LECTURE 2: DATA (PRE-)PROCESSING

Dr. Dhaval Patel

CSE, IIT-Roorkee

….

- In Previous Class,
  - We discuss various type of Data with examples

- In this Class,
  - We focus on Data pre-processing – “an important milestone of the Data Mining Process”

Data analysis pipeline

- Mining is not the only step in the analysis process

- Preprocessing: real data is noisy, incomplete and inconsistent. Data cleaning is required to make sense of the data
  - Techniques: Sampling, Dimensionality Reduction, Feature Selection.

- Post-Processing: Make the data actionable and useful to the user : Statistical analysis of importance & Visualization.

Data Preprocessing

- Attribute Values
- Attribute Transformation
  - Normalization (Standardization)
  - Aggregation
  - Discretization
- Sampling
- Dimensionality Reduction
- Feature subset selection
- Distance/Similarity Calculation
- Visualization

Attribute Values

Data is described using attribute values

Attribute Values

- Attribute values are **numbers** or **symbols** assigned to an attribute

- Distinction between attributes and attribute values
  - Same attribute can be mapped to different attribute values
    - Example: height can be measured in feet or meters

  - Different attributes can be mapped to the same set of values
    - Example: Attribute values for ID and age are integers
    - But properties of attribute values can be different
      - ID has no limit but age has a maximum and minimum value

Types of Attributes

- There are different types of attributes
  - Nominal
    - Examples: ID numbers, eye color, zip codes
  - Ordinal
    - Examples: rankings (e.g., taste of potato chips on a scale from 1-10), grades, height in {tall, medium, short}
  - Interval
    - Examples: calendar dates
  - Ratio
    - Examples: length, time, counts

Types of Attributes

Attribute Level	Transformation	Comments
Nominal	Any permutation of values	If all employee ID numbers were reassigned, would it make any difference?
Ordinal	An order preserving change of values, i.e., $new\_value = f(old\_value)$ where f is a monotonic function.	An attribute encompassing the notion of good, better best can be represented equally well by the values
Interval	$new\_value = a * old\_value + b$ where a and b are constants	Calendar dates can be converted – financial vs. Gregorian etc.
Ratio	$new\_value = a * old\_value$	Length can be measured in meters or feet.

Discrete and Continuous Attributes

- Discrete Attribute
  - Has only a finite or countable infinite set of values
  - Examples: zip codes, counts, or the set of words in a collection of documents
  - Often represented as integer variables.
- Continuous Attribute
  - Has real numbers as attribute values
  - Examples: temperature, height, or weight.
  - Practically, real values can only be measured and represented using a finite number of digits.

Data Quality

Data has attribute values

Then,

How good our Data w.r.t. these attribute values?

Data Quality

- Examples of data quality problems:
  - Noise and outliers
  - Missing values
  - Duplicate data

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	10000K	Yes
6	No	NULL	60K	No
7	Yes	Divorced	220K	NULL
8	No	Single	85K	Yes
9	No	Married	90K	No
9	No	Single	90K	No

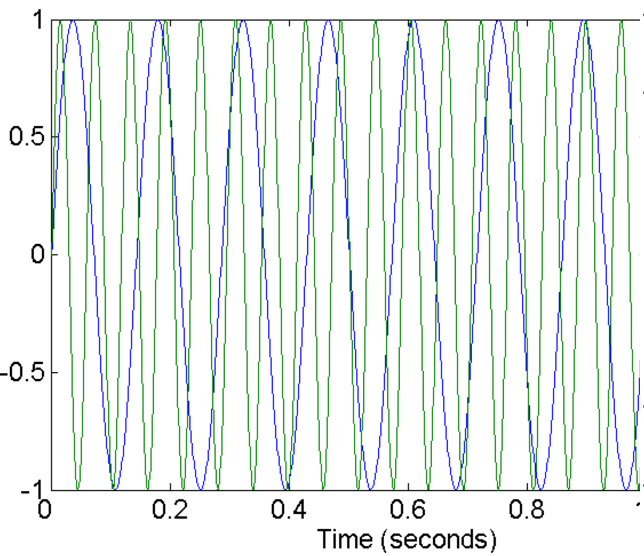
A mistake or a millionaire?

Missing values

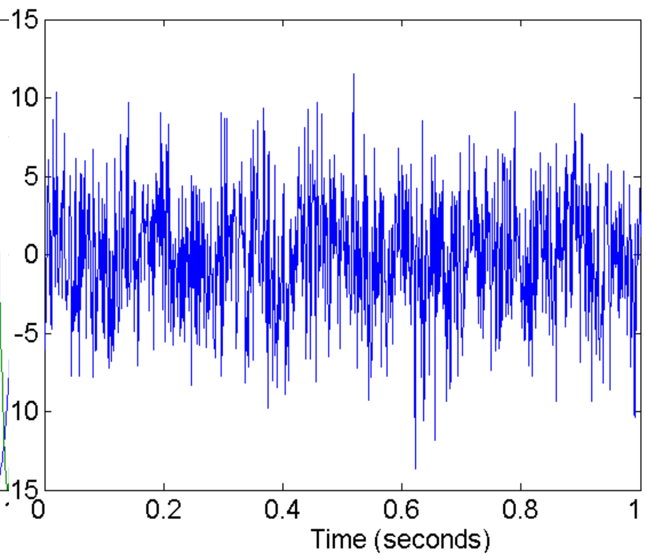
Inconsistent duplicate entries

Data Quality: Noise

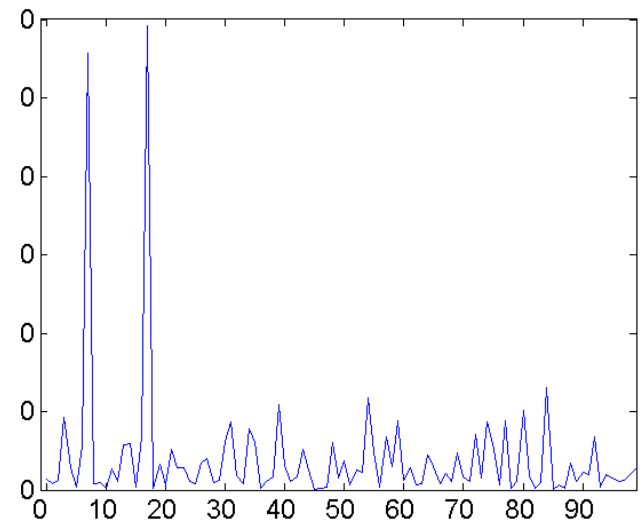
- Noise refers to modification of original values
  - Examples: distortion of a person's voice when talking on



Two Sine Waves



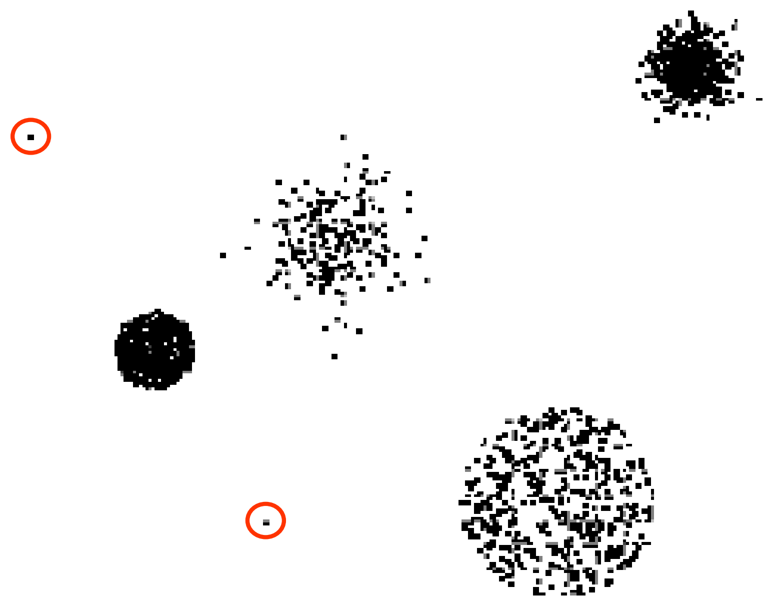
Two Sine Waves + Noise



Frequency Plot (FFT)

Data Quality: Outliers

- Outliers are data objects with characteristics that are considerably different than most of the other data objects in the data set



Data Quality: Missing Values

- Reasons for missing values
  - Information is not collected
    (e.g., people decline to give their age and weight)
  - Attributes may not be applicable to all cases
    (e.g., annual income is not applicable to children)

- Handling missing values
  - Eliminate Data Objects
  - Estimate Missing Values
  - Ignore the Missing Value During Analysis
  - Replace with all possible values (weighted by their probabilities)

Data Quality: Duplicate Data

- Data set may include data objects that are duplicates, or almost duplicates of one another
  - Major issue when merging data from heterogeous sources

- Examples:
  - Same person with multiple email addresses

- Data cleaning
  - Process of dealing with duplicate data issues

Data Quality: Handle Noise(Binning)

- Binning
  - sort data and partition into (equi-depth) bins
  - smooth by bin means, bin median, bin boundaries, etc.
- Regression
  - smooth by fitting a regression function
- Clustering
  - detect and remove outliers
- Combined computer and human inspection
  - detect suspicious values automatically and check by human

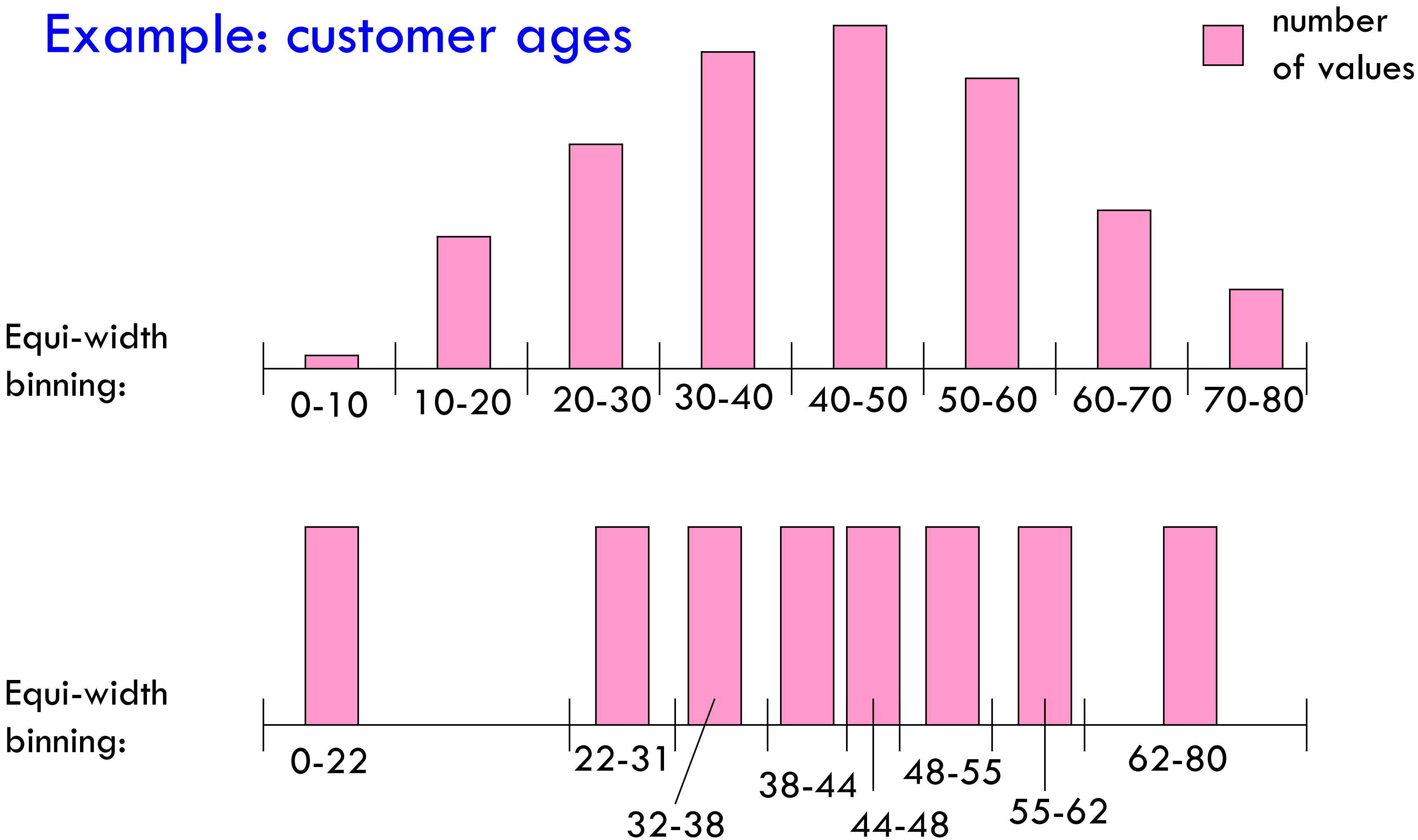
Data Quality: Handle Noise(Binning)

- Equal-width binning
  - Divides the range into *N* intervals of *equal size*
  - Width of intervals:
  - Simple
  - Outliers may dominate result
- Equal-depth binning
  - Divides the range into *N* intervals,

    each containing approximately *same number* of records
  - Skewed data is also handled well

Simple Methods: Binning

Example: customer ages



Data Quality: Handle Noise(Binning)

Example: Sorted price values 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

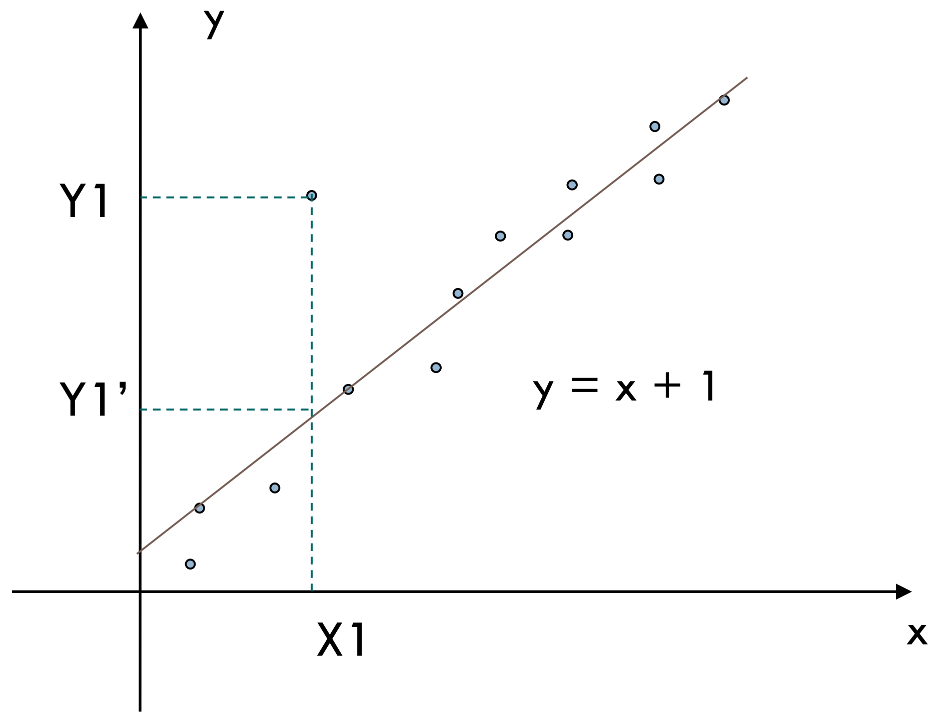
* Partition into three (equi-depth) bins
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34

* Smoothing by bin means
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29

* Smoothing by bin boundaries
  - Bin 1: 4, 4, 4, 15
  - Bin 2: 21, 21, 25, 25
  - Bin 3: 26, 26, 26, 34

Data Quality: Handle Noise(*Regression*)

- Replace noisy or missing values by predicted values
- Requires model of attribute dependencies (maybe wrong!)
- Can be used for data smoothing or for handling missing data



Data Quality

There are many more noise handling techniques ….

> Imputation

Data Transformation

Data has an attribute values

Then,

Can we compare these attribute values?

For Example: Compare following two records

(1) (5.9 ft, 50 Kg)

(2) (4.6 ft, 55 Kg)

Vs.

(3) (5.9 ft, 50 Kg)

(4) (5.6 ft, 56 Kg)

We need Data Transformation to makes different dimension(attribute) records comparable …

Data Transformation Techniques

- Normalization: scaled to fall within a small, specified range.
  - min-max normalization
  - z-score normalization
  - normalization by decimal scaling

- Centralization:
  - Based on fitting a distribution to the data
  - Distance function between distributions
    - KL Distance
    - Mean Centering

Data Transformation: Normalization

- min-max normalization

$$v' = \frac{v - min}{max - min}(new\_max - new\_min) + new\_min$$

- z-score normalization

$$v' = \frac{v - mean}{stand\_dev}$$

- normalization by decimal scaling

$$v' = \frac{v}{10^j}$$ Where j is the smallest integer such that Max(| v' |)<1

Example: Data Transformation

- Assume, min and max value for height and weight.
- Now, apply Min-Max normalization to both attributes as given follow

(1) (5.9 ft, 50 Kg)

(2) (4.6 ft, 55 Kg)

Vs.

(3) (5.9 ft, 50 Kg)

(4) (5.6 ft, 56 Kg)

- Compare your results…

Data Transformation: Aggregation

- Combining two or more attributes (or objects) into a single attribute (or object)

- Purpose
  - Data reduction
    - Reduce the number of attributes or objects
  - Change of scale
    - Cities aggregated into regions, states, countries, etc
  - More "stable" data
    - Aggregated data tends to have less variability

Data Transformation: Discretization

- Motivation for Discretization
  - Some data mining algorithms only accept categorical attributes
  - May improve understandability of patterns

Data Transformation: Discretization

- Task
  - Reduce the number of values for a given continuous attribute by partitioning the range of the attribute into intervals
  - Interval labels replace actual attribute values

- Methods
  - Binning (as explained earlier)
  - Cluster analysis (will be discussed later)
  - Entropy-based Discretization (Supervised)

Simple Discretization Methods: Binning

- Equal-width (distance) partitioning:
  - Divides the range into N intervals of equal size: uniform grid
  - if A and B are the lowest and highest values of the attribute, the width of intervals will be: $W = (B - A)/N$.
  - The most straightforward, but outliers may dominate presentation
  - Skewed data is not handled well.

- Equal-depth (frequency) partitioning:
  - Divides the range into N intervals, each containing approximately same number of samples
  - Good data scaling
  - Managing categorical attributes can be tricky.

Information/Entropy

- Given probabilitites p_1, p_2, .., p_s whose sum is 1, ***Entropy*** is defined as:

$$\boxed{H\left(p_1, p_2, \ldots, p_s\right) = \sum_{i=1}^{s}\left(p_i log\left(1/p_i\right)\right)}$$

- Entropy measures the amount of randomness or surprise or uncertainty.

- Only takes into account non-zero probabilities

Entropy-Based Discretization

- Given a set of samples S, if S is partitioned into two intervals S1 and S2 using boundary T, the entropy after partitioning is

$$E(S,T) = \frac{|S_1|}{|S|} Ent(S_1) + \frac{|S_2|}{|S|} Ent(S_2)$$

- The boundary that minimizes the entropy function over all possible boundaries is selected as a binary discretization.
- The process is recursively applied to partitions obtained until some stopping criterion is met, e.g.,

$$Ent(S) - E(T,S) > \delta$$

- Experiments show that it may reduce data size and improve classification accuracy

Data Sampling

Data may be **Big**

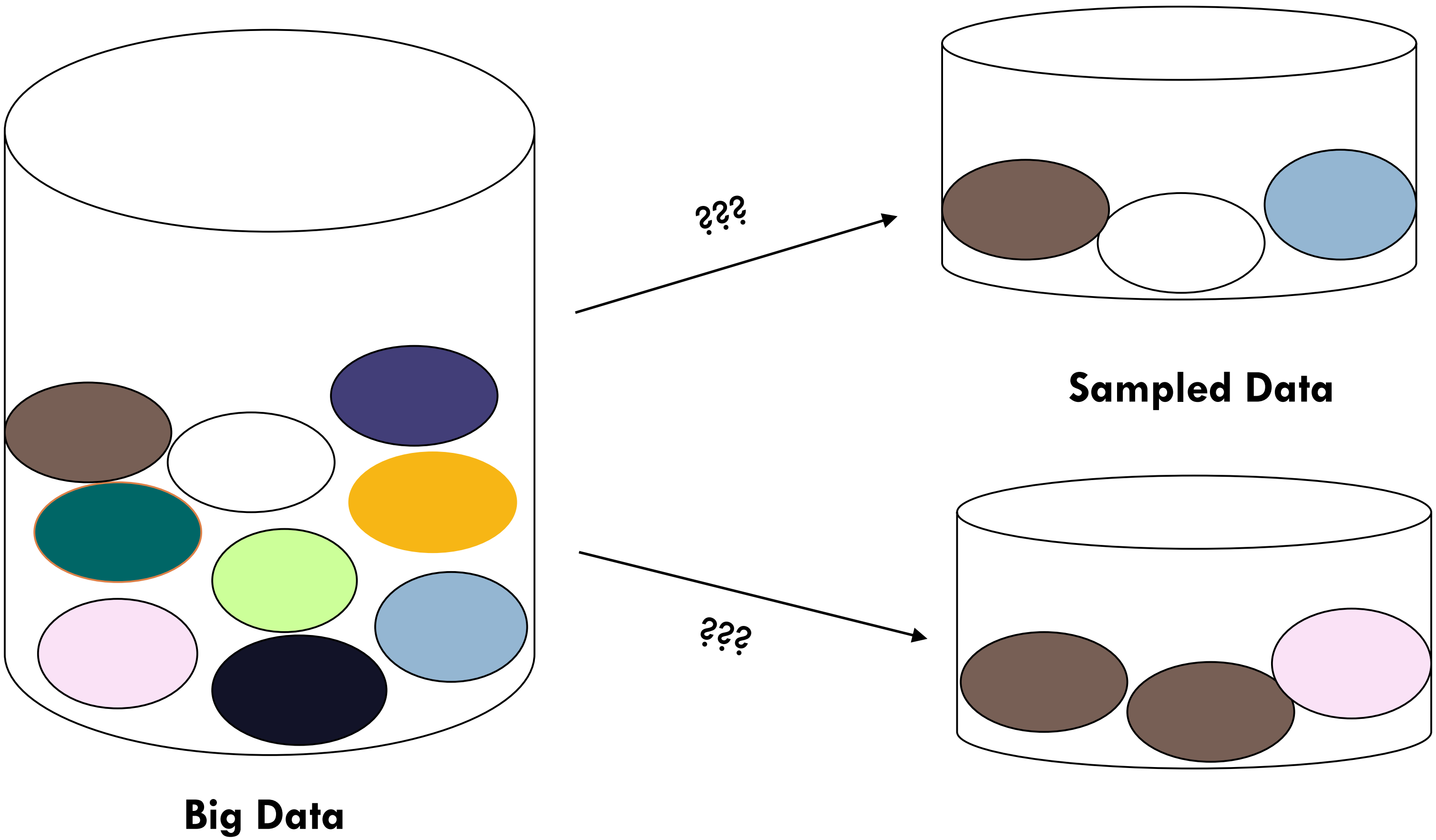
Then,

Can we make is it **Small** by selecting some part of it?

Data Sampling can do this…

"Sampling is the main technique employed for data selection."

Data Sampling



Data Sampling

- Statisticians sample because obtaining the entire set of data of interest is too expensive or time consuming.
  - Example: What is the average height of a person in Ioannina?
    - We cannot measure the height of everybody

- Sampling is used in data mining because processing the entire set of data of interest is too expensive or time consuming.
  - Example: We have 1M documents. What fraction has at least 100 words in common?
    - Computing number of common words for all pairs requires 10^12 comparisons

Data Sampling …

- The key principle for effective sampling is the following:
  - Using a sample will work almost as well as using the entire data sets, if the sample is representative
  - A sample is representative if it has approximately the same property (of interest) as the original set of data
  - Otherwise we say that the sample introduces some bias
  - What happens if we take a sample from the university campus to compute the average height of a person at Ioannina?

Types of Sampling

- Simple Random Sampling
  - There is an equal probability of selecting any particular item
- Sampling without replacement
  - As each item is selected, it is removed from the population
- Sampling with replacement
  - Objects are not removed from the population as they are selected for the sample.
    - In sampling with replacement, the same object can be picked up more than once
- Stratified sampling
  - Split the data into several partitions; then draw random samples from each partition

Types of Sampling

- Simple Random Sampling
  - There is an equal probability of selecting any particular item
- Sampling without replacement
  - As each item is selected, it is removed from the population
- Sampling with replacement
  - Objects are not removed from the population as they are selected for the sample.
    - In sampling with replacement, the same object can be picked up more than once. This makes analytical computation of probabilities easier
    - E.g., we have 100 people, 51 are women $P(W) = 0.51$, 49 men $P(M) = 0.49$. If I pick two persons what is the probability $P(W,W)$ that both are women?
      - Sampling with replacement: $P(W,W) = 0.51^2$
      - Sampling without replacement: $P(W,W) = 51/100 * 50/99$

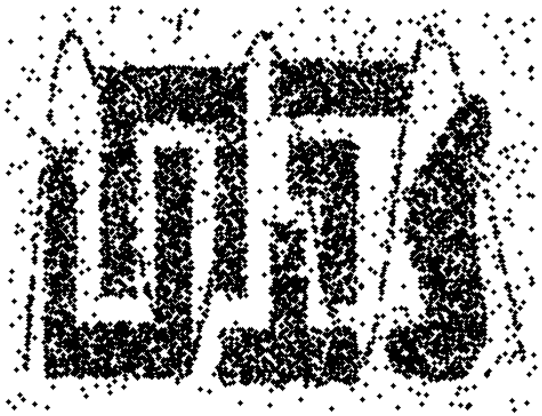
Types of Sampling

- Stratified sampling
  - Split the data into several groups; then draw random samples from each group.
    - Ensures that both groups are represented.
  - Example 1. I want to understand the differences between legitimate and fraudulent credit card transactions. 0.1% of transactions are fraudulent. What happens if I select 1000 transactions at random?
    - I get 1 fraudulent transaction (in expectation). Not enough to draw any conclusions. Solution: sample 1000 legitimate and 1000 fraudulent transactions

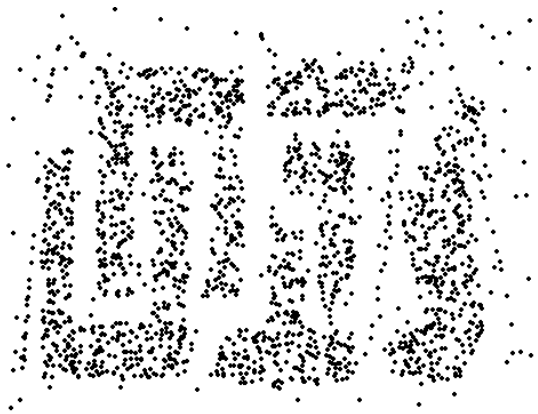
Probability Reminder: If an event has probability p of happening and I do N trials, the expected number of times the event occurs is pN

  - Example 2. I want to answer the question: Do web pages that are linked have on average more words in common than those that are not? I have 1M pages, and 1M links, what happens if I select 10K pairs of pages at random?
    - Most likely I will not get any links. Solution: sample 10K random pairs, and 10K links

Sample Size



8000 points

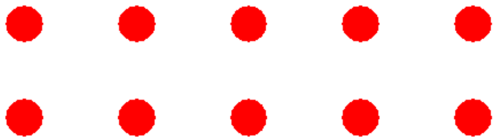


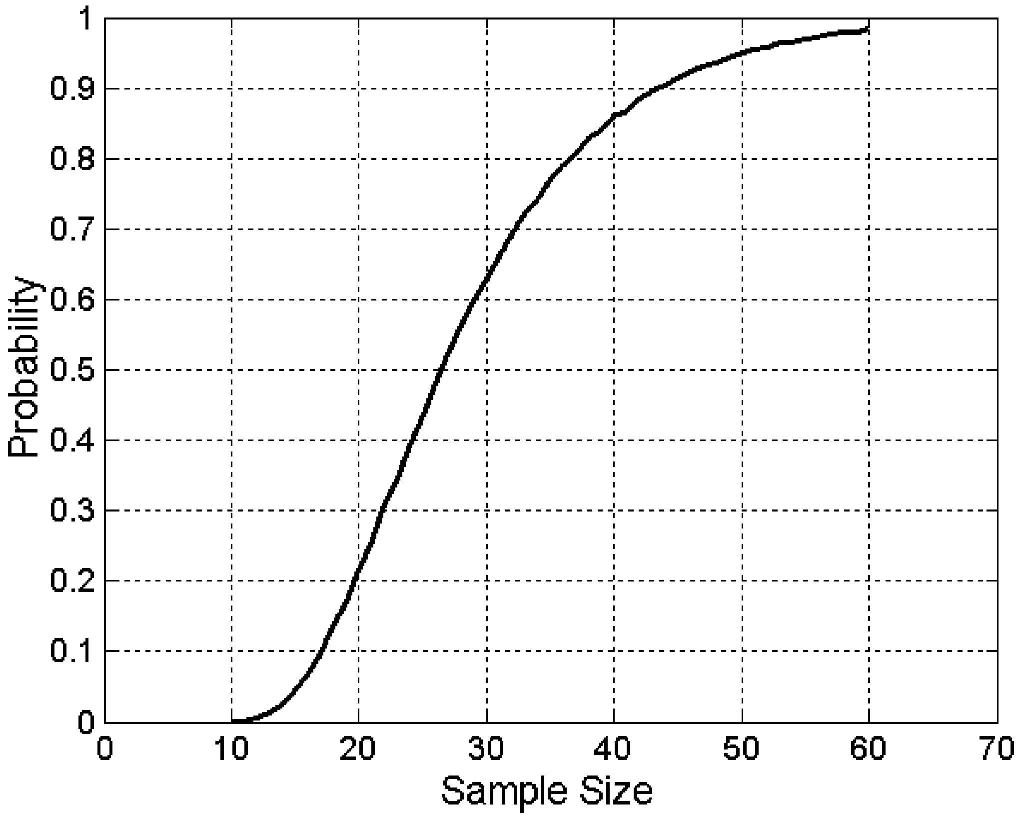
2000 Points

500 Points

Sample Size

- **What sample size is necessary to get at least one object from each of 10 groups.**





A data mining challenge

- You have N integers and you want to sample **one integer** uniformly at random. How do you do that?

- The integers are coming in a stream: you do not know the size of the stream in advance, and there is not enough memory to store the stream in memory. You can only keep a constant amount of integers in memory
- How do you sample?
  - Hint: if the stream ends after reading n integers the last integer in the stream should have probability $1/n$ to be selected.

- Reservoir Sampling:
  - Standard interview question for many companies

Reservoir Sampling

```
array R[k]; //
result integer i, j;
// fill the reservoir array
for each i in 1 to k do
      R[i] := S[i]
done;
for each i in k+1 to length(S) do
      j := random(1, i);
      if j <= k then
         R[j] := S[i]
      fi done
```

Reservoir Sampling

- Algorithm: With probability 1/n select the n-th item of the stream and replace the previous choice.

- Claim: Every item has probability 1/N to be selected after N items have been read.

- Proof
  - What is the probability of the n-the item to be selected?
    - $\frac{1}{n}$
  - What is the probability of the n-th items to survive for N-n rounds?
    - $\left(1-\frac{1}{n+1}\right)\left(1-\frac{1}{n+2}\right)\cdots\left(1-\frac{1}{N}\right)$

Reservoir sampling

- Do you know "**Fisher-Yates shuffle**"
  - S is an array with n number, a is also an array of size n
  - $a[0] \leftarrow S[0]$

    for i **from** 1 **to** n - 1 **do**

    $r \leftarrow random$ $(0 .. i)$

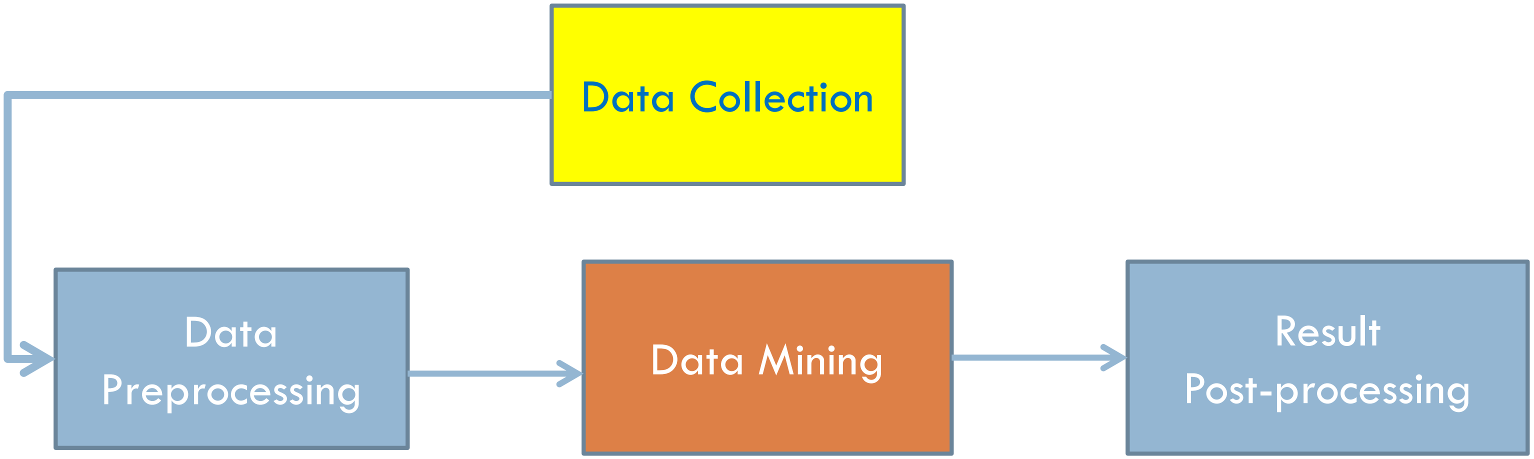
    $a[i] \leftarrow a[r]$

    $a[r] \leftarrow S[i]$

A (detailed) data preprocessing example

- Suppose we want to mine the comments/reviews of people on Yelp and Foursquare.

Example: Data Collection



- Today there is an abundance of data online
  - Facebook, Twitter, Wikipedia, Web, etc…
- We can extract interesting information from this data, but first we need to collect it
  - Customized crawlers, use of public APIs
  - Additional cleaning/processing to parse out the useful parts
  - Respect of crawling etiquette

Example: Mining Task

- Collect all reviews for the top-10 most reviewed restaurants in NY in Yelp
  - (thanks to Sahishnu)
- Find few terms that best describe the restaurants.
- Algorithm?

Example: Data

- I heard so many good things about this place so I was pretty juiced to try it. I'm from Cali and I heard Shake Shack is comparable to IN-N-OUT and I gotta say, Shake Shake wins hands down. Surprisingly, the line was short and we waited about 10 MIN. to order. I ordered a regular cheeseburger, fries and a black/white shake. So yummerz. I love the location too! It's in the middle of the city and the view is breathtaking. Definitely one of my favorite places to eat in NYC.

- I'm from California and I must say, Shake Shack is better than IN-N-OUT, all day, err'day.

- Would I pay $15+ for a burger here? No. But for the price point they are asking for, this is a definite bang for your buck (though for some, the opportunity cost of waiting in line might outweigh the cost savings) Thankfully, I came in before the lunch swarm descended and I ordered a shake shack (the special burger with the patty + fried cheese & portabella topping) and a coffee milk shake. The beef patty was very juicy and snugly packed within a soft potato roll. On the downside, I could do without the fried portabella-thingy, as the crispy taste conflicted with the juicy, tender burger. How does shake shack compare with in-and-out or 5-guys? I say a very close tie, and I think it comes down to personal affliations. On the shake side, true to its name, the shake was well churned and very thick and luscious. The coffee flavor added a tangy taste and complemented the vanilla shake well. Situated in an open space in NYC, the open air sitting allows you to munch on your burger while watching people zoom by around the city. It's an oddly calming experience, or perhaps it was the food coma I was slowly falling into. Great place with food at a great price.

Example: First cut

- Do simple processing to "normalize" the data (remove punctuation, make into lower case, clear white spaces, other?)
- Break into words, keep the most popular words

the 27514	the 16710	the 16010	the 14241
and 14508	and 9139	and 9504	and 8237
i 13088	a 8583	i 7966	a 8182
a 12152	i 8415	to 6524	i 7001
to 10672	to 7003	a 6370	to 6727
of 8702	in 5363	it 5169	of 4874
ramen 8518	it 4606	of 5159	you 4515
was 8274	of 4365	is 4519	it 4308
is 6835	is 4340	sauce 4020	is 4016
it 6802	burger 432	in 3951	was 3791
in 6402	was 4070	this 3519	pastrami 3748
for 6145	for 3441	was 3453	in 3508
but 5254	but 3284	for 3327	for 3424
that 4540	shack 3278	you 3220	sandwich 2928
you 4366	shake 3172	that 2769	that 2728
with 4181	that 3005	but 2590	but 2715
pork 4115	you 2985	food 2497	on 2247
my 3841	my 2514	on 2350	this 2099
this 3487	line 2389	my 2311	my 2064
wait 3184	this 2242	cart 2236	with 2040
not 3016	fries 2240	chicken 2220	not 1655
we 2984	on 2204	with 2195	your 1622
at 2980	are 2142	rice 2049	so 1610
on 2922	with 2095	so 1825	have 1585

Example: First cut

- Do simple processing to "normalize" the data (remove punctuation, make into lower case, clear white spaces, other?)
- Break into words, keep the most popular words

the 27514	the 16710	the 16010	the 14241
and 14508	and 9139	and 9504	and 8237
i 13088	a 8583	i 7966	a 8182
a 12152	i 8415	to 6524	i 7001
to 10672	to 7003	a 6370	to 6727
of 8702	in 5363	it 5169	of 4874
ramen 8518	it 4606	of 5159	you 4515
was 8274	of 4365	is 4519	it 4308
is 6835	is 4340	**sauce 4020**	is 4016
it 6802	**burger 432**	in 3951	was 3791
in 6402	was 4070	this 3519	**pastrami 3748**
for 6145	for 3441	was 3453	in 3508
but 5254	but 3284	for 3327	for 3424
that 4540	**shack 3278**	you 3220	**sandwich 2928**
you 4366	**shake 3172**	that 2769	that 2728
with 4181	that 3005	but 2590	but 2715
pork 4115	you 2985	food 2497	on 2247
my 3841	my 2514		
this 3487	line 2389		
wait 3184	this 2242	**cart 2236**	
not 3016	**fries 2240**	**chicken 2220**	not 1655
we 2984	on 2204	with 2195	your 1622
at 2980	are 2142	rice 2049	so 1610
on 2922	with 2095	so 1825	have 1585

Most frequent words are stop words

Example: Second cut

- Remove stop words
  - Stop-word lists can be found online.

```
a,about,above,after,again,against,all,am,an,and,any,are,aren't,as,at,be,be
cause,been,before,being,below,between,both,but,by,can't,cannot,could,could
n't,did,didn't,do,does,doesn't,doing,don't,down,during,each,few,for,from,f
urther,had,hadn't,has,hasn't,have,haven't,having,he,he'd,he'll,he's,her,he
re,here's,hers,herself,him,himself,his,how,how's,i,i'd,i'll,i'm,i've,if,in
,into,is,isn't,it,it's,its,itself,let's,me,more,most,mustn't,my,myself,no,
nor,not,of,off,on,once,only,or,other,ought,our,ours,ourselves,out,over,own
,same,shan't,she,she'd,she'll,she's,should,shouldn't,so,some,such,than,tha
t,that's,the,their,theirs,them,themselves,then,there,there's,these,they,th
ey'd,they'll,they're,they've,this,those,through,to,too,under,until,up,very
,was,wasn't,we,we'd,we'll,we're,we've,were,weren't,what,what's,when,when's
,where,where's,which,while,who,who's,whom,why,why's,with,won't,would,would
n't,you,you'd,you'll,you're,you've,your,yours,yourself,yourselves,
```

Example: Second cut

- Remove stop words
  - Stop-word lists can be found online.

ramen 8572	burger 4340	sauce 4023	pastrami 3782
pork 4152	shack 3291	food 2507	sandwich 2934
wait 3195	shake 3221	cart 2239	place 1480
good 2867	line 2397	chicken 2238	good 1341
place 2361	fries 2260	rice 2052	get 1251
noodles 2279	good 1920	hot 1835	katz's 1223
ippudo 2261	burgers 1643	white 1782	just 1214
buns 2251	wait 1508	line 1755	like 1207
broth 2041	just 1412	good 1629	meat 1168
like 1902	cheese 1307	lamb 1422	one 1071
just 1896	like 1204	halal 1343	deli 984
get 1641	food 1175	just 1338	best 965
time 1613	get 1162	get 1332	go 961
one 1460	place 1159	one 1222	ticket 955
really 1437	one 1118	like 1096	food 896
go 1366	long 1013	place 1052	sandwiches 813
food 1296	go 995	go 965	can 812
bowl 1272	time 951	can 878	beef 768
can 1256	park 887	night 832	order 720
great 1172	can 860	time 794	pickles 699
best 1167	best 849	long 792	time 662
		people 790	

Example: Second cut

- Remove stop words
  - Stop-word lists can be found online.

ramen 8572	burger 4340	sauce 4023	pastrami 3782
pork 4152	shack 3291	food 2507	sandwich 2934
wait 3195	shake 3221	cart 2239	place 1480
good 2867	line 2397	chicken 2238	good 1341
place 2361	fries 2260	rice 2052	get 1251
noodles 2279	good 1920	hot 1835	katz's 1223
ippudo 2261	burgers 1643	white 1782	just 1214
buns 2251	wait 1508	line 1755	like 1207
broth 2041	just 1412	good 1629	meat 1168
like 1902	cheese 1307	lamb 1422	one 1071
just 1896	like 1204	halal 1343	deli 984
get 1641	food 1175	just 1338	best 965
time 1613	get 1162	get 1332	go 961
one 1460			
really 1437			
go 1366	long 1013	place 1052	sandwiches 813
food 1296	go 995	go 965	can 812
bowl 1272	time 951	can 878	beef 768
can 1256	park 887	night 832	order 720
great 1172	can 860	time 794	pickles 699
best 1167	best 849	long 792	time 662
		people 790	

Commonly used words in reviews, not so interesting

Example: IDF

- Important words are the ones that are unique to the document (differentiating) compared to the rest of the collection
  - All reviews use the word "like". This is not interesting
  - We want the words that characterize the specific restaurant

- Document Frequency $DF(w)$: fraction of documents that contain word w.

$$DF(w) = \frac{D(w)}{D}$$

$D(w)$: num of docs that contain word w

D: total number of documents

- Inverse Document Frequency $IDF(w)$:

$$IDF(w) = \log\left(\frac{1}{DF(w)}\right)$$

- Maximum when unique to one document : $IDF(w) = \log(D)$
- Minimum when the word is common to all documents: $IDF(w) = 0$

Example: TF-IDF

- The words that are best for describing a document are the ones that are important for the document, but also unique to the document.

- TF(w,d): term frequency of word w in document d
  - Number of times that the word appears in the document
  - Natural measure of importance of the word for the document

- IDF(w): inverse document frequency
  - Natural measure of the uniqueness of the word w

- TF-IDF(w,d) = TF(w,d) × IDF(w)

Example: Third cut

- Ordered by TF-IDF

ramen 3057.4176194	fries 806.08537330	lamb 985.655290756243	pastrami 1931.94250908298 6
akamaru 2353.24196	custard 729.607519	halal 686.038812717726	katz's 1120.62356508209 4
noodles 1579.68242	shakes 628.4738038	53rd 375.685771863491	rye 1004.28925735888 2
broth 1414.7133955	shroom 515.7790608	gyro 305.809092298788	corned 906.113544700399 2
miso 1252.60629058	burger 457.2646379	pita 304.984759446376	pickles 640.487221580035 4
hirata 709.1962086	crinkle 398.347221	cart 235.902194557873	reuben 515.779060830666 1
hakata 591.7643688	burgers 366.624854	platter 139.45990308004	matzo 430.583412389887 1
shiromaru 587.1591	madison 350.939350	chicken/lamb 135.852520	sally 428.110484707471 2
noodle 581.8446147	shackburger 292.42	carts 120.274374158359	harry 226.323810772916 4
tonkotsu 529.59457	'shroom 287.823136	hilton 84.2987473324223	mustard 216.079238853014 6
ippudo 504.5275695	portobello 239.806	lamb/chicken 82.8930633	cutter 209.535243462458 1
buns 502.296134008	custards 211.83782	yogurt 70.0078652365545	carnegie 198.655512713779 3
ippudo's 453.60926	concrete 195.16992	52nd 67.5963923222322	katz 194.387844446609 7
modern 394.8391629	bun 186.9621782983	6th 60.7930175345658 9	knish 184.206807439524 1
egg 367.3680056967	milkshakes 174.996	4am 55.4517744447956 5	sandwiches 181.415707218 8
shoyu 352.29551922	concretes 165.7861	yellow 54.4470265206673	brisket 131.945865389878 4
chashu 347.6903490	portabello 163.483	tzatziki 52.9594571388	fries 131.613054313392 7
karaka 336.1774235	shack's 159.334353	lettuce 51.323016802268	salami 127.621117258549 3
kakuni 276.3102111	patty 152.22603588	sammy's 50.656872045869	knishes 124.339595021678 1
ramens 262.4947006	ss 149.66803104461	sw 50.5668577816893 3	delicatessen 117.488967607 2
bun 236.5122638036	patties 148.068287	platters 49.906597000310	deli's 117.431839742696 1
wasabi 232.3667512	cam 105.9496067806	falafel 49.479699521204	carver 115.129254649702 1
dama 221.048168927	milkshake 103.9720	sober 49.2211422635451	brown's 109.441778045519 2
brulee 201.1797390	lamps 99.011158998	moma 48.1589121730374	matzoh 108.22149937072 1

Example: Third cut

- TF-IDF takes care of stop words as well
- We do not need to remove the stop words since they will get $IDF(w) = 0$

Example: Decisions, decisions…

- When mining real data you often need to make some
  - What data should we collect? How much? For how long?
  - Should we throw out some data that does not seem to be useful?

An actual review

```
AAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAAAA AAA
```

    - Too frequent data (stop words), too infrequent (errors?), erroneous data, missing data, outliers
  - How should we weight the different pieces of data?

- Most decisions are application dependent. Some information may be lost but we can usually live with it (most of the times)

- Dealing with real data is hard…

Dimensionality Reduction

Each record has many attributes

- useful, useless or correlated

Then,

Can we select some small subset of attributes?

Dimensionality Reduction can do this….

Dimensionality Reduction

- Why?
  - When dimensionality increases, data becomes increasingly sparse in the space that it occupies
  - **Curse of Dimensionality** : Definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful
- Objectives:
  - Avoid curse of dimensionality
  - Reduce amount of time and memory required by data mining algorithms
  - Observation: **Certain Dimensions are correlated**

Dimensionality Reduction

- Allow data to be more easily visualized

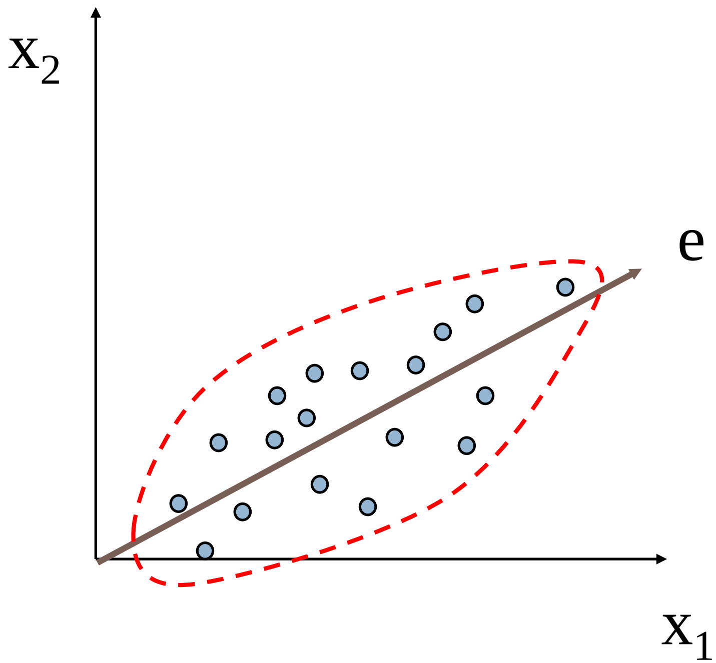
- May help to eliminate irrelevant features or reduce noise

- Techniques

  - Principle Component Analysis or Singular Value Decomposition
  - (Mapping Data to New Space) : Wavelet Transform
  - Others: supervised and non-linear techniques

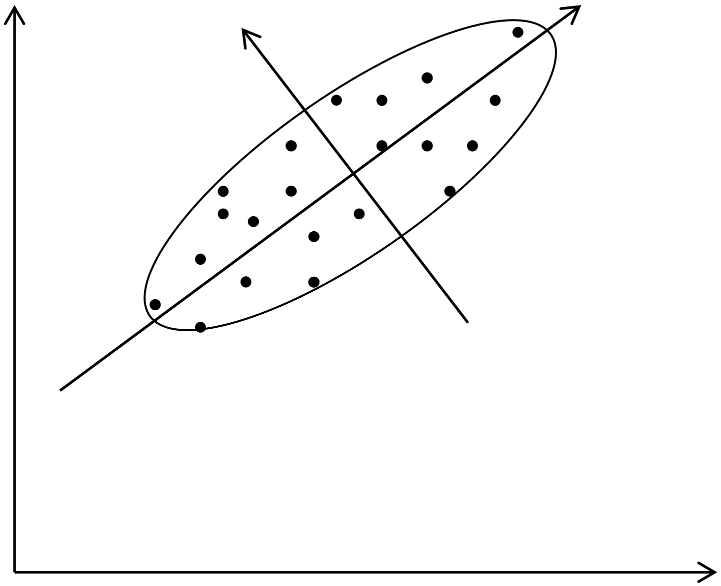
Principal Components Analysis: Intuition

- Goal is to find a projection that captures the largest amount of variation in data
- Find the eigenvectors of the covariance matrix
- The eigenvectors define the new space



Principal Component Analysis (PCA)

- Eigen Vectors show the direction of axes of a fitted ellipsoid
- Eigen Values show the significance of the corresponding axis
- The larger the Eigen value, the more separation between mapped data
- For high dimensional data, only few of Eigen values are significant



PCA: Principle Component Analysis

PCA (Principle Component Analysis) is defined as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance comes to lie on the first coordinate, the second greatest variance on the second coordinate and so on.

PCA: Principle Component

- Each Coordinate in Principle Component Analysis is called Principle Component.

$$C_i = b_{i1}(x_1) + b_{i2}(x_2) + \ldots + b_{in}(x_n)$$

where, C_i is the i^{th} principle component, b_{ij} is the regression coefficient for observed variable j for the principle component i and x_i are the variables/dimensions.

PCA: Overview

- Variance and Covariance
- Eigenvector and Eigenvalue
- Principle Component Analysis
- Application of PCA in Image Processing

PCA: Variance and Covariance(1/2)

- The **variance** is a measure of how far a set of numbers is spread out.
- The equation of variance is

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

PCA: Variance and Covariance(2/2)

- Covariance is a measure of how much two random variables change together.
- The equation of variance is

$$\mathrm{cov}(x, y) = \frac{\sum_{i=1}^{n} (x_i - \overline{x})(y_i - \overline{y})}{n-1}$$

PCA: Covariance Matrix

- Covariance Matrix is a n*n matrix where each element can be define as

$$M_{ij} = \operatorname{cov}(i, j)$$

- A covariance matrix over 2 dimensional dataset is

$$M = \begin{bmatrix} \operatorname{cov}(x, x) & \operatorname{cov}(x, y) \\ \operatorname{cov}(y, x) & \operatorname{cov}(y, y) \end{bmatrix}$$

PCA: Eigenvector

- The **eigenvectors** of a square matrix *A* are the non-zero vectors *x* such that, after being multiplied by the matrix, remain parallel to the original vector.

$$\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ -3 \end{bmatrix} = \begin{bmatrix} 3 \\ -3 \end{bmatrix}$$

PCA: Eigenvalue

- For each Eigenvector, the corresponding **Eigenvalue** is the factor by which the eigenvector is scaled when multiplied by the matrix.

$$\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ -3 \end{bmatrix} = 1 \bullet \begin{bmatrix} 3 \\ -3 \end{bmatrix}$$

PCA: Eigenvector and Eigenvalue (1/2)

- The vector *x* is an eigenvector of the matrix *A* with eigenvalue λ (lambda) if the following equation holds:

$$Ax = \lambda x$$

$$or, Ax - \lambda x = 0$$

$$or, (A - \lambda I)x = 0$$

PCA: Eigenvector and Eigenvalue (2/2)

- Calculating Eigenvalues

$$\left| A - \lambda I \right| = 0$$

- Calculating Eigenvector

$$(A - \lambda I)x = 0$$

PCA: Eigenvector and Principle Component

- It turns out that the Eigenvectors of covariance matrix of the data set are the principle components of the data set.
- Eigenvector with the highest eigenvalue is first principle component and with the 2nd highest eigenvalue is the second principle component and so on.

PCA: Steps to find Principle Components

1. Adjust the dataset to zero mean dataset.
2. Find the Covariance Matrix M
3. Calculate the normalized Eigenvectors and Eigenvalues of M
4. Sort the Eigenvectors according to Eigenvalues from highest to lowest
5. Form the Feature vector F using the transpose of Eigenvectors.
6. Multiply the transposed dataset with F

PCA: Example

AdjustedDataSet = OriginalDataSet - Mean

X	Y
2.5	2.4
0.5	0.7
2.2	2.9
1.9	2.2
3.1	3.0
2.3	2.7
2	1.6
1	1.1
1.5	1.6
1.1	0.9

Original Data

X	Y
0.69	0.49
-1.31	-1.21
0.39	0.99
0.09	0.29
1.29	1.09
0.49	0.79
0.19	-0.31
-0.81	-0.81
-0.31	-0.31
-0.71	-1.01

Adjusted Dataset

PCA: Covariance Matrix

$$M = \begin{bmatrix} 0.616555556 & 0.615444444 \\ 0.615444444 & 0.716555556 \end{bmatrix}$$

PCA: Eigenvalues and Eigenvectors

- The eigenvalues of matrix *M* are

$$eigenvalues = \begin{pmatrix} 0.0490833989 \\ 1.28402771 \end{pmatrix}$$

- Normalized Eigenvectors with corresponding eigenvales are

$$eigenvectors = \begin{pmatrix} -0.735178656 & -0.677873399 \\ 0.677873399 & -0.735178656 \end{pmatrix}$$

PCA: Feature Vector

- Sorted eigenvector

$$eigenvectors = \begin{pmatrix} -0.677873399 & -0.735178656 \\ -0.735178656 & 0.677873399 \end{pmatrix}$$

- Feature vector

$$F = \begin{pmatrix} -0.677873399 & -0.735178656 \\ -0.735178656 & 0.677873399 \end{pmatrix}^T$$

$$or, F = \begin{pmatrix} -0.677873399 & -0.735178656 \\ -0.735178656 & 0.677873399 \end{pmatrix}$$

PCA: Final Data (1/2)

FinalData = F x AdjustedDataSetTransposed

X	Y
-0.827970186	-0.175115307
1.77758033	0.142857227
-0.992197494	0.384374989
-0.274210416	0.130417207
-1.67580142	-0.209498461
-0.912949103	0.175282444
-0.0991109437	-0.349824698
1.14457216	0.0464172582
0.438046137	0.0177646297
1.22382056	-0.162675287

PCA: Final Data (2/2)

FinalData = F x AdjustedDataSetTransposed

x
-0.827970186
1.77758033
-0.992197494
-0.274210416
-1.67580142
-0.912949103
0.0991094375
1.14457216
0.438046137
1.22382056

PCA: Retrieving Original Data

FinalData = F x AdjustedDataSetTransposed

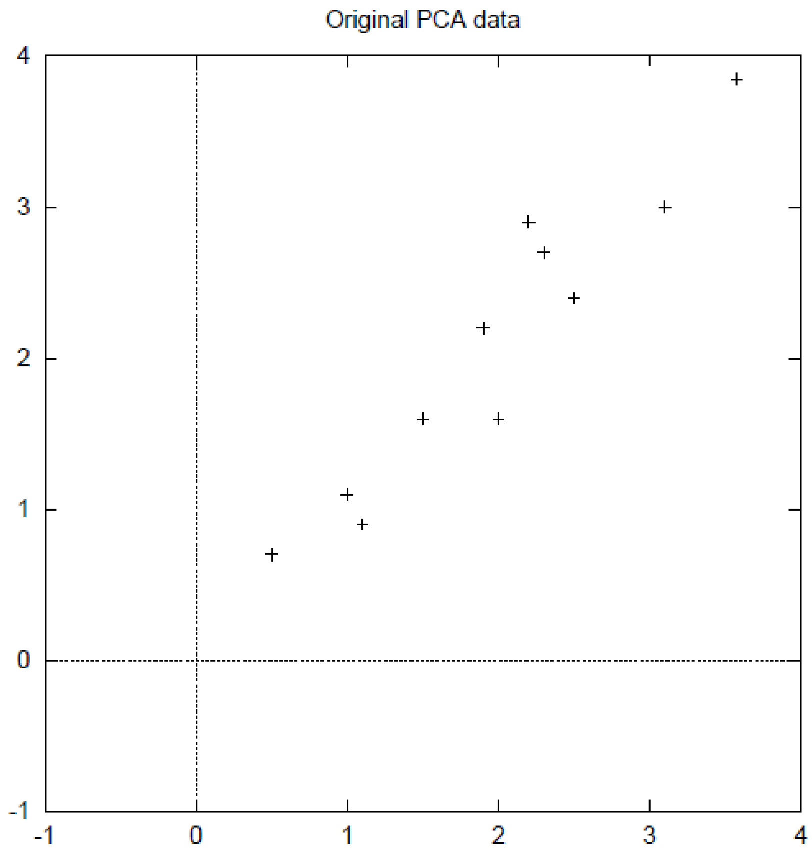
AdjustedDataSetTransposed = F^{-1} *x FinalData*

but, $F^{-1} = F^{T}$

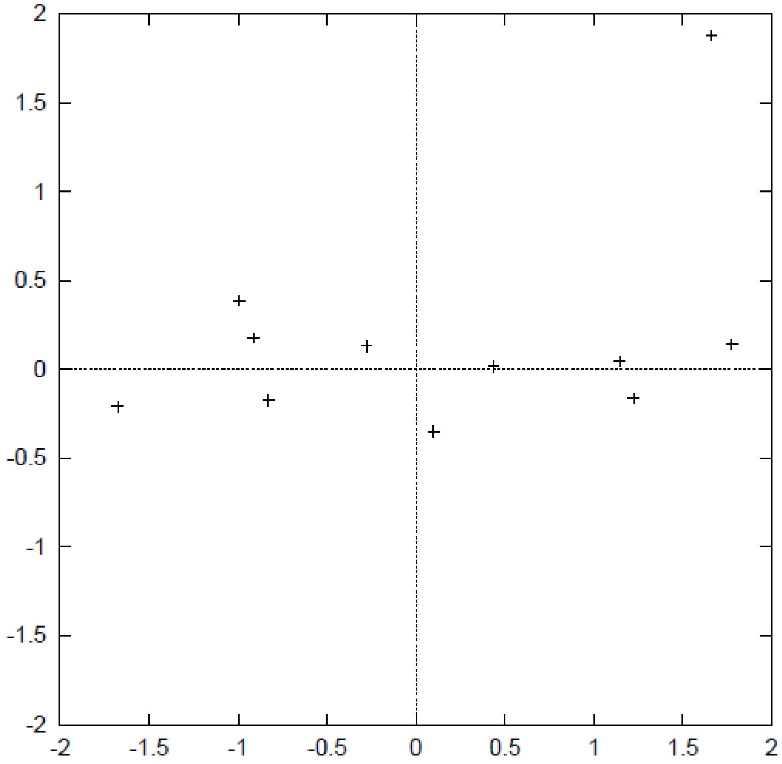
So, AdjustedDataSetTransposed $= F^{T}$ *x FinalData*

and, OriginalDataSet = AdjustedDataSet + Mean

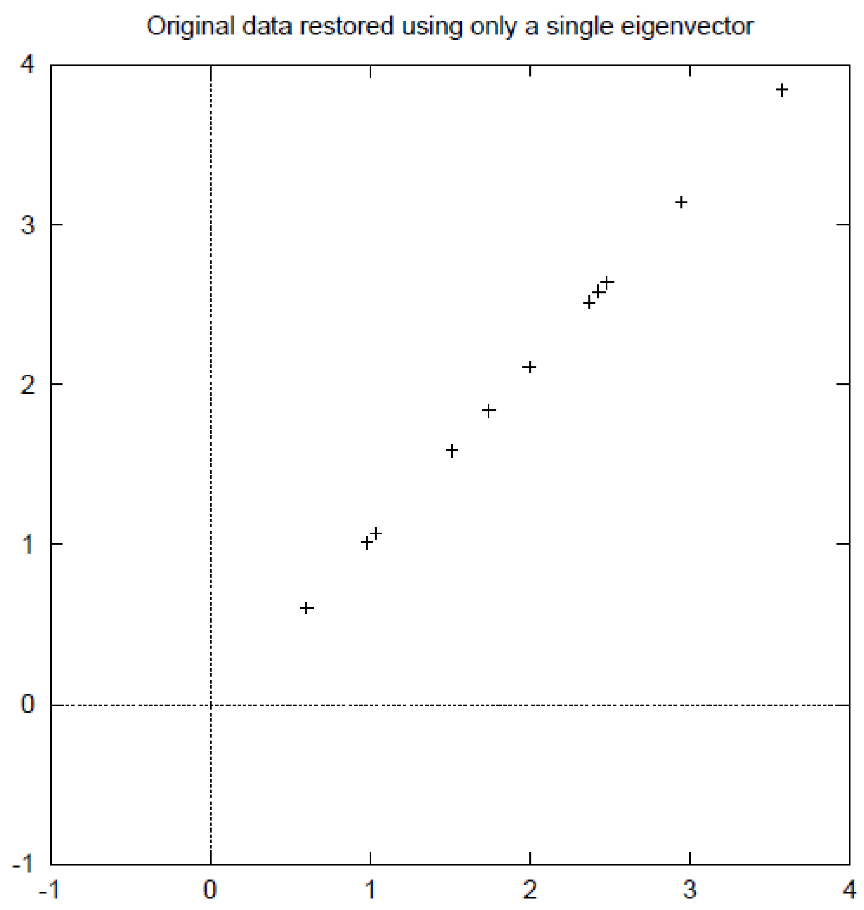
PCA: Principle Component Analysis



PCA: Principle Component Analysis



PCA: Retrieving Original Data(2/2)



PCA Demo

- http://www.cs.mcgill.ca/~sqrt/dimr/dimreduction.html

Applying the PCs to transform data

- Using all PCs

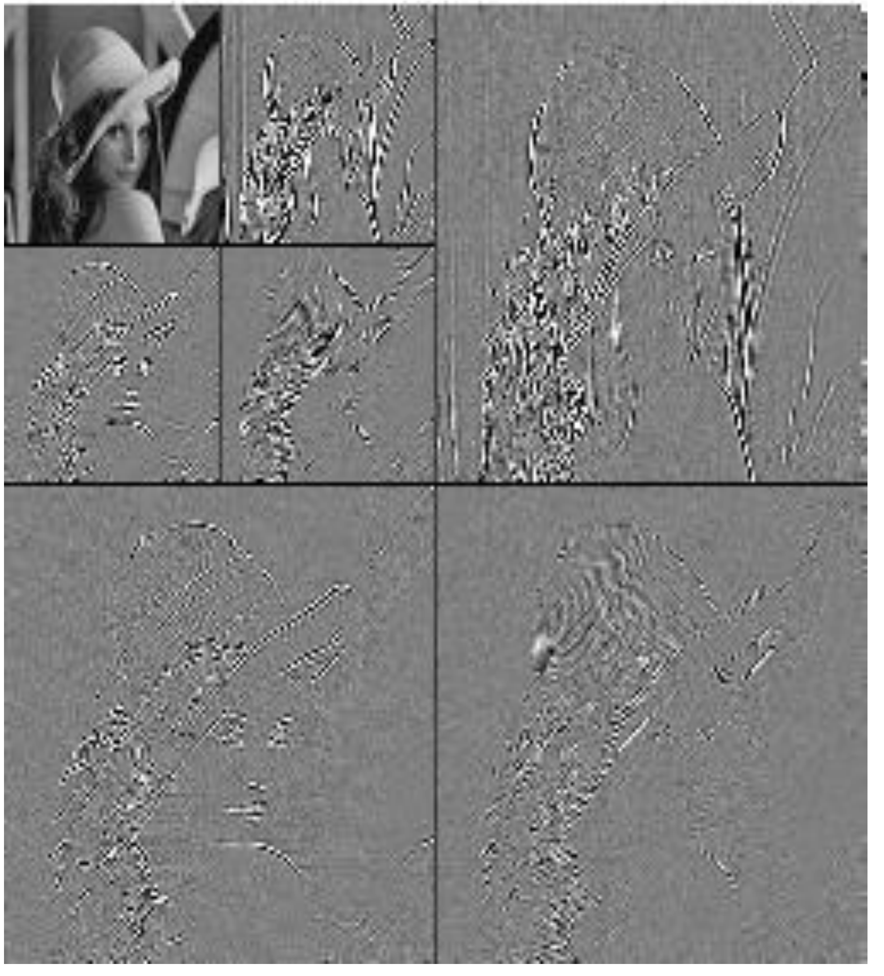
$$\begin{pmatrix} z_{11} & z_{12} & . & . & z_{1n} \\ z_{21} & z_{22} & . & . & z_{2n} \\ . & & & & \\ . & & & & \\ z_{n1} & z_{n2} & , & , & z_{nn} \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ . \\ . \\ x_n \end{pmatrix} = \begin{pmatrix} x'_1 \\ x'_2 \\ . \\ . \\ x'_n \end{pmatrix}$$

- Using only 2 PCs

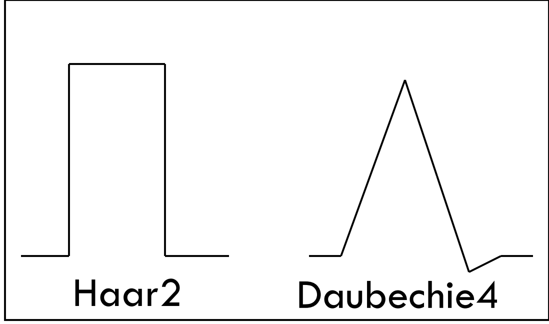
$$\begin{pmatrix} z_{11} & z_{12} & . & . & z_{1n} \\ z_{21} & z_{22} & . & . & z_{2n} \end{pmatrix} \times \begin{pmatrix} x_1 \\ x_2 \\ . \\ . \\ x_n \end{pmatrix} = \begin{pmatrix} x'_1 \\ x'_2 \end{pmatrix}$$

What Is Wavelet Transform?

- Decomposes a signal into different frequency subbands
  - Applicable to n-dimensional signals
- Data are transformed to preserve relative distance between objects at different levels of resolution
- Allow natural clusters to become more distinguishable
- Used for image compression



Wavelet Transformation



- Discrete wavelet transform (DWT) for linear signal processing, multi-resolution analysis
- Compressed approximation: store only a small fraction of the strongest of the wavelet coefficients
- Similar to discrete Fourier transform (DFT), but better lossy compression, localized in space
- Method:
  - Length, L, must be an integer power of 2 (padding with 0's, when necessary)
  - Each transform has 2 functions: smoothing, difference
  - Applies to pairs of data, resulting in two set of data of length L/2
  - Applies two functions recursively, until reaches the desired length

Wavelet Decomposition

- Wavelets: A math tool for space-efficient hierarchical decomposition of functions
- $S = [2, 2, 0, 2, 3, 5, 4, 4]$ can be transformed to $S_{\wedge} = [2^3/_4, -1^1/_4, {}^1/_2, 0, 0, -1, -1, 0]$
- Compression: many small detail coefficients can be replaced by 0's, and only the significant coefficients are retained

Resolution	Averages	Detail Coefficients
8	$[2, 2, 0, 2, 3, 5, 4, 4]$	
4	$[2, 1, 4, 4]$	$[0, -1, -1, 0]$
2	$[1\frac{1}{2}, 4]$	$[\frac{1}{2}, 0]$
1	$[2\frac{3}{4}]$	$[-1\frac{1}{4}]$

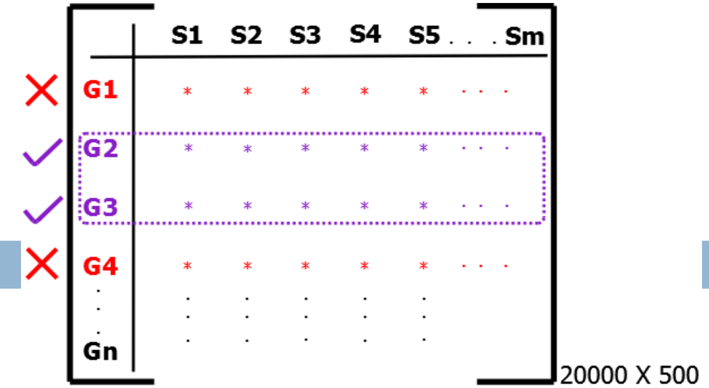
Feature Subset Selection

- Another way to reduce dimensionality of data

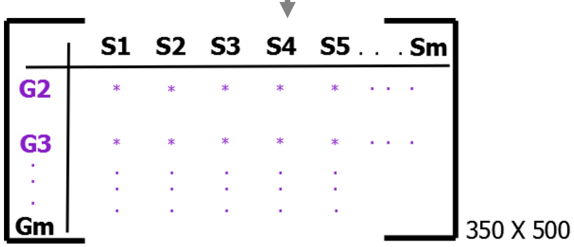
- Redundant features
  - duplicate much or all of the information contained in one or more other attributes
  - Example: purchase price of a product and the amount of sales tax paid

- Irrelevant features
  - contain no information that is useful for the data mining task at hand
  - Example: students' ID is often irrelevant to the task of predicting students' GPA

1-2 Opening…. For … M.Tech. Dissertation in the Area of Feature Subset Selection



Feature Subset Selection from High Dimensional Biological Data



Abhinna Agarwal
M.Tech.(CSE)

Guided by
Dr. Dhaval Patel

Outline…..

So far, our Trajectory on **Data Preprocessing** is as follow:

1. Data has **attributes** and their **values**
   - Noise, Quality, Inconsistent, Incomplete, …

2. Data has **many records**
   - Data Sampling

3. Data has **many attributes/dimensions**
   - Feature Selections or Dimensionality Reduction

4. Can you guess **What is next**?

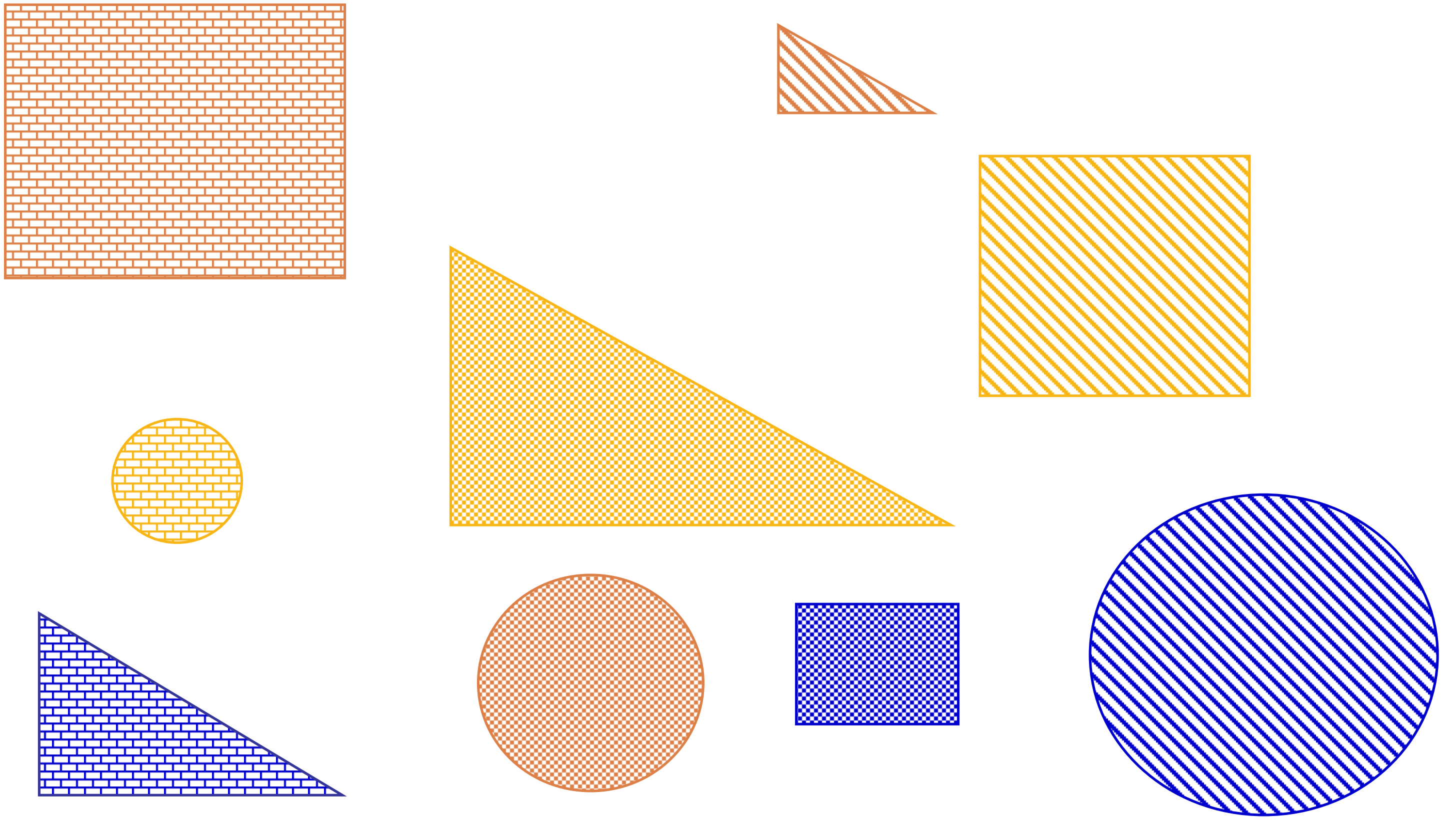
Distance/Similarity

Data has many records

Then,

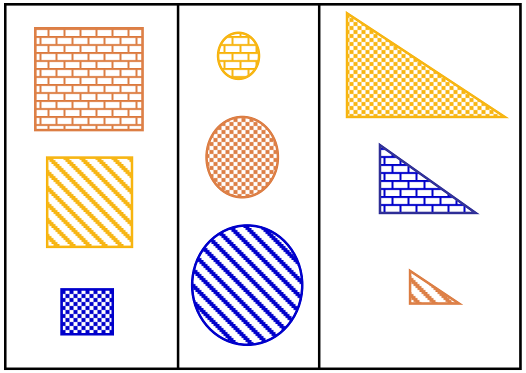
Can we find **similar records**?

Distance and Similarity are commonly used….

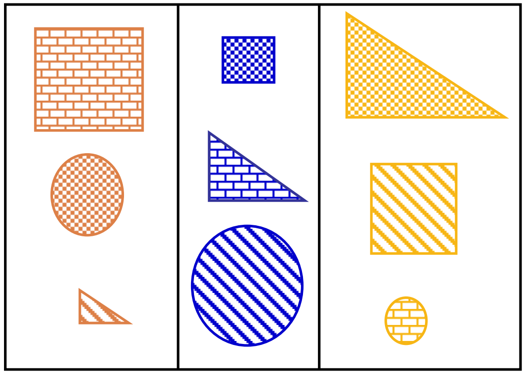


What is similar?

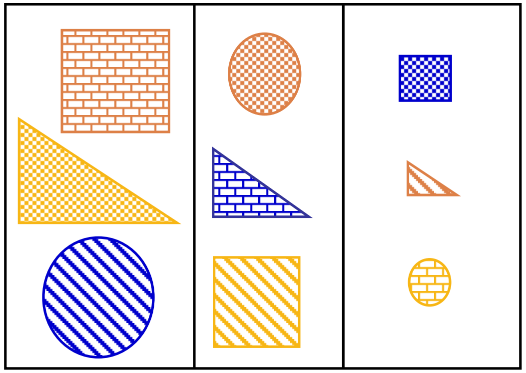
Shape



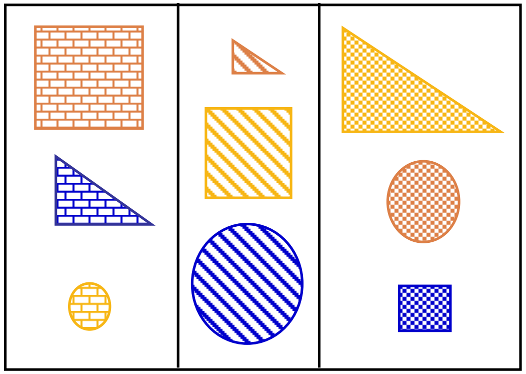
Colour



Size



Pattern



Similarity and Dissimilarity

- Similarity
  - Numerical measure of how alike two data objects are.
  - Is higher when objects are more alike.
  - Often falls in the range [0,1]
- Dissimilarity
  - Numerical measure of how different are two data objects
  - Lower when objects are more alike
  - Minimum dissimilarity is often 0
  - Upper limit varies
- Proximity refers to a similarity or dissimilarity

Euclidean Distance

- Euclidean Distance

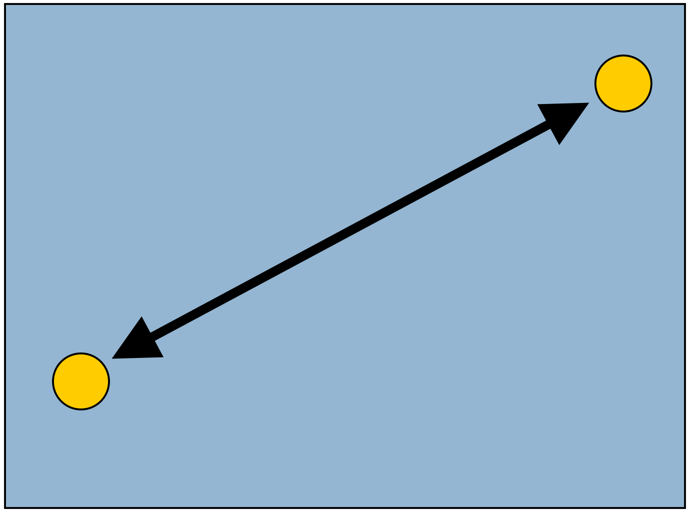
$$dist = \sqrt{\sum_{k=1}^{n} (p_k - q_k)^2}$$

Where n is the number of dimensions (attributes) and p_k and q_k are, respectively, the k^{th} attributes (components) or data objects p and q.

- Standardization is necessary, if scales differ.

Euclidean Distance (Metric)

Euclidean distance:



Point 1 is: $(x_1, x_2, \ldots, x_n)$

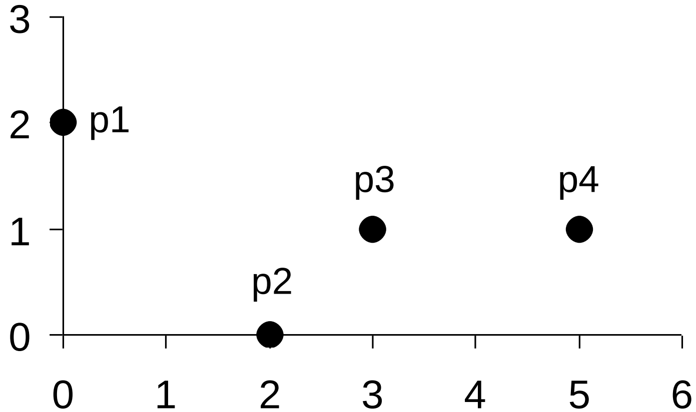
Point 2 is: $(y_1, y_2, \ldots, y_n)$

Euclidean distance is:

$$\sqrt{(y_1 - x_1)^2 + (y_2 - x_2)^2 + \ldots + (y_n - x_n)^2}$$

David Corne, and Nick Taylor, Heriot-Watt University - dwcorne@gmail.com

Euclidean Distance



point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix

Minkowski Distance

- Minkowski Distance is a generalization of Euclidean Distance

$$dist = \left(\sum_{k=1}^{n} | p_k - q_k |^r \right)^{\frac{1}{r}}$$

Where *r* is a parameter, *n* is the number of dimensions (attributes) and p_k and q_k are, respectively, the kth attributes (components) or data objects *p* and *q*.

Minkowski Distance: Examples

- $r = 1$. City block (Manhattan, taxicab, L_1 norm) distance.
  - A common example of this is the Hamming distance, which is just the number of bits that are different between two binary vectors

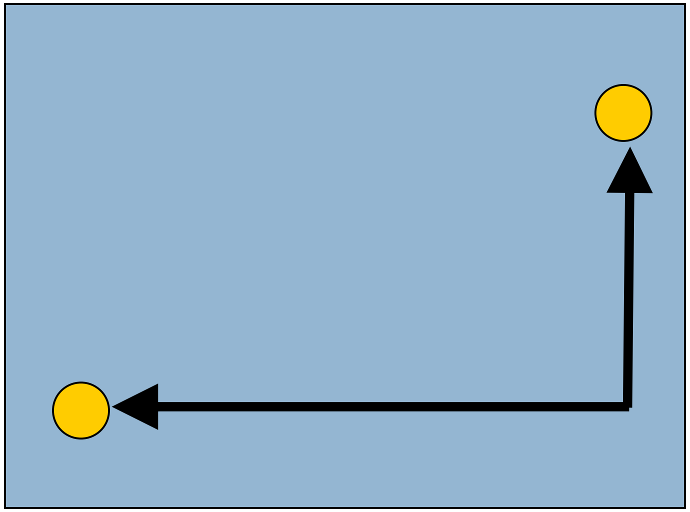
- $r = 2$. Euclidean distance

- $r \rightarrow \infty$. "supremum" (L_{max} norm, L_∞ norm) distance.
  - This is the maximum difference between any component of the vectors
  - Example: L_infinity of (1, 0, 2) and (6, 0, 3) = ??

  - Do not confuse *r* with *n*, i.e., all these distances are defined for all numbers of dimensions.

Manhattan Distance

Manhattan distance
(aka city-block distance)



Point 1 is: $(x_1, x_2, \ldots, x_n)$

Point 2 is: $(y_1, y_2, \ldots, y_n)$

Manhattan distance is:

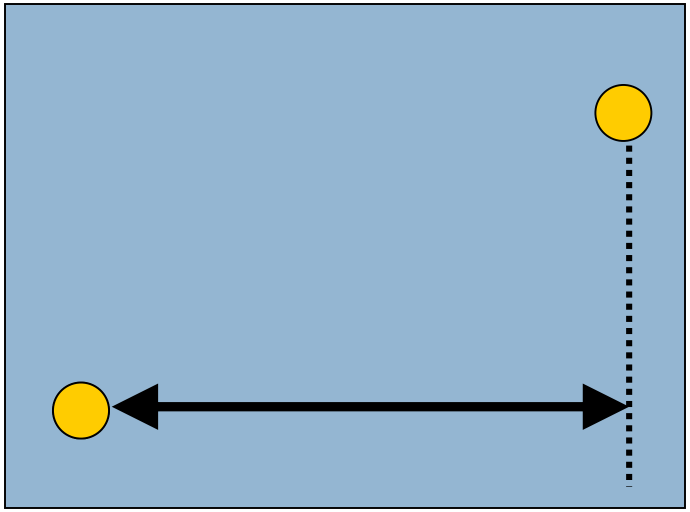
$$| y_1 - x_1 | + | y_2 - x_2 | + \ldots + | y_n - x_n |$$

(in case you don't know: $| x |$ is the absolute value of x.)

David Corne, and Nick Taylor, Heriot-Watt University - dwcorne@gmail.com

Chebychev Distance

Chebychev distance



Point 1 is: $(x_1, x_2, \ldots, x_n)$

Point 2 is: $(y_1, y_2, \ldots, y_n)$

Chebychev distance is:

$$\max\{|\, y_1 - x_1 \,|, |\, y_2 - x_2 \,|, \ldots, |\, y_n - x_n \,|\}$$

L1-L2-... Distances

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

L_∞	p1	p2	p3	p4
p1	0	2	3	5
p2	2	0	1	3
p3	3	1	0	2
p4	5	3	2	0

Distance Matrix

Additive Distances

- Each variable contributes independently to the measure of distance.
- May not always be appropriate… e.g., think of nearest neighbor classifier

object i

object j

object i	object j
diameter(i)	diameter(j)
height(i)	height(j)
$\text{height}_2(i)$	$\text{height}_2(j)$
⋮	⋮
$\text{height}_{100}(i)$	$\text{height}_{100}(j)$

Dependence among Variables

- Covariance and correlation measure linear dependence (distance between variables, not objects)

- Assume we have two variables or attributes X and Y and n objects taking on values x(1), …, x(n) and y(1), …, y(n). The sample covariance of X and Y is:

$$Cov(X,Y) = \frac{1}{n}\sum_{i=1}^{n}(x(i)-\bar{x})(y(i)-\bar{y})$$

- The covariance is a measure of how X and Y vary together.
  - it will be large and positive if large values of X are associated with large values of Y, and small X $\Rightarrow$ small Y

Correlation coefficient

- Covariance depends on ranges of X and Y
- Standardize by dividing by standard deviation
- Linear correlation coefficient is defined as:

$$\rho(X,Y) = \frac{\sum_{i=1}^{n}(x(i)-\bar{x})(y(i)-\bar{y})}{\left(\sum_{i=1}^{n}(x(i)-\bar{x})^2 \sum_{i=1}^{n}\ (y(i)-\bar{y})^2\right)^{\frac{1}{2}}}$$

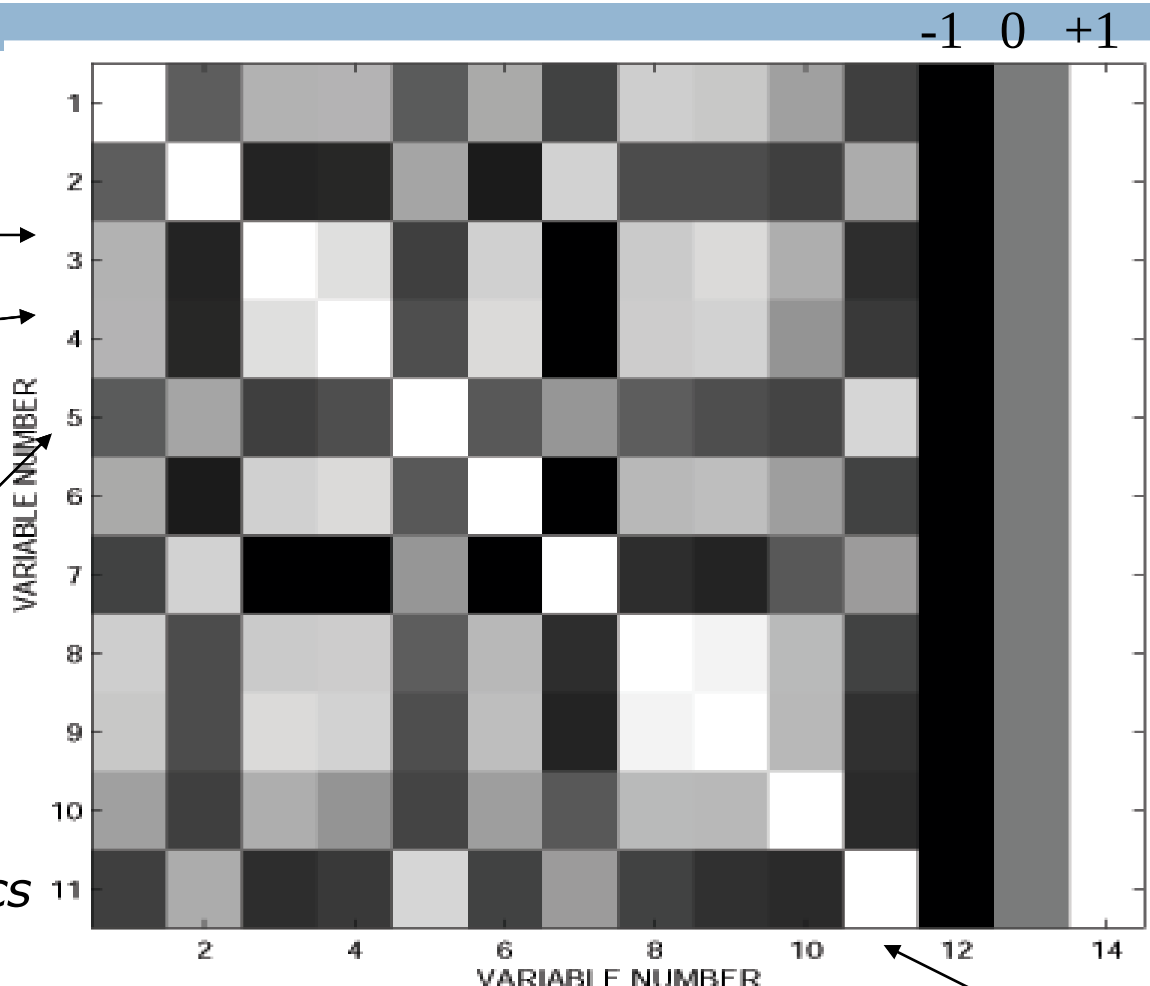
Sample Correlation Matrix

business acreage

nitrous oxide

average # rooms

Data on characteristics of Boston surburbs



Median house value

percentage of large residential lots

Mahalanobis distance (between objects)

$$d_{MH}(x, y) = \left((x-y)^T \Sigma^{-1} (x-y) \right)^{\frac{1}{2}}$$

Evaluates to a scalar distance

Vector difference in p-dimensional space

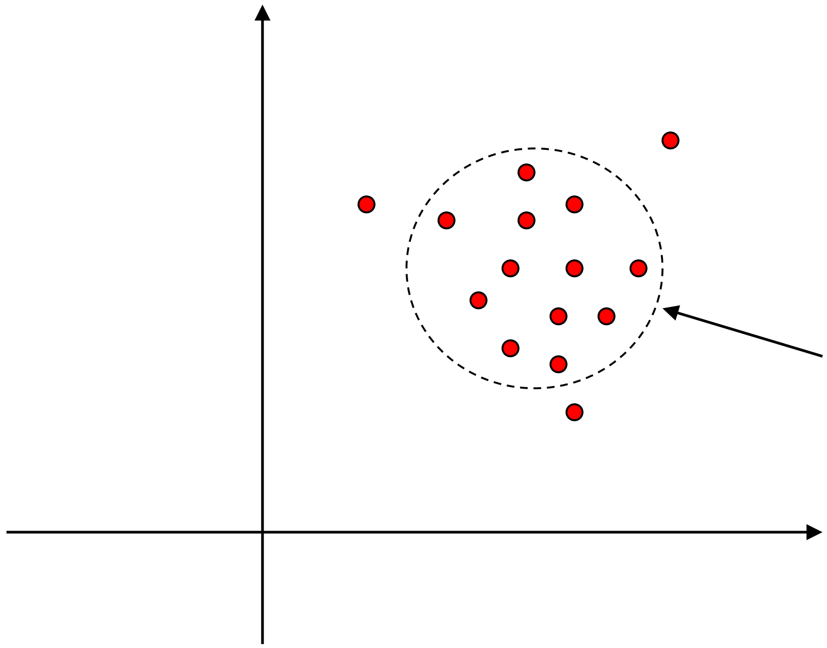
Inverse covariance matrix

1. It automatically accounts for the scaling of the coordinate axes
2. It corrects for correlation between the different features

Cost:

1. The covariance matrices can be hard to determine accurately
2. The memory and time requirements grow quadratically, $O(p^2)$, rather than linearly with the number of features.

Example 1 of Mahalonobis distance

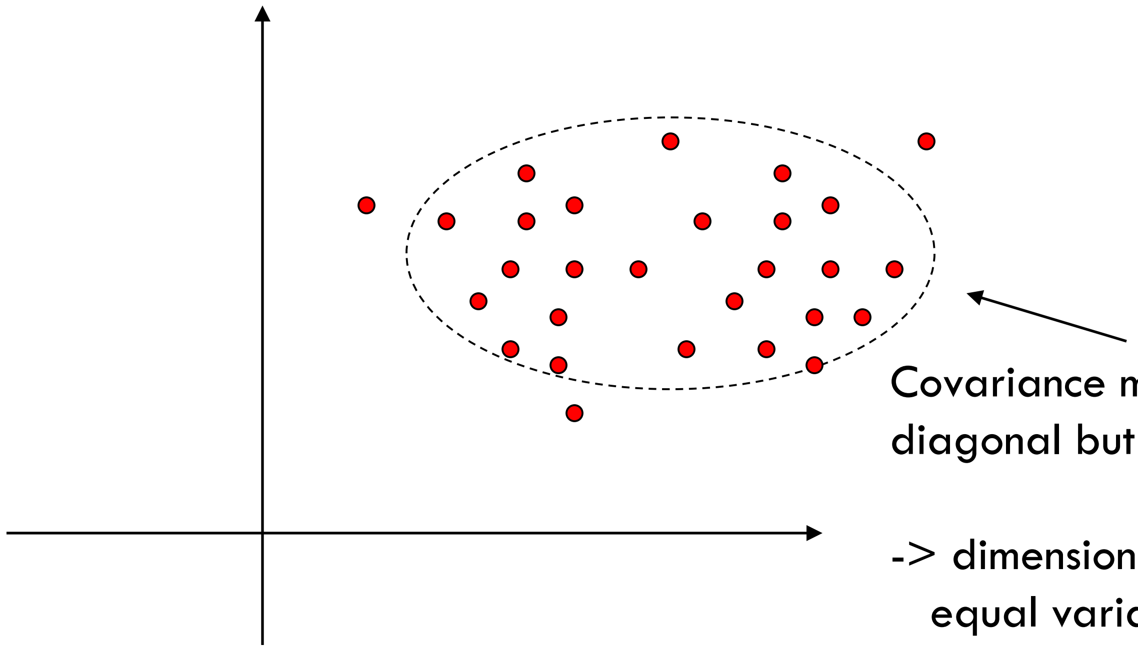


Covariance matrix is diagonal and isotropic

-> all dimensions have equal variance

-> MH distance reduces to Euclidean distance

Example 2 of Mahalonobis distance

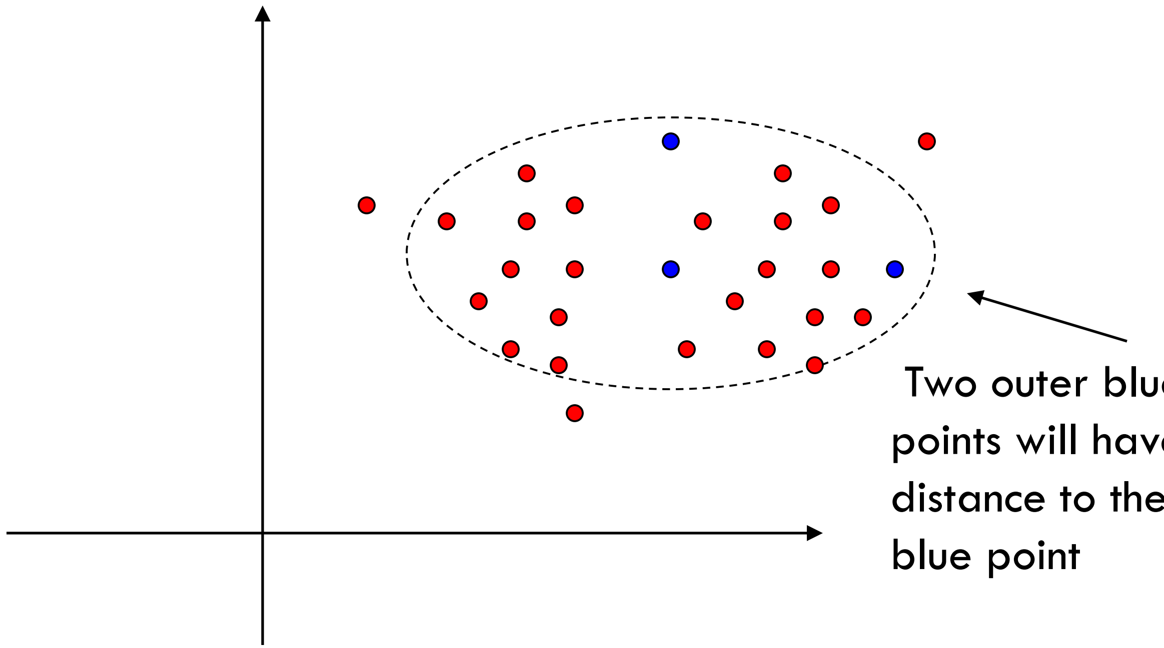


Covariance matrix is diagonal but non-isotropic

-> dimensions do not have equal variance

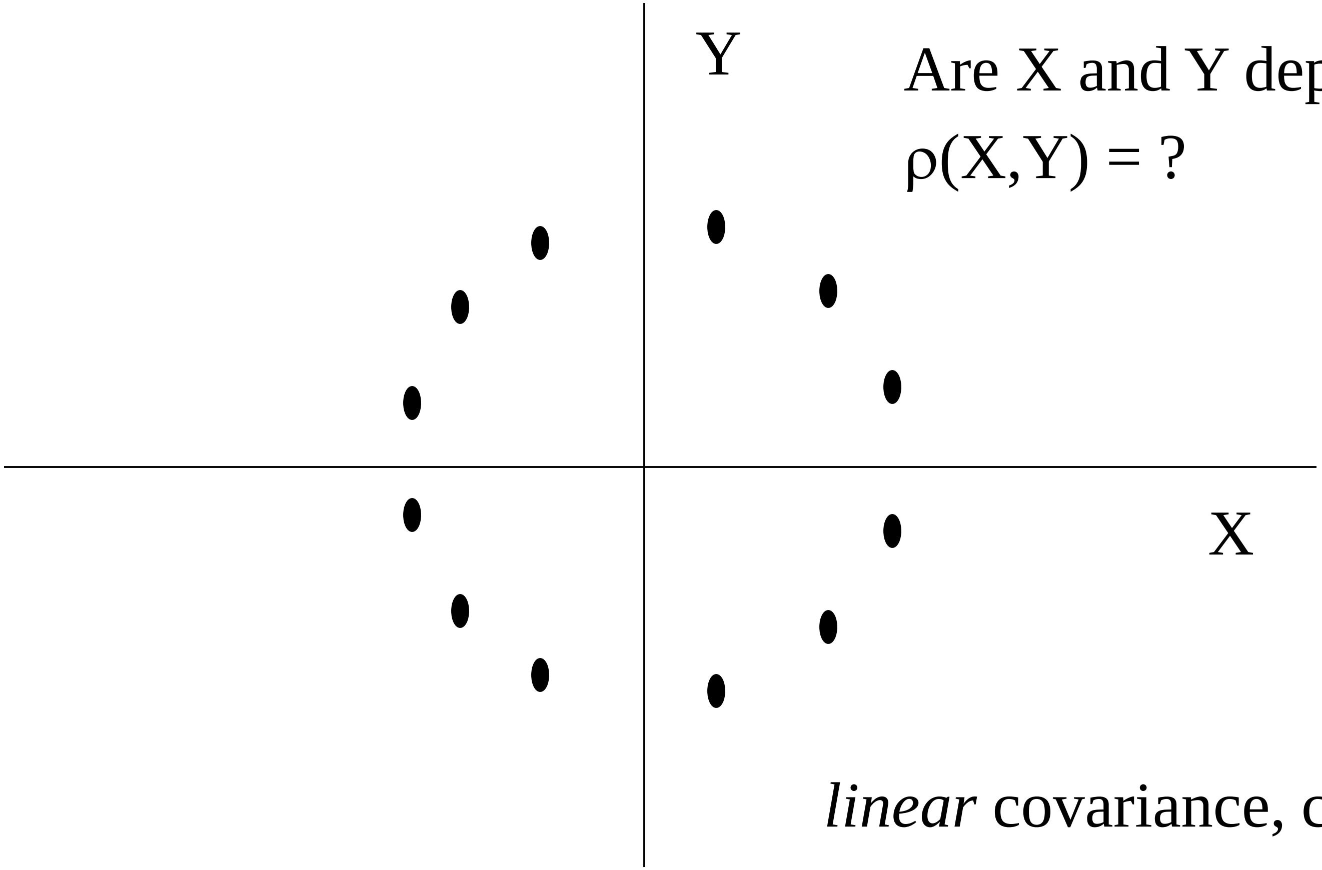
-> MH distance reduces to weighted Euclidean distance with weights = inverse variance

Example 2 of Mahalonobis distance



Two outer blue
points will have same MH
distance to the center
blue point

What about…



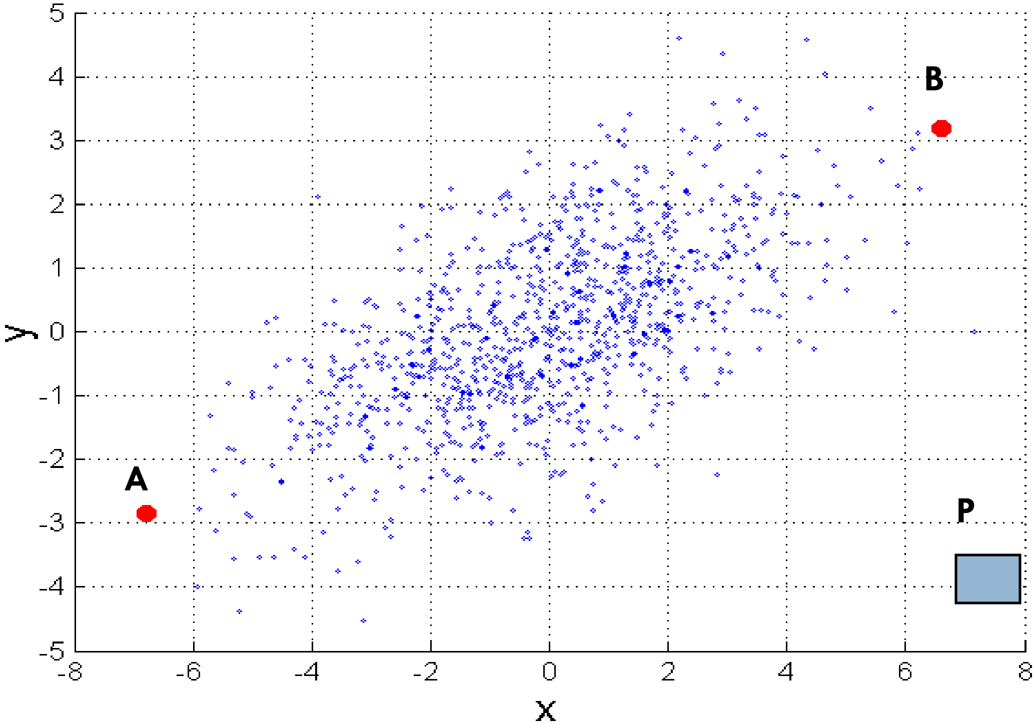
Are X and Y dependent?

$\rho(X,Y) = ?$

linear covariance, correlation

Mahalanobis Distance

$$mahalanobis(p,q) = (p-q)\Sigma^{-1}(p-q)^T$$

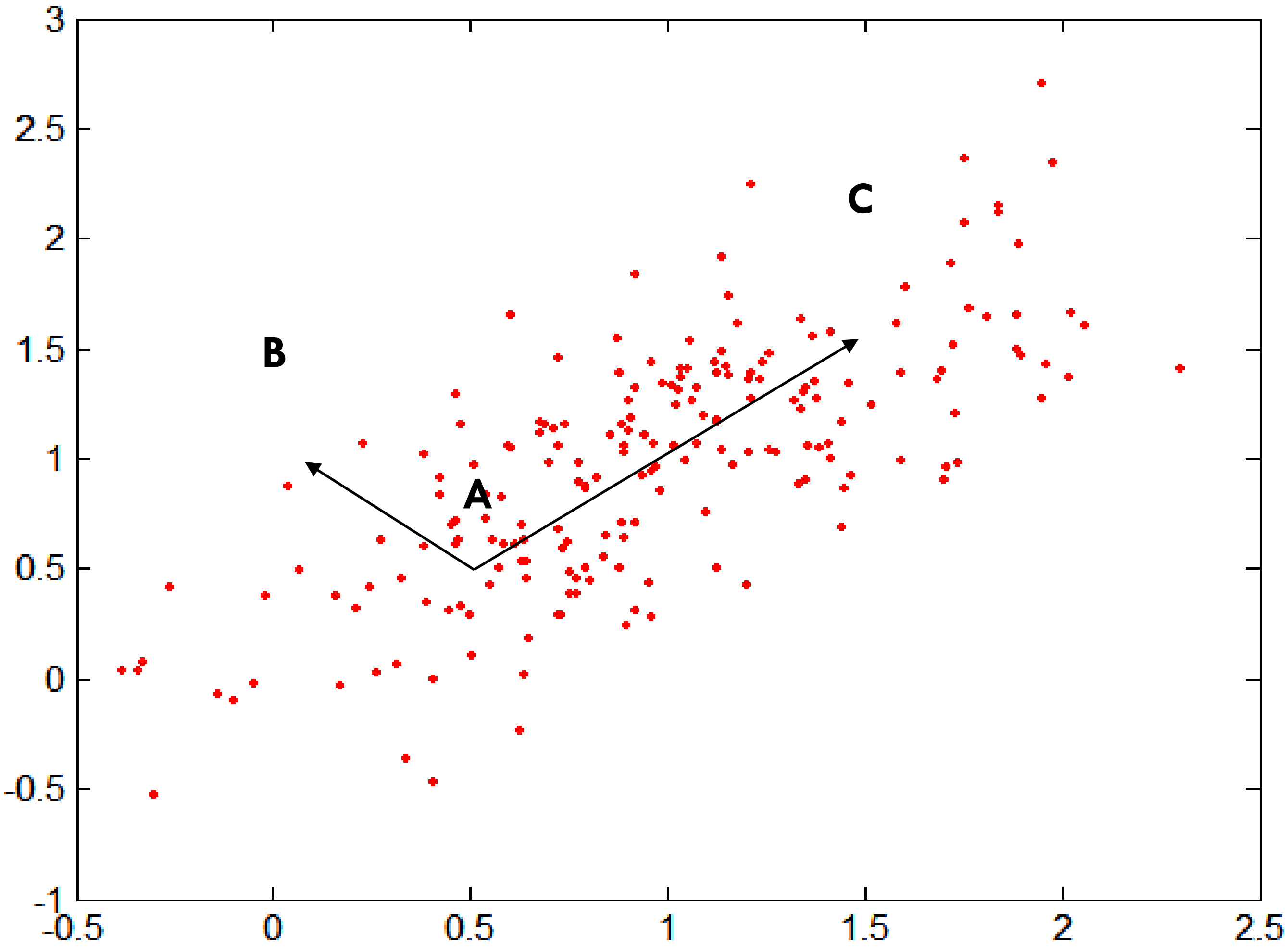


Σ is the covariance matrix of the input data *X*

$$\Sigma_{j,k} = \frac{1}{n-1}\sum_{i=1}^{n}(X_{ij}-\overline{X}_j)(X_{ik}-\overline{X}_k)$$

For red points, the Euclidean distance is 14.7, Mahalanobis distance is 6.

Mahalanobis Distance



Covariance Matrix:

$$\Sigma = \begin{bmatrix} 0.3 & 0.2 \\ 0.2 & 0.3 \end{bmatrix}$$

A: (0.5, 0.5)

B: (0, 1)

C: (1.5, 1.5)

Mahal(A,B) = 5

Mahal(A,C) = 4

Distances between Categorical Vectors

Proportion different

Point 1 is: $(x_1, x_2, \ldots, x_n)$

Point 2 is: $(y_1, y_2, \ldots, y_n)$

Proportion different is:

(red, male, big, hot)

(green, male, small, hot)

$$d = 0$$

$$\text{for each field f}$$

$$\text{if } (y_f \neq x_f) \text{ then } d = d + 1$$

$$\text{proportion different is } d / n$$

Distances between Categorical Vectors

Jaccard coefficient

(bread, cheese, milk, nappies)

(batteries, cheese)

Point 1 is a set: *A*

Point 2 is a set: *B*

Jaccard Coefficient is:

$$\frac{|A \cap B|}{|A \cup B|}$$

The number of things that appear in **both** (1 - cheese), divided by the total number of different things (5))

Using common sense

Data vectors are: (colour, manufacturer, top-speed)

e.g.: (red, ford, 180)
(yellow, toyota, 160)
(silver, bugatti, 300)

What distance measure will you use?

Using common sense

Data vectors are : (colour, manufacturer, top-speed)

e.g.: (dark, ford, high)
(medium, toyota, high)
(light, bugatti, very-high)

What distance measure will you use?

Using common sense

With different types of fields, e.g.

p1 = (red, high, 0.5, UK, 12)
p2 = (blue, high, 0.6, France, 15)

You could simply define a distance measure for each field
Individually, and add them up.

Similarly, you could divide the vectors into ordinal and numeric parts:

p1a = (red, high, UK) p1b = (0.5, 12)
p2a = (blue, high, France) p2b = (0.6, 15)

and say that dist(p1, p2) = dist(p1a,p2a)+d(p1b,p2b)
using appropriate measures for the two kinds of vector.

Using common sense…

Suppose one field varies hugely (standard deviation is 100), and one field varies a tiny amount (standard deviation 0.001) – why is Euclidean distance a bad idea? What can you do?

What is the distance between these two?

“Star Trek: Voyager”

“Satr Trek: Voyagger”

Normalising fields individually is often a good idea – when a numerical field is normalised, that means you scale it so that the mean is 0 and the standard deviation is 1.

Edit distance is useful in many applications: see

http://www.merriampark.com/ld.htm

Cosine Similarity

- If d_1 and d_2 are two document vectors, then

$$\cos(d_1, d_2) = (d_1 \bullet d_2) / ||d_1|| \; ||d_2|| ,$$

where • indicates vector dot product and || d || is the length of vector d.

- Example:

d_1 = **3 2 0 5 0 0 0 2 0 0**
d_2 = **1 0 0 0 0 0 0 1 0 2**

$d_1 \bullet d_2$= 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5

$||d_1||$ = (3*3+2*2+0*0+5*5+0*0+0*0+0*0+2*2+0*0+0*0)$^{0.5}$ = (42) $^{0.5}$ = 6.481

$||d_2||$ = (1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2) $^{0.5}$ = (6) $^{0.5}$ = 2.245

$\cos(d_1, d_2)$ = .3150, distance=1-cos(d1,d2)

Nominal Variables

- A generalization of the binary variable in that it can take more than 2 states, e.g., red, yellow, blue, green
- Method 1: Simple matching
  - *m*: # of matches, *p*: total # of variables

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

- Method 2: use a large number of binary variables
  - creating a new binary variable for each of the *M* nominal states

Ordinal Variables

- An ordinal variable can be discrete or continuous
- order is important, e.g., rank
- Can be treated like interval-scaled
  - replacing x_{if} by their rank $r_{if} \in \{1, \ldots, M_f\}$
  - map the range of each variable onto [0, 1] by replacing *i*-th object in the *f*-th variable by

$$z_{if} = \frac{r_{if} - 1}{M_f - 1}$$

  - compute the dissimilarity using methods for interval-scaled variables

Common Properties of a Distance

- Distances, such as the Euclidean distance, have some well known properties.

  1. $d(p, q) \geq 0$ for all p and q and $d(p, q) = 0$ only if $p = q$. (Positive definiteness)
  2. $d(p, q) = d(q, p)$ for all p and q. (Symmetry)
  3. $d(p, r) \leq d(p, q) + d(q, r)$ for all points p, q, and r. (Triangle Inequality)

  where $d(p, q)$ is the distance (dissimilarity) between points (data objects), p and q.

- A distance that satisfies these properties is a metric

Common Properties of a Similarity

- Similarities, also have some well known properties.

1. $s(p, q) = 1$ (or maximum similarity) only if $p = q$.
2. $s(p, q) = s(q, p)$ for all p and q. (Symmetry)

where $s(p, q)$ is the similarity between points (data objects), p and q.