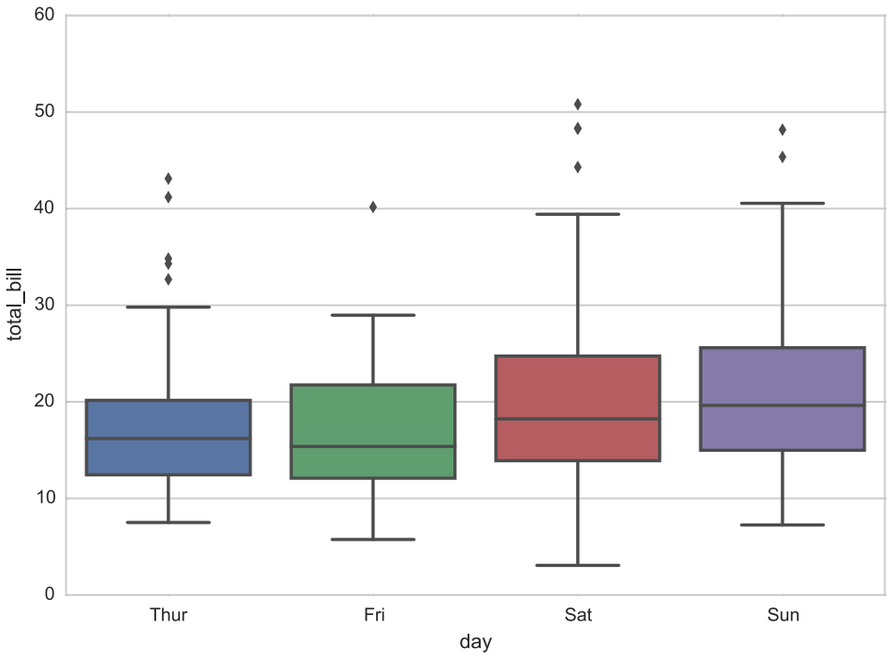
Bars can be reordered in bar charts but not in histograms.

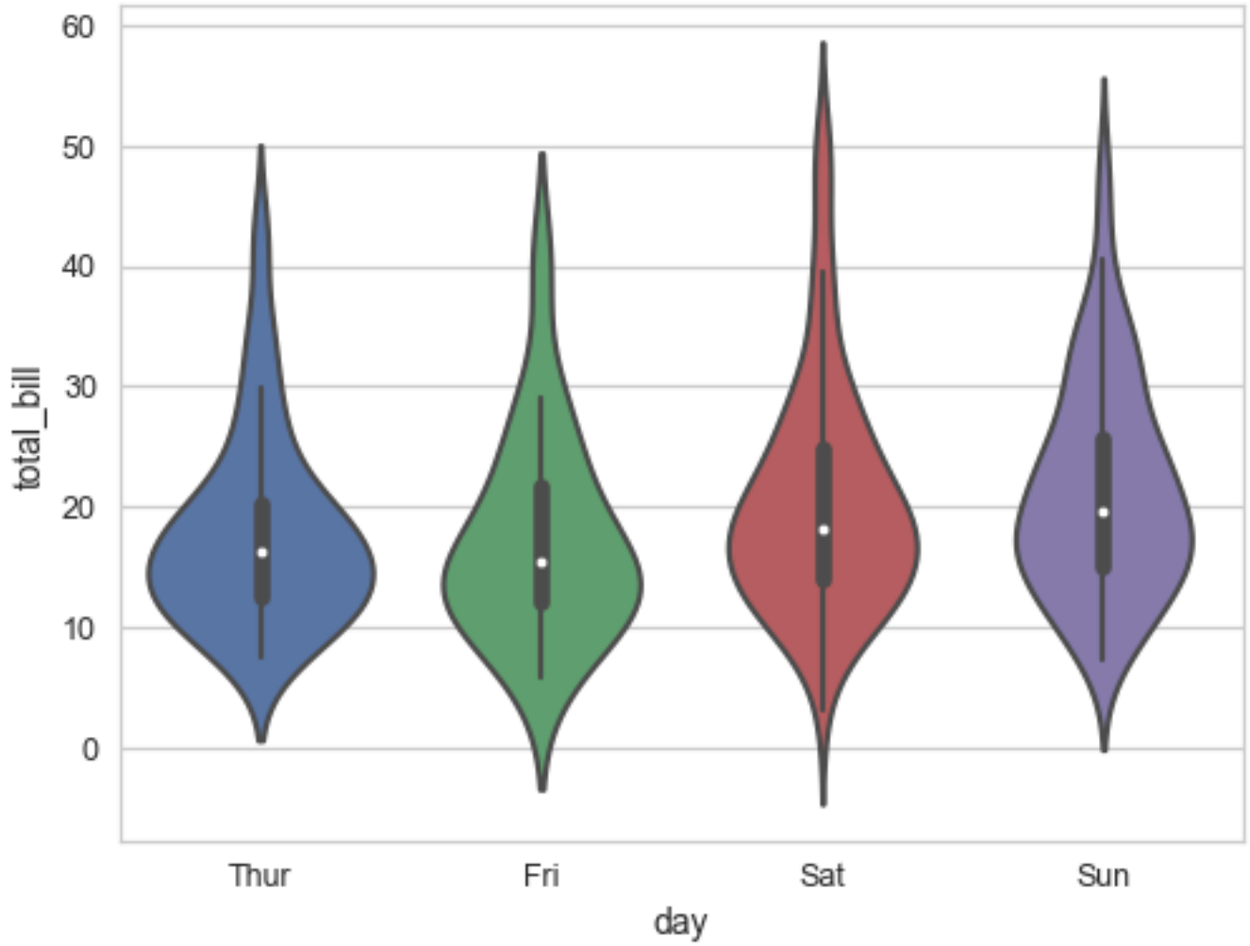
Differs from a bar chart in that it is the area of the bar that denotes the value, not the height as in bar charts, a crucial distinction when the categories are not of uniform width

histograms tell more than box plots



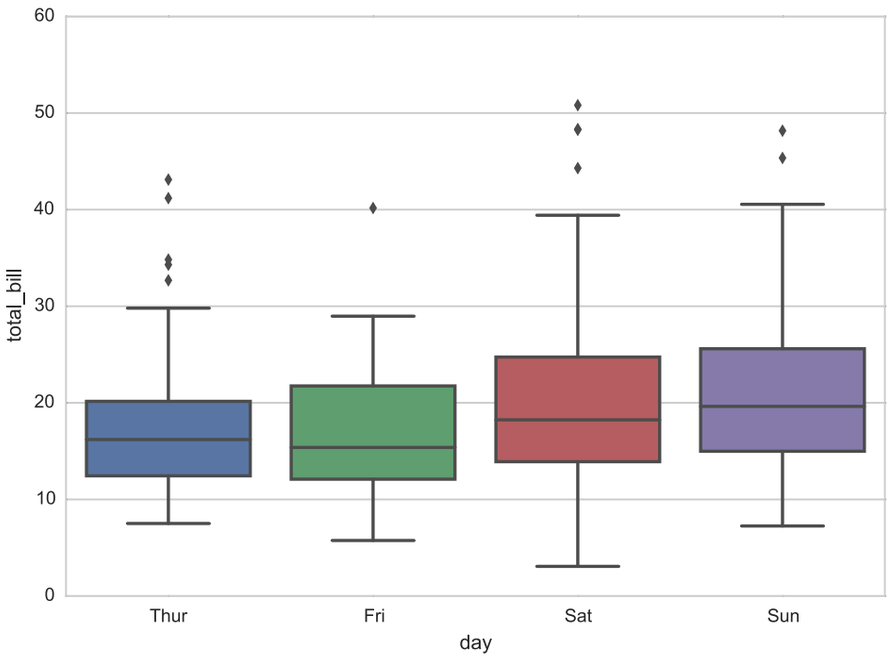
different distributions, with identical box plots

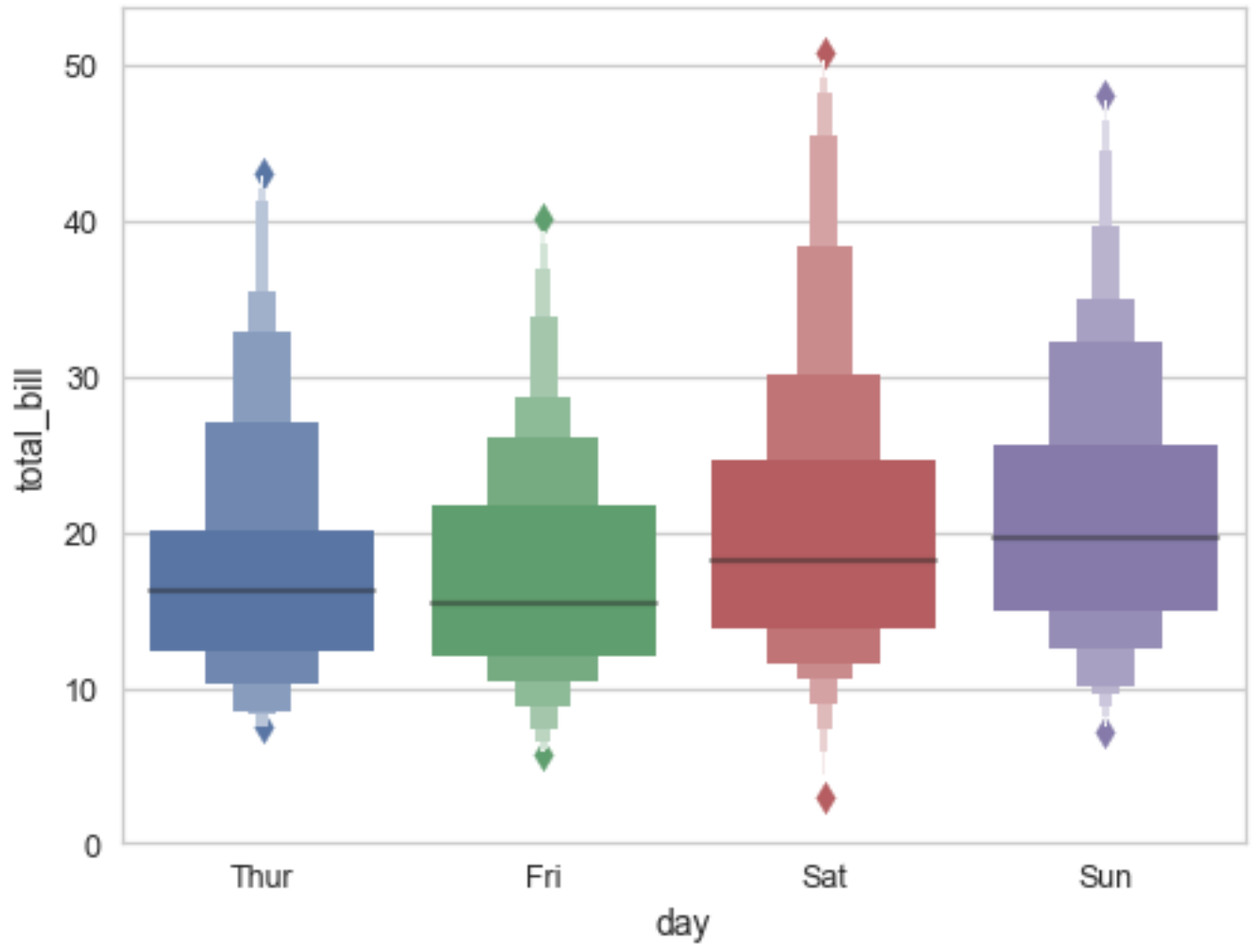




Violin Plots

http://bit.ly/2Dv5UFb

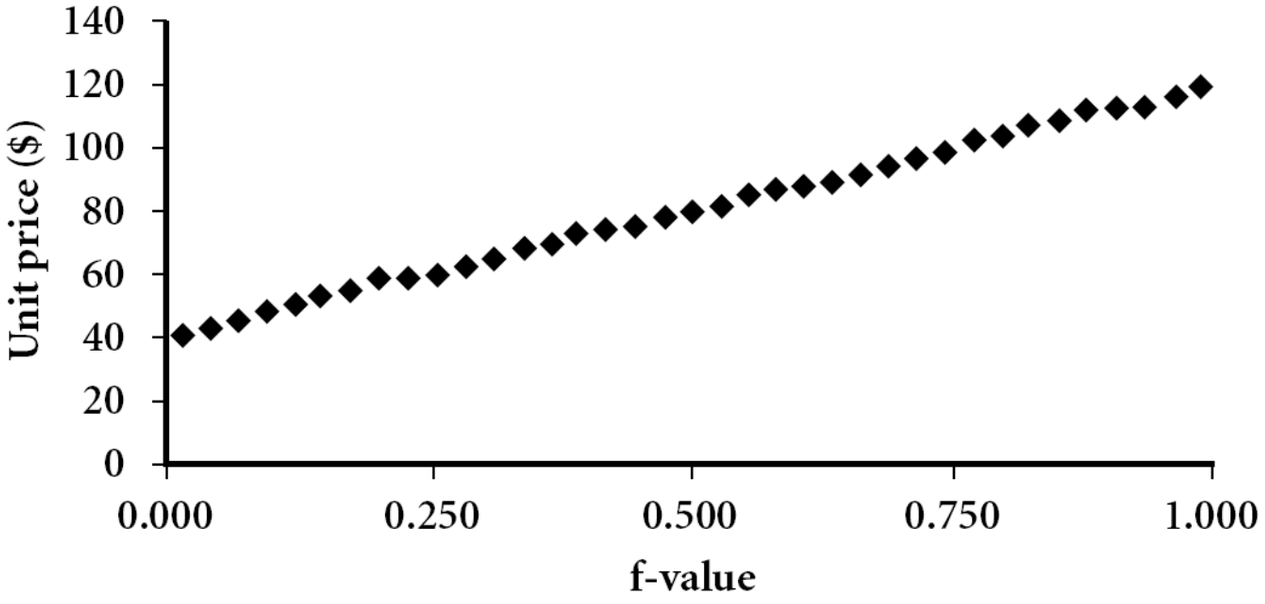




Letter Value Plots

http://bit.ly/2DtPRY3

QUANTILES

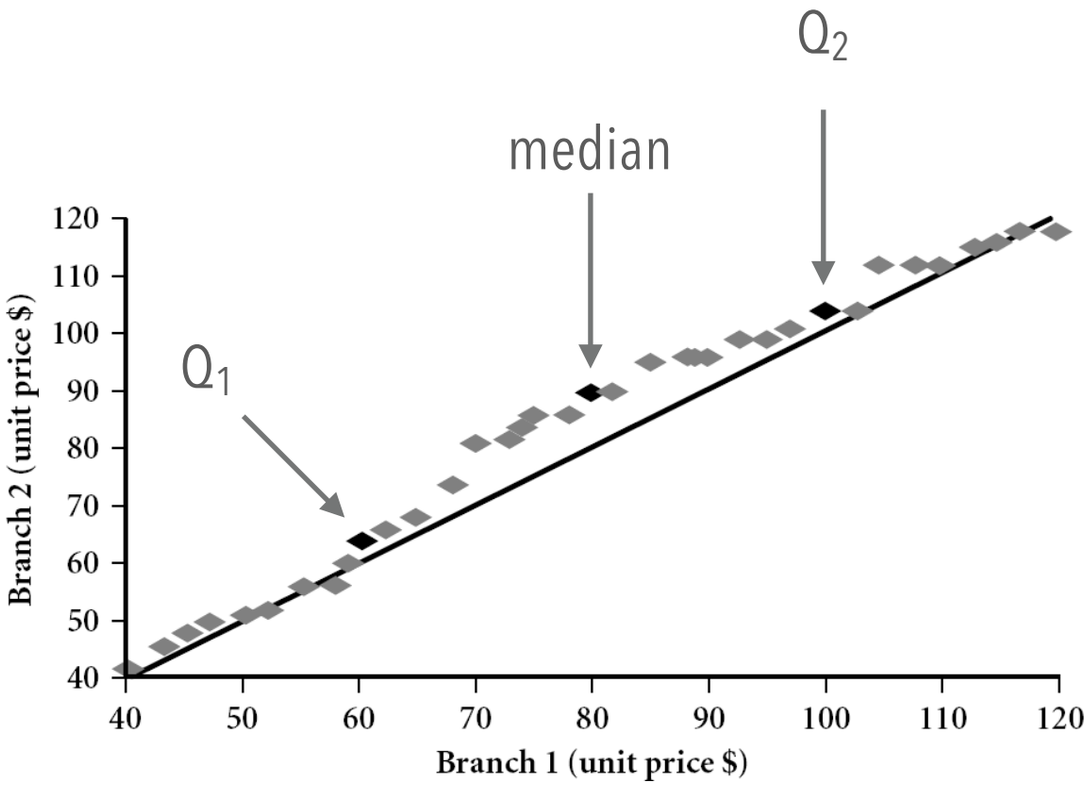


Displays all of the data (allowing the user to assess both the overall behavior and unusual occurrences)

Plots quantile information

For a data x_i data sorted in increasing order, f_i indicates that approximately $100f_i\%$ of the data are below or equal to the value x_i

Q-Q PLOT



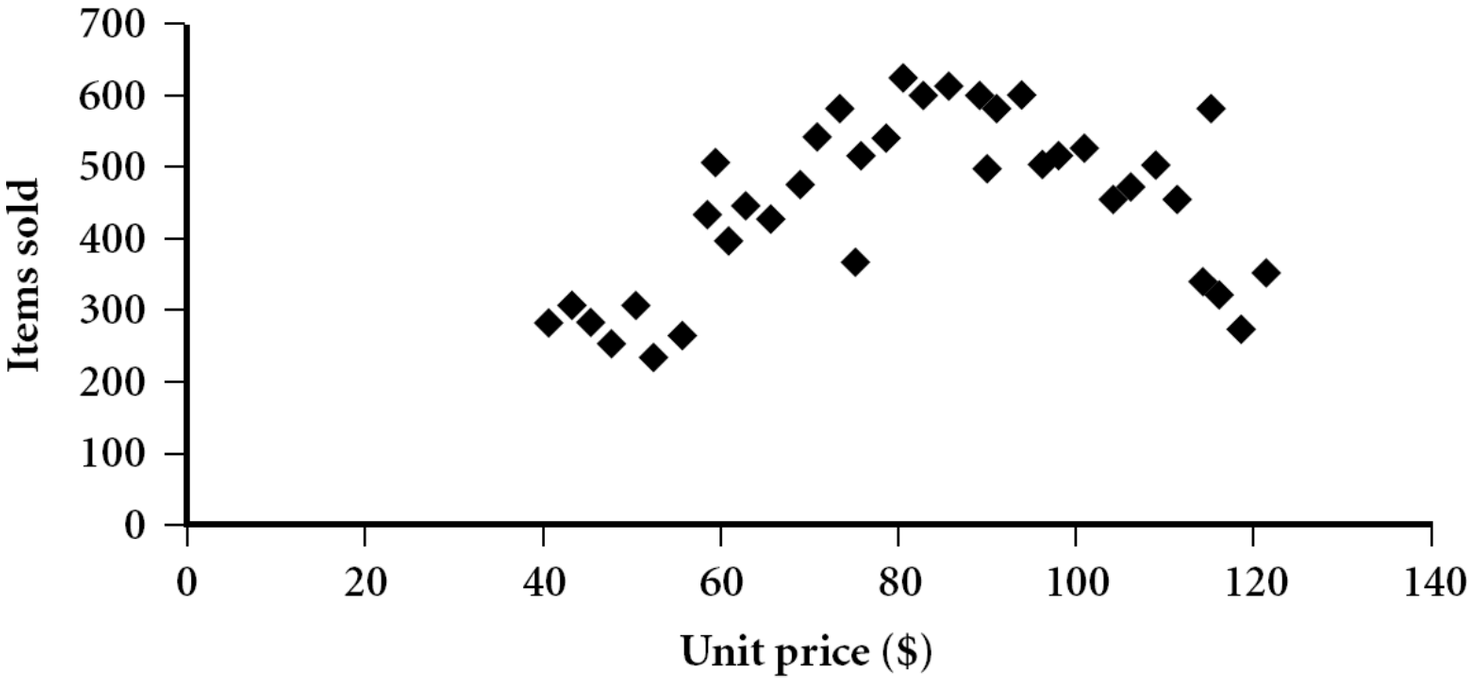
Graphs the quantiles of one univariate distribution against the corresponding quantiles of another

Is there is a shift in going from one distribution to another?

Example shows unit price of items sold at Branch 1 vs. Branch 2 for each quantile. Unit prices of items sold at Branch 1 tend to be lower than those at Branch 2.

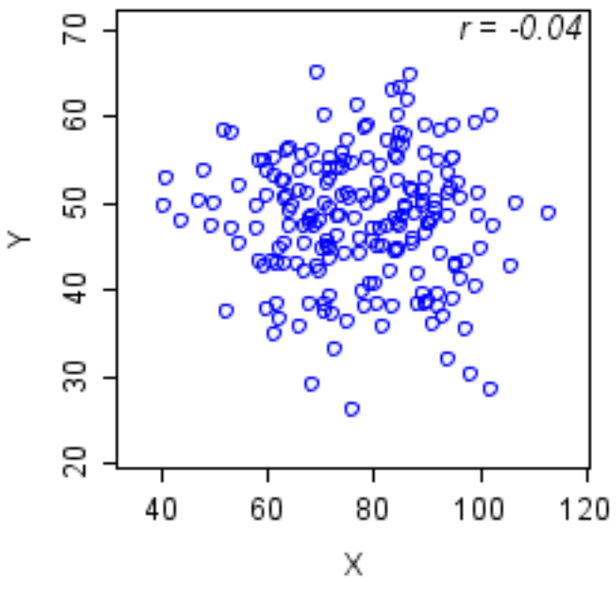
Scatter Plot

Each pair of values is treated as a pair of coordinates and plotted as points in the plane

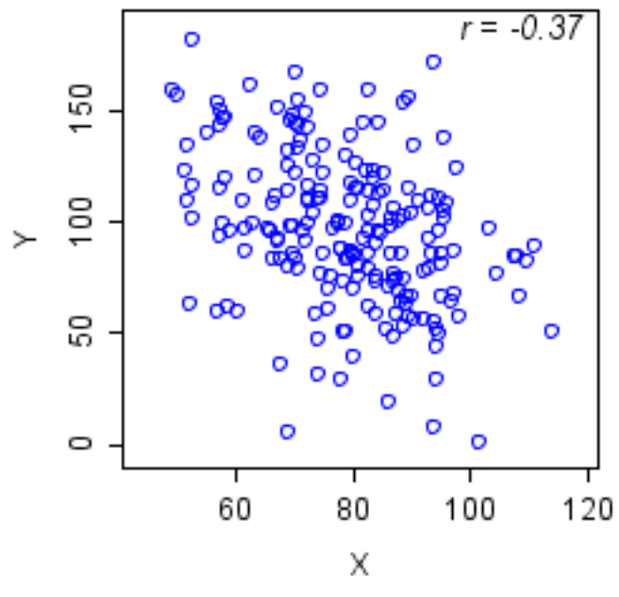


Provides a first look at bivariate data to see clusters of points, outliers, etc.

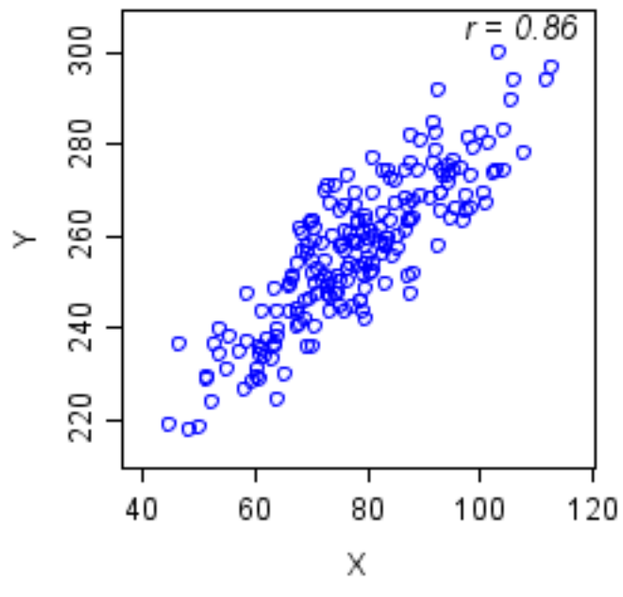
types of correlation



almost uncorrelated



negatively correlated



positively correlated

On writing reference letters

Extra Credit on Piazza

Students get points for making **substantial** contributions to answers **endorsed by TA**

TA will look at questions **8** hours after being posted

Person with **most** endorsements gets **5** extra credit points

Everyone else is **proportionally** scaled down

Arjun gets **5** points

Arjun has **10** endorsements

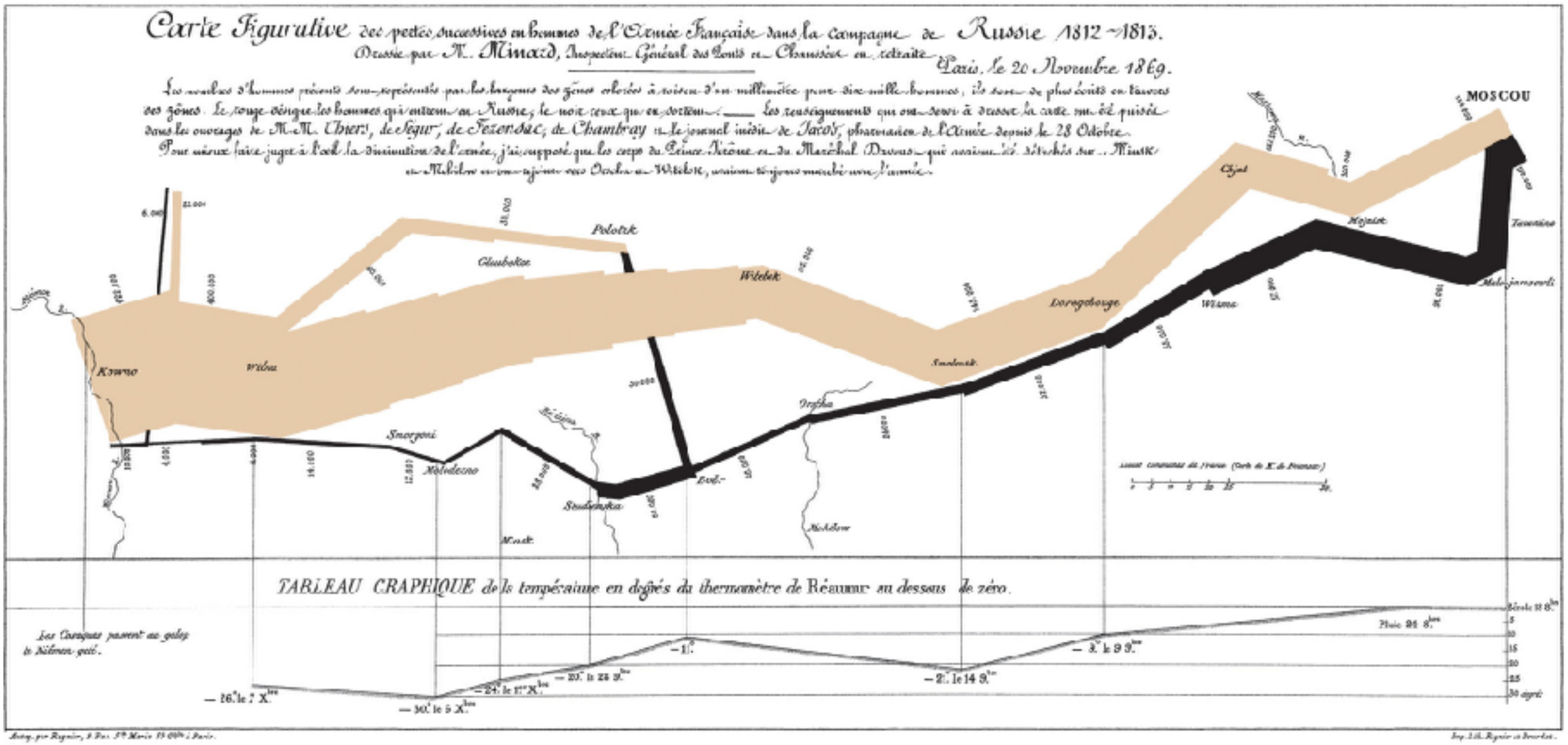
John gets **3** points

John has **6** endorsements

Off topic questions

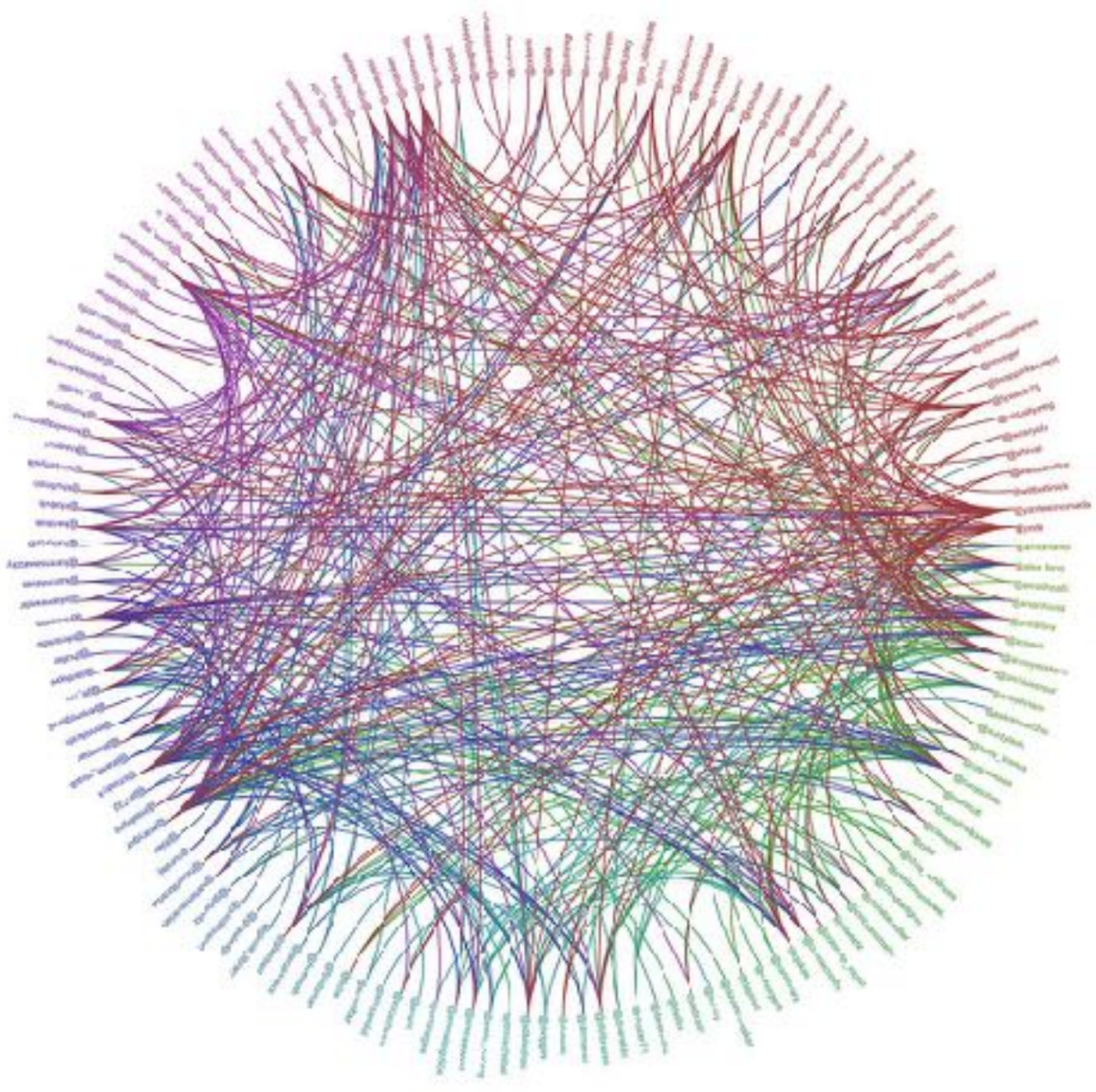
Gaming answers will get you a 0

TA will keep an excel sheet



DATA VISUALIZATION

Minard's plot of Napoleon's march



Enron email corpus

WHY?

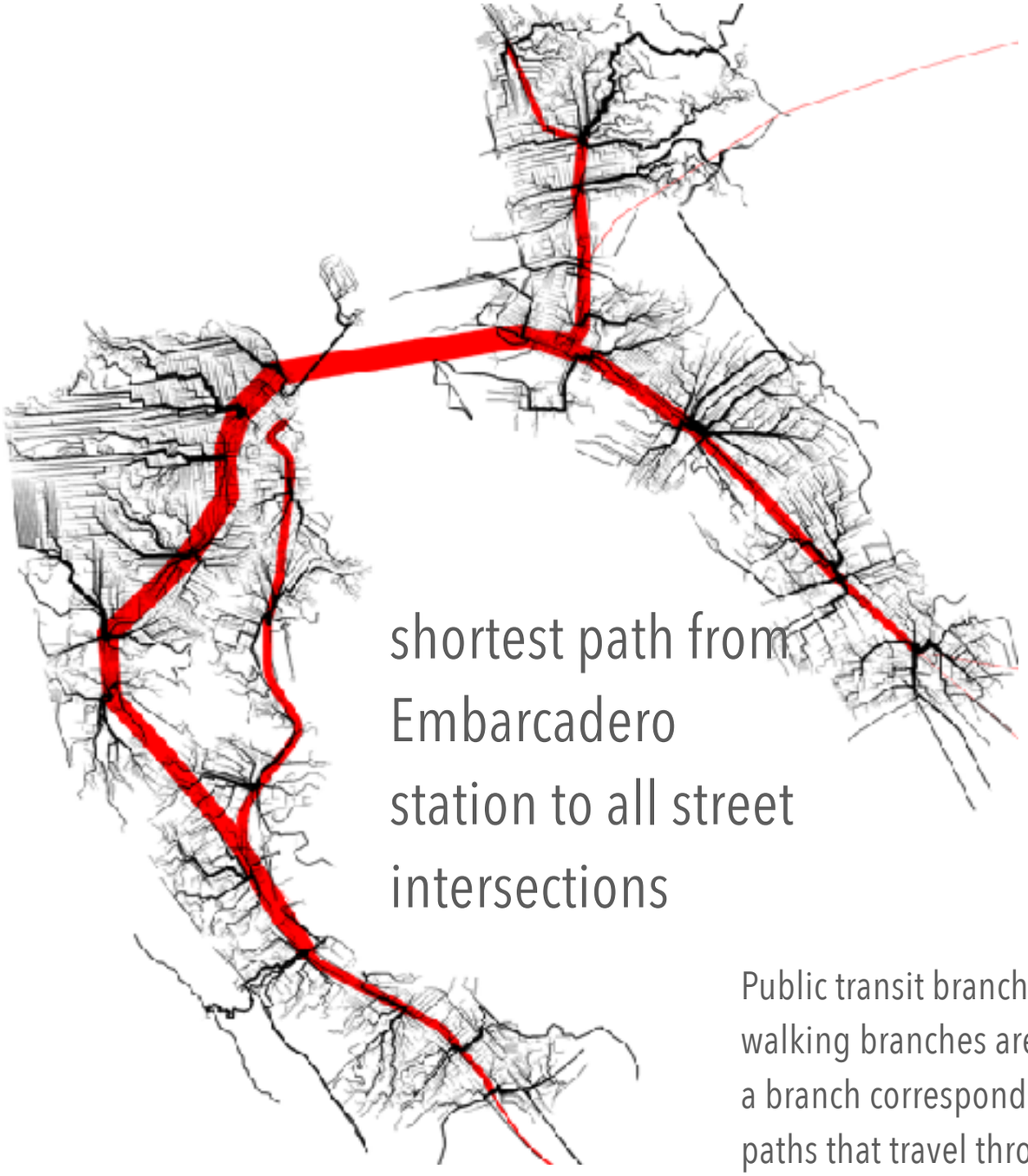
Gain insight into an information space by mapping data onto graphical primitives

Provide qualitative overview of large data sets

Search for patterns, trends, structure, irregularities, relationships among data

Help find interesting regions and suitable parameters for further quantitative analysis

Provide a visual proof of computer representations derived



shortest path from Embarcadero station to all street intersections

WHY?

This image shows the **shortest path** from Embarcadero Station on August 1st, 2008 11:43:33 AM local time to all street intersections in the San Francisco Bay area, assuming travel by public transit (namely, BART and Caltrain) and walking.

Public transit branches are colored red, while walking branches are colored black. The width of a branch corresponds to the number of shortest paths that travel through that branch. For example, the shortest route from Embarcadero to almost all destinations involves taking the BART. As a result, the BART lines are very thick.

Pixel-oriented

Hierarchical

Icon-based

Geometric

Visualization Categories

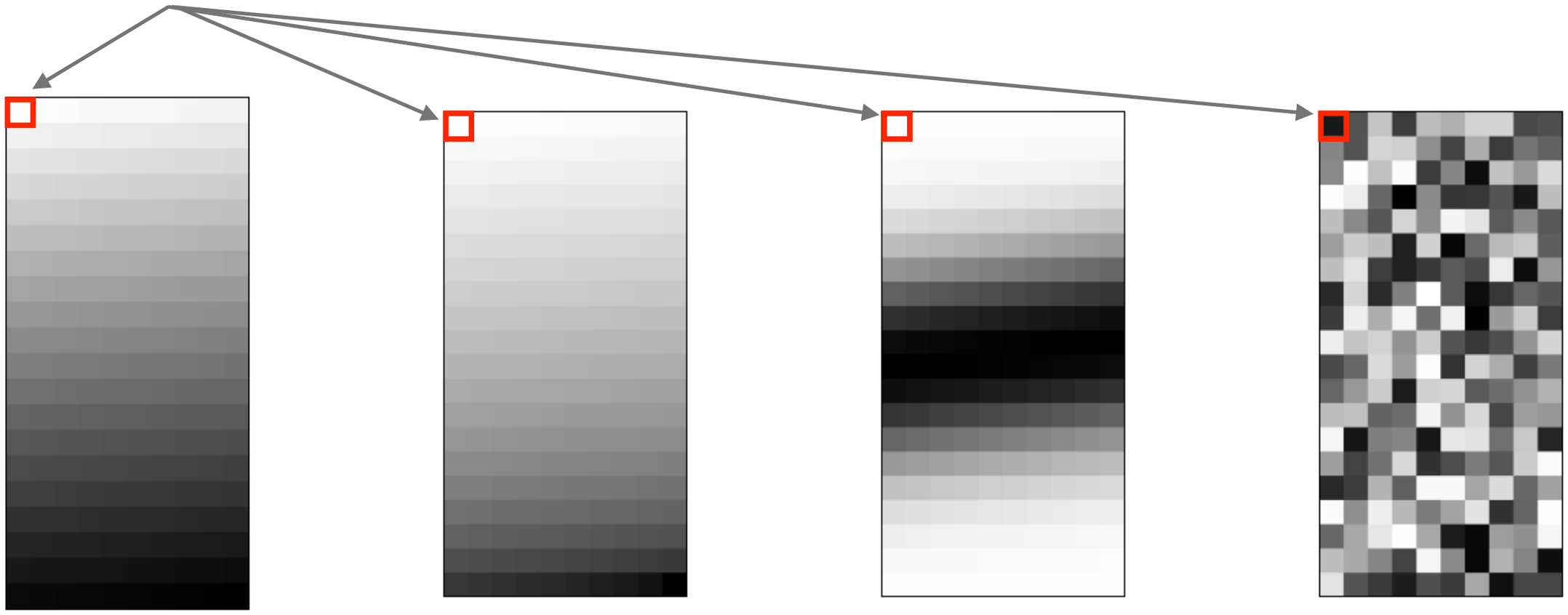
Visualizing complex data & relationships

pixel oriented visualization techniques

For a data set of m dimensions, create m windows on the screen, one for each dimension

The m dimension values of a record are mapped to m pixels at the corresponding positions in the windows

The colors of the pixels reflect the corresponding values



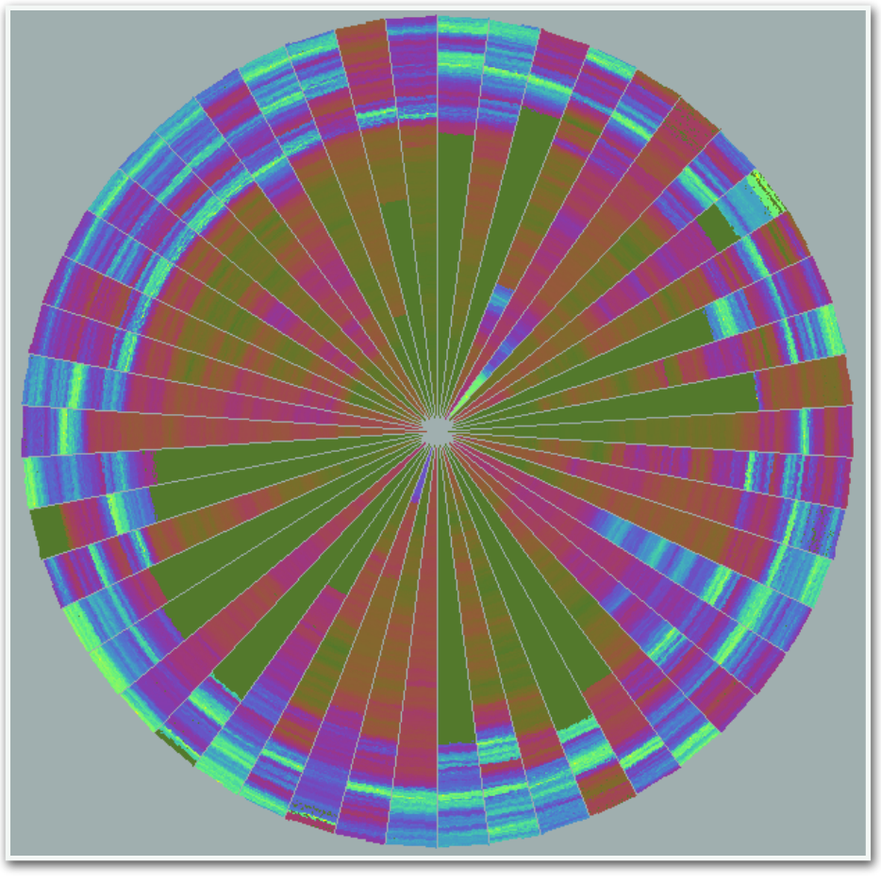
income

credit limit

transaction volume

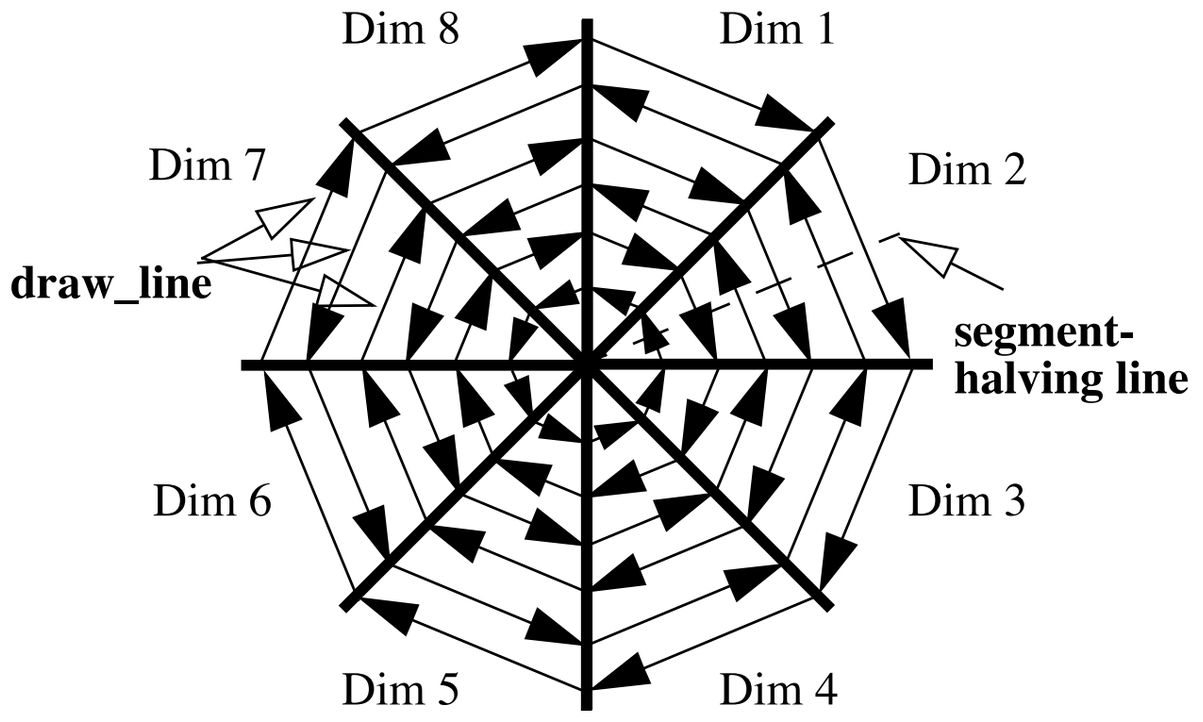
age

Space Filling Curves



Representing about 265,000 50-dimensional Data Items with the 'Circle Segments' Technique

M. Ankerst, D. A. Keim, and H.-P. Kriegel. Circle segments: A technique for visually exploring large multidimensional data sets. In Visualization, 1996.



To save space and show the connections among multiple dimensions, space filling is often done in a circle segment

Circular Segments

Direct visualization

Scatterplot matrices

Projection views

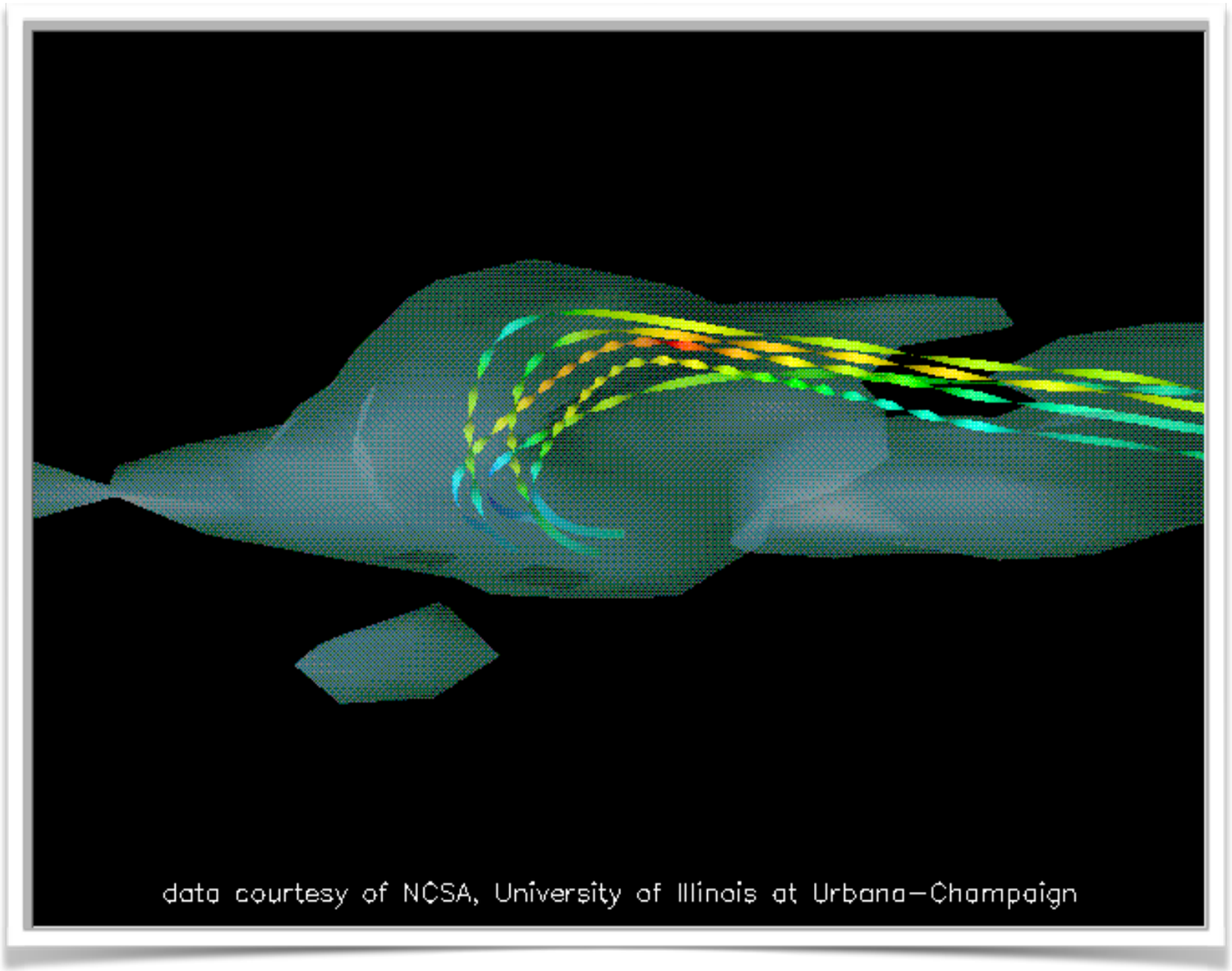
Landscapes

Geometric Projection Visualization

Hyperslice

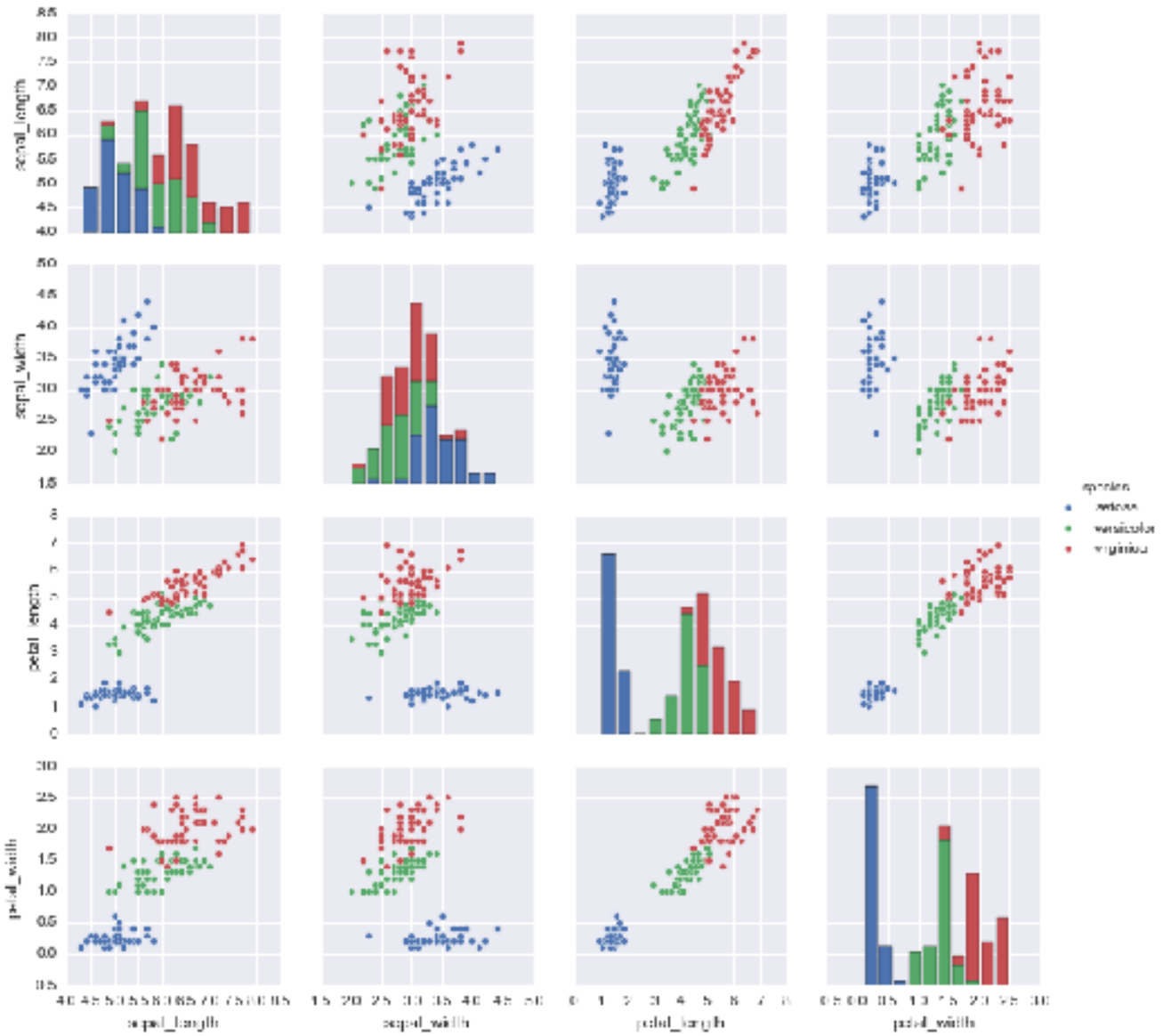
Parallel coordinates

Projection pursuit technique: Help users find meaningful projections of multidimensional data



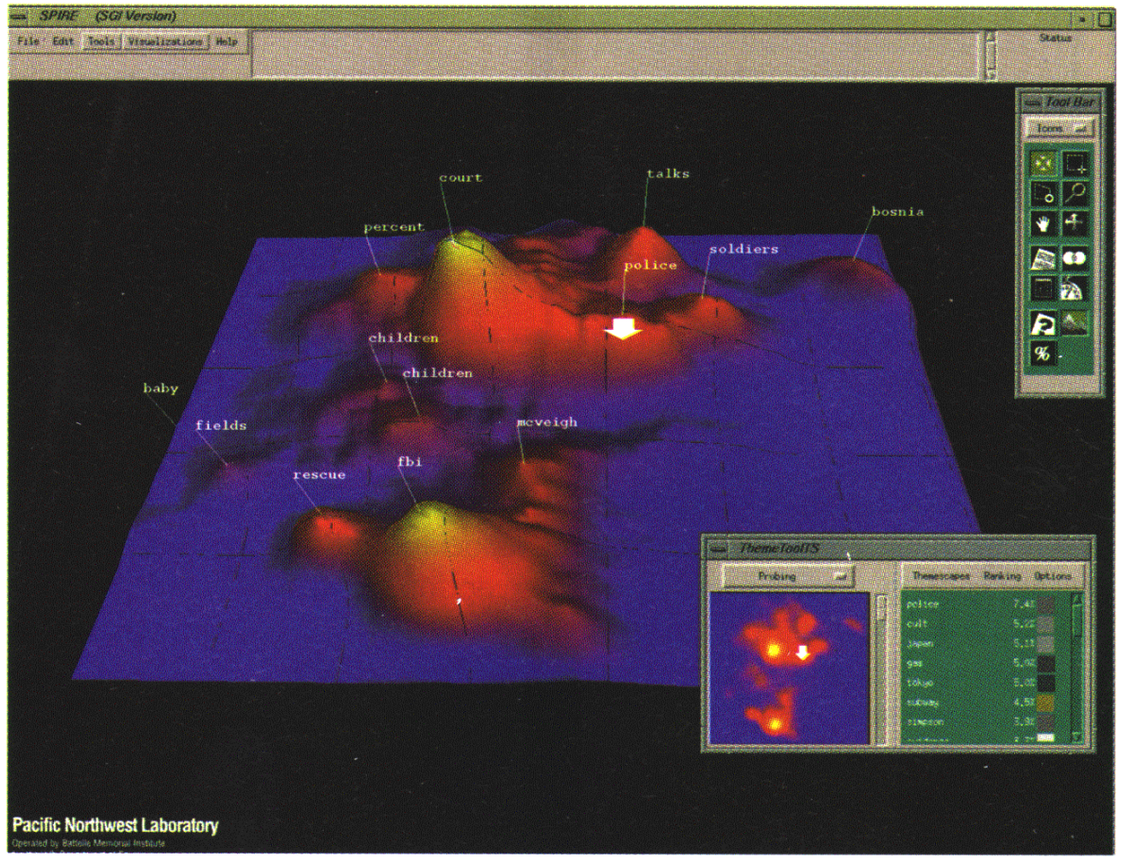
direct data visualization

Matrix of scatterplots (x-y-diagrams) of the k-dim. data [total of k*(k-1)/2 scatterplots]



scatter plot matrices

Visualization of the data as perspective landscape

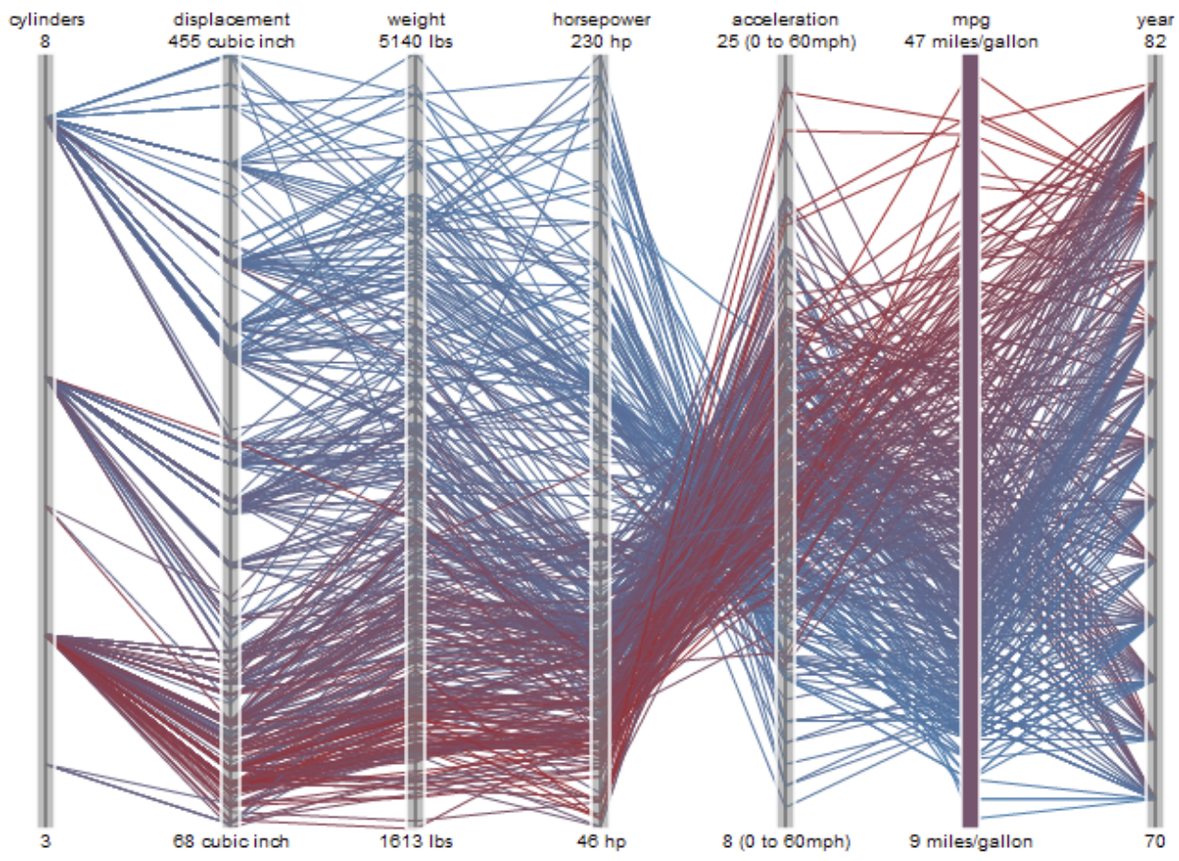


news articles visualized as a landscape

The data needs to be transformed into a (possibly artificial) 2D spatial representation which preserves the characteristics of the data

landscapes

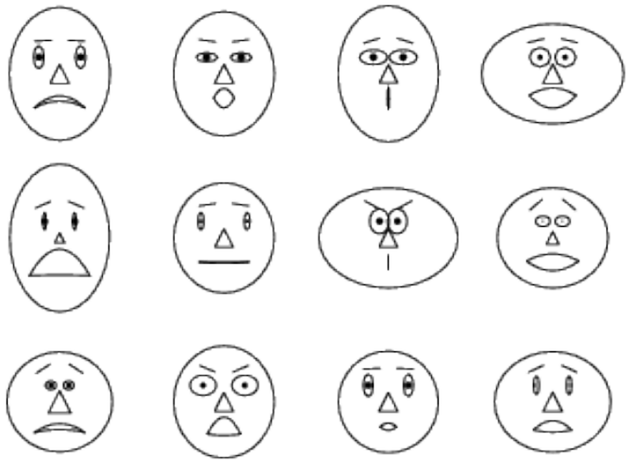
PARALLEL COORDINATES

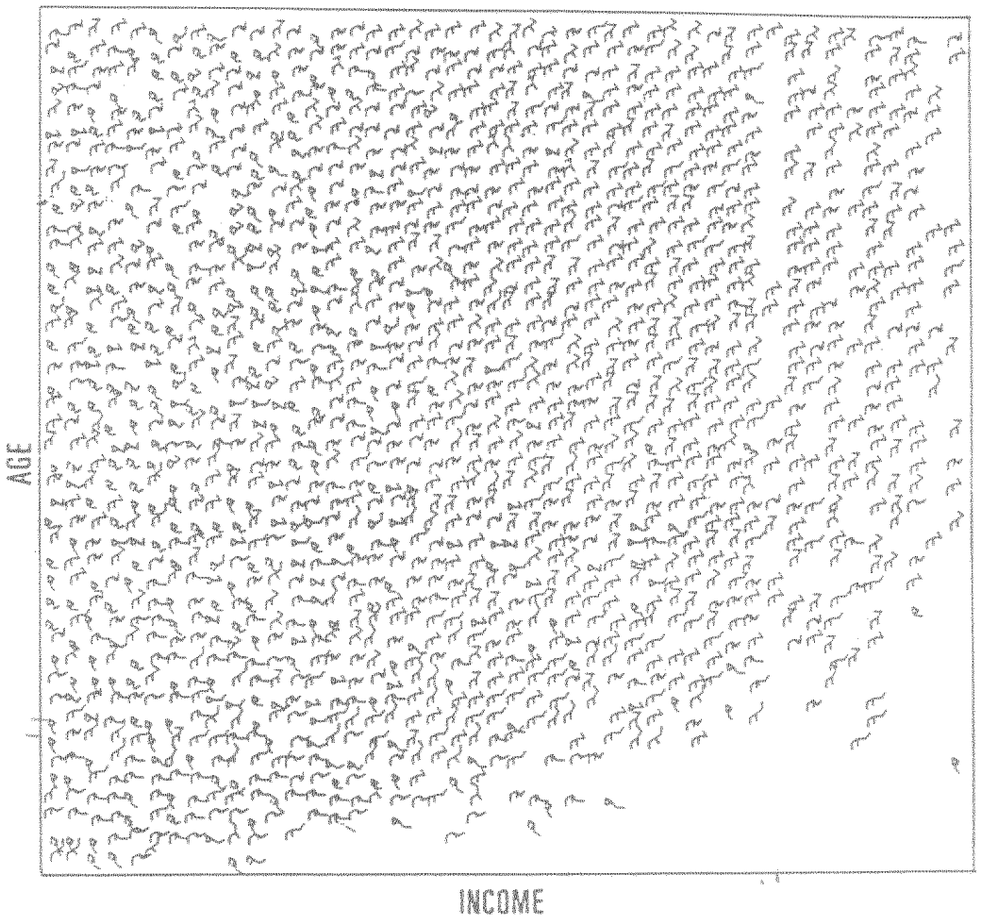


n equidistant axes which are parallel to one of the screen axes and correspond to the attributes

The axes are scaled to the [minimum, maximum]: range of the corresponding attribute

Every data item corresponds to a polygonal line which intersects each of the axes at the point which corresponds to the value for the attribute





ICONS

Visualization of the data values as features of icons

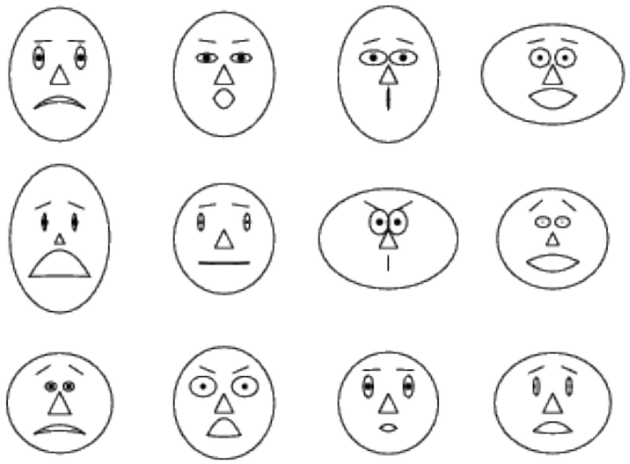
Typical visualization methods

- Chernoff Faces
- Stick Figures

General techniques

- Shape coding: Use shape to represent certain information encoding
- Color icons: Use color icons to encode more information
- Tile bars: Use small icons to represent the relevant feature vectors in document retrieval

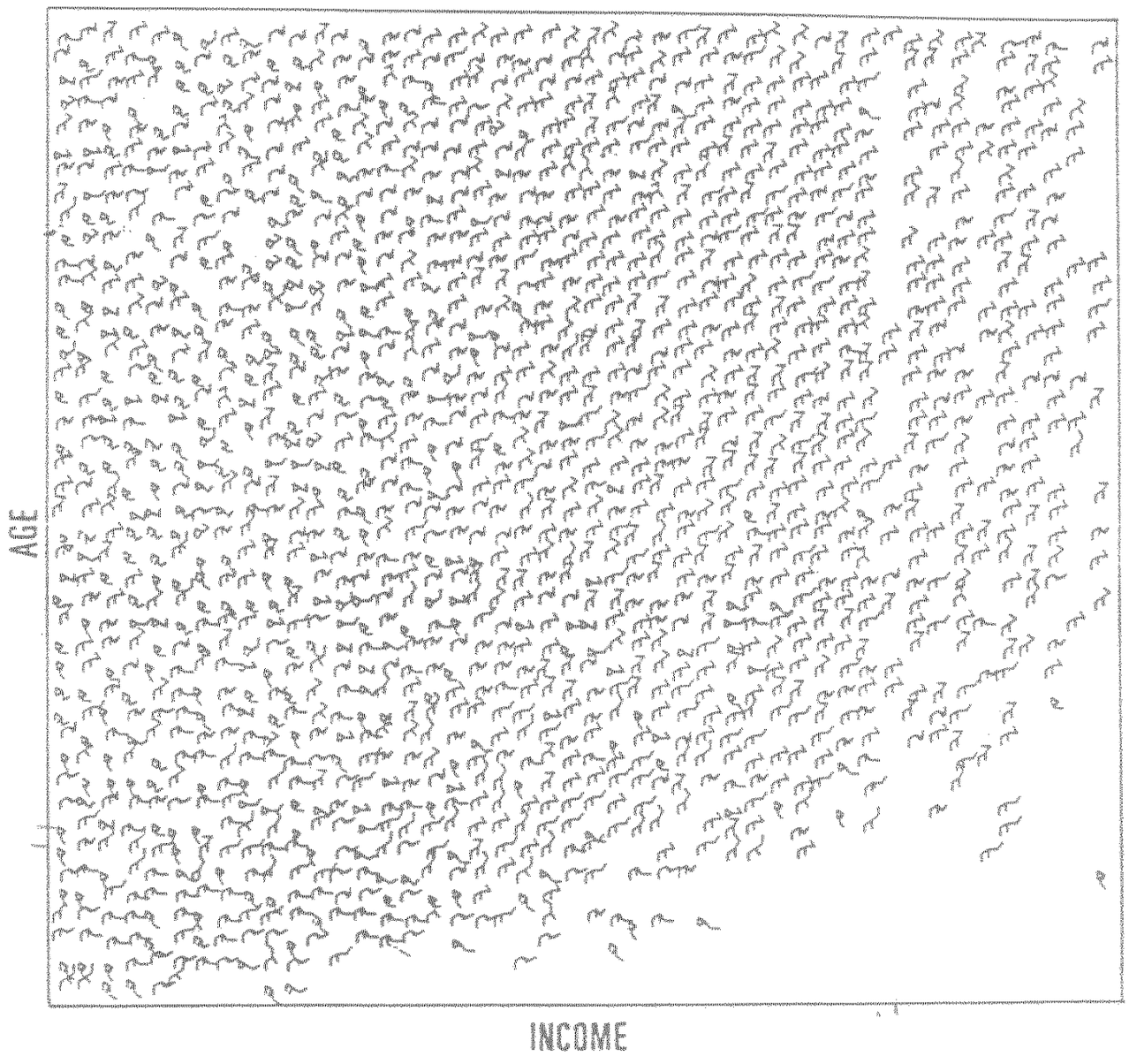
CHERNOFF FACES



A way to display variables on a two-dimensional surface, e.g., let x be eyebrow slant, y be eye size, z be nose length, etc.

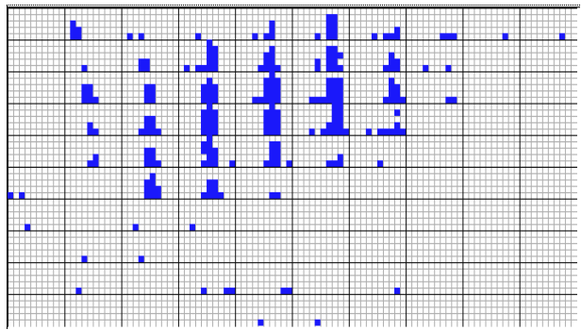
The figure shows faces produced using 10 characteristics–head eccentricity, eye size, eye spacing, eye eccentricity, pupil size, eyebrow slant, nose size, mouth shape, mouth size, and mouth opening; Each assigned one of 10 possible values

stick figures



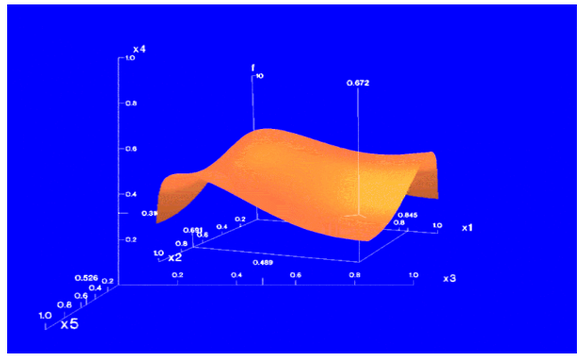
A census data figure showing age, income, gender, education, etc.

A 5-piece stick figure (1 body and 4 limbs w. different angle/length)



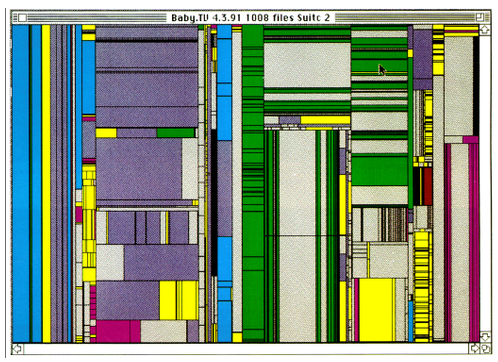
Dimensional Stacking

Worlds-within-Worlds

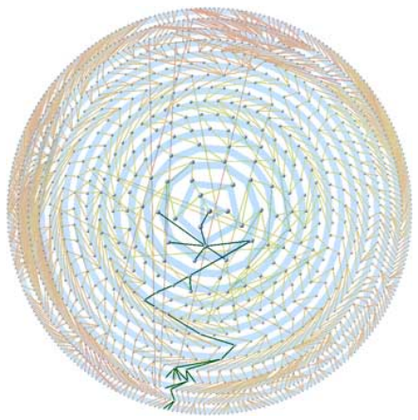


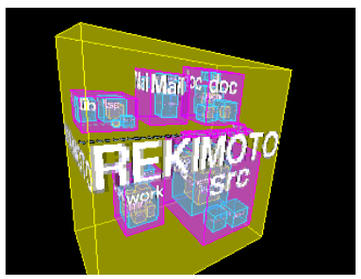
hierarchical visualization

Tree-Map



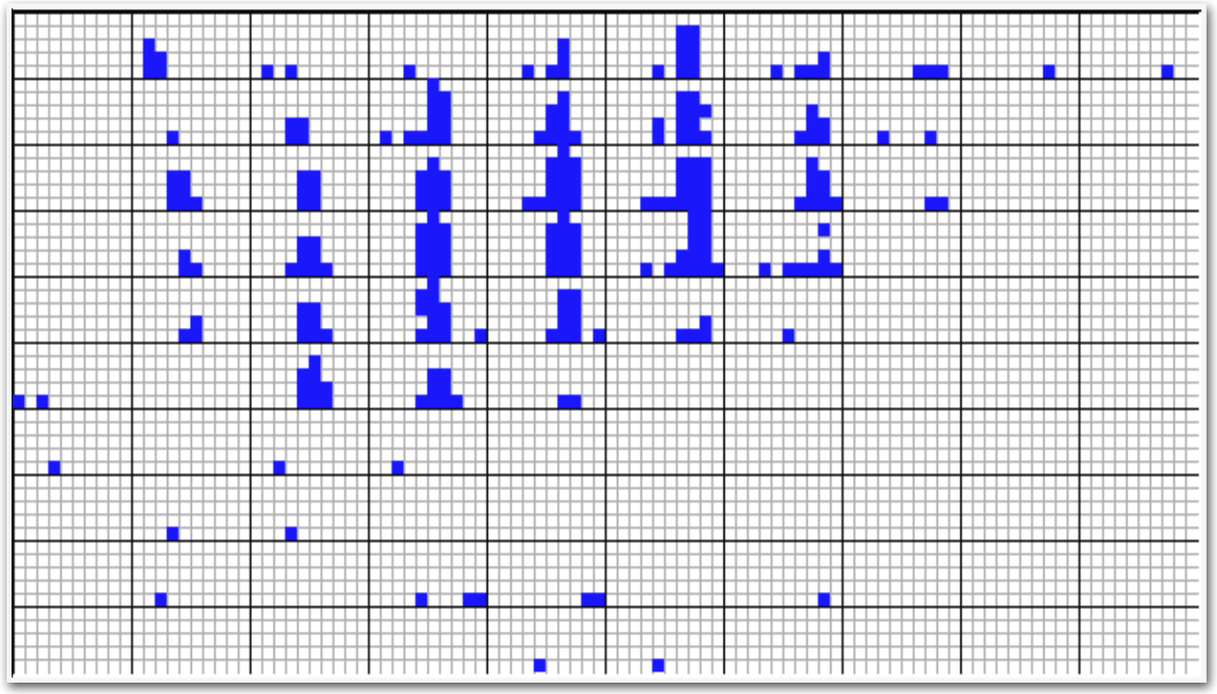
Cone Trees





InfoCube

DIMENSIONAL STACKING



Visualization of oil mining data with longitude and latitude mapped to the outer x-, y-axes and ore grade and depth mapped to the inner x-, y-axes

Partitioning of the n-dimensional attribute space in 2-D subspaces, which are ‘stacked’ into each other

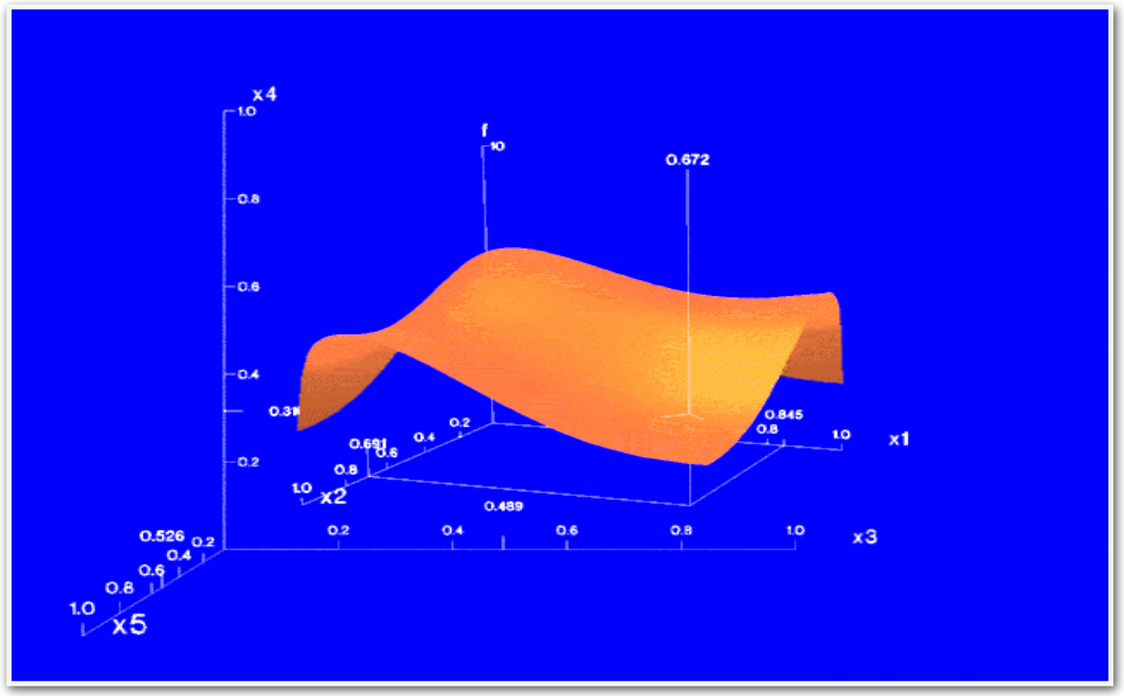
Partitioning of the attribute value ranges into classes. The important attributes should be used on the outer levels.

Adequate for data with ordinal attributes of low cardinality

But, difficult to display more than nine dimensions

Important to map dimensions appropriately

WORLDS WITHIN WORLDS



Assign the function and two most important parameters to innermost world

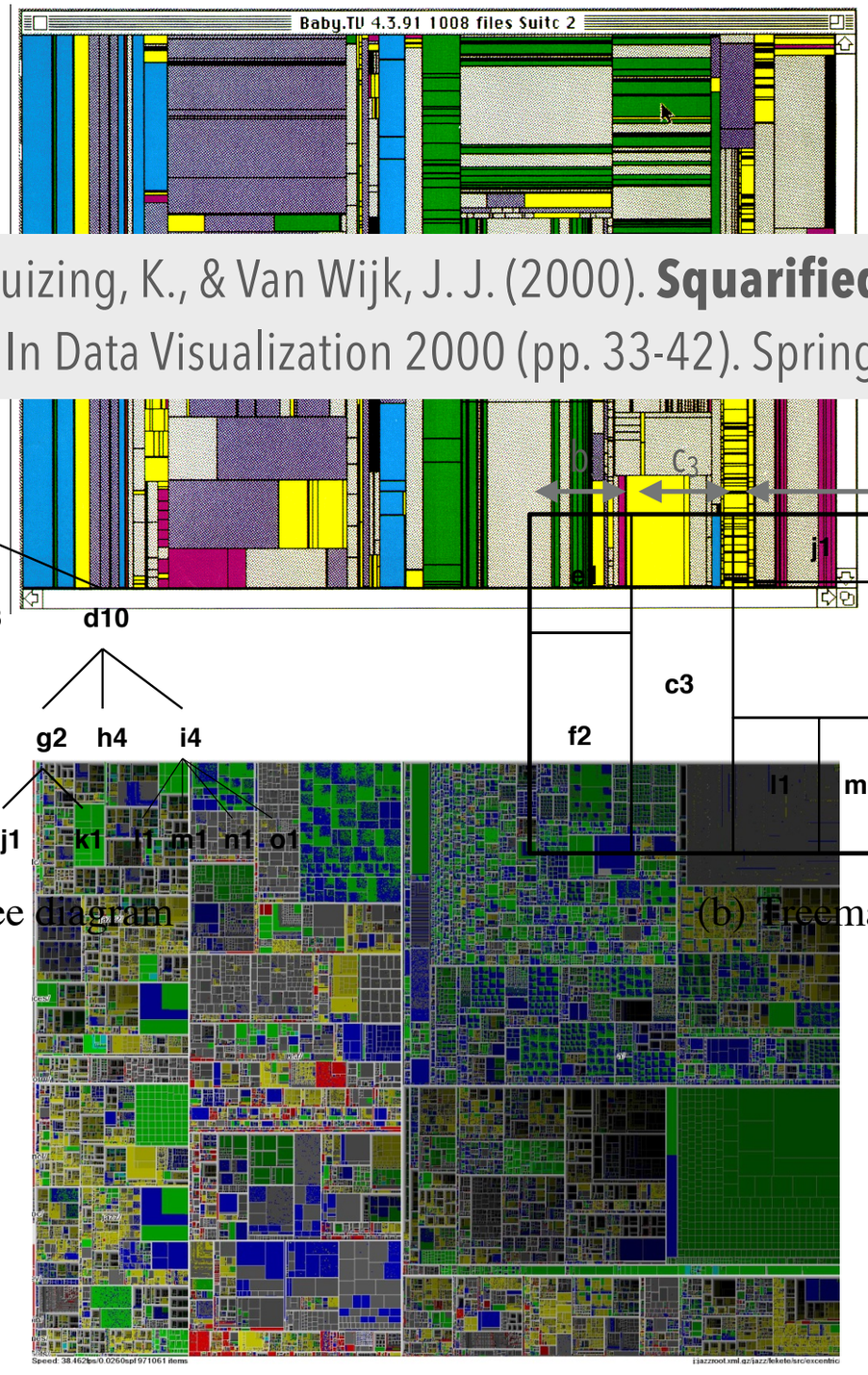
Fix all other parameters at constant values—draw other (1 or 2 or 3 dimensional worlds choosing these as the axes)

TREEMAP

Screen-filling method which uses a hierarchical partitioning of the screen into regions depending on the attribute values

The x- and y-dimension of the screen are partitioned alternately according to the attribute values (classes)

treemap of a file system



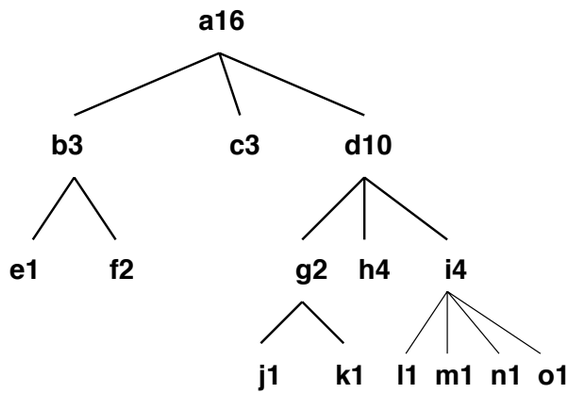
Bruls, M., Huizing, K., & Van Wijk, J. J. (2000). **Squarified treemaps**. In Data Visualization 2000 (pp. 33-42). Springer Vienna.

(a) Tree diagram (b) Treemap

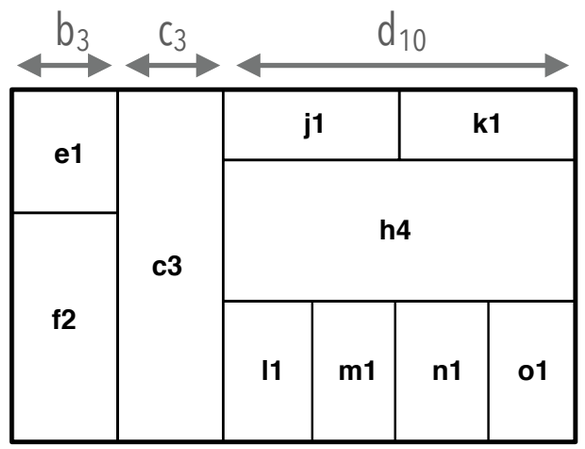
http://newsmap.jp

Martin Wattenberg. Visualizing the stock market. In CHI'99 extended abstracts on Human factors in computing systems, pages 188–189. ACM, 1999.

Bruls, M., Huizing, K., & Van Wijk, J. J. (2000). **Squarified treemaps**. In Data Visualization 2000 (pp. 33-42). Springer Vienna.

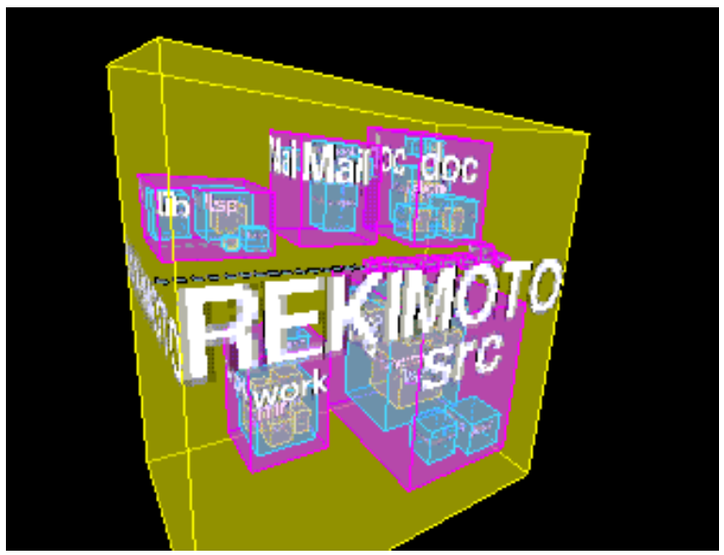


(a) Tree diagram



(b) Treemap

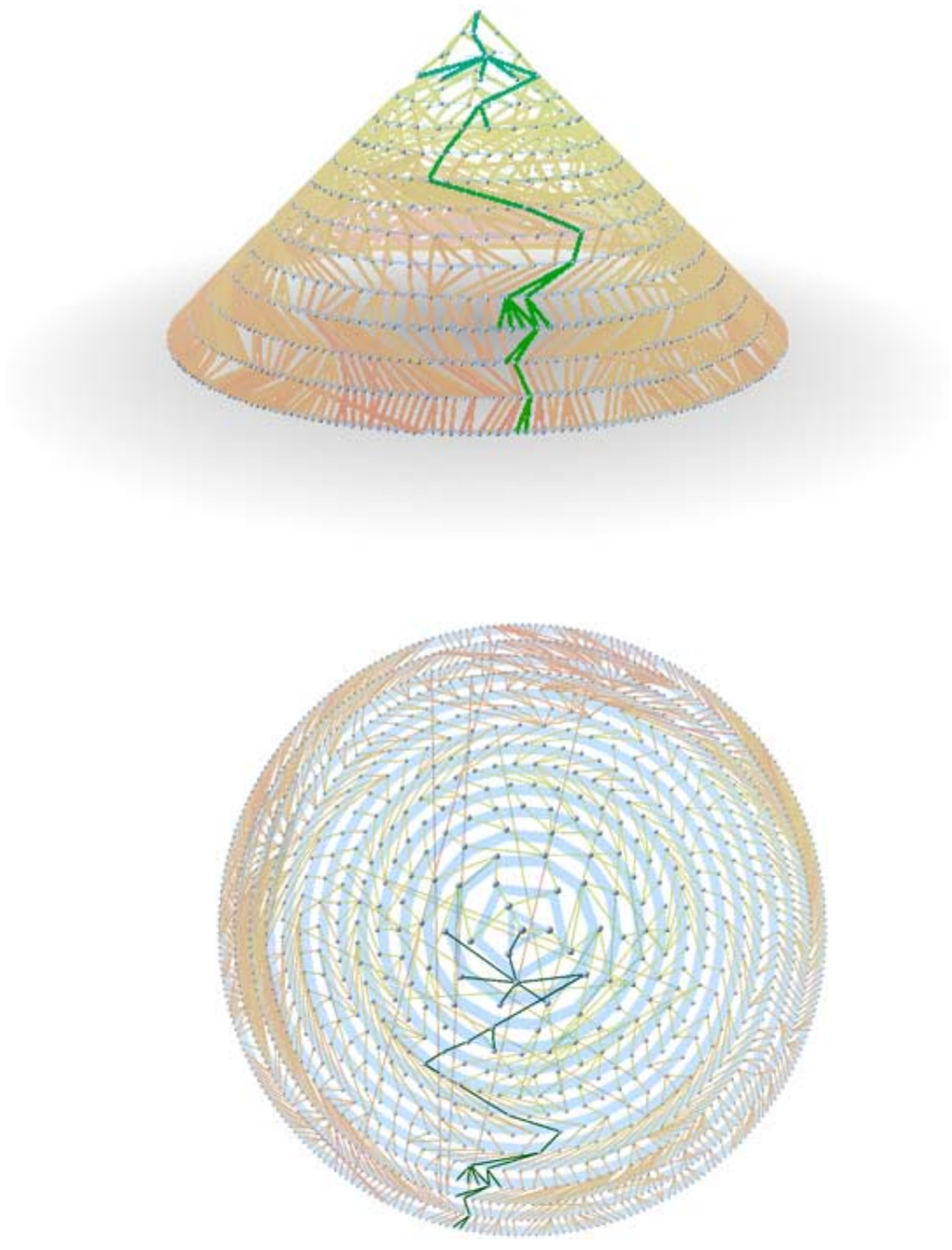
INFOCUBE



A 3-D visualization technique where hierarchical information is displayed as nested semi-transparent cubes

The outermost cubes correspond to the top level data, while the sub nodes or the lower level data are represented as smaller cubes inside the outermost cubes, and so on

3D CONE TREES



Visualize a social network data set that models the way an infection spreads from one person to the next

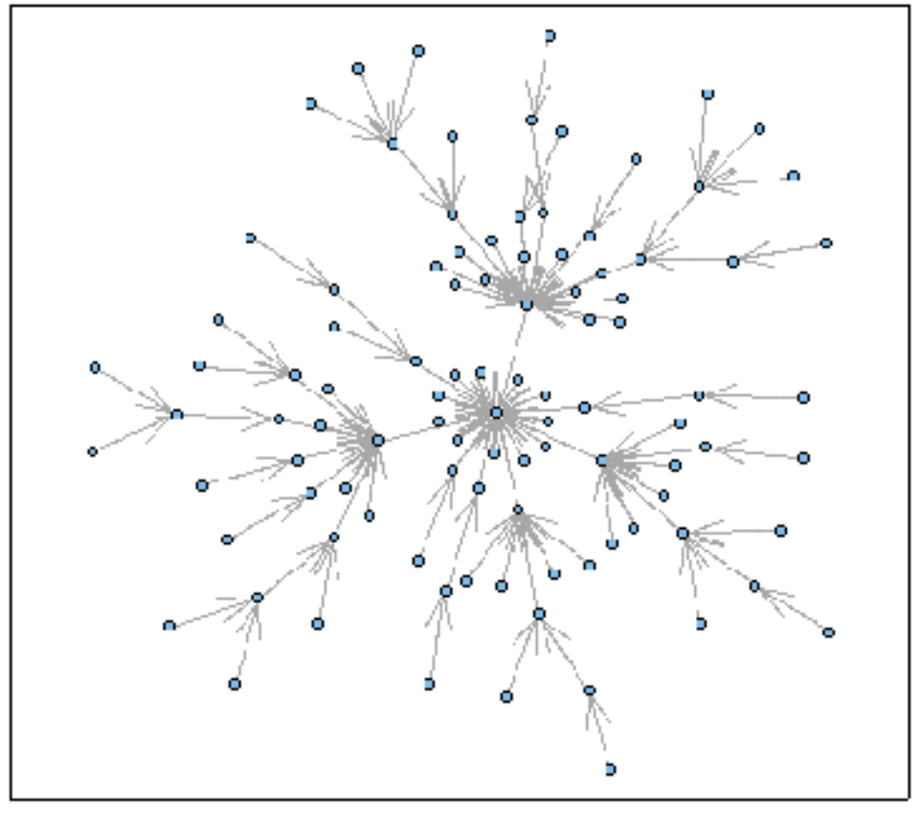
3D cone tree visualization technique works well for data with hierarchical structure for up to a thousand nodes or so

First build a 2D circle tree that arranges its nodes in concentric circles centered on the root node

Cannot avoid overlaps when projected to 2D

non-numerical data

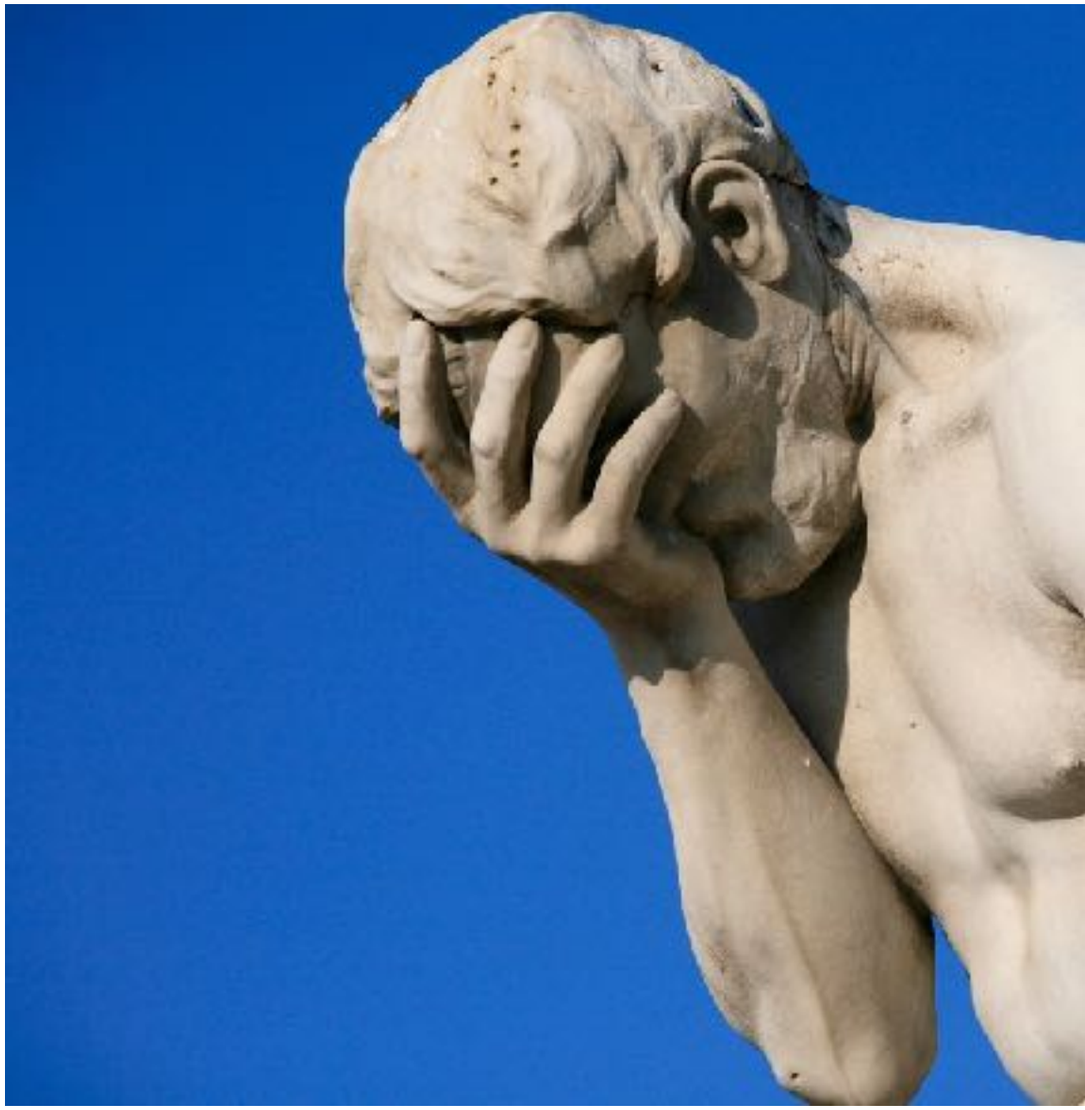
tag clouds



many network visualizations

WHAT CAN YOU DO RIGHT NOW?

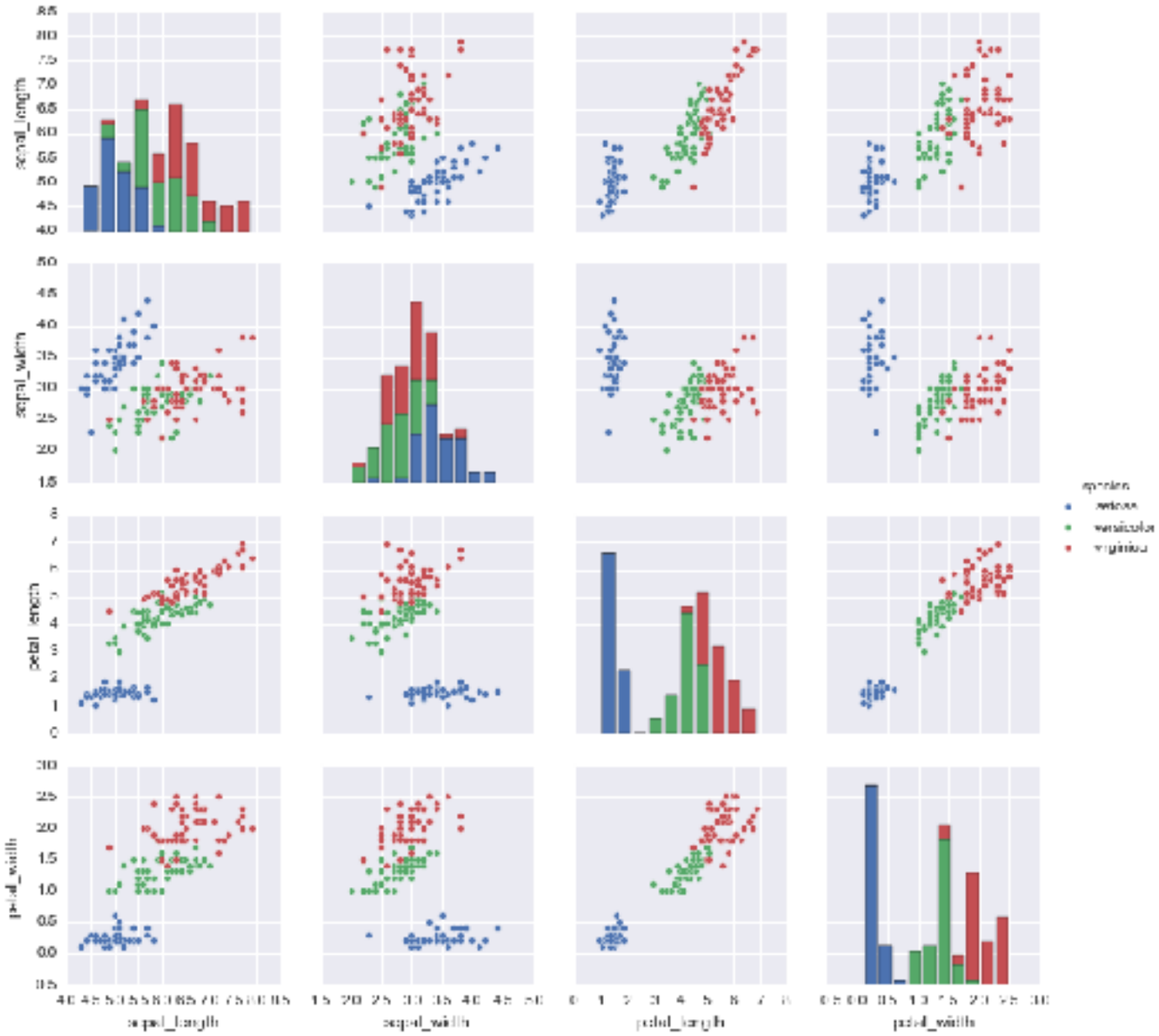
tools and code



http://www.tableau.com

seaborn

http://stanford.io/1PhonDZ

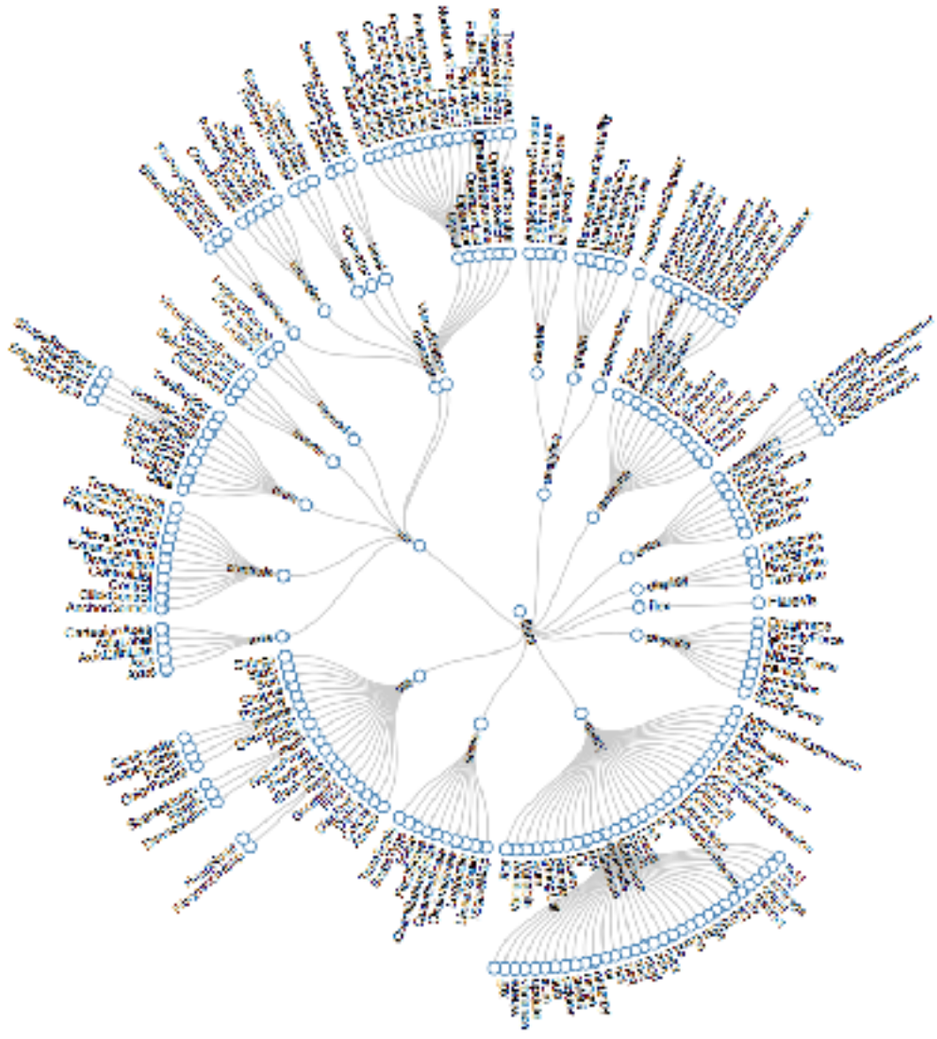


http://d3js.org

http://bokeh.pydata.org/en/latest/

mbostock's block #4063550 November 13, 2012

Radial Reingold–Tilford Tree



The tree layout implements the Reingold-Tilford algorithm for efficient, tidy arrangement of layered nodes. The depth of nodes is computed by distance from the root, leading to a ragged appearance. Cartesian orientations are also supported. Implementation based on work by Jeff Heer and Jason Davies using Buchheim et al.'s linear-time variant of the Reingold-Tilford algorithm. Data shows the Flare class hierarchy, also courtesy Jeff Heer.

Open in a new window.

Compare to this Cartesian layout.

SeeDB: Visualizing Database Queries Efficiently

[Vision Paper]

Aditya Parameswaran
Stanford University
adityagp@cs.stanford.edu

Neoklis Polyzotis
Google & UCSC
alkis@cs.ucsc.edu

Hector Garcia-Molina
Stanford University
hector@cs.stanford.edu

ABSTRACT

Data scientists rely on visualizations to interpret the data returned by queries, but finding the right visualization remains a manual task that is often laborious. We propose a DBMS that partially automates the task of finding the right visualizations for a query. In a nutshell, given an input query Q, the new DBMS optimizer will explore not only the space of physical plans for Q, but also the space of possible visualizations for the results of Q. The output will comprise a recommendation of potentially "interesting" or "useful" visualizations, where each visualization is coupled with a suitable query execution plan. We discuss the technical challenges in building this system and outline an agenda for future research.

1. INTRODUCTION

... today's researchers must consume ever higher volumes of numbers that gush, as if from a fire hose ...

— R.M. Friedhoff and T. Kiely

Data analysts must sift through huge volumes of data looking for valuable data-specific insights, trends, or anomalies. This process involves selecting the "right" subset of the data, and the "right" way to view it, so that the "insights" become apparent. Moreover, the process is often ad-hoc and consumes a lot of the analyst's time. Our vision is that *some especially cumbersome aspects* of this search for interesting insights can be automated.

To illustrate, consider the following interactive exploration workflow, which we believe is often used in practice.

Step (1): First, the analyst poses a relational query to extract some subset of data they are interested in exploring. For example, the analyst may select all records associated with "stapler" products.

Step (2): Then, the analyst considers several candidate views over this subset of data, formed by, say, aggregation and grouping; the analyst must study all of these views one by one. For example, one view may be total stapler sales by year, while another view may be the quantity in stock by sales region. Since these views have two-attributes each, we can view them as 2-dimensional graphs. For example, Figure 1(a) may be the stapler sales (y-axis) by year (x-axis), while Figure 1(c) may be the quantity (y-axis) by region (x-axis). (Figures 1(b) and 1(d) are discussed below.)

Proceedings of the VLDB Endowment, Vol. 7, No. X

demo on Friday

Step (3): Next, the analyst steps through each view, and decides which views are "interesting." This of course is the critical and time-consuming step. What makes a view like Figure 1(a) interesting or not? Well, it all depends on the application semantics and what we are comparing against. For example, Figure 1(a) shows decreasing sales over time. If we are in a recession and all product sales are down then this observation is not very interesting. However, say that Figure 1(b) shows the aggregate (all) product sales over the same time periods. Then the stapler sales view goes against the general trend: overall sales are up, but stapler sales are going down. In this case, the view is *potentially* "interesting" because it depicts a trend in the subset of data that the analyst is interested in (i.e., stapler-related data) that *deviates* from the trend in the overall data. Of course, the analyst must decide if this deviation is truly an insight for this application. Even so, our key insight is that we may be able to *identify and highlight to the analyst potentially interesting views using automated mechanisms based on deviation*. By doing so, we eliminate the laborious process of stepping through all possible views that the analyst currently performs. Once we recommend potentially interesting views, we can let the analyst make the final decision.

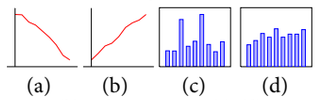
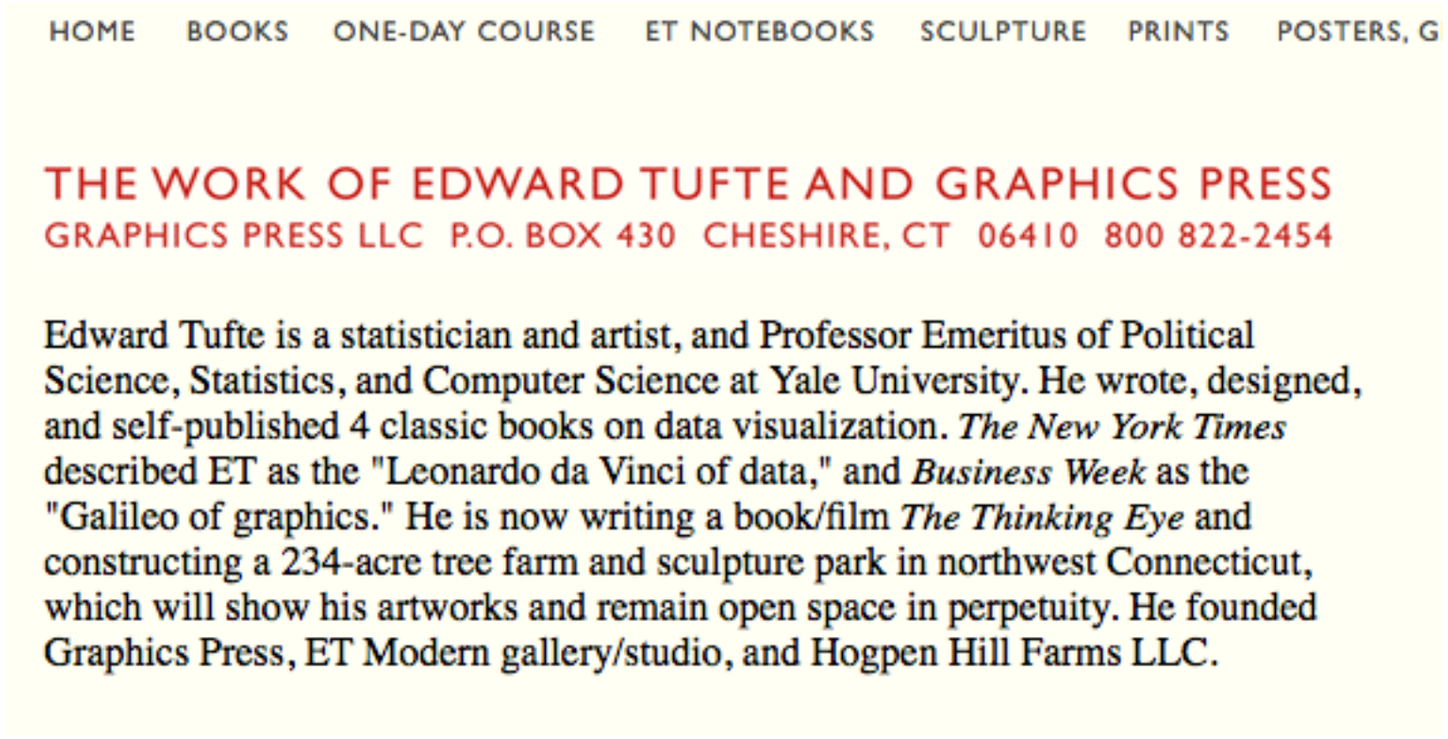


Figure 1: Views (a), (b): Sales over Time. (c), (d): Quantity by Region.

Figures 1(c) and 1(d) illustrate a different type of deviation. The first figure shows the distribution of staplers across regions, while the second figure shows the overall product distribution. Again, the stapler view does not follow the general trend: the regions that have the most staplers are not the larger regions that have most product in stock. The analysis must decide if this observation is interesting: perhaps the region that has many staplers is near the world-famous stapler-gun-wrestling contest, in which case the observation is expected. But perhaps there is a problem with the product shipping strategy, in which case the deviation is very important.

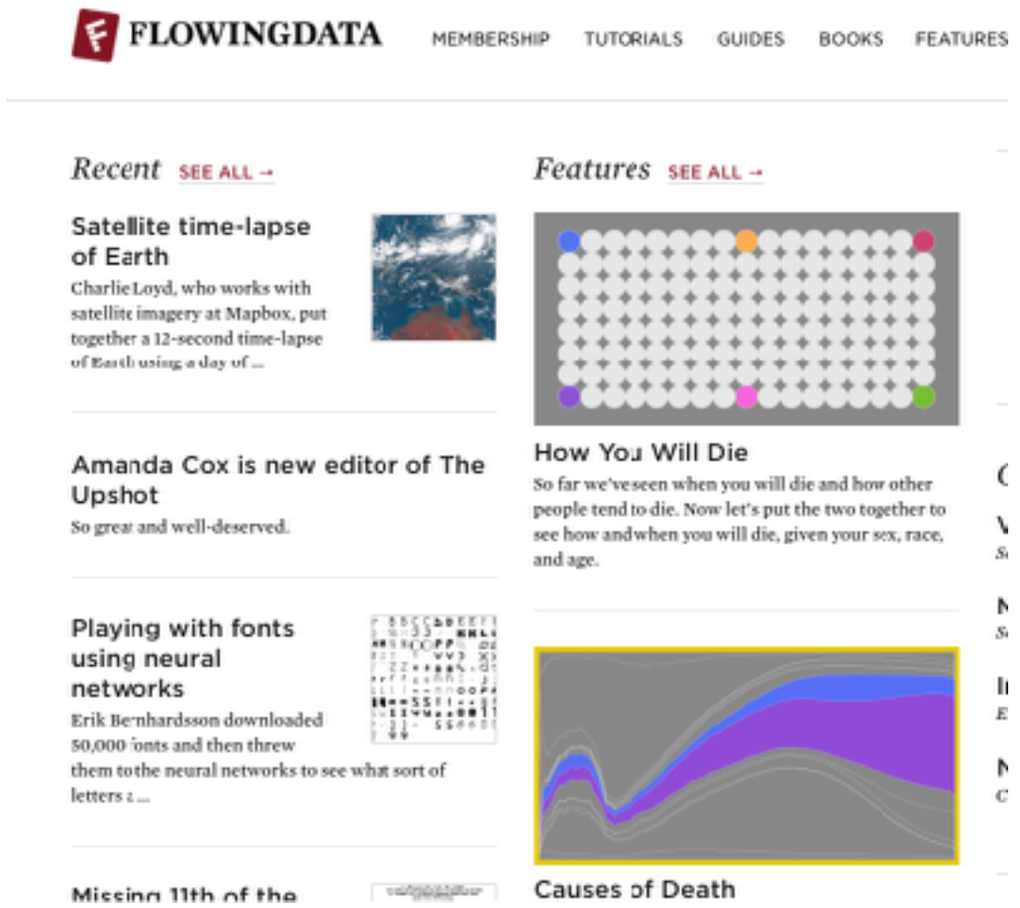
In this vision paper, we sketch our design for a new data base management system (DBMS), SeeDB, that automates the especially laborious aspects of the search for useful data insights. Figure 2 depicts SeeDB, together with a conventional DBMS. In the conventional system, the user or analyst submits a query Q and obtains data subsets. Thus, conventional systems do not provide any means for the analyst to get intuitive visual insights directly. In SeeDB, the analyst also submits a query, but instead automatically obtains views (or visualizations) of the query result that are potentially of interest. As illustrated in our examples, these visualizations help the analyst quickly interpret and understand specific "interesting/useful" aspects of the query result. Thus, with SeeDB, we fundamentally modify the query-result paradigm of databases: SeeDB is provided a query Q, and outputs visualizations of interesting aspects of Q.

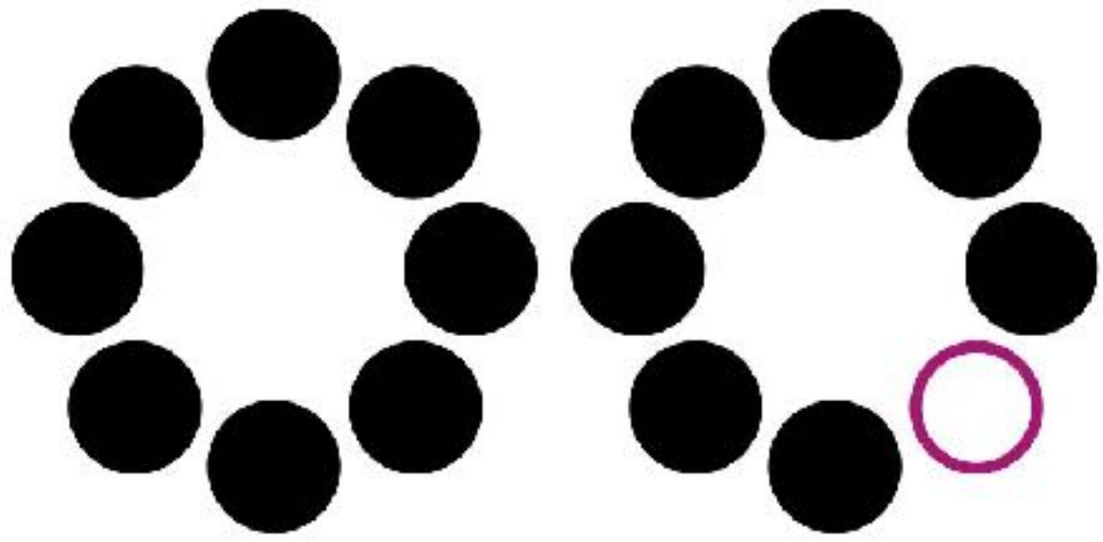


http://www.edwardtufte.com/tufte/

etc.

http://flowingdata.com





SIMILARITY

Numerical measure of how different two data objects are

Minimum dissimilarity is often 0

Upper limit varies

dis-similarity

Lower when objects are more alike

Numerical measure of how alike two data objects are

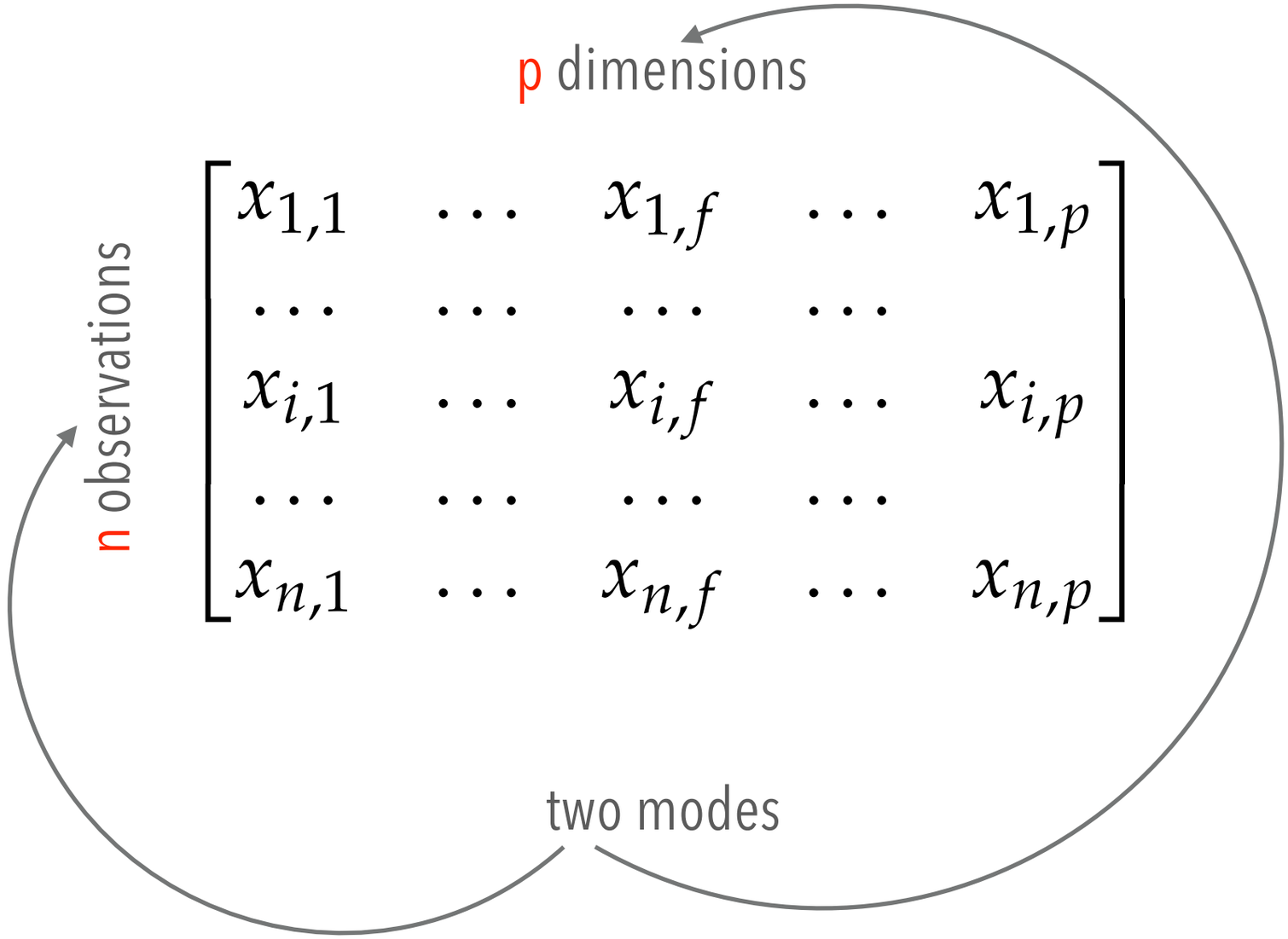
Often falls in the range [0,1]

similarity

Value is higher when objects are more alike

Proximity refers to similarity or to dissimilarity

since similarity and dis-similarity are related


p dimensions
n observations
$\begin{bmatrix} x_{1,1} & \dots & x_{1,f} & \dots & x_{1,p} \\ \dots & \dots & \dots & \dots & \\ x_{i,1} & \dots & x_{i,f} & \dots & x_{i,p} \\ \dots & \dots & \dots & \dots & \\ x_{n,1} & \dots & x_{n,f} & \dots & x_{n,p} \end{bmatrix}$
two modes

data matrix

n objects

n objects

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & \vdots & \\ d(n,1) & d(n,2) & \dots & \dots & 0 \end{bmatrix}$$

a triangular matrix

only registers distance → one mode

dissimilarity matrix

Proximity for Nominal Attributes

1

two objects

$$d(i,j) = \frac{p - m}{p}$$

m: #matches p: #variables

2

binary

Object j

Object i

	1	0	sum
1	q	r	$q+r$
0	s	t	$s+t$
sum	$q+s$	$r+t$	p

p: #number of attributes

symmetric distance $$d(i,j) = \frac{r+s}{q+r+s+t}$$

binary attributes

Object j

Object i

	1	0	sum
1	q	r	$q+r$
0	s	t	$s+t$
sum	$q+s$	$r+t$	p

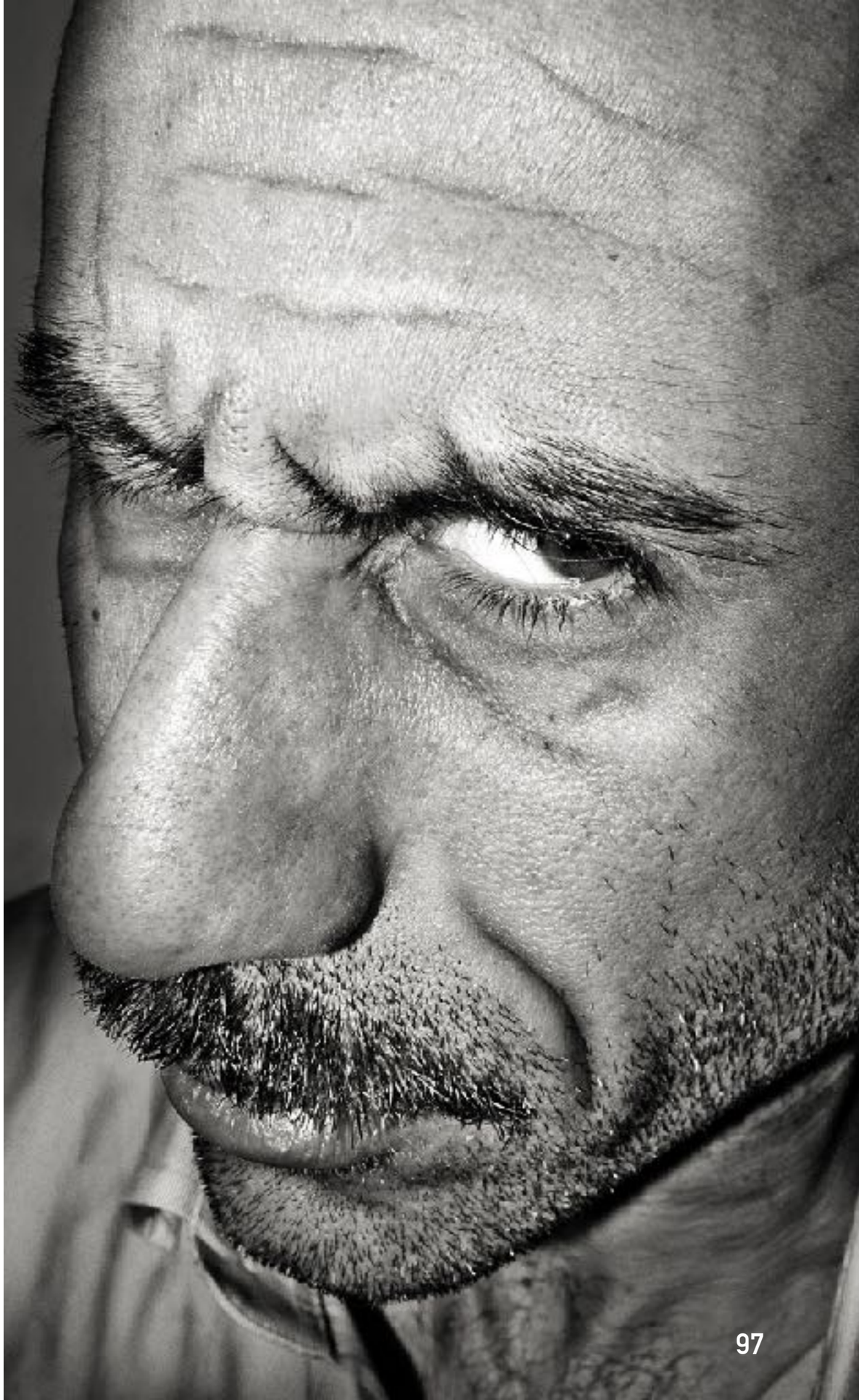
p: #number of attributes

both objects register 0 for these attributes

asymmetric distance $$d(i,j) = \frac{r+s}{q+r+s}$$

binary attributes

Why drop the case when **both** attributes register False?



Object j

Object i	1	0	*sum*
1	q	r	$q+r$
0	s	t	$s+t$
sum	$q+s$	$r+t$	p

p: #number of attributes

drop the case when both objects register 0

asymmetric similarity

Jacquard coefficient

$$sim(i,j) = 1 - d(i,j) = \frac{q}{q+r+s}$$

binary attributes

Exercise!

compute dissimilarity between Jack and Jim on asymmetric attributes

Y:yes N:no, negative P: positive

Name	*Gender*	*Fever*	*Cough*	*Test-1*	*Test-2*	*Test-3*	*Test-4*
Jack	M	Y	N	P	N	N	N
Mary	F	Y	N	P	N	P	N
Jim	M	Y	P	N	N	N	N

$$d(Jack, Jim) = \frac{r+s}{q+r+s} = \frac{2}{3}$$

on how many attributes do they disagree?

/

what are the attributes that we care about?

raw score

z is zero mean, unit variance

$$z = \frac{x - \bar{x}}{s}$$

mean

standard deviation

z-score

$$z = \frac{x - \bar{x}}{s_f}$$

$$s_f = \frac{1}{n} \sum_{i}^{n} |x_i - \bar{x}|$$

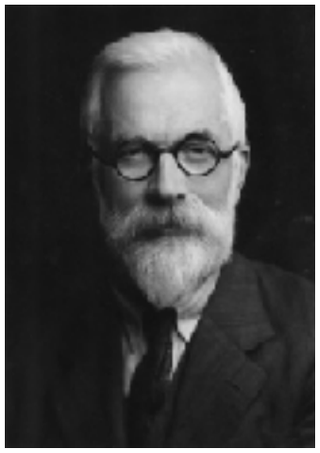
mean absolute deviation

Apply z-score normalization **prior** to computing distance, when **combining** attributes with different ranges


you
cannot
be
serious!

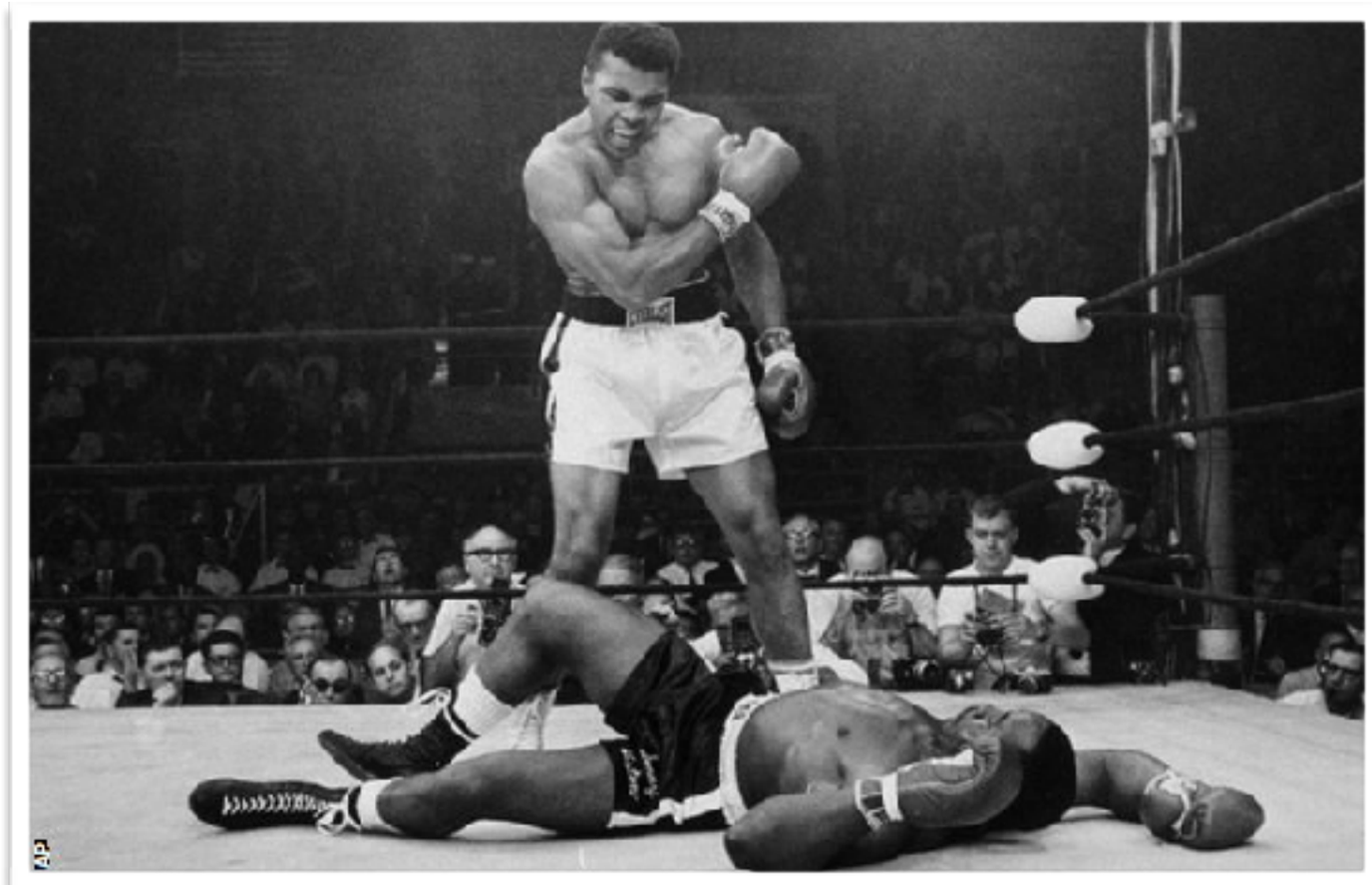
consistent sufficient efficient

standard deviation is more efficient for **normally** distributed data than MAD



RA Fisher

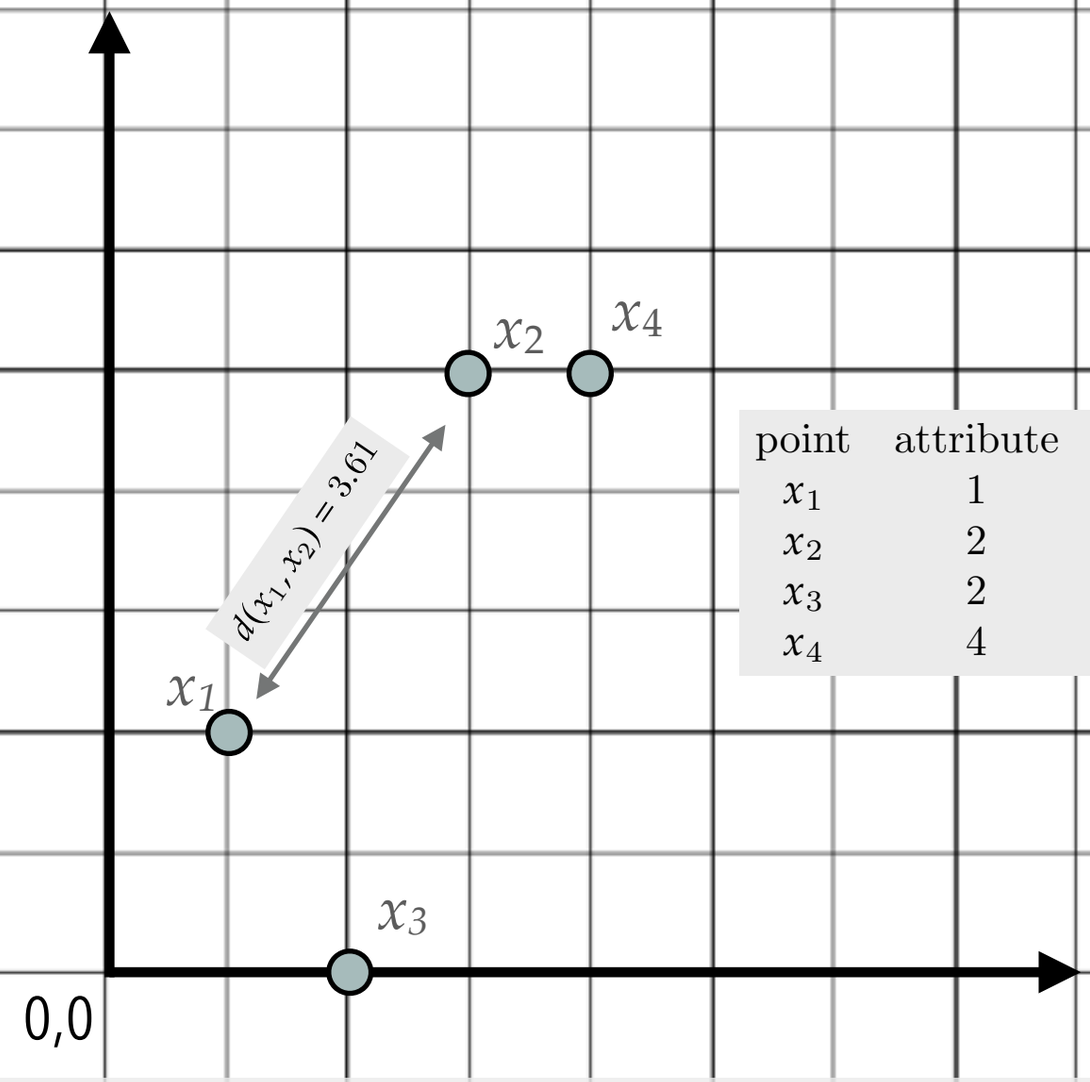
Eddington



RA Fisher's work settled the debate in favor of σ

noise, non-normal data

Gorard, S. (2005). Revisiting a 90-Year-Old Debate: The Advantages of the Mean Deviation. British Journal of Educational Studies, 53(4), 417–430.



point	attribute	attribute
x_1	1	2
x_2	2	5
x_3	2	0
x_4	4	5

euclidean dissimilarity

$$d(x, y) = \left(\sum_{i}^{p} |x_i - y_i|^h \right)^{\frac{1}{h}} \quad L_h \text{ norm}$$

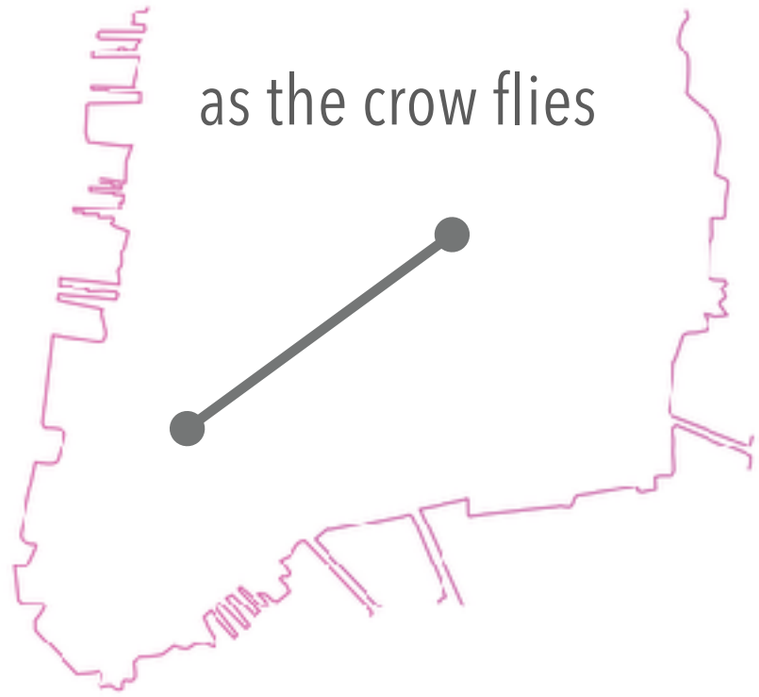
Minkowski distance

positive $d(x, y) \geq 0$

symmetry $d(x, y) = d(y, x)$

triangle inequality $d(x, y) \leq d(x, z) + d(z, y)$

$$d(x, y) = \left(\sum_{i}^{p} |x_i - y_i|^2 \right)^{\frac{1}{2}}$$



Special cases: **Euclidean distance**

$$d(x, y) = \sum_{i}^{p} |x_i - y_i|$$

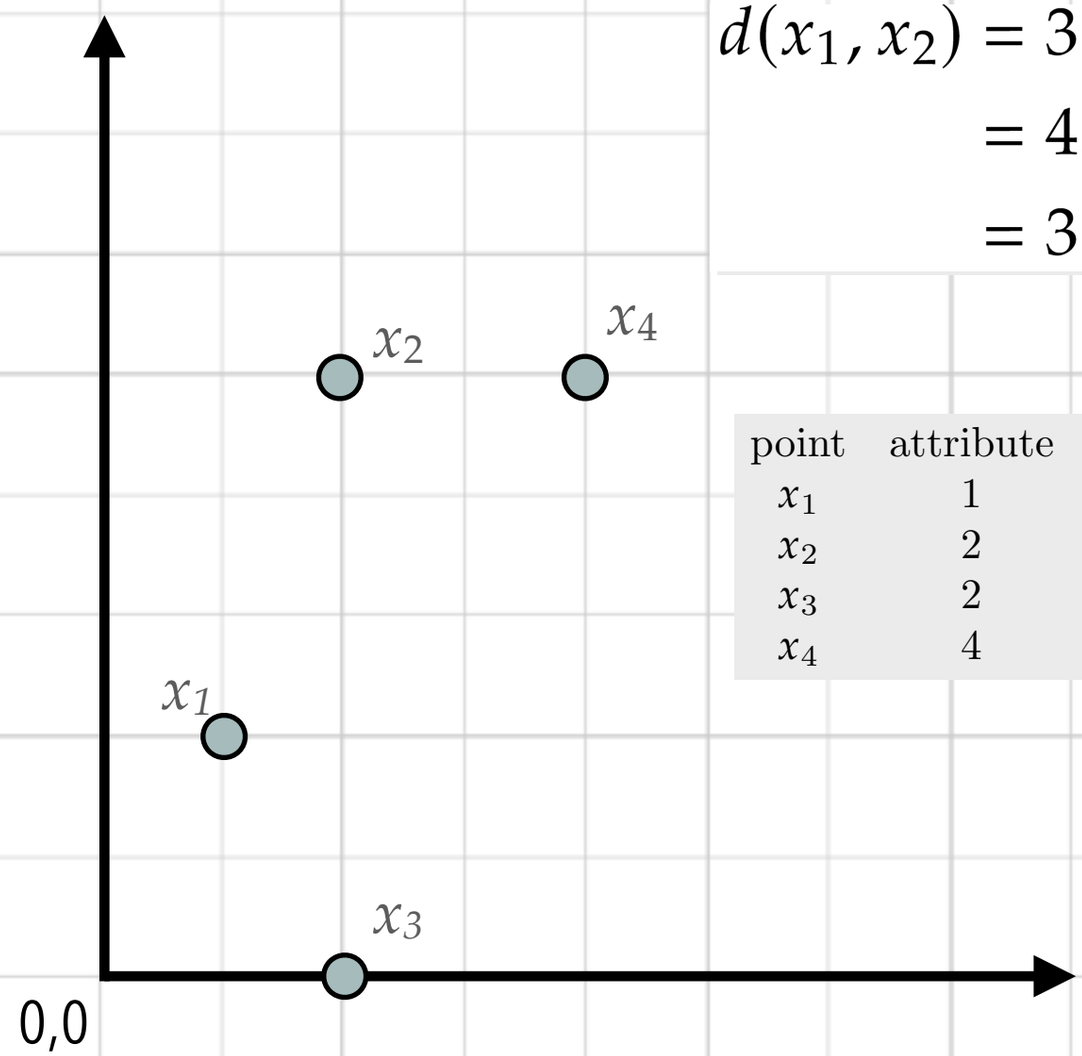
Special cases:
Manhattan distance

$$d(x, y) = \lim_{h \to \infty} \left(\sum_{i}^{p} |x_i - y_i|^h \right)^{\frac{1}{h}}$$

$$= \max |x_i - y_i|$$

This is the maximum difference between any component (attribute) of the vectors

Special cases: **Supremum norm**



$$d(x_1, x_2) = 3.16 \quad L_2$$
$$= 4.00 \quad L_1$$
$$= 3.00 \quad L_\infty$$

point	attribute	attribute
x_1	1	2
x_2	2	5
x_3	2	0
x_4	4	5

Minkowski comparison

An ordinal variable can be discrete or continuous

Order is important, e.g., rank

Can be treated like interval-scaled

ordinal variables

$$x_i \rightarrow r_i, r_i \in \{1, \ldots, M\}$$

replace x_i by their rank

$$z_i = \frac{r_i - 1}{M - 1}$$

map

compute dissimilarity using an appropriate Minkowski distance

use the distance corresponding to the attribute type

attribute

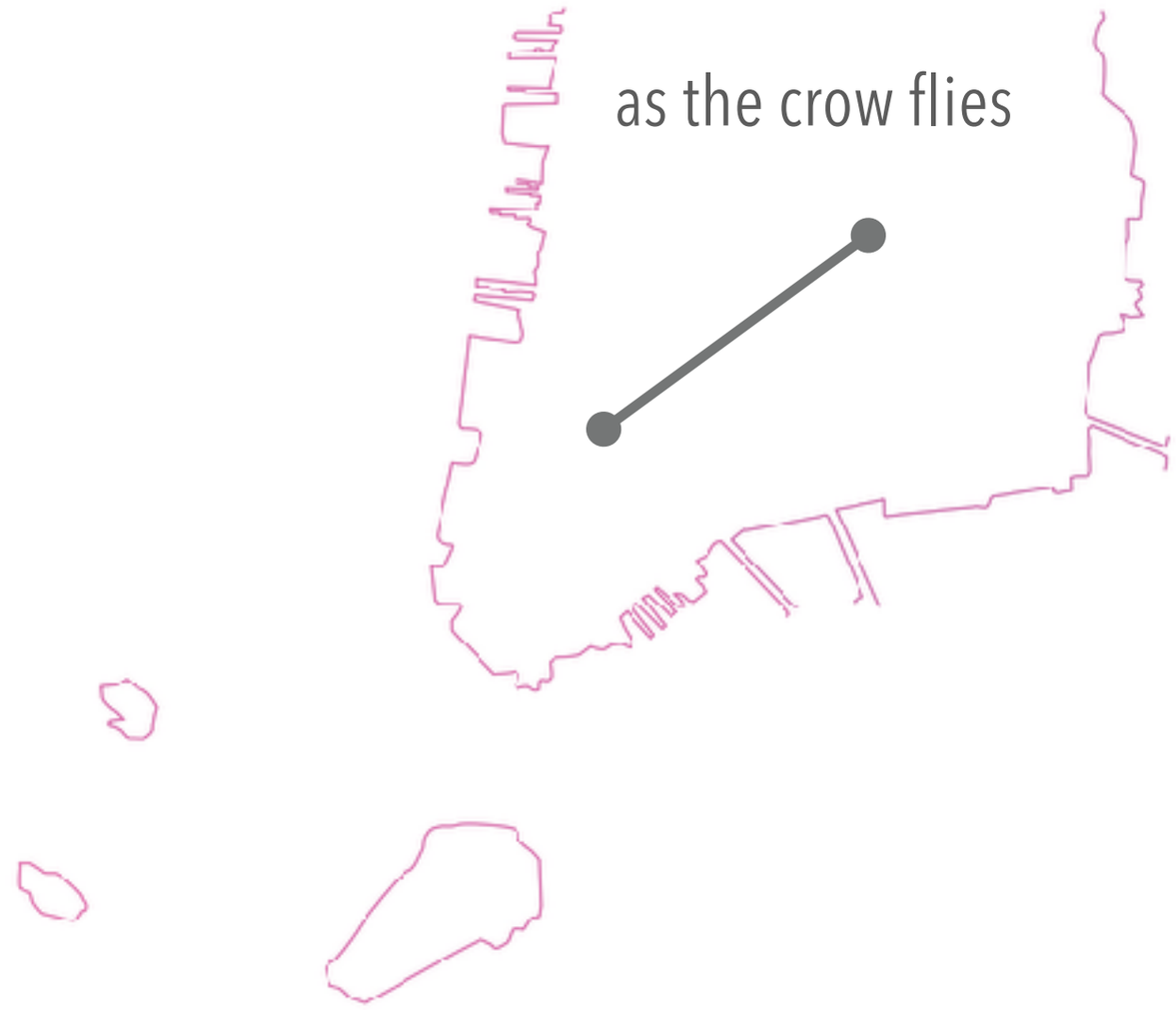
$$d(x, y) = \frac{\sum_{i=1}^{p} \omega_i d(x, y)^{(i)}}{\sum_{i=1}^{p} \omega_i}$$

mixed attributes

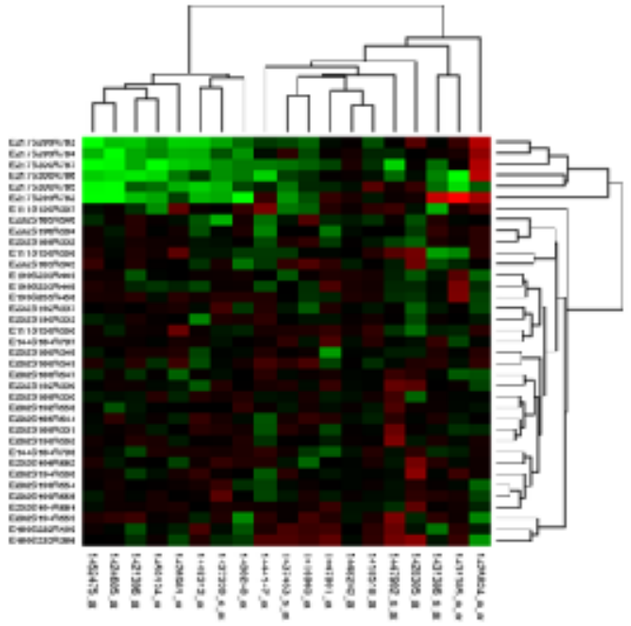
Nominal, symmetric
binary, asymmetric
binary, numeric, ordinal

Exercise!

$$d(x, y) = \left(\sum_{i}^{p} |x_i - y_i|^2 \right)^{\frac{1}{2}}$$



what happens when the number of dimensions is very large?



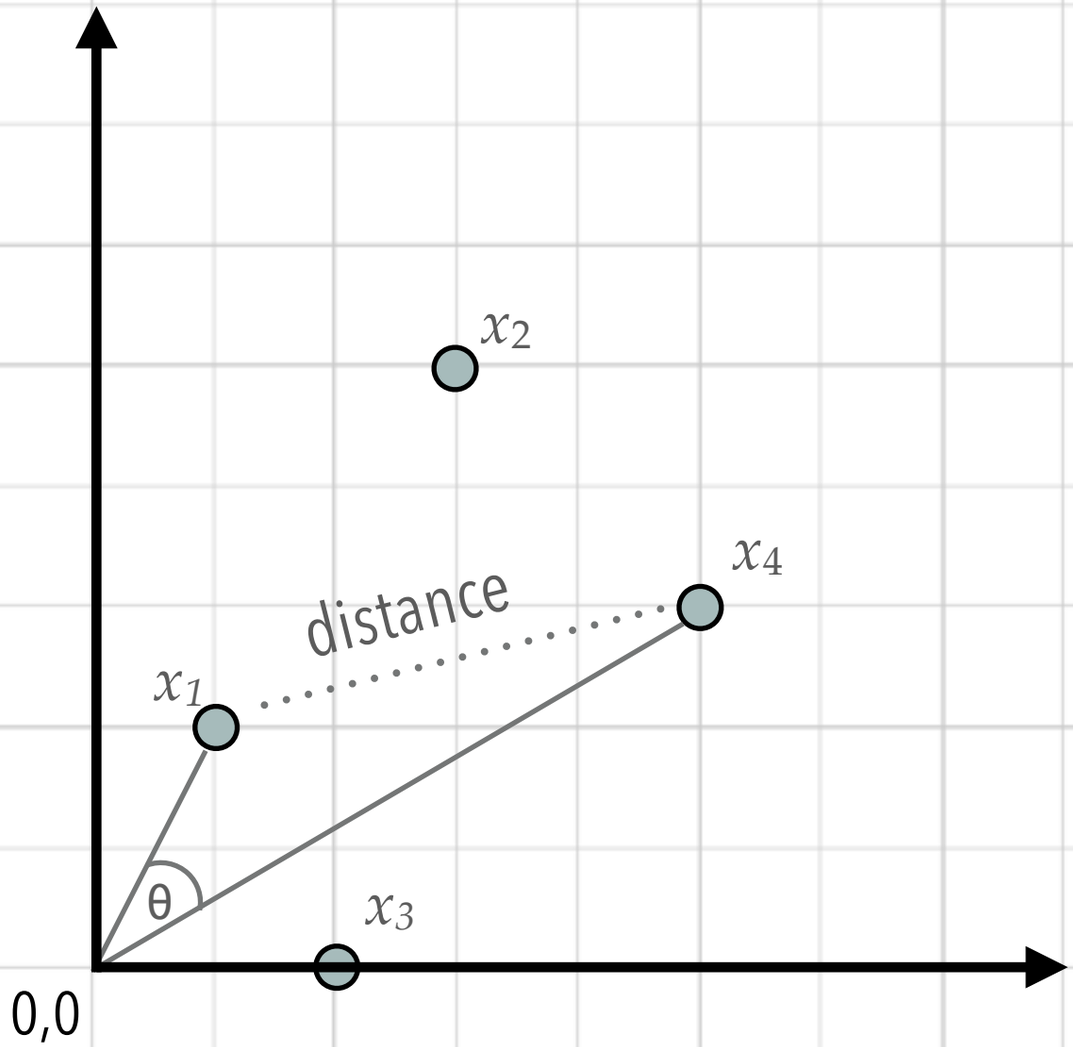
Micro-array may have tens of thousands of dimensions

Document	*team*	*coach*	*hockey*	*baseball*	*soccer*	*penalty*	*score*	*win*	*loss*	*season*
Document1	5	0	3	0	2	0	0	2	0	0
Document2	3	0	2	0	1	1	0	1	0	1
Document3	0	7	0	2	1	0	0	3	0	0
Document4	0	1	0	0	1	2	2	0	3	0

A document can be represented by thousands of attributes, each recording the frequency of a particular word (such as keywords) or phrase in the document.

cosine similarity is used for high dimensional data

instead of measuring **length**, we measure the **angle** between two objects



cosine similarity $s(x, y) = \cos(\theta) = \dfrac{< x, y >}{||x||||y||}$

dot product

L_2 norm

cosine distance $d(x, y) = 1 - s(x, y)$

Exercise!

$$x = (5, 0, 3, 0, 2, 0, 0, 2, 0, 0)$$
$$y = (3, 0, 2, 0, 1, 1, 0, 1, 0, 1)$$

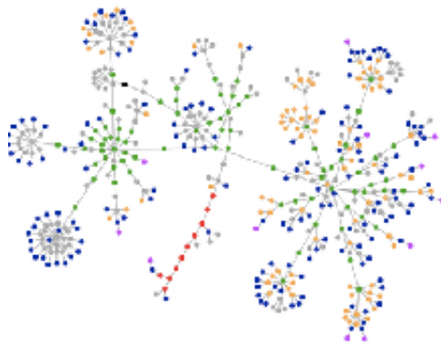
compute the cosine similarity

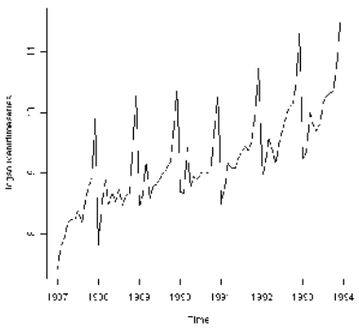
$$< x, y > = 5 \times 3 + 0 \times 0 + 3 \times 2 + 0 \times 0 + 2 \times 1 + 0 \times 1 + 2 \times 1 + 0 \times 0 + 0 \times 1$$
$$= 25$$
$$||x|| = 6.481$$
$$||y|| = 4.12$$

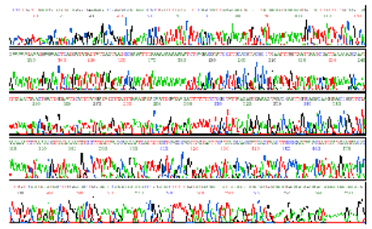
$$s(x, y) = \cos(\theta) = \frac{< x, y >}{||x||||y||}$$

$$d(x, y) = 1 - s(x, y)$$

What distance **measure** should we use and when?







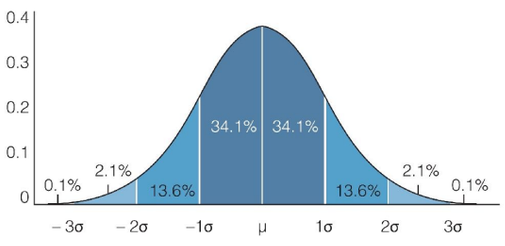
TID	Items
1	Bread, Coke, Milk
2	Beer, Bread
3	Beer, Coke, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Coke, Diaper, Milk

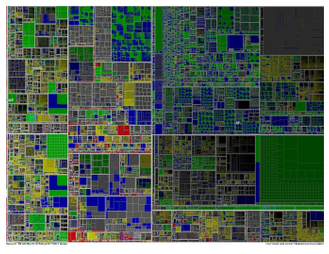
Many types of data sets, e.g., numerical, text, graph, Web, image.

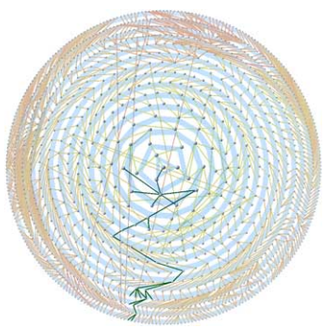
Data attribute types: nominal, binary, ordinal, interval-scaled, ratio-scaled

Gain insight into the data by:

- Basic statistical data description: central tendency, dispersion, graphical displays
- Data visualization: map data onto graphical primitives
- Measure data similarity







SUMMARY

Above steps are the beginning of data preprocessing

Many methods have been developed but still an active area of research

$$d(x, y) = \left(\sum_{i}^{p} |x_i - y_i|^h \right)^{\frac{1}{h}} \qquad d(x, y) = \cos(\theta) = \frac{\langle x, y \rangle}{||x|| ||y||}$$