CSE 332
Introduction to Visualization

Data Reduction & Similarity metrics

Klaus Mueller

Computer Science Department
Stony Brook University

Lecture	Topic	Projects
1	Intro, schedule, and logistics	
2	Intro continued	
3	Applications of visual analytics, data, and basic tasks	
4	Data preparation and reduction	
5	Data reduction and similarity metrics	Project 1 out
6	Dimension reduction	
7	Introduction to D3	Project 2 out
8	Bias in visualization	
9	Perception and cognition	
10	Visual design and aesthetics	
11	Cluster and pattern analysis	
12	High-Dimensional data visualization: linear methods	
13	High-D data vis.: non-linear methods, categorical data	Project 3 out
14	Principles of interaction	
15	Visual analytics and the visual sense making process	
16	VA design and evaluation	
17	Visualization of graphs and hierarchies	
18	Visualization of time-varying and time-series data	Project 4 out
19	Midterm	
20	Maps and geo-vis	
21	Computer graphics and volume rendering	
22	Techniques to visualize spatial (3D) data	Project 4 halfway report due
23	Scientific and medical visualization	
24	Scientific and medical visualization	
25	Non-photorealistic rendering	
26	Memorable visualizations, visual embellishments	Project 5 out
27	Infographics design	
28	Projects Hall of Fame demos	

Recall: The Rectangular Dataset

One data item

The variables
→ the attributes or properties we measured

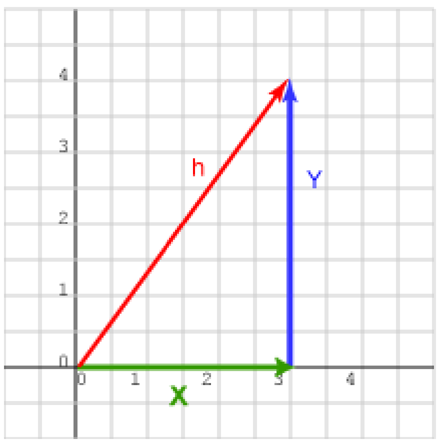
The data items
→ the samples (observations) we obtained from the population of all instances

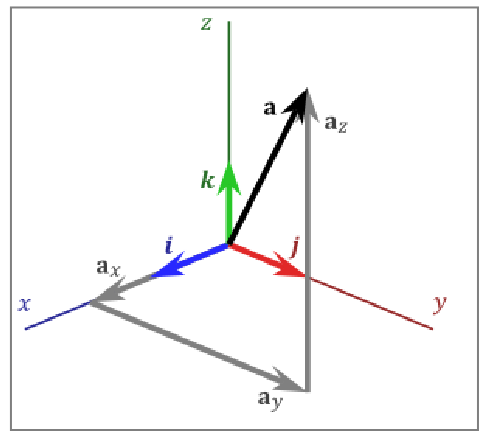
	A	B	C	D	E	F	G	H	I
1	**Name**	**Country**	**Miles Per Gallon**	**Accceleration**	**Horsepower**	**weight**	**cylinders**	**year**	**price**
2	Volkswagen Rabbit Dl	Germany	43,1	21,5	48	1985	4	78	2400
3	Ford Fiesta	Germany	36,1	14,4	66	1800	4	78	1900
4	Mazda GLC Deluxe	Japan	32,8	19,4	52	1985	4	78	2200
5	Datsun B210 GX	Japan	39,4	18,6	70	2070	4	78	2725
6	Honda Civic CVCC	Japan	36,1	16,4	60	1800	4	78	2250
7	Oldsmobile Cutlass	USA	19,9	15,5	110	3365	8	78	3300
8	Dodge Diplomat	USA	19,4	13,2	140	3735	8	78	3125
9	Mercury Monarch	USA	20,2	12,8	139	3570	8	78	2850
10	Pontiac Phoenix	USA	19,2	19,2	105	3535	6	78	2800
11	Chevrolet Malibu	USA	20,5	18,2	95	3155	6	78	3275
12	Ford Fairmont A	USA	20,2	15,8	85	2965	6	78	2375
13	Ford Fairmont M	USA	25,1	15,4	88	2720	4	78	2275
14	Plymouth Volare	USA	20,5	17,2	100	3430	6	78	2700
15	AMC Concord	USA	19,4	17,2	90	3210	6	78	2300
16	Buick Century	USA	20,6	15,8	105	3380	6	78	3300
17	Mercury Zephyr	USA	20,8	16,7	85	3070	6	78	2425
18	Dodge Aspen	USA	18,6	18,7	110	3620	6	78	2700
19	AMC Concord D1	USA	18,1	15,1	120	3410	6	78	2425
20	Chevrolet MonteCarlo	USA	19,2	13,2	145	3425	8	78	3900
21	Buick RegalTurbo	USA	17,7	13,4	165	3445	6	78	4400
22	Ford Futura	Germany	18,1	11,2	139	3205	8	78	2525
23	Dodge Magnum XE	USA	17,5	13,7	140	4080	8	78	3000
24	Chevrolet Chevette	USA	30	16,5	68	2155	4	78	2100

Representation

Each data item is an N-dimensional vector (N variables)

- recall 2D and 3D vectors in 2D and 3D space, respectively





Now we have N-D attribute space

- now the data axes extend into more than 3 orthogonal directions
- hard to imagine?
- that's why need good visualization methods (will see some soon…)

Today's Theme

Data Reduction

Data Reduction – Why?

Because...

- need to reduce the data so they can be feasibly stored
- need to reduce the data so a mining algorithm can be feasibly run

What else could we do

- buy more storage
- buy more computers or faster ones
- develop more efficient algorithms (look beyond O-notation)

However, in practice, all of this is happening at the same time

- unfortunately, the growth of data and complexities is always faster
- and so, data reduction will always be important

Data Reduction – How?

Reduce the number of data items (samples):

- random sampling
- stratified sampling

Reduce the number of attributes (dimensions):

- dimension reduction by transformation
- dimension reduction by elimination

Usually do both

Utmost goal

- keep the gist of the data
- only throw away what is redundant or superfluous
- it's a one way street – once it's gone, it's gone

Which Samples to Discard?

Good candidates are *redundant* data

- how many cans of ravioli will you buy?

Sampling Principles

Keep a representative number of samples:

- pick one of each
- or maybe a few more depending on importance

How to Pick?

You are faced with collections of many different data

- they are usually not nicely organized like this:

- but more like this:

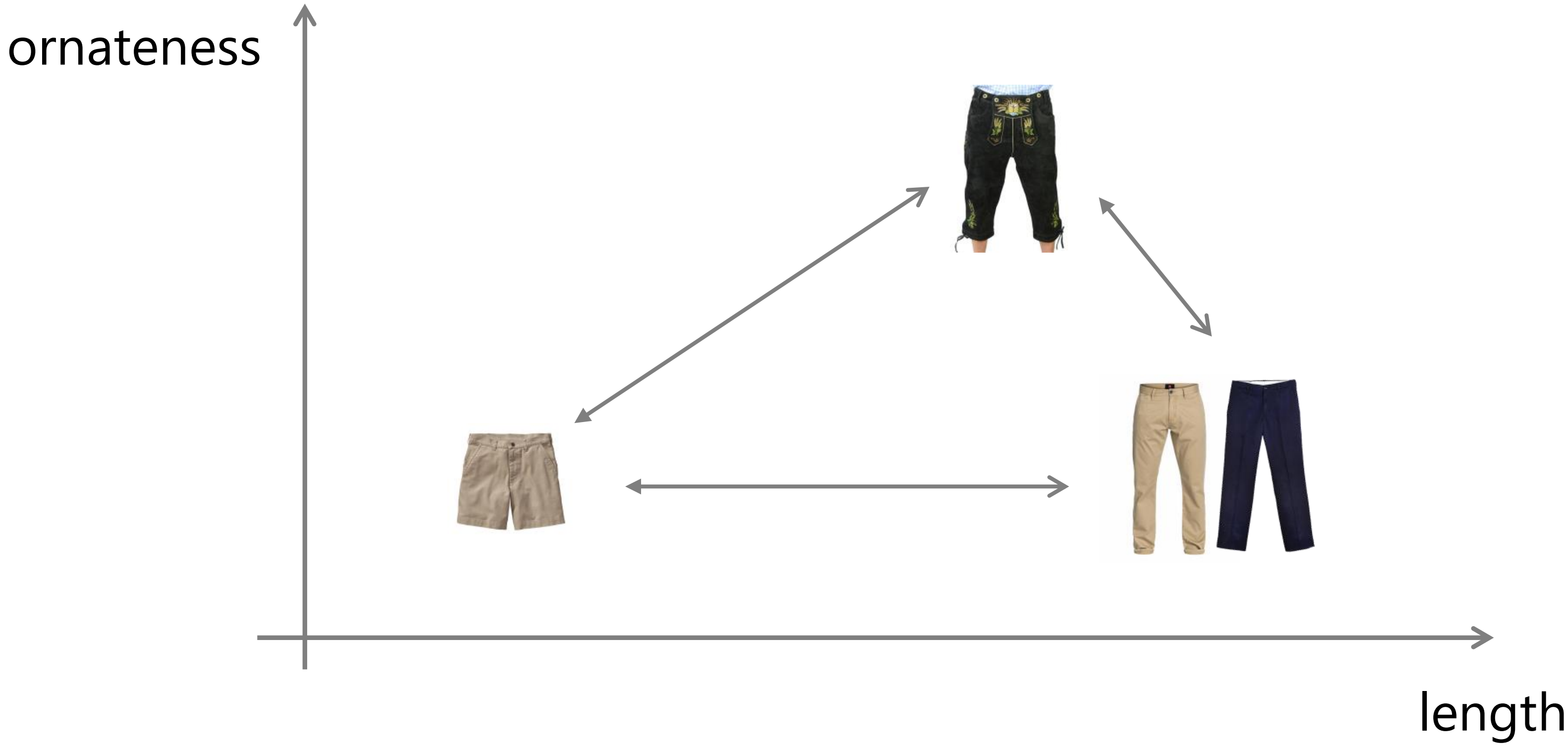
Measure of Similarity

Are all of these items pants?

- need a measure of similarity
- it’s a distance measure in high-dimensional feature space

Feature Space



We did not consider color, texture, size, etc…

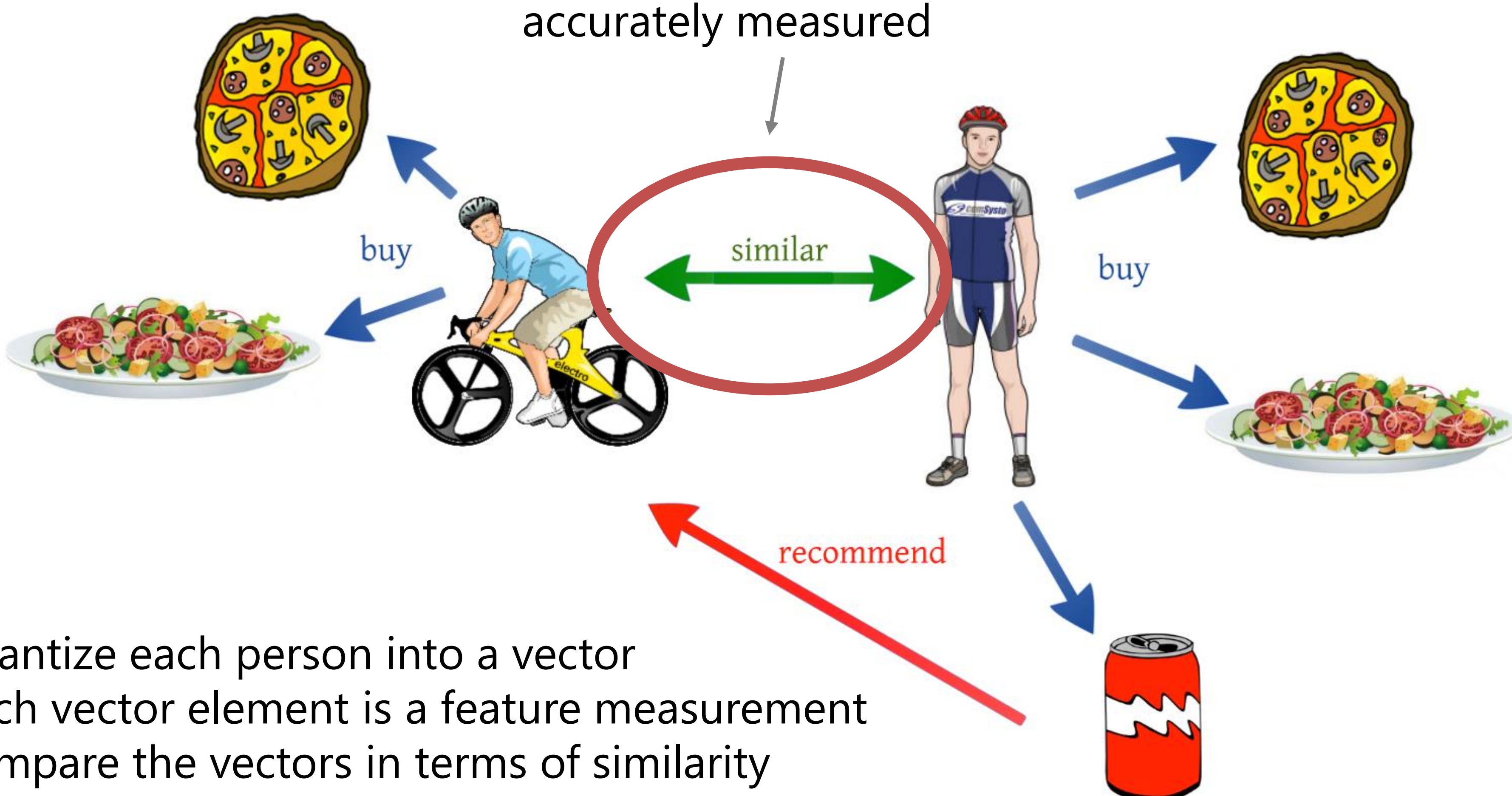
- this would have brought more differentiation (blue vs. tan pants)
- the more features, the better the differentiation

How Many Features Do We Need?

Measuring similarity can be difficult

Back To Similarity Functions

needs to be
accurately measured



quantize each person into a vector
each vector element is a feature measurement
compare the vectors in terms of similarity
similarity is also called a distance function

Data Vectors

Pant:

<length, ornateness, color>

Food delivery customer:

<type-pizza, type-salad, type-drink>

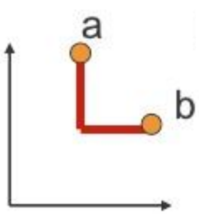
Examples:

- pants: <long, plain, tan>, <short, ornate, blue>, …
  expressed in numbers: <30”, 1, 2>, <15”, 2, 5>

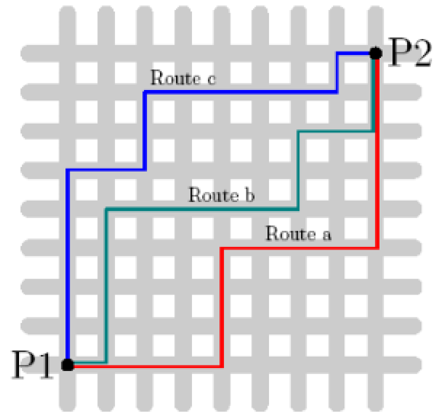
- food: <pepperoni, tossed, none>, <pepperoni, tossed, coke>, …
  expressed in numbers: <1, 1, 0>, <1, 1, 3>

Metric Distances

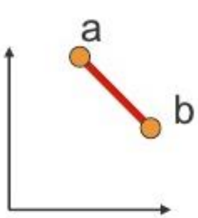
Manhattan distance



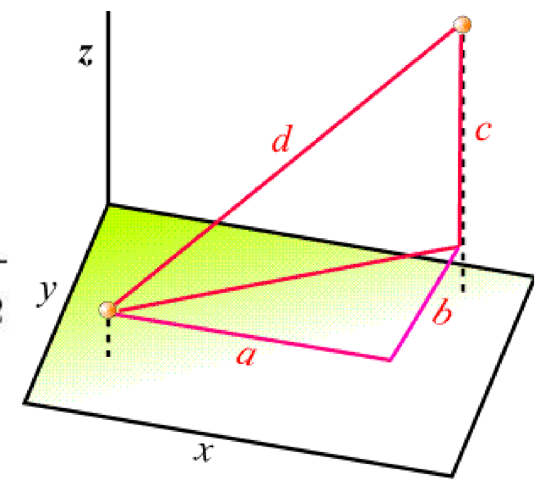
$$\text{dist(a,b)} = \|a - b\|_1 = \sum_i |a_i - b_i|$$



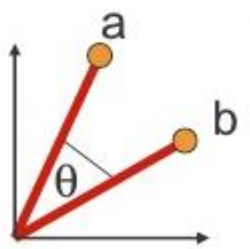
Euclidian distance



$$\text{dist(a,b)} = \|a - b\|_2 = \sqrt{\sum_i (a_i - b_i)^2}$$



Cosine Similarity



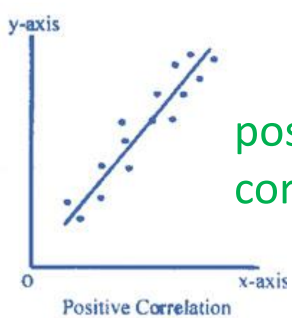
$$s(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x} \cdot \mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|} = \frac{\sum_{i=0}^{n-1} x_i y_i}{\sqrt{\sum_{i=0}^{n-1} (x_i)^2} \times \sqrt{\sum_{i=0}^{n-1} (y_i)^2}}$$

how is this related to correlation?

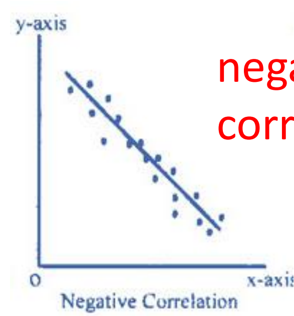
Interlude – Correlation

What is *correlation*

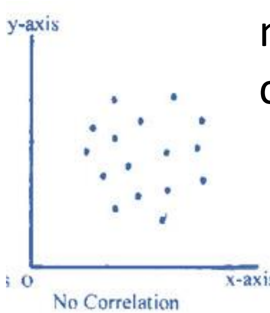
- correlation is a statistical measure that indicates the extent to which two or more variables fluctuate together
- a positive correlation indicates the extent to which those variables increase or decrease in parallel
- a negative correlation indicates the extent to which one variable increases as the other decreases



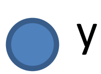
positive correlation



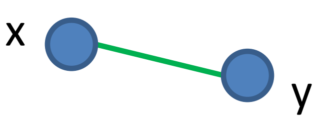
negative correlation

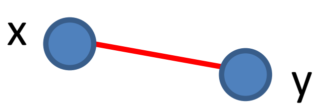


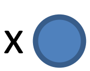
no correlation



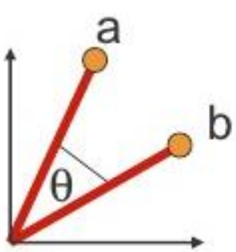
spatial proximity representation







Cosine Similarity



$$s(\mathbf{x},\mathbf{y}) = \frac{\mathbf{x}\cdot\mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|} = \frac{\sum_{i=0}^{n-1} x_i y_i}{\sqrt{\sum_{i=0}^{n-1} (x_i)^2} \times \sqrt{\sum_{i=0}^{n-1} (y_i)^2}}$$

how is this related to correlation?

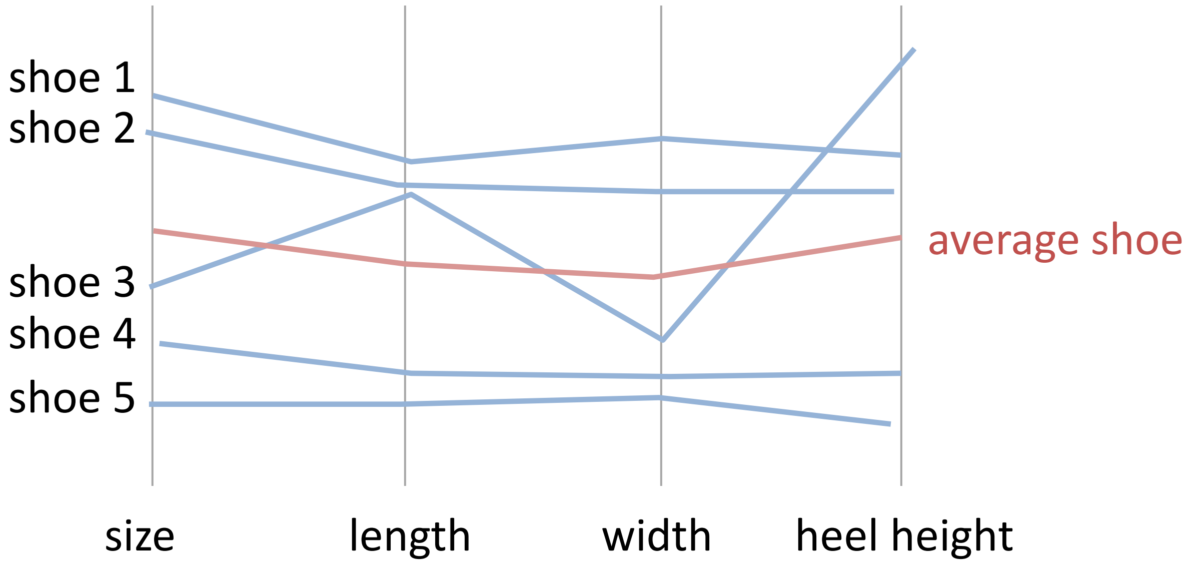
Pearson's Correlation = correlation similarity

mean across all variable values for data items x, y

$$r = r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

e.g. the "average looking" pair of pants or shoes

Correlation Demonstration



Correlations: (5x5-5)/2=10 pairs

- positively correlated: shoes 1 and 2, shoes 4 and 5
- negatively correlated: shoes 1 and 4, 1 and 5, 2 and 4, 2 and 5
- fairly uncorrelated: shoe 3 with all others 1, 2, 4, 5

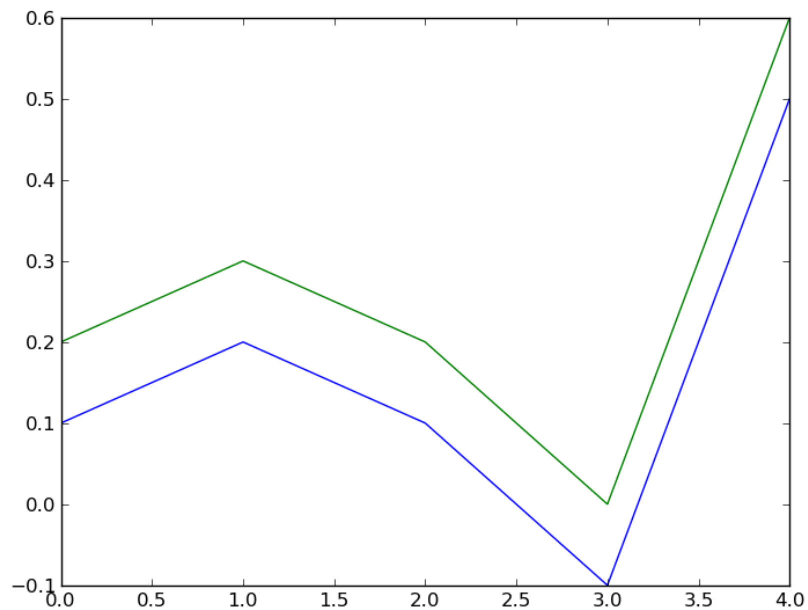
Correlation vs. Cosine Distance

Correlation distance is invariant to addition of a constant

- subtracts out by construction
- green and blue curve have correlation of 1
- but cosine similarity is < 1
- correlated vectors just vary in the same way
- cosine similarity is stricter

Both correlation and cosine similarity are invariant to multiplication with a constant

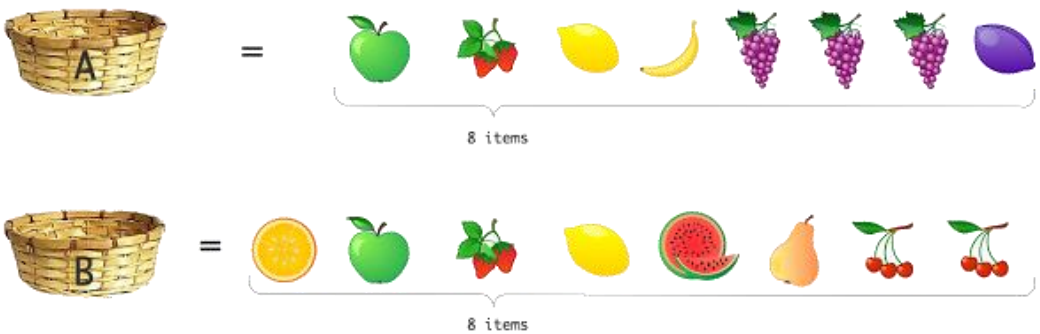
- invariant to scaling

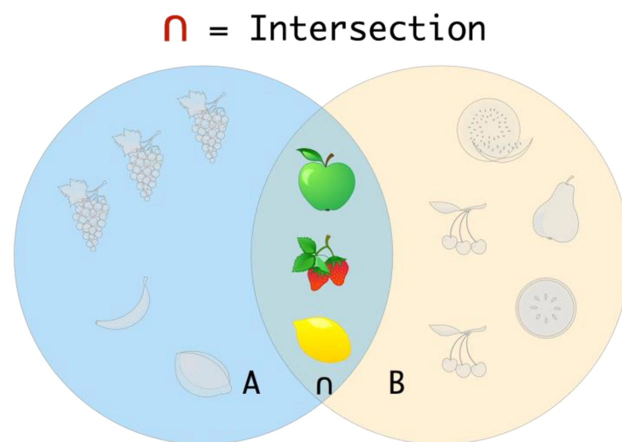


green = blue + 0.1

Jaccard Distance

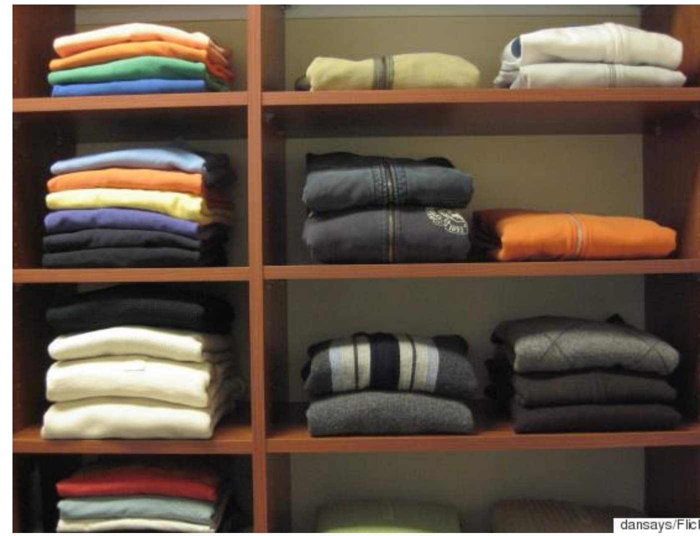
$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$





What's the Jaccard similarity of the two baskets A and B?
3/13 = 0.23

Organizing the Shelf



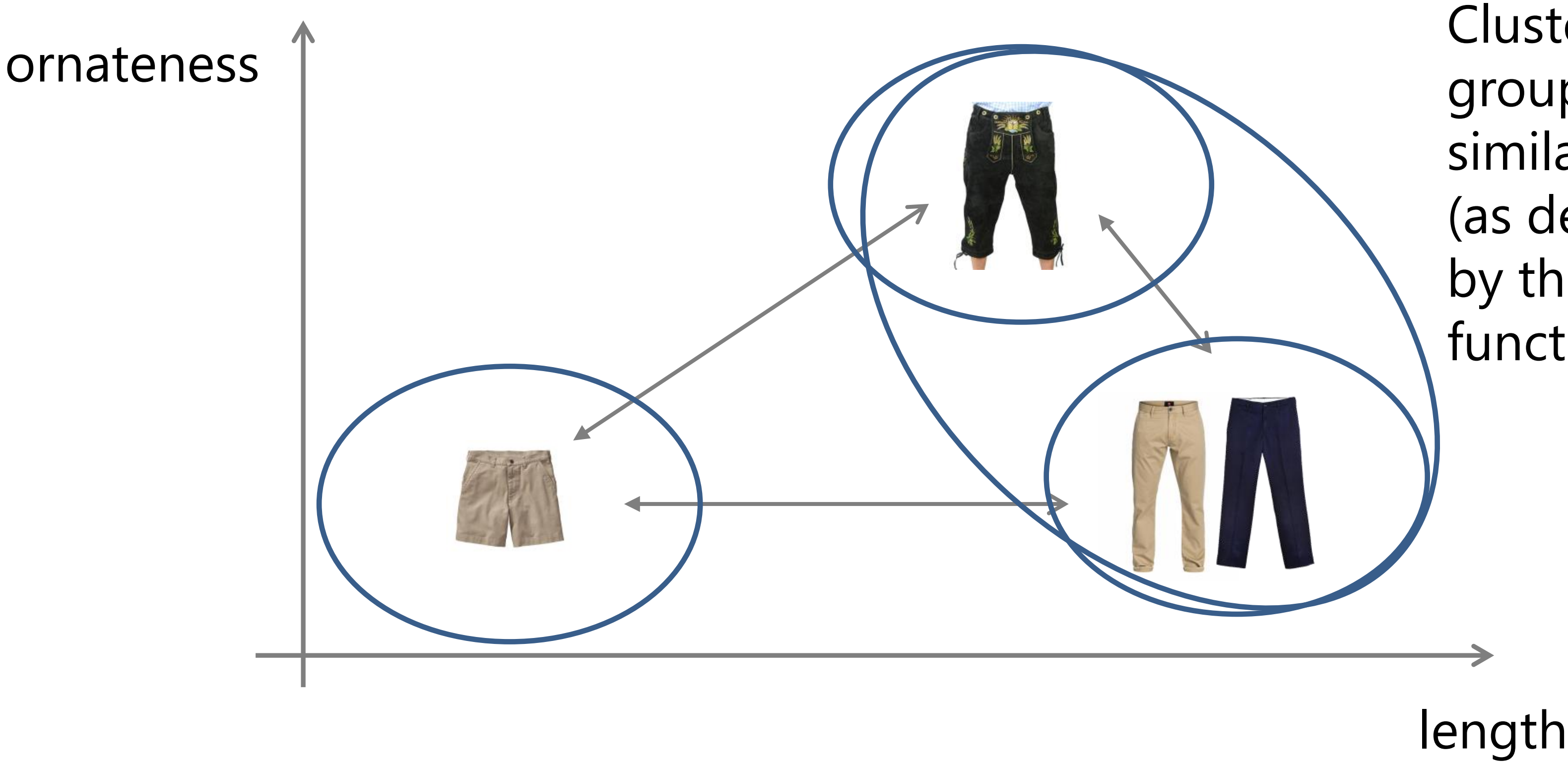
This process is called *clustering*

- and in contrast to a real store, we can make the computer do it for us

WHAT IS CLUSTERING?

Note:

- in data mining similarity and distance are the same thing
- so we will use these terms interchangeably



Clustering = grouping of similar items (as determined by the distance function)

What is a Good Cluster?

A cluster is a group of objects that are similar

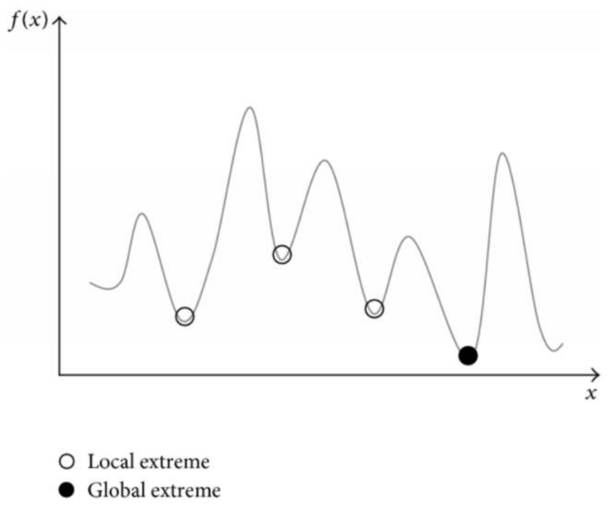
- and dissimilar from other groups of objects at the same time

We need an objective function to capture this mathematically

- the computer will evaluate this function within an algorithm
- one such function is the mean-squared error (MSE)
- and the objective is to minimize the MSE

It's not that easy in practice

- there is only one global minimum
- but often there are many local minima
- need to find the global minimum



Objective – Minimize Squared Error

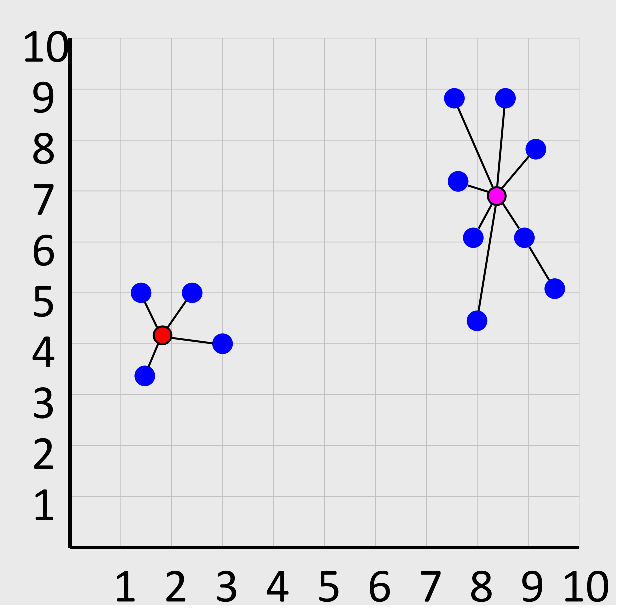
number of clusters number of cases centroid for cluster *j*

case *i*

$$\text{objective function} \leftarrow J = \sum_{j=1}^{k} \sum_{i=1}^{n} \underbrace{\left\| x_i^{(j)} - c_j \right\|^2}_{\text{Distance function}}$$

In this case

- n=12 (blue points)
- k=2 (red points, the computed centroids)
- distance metric used: Euclidian
- minimization seems to be achieved



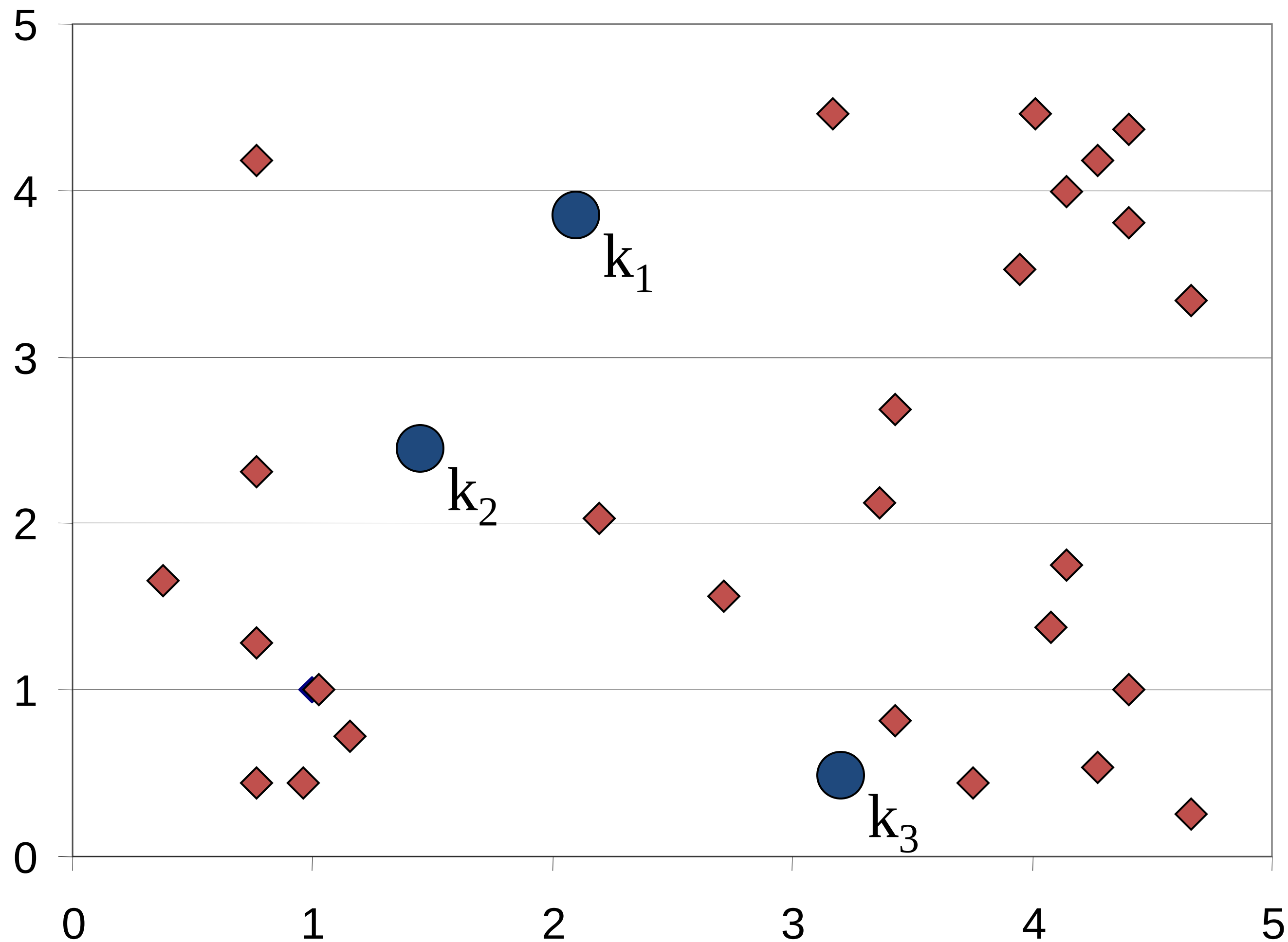
The K-Means Clustering Algorithm

1. Decide on a value for k
2. Initialize the k cluster centers (randomly, if necessary)
3. Decide the class memberships of the N objects by assigning them to the nearest cluster center
4. Re-estimate the k cluster centers, by assuming the memberships found above are correct
5. If none of the N objects changed membership in the last iteration, exit. Otherwise goto 3

The last slide and the next 8 slides contain figures courtesy of Eamonn Keogh, UC Riverside

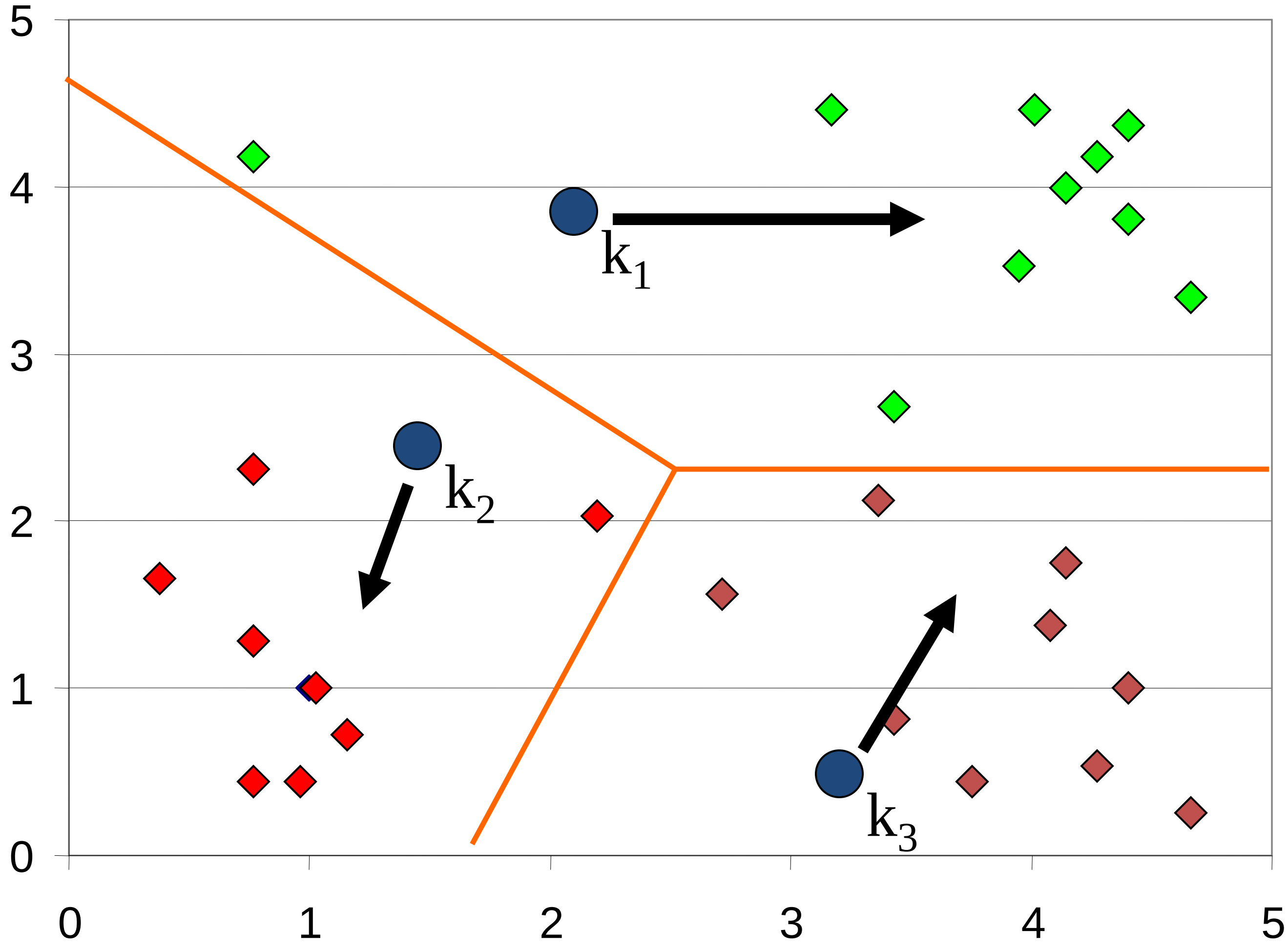
K-means Clustering: Step 1

Algorithm: k-means, Distance Metric: Euclidean Distance



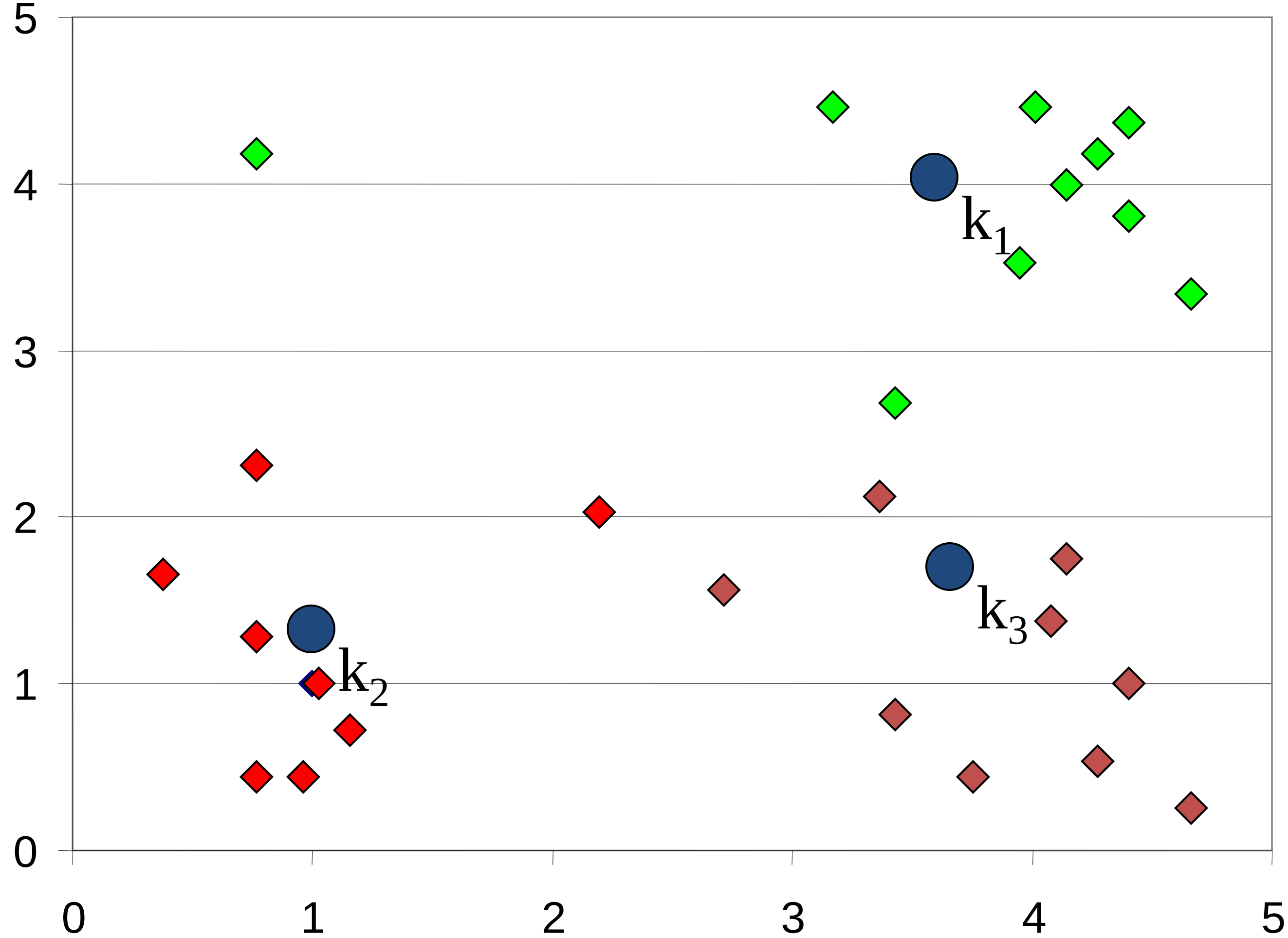
K-means Clustering: Step 2

Algorithm: k-means, Distance Metric: Euclidean Distance



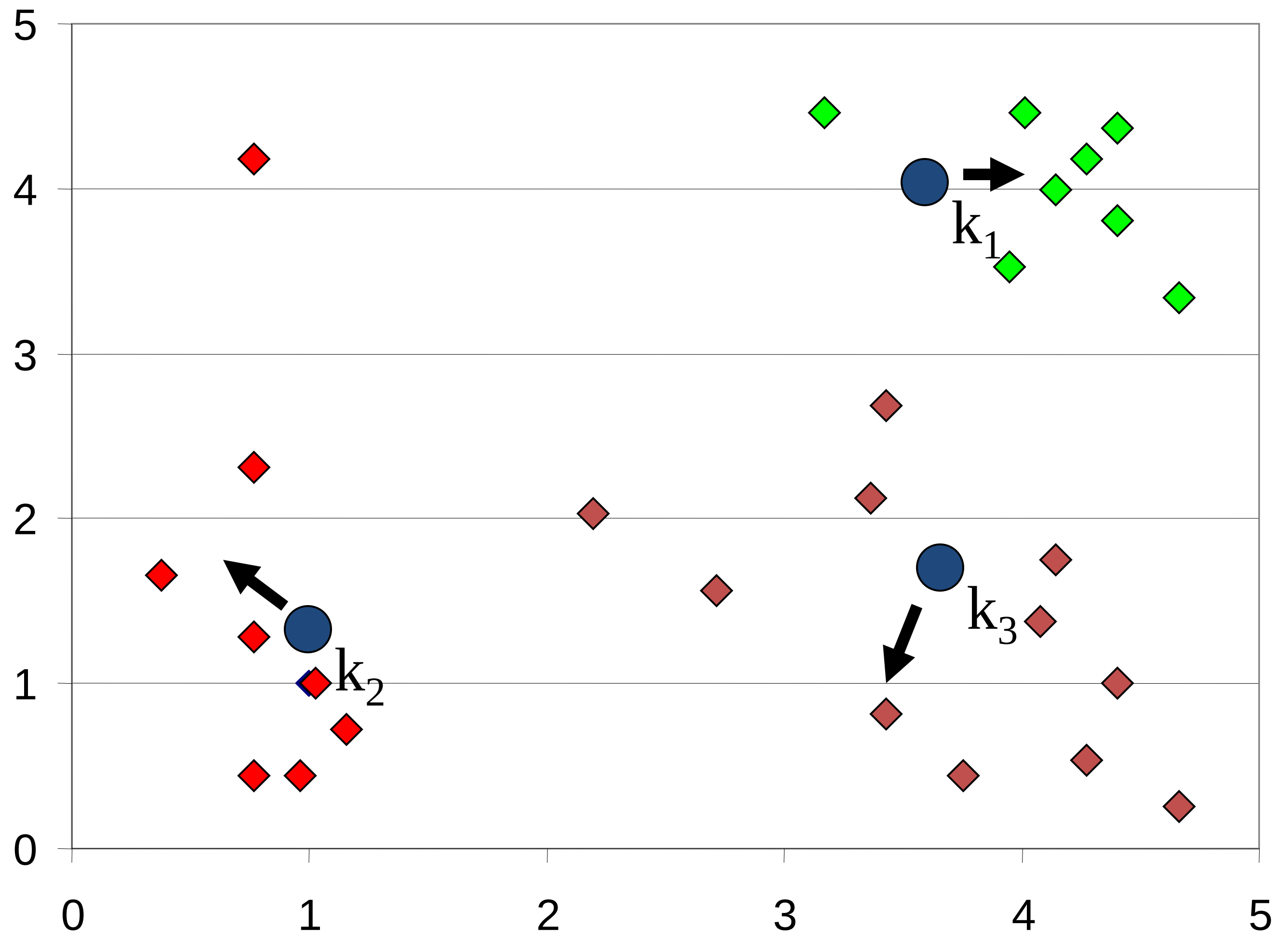
K-means Clustering: Step 3

Algorithm: k-means, Distance Metric: Euclidean Distance



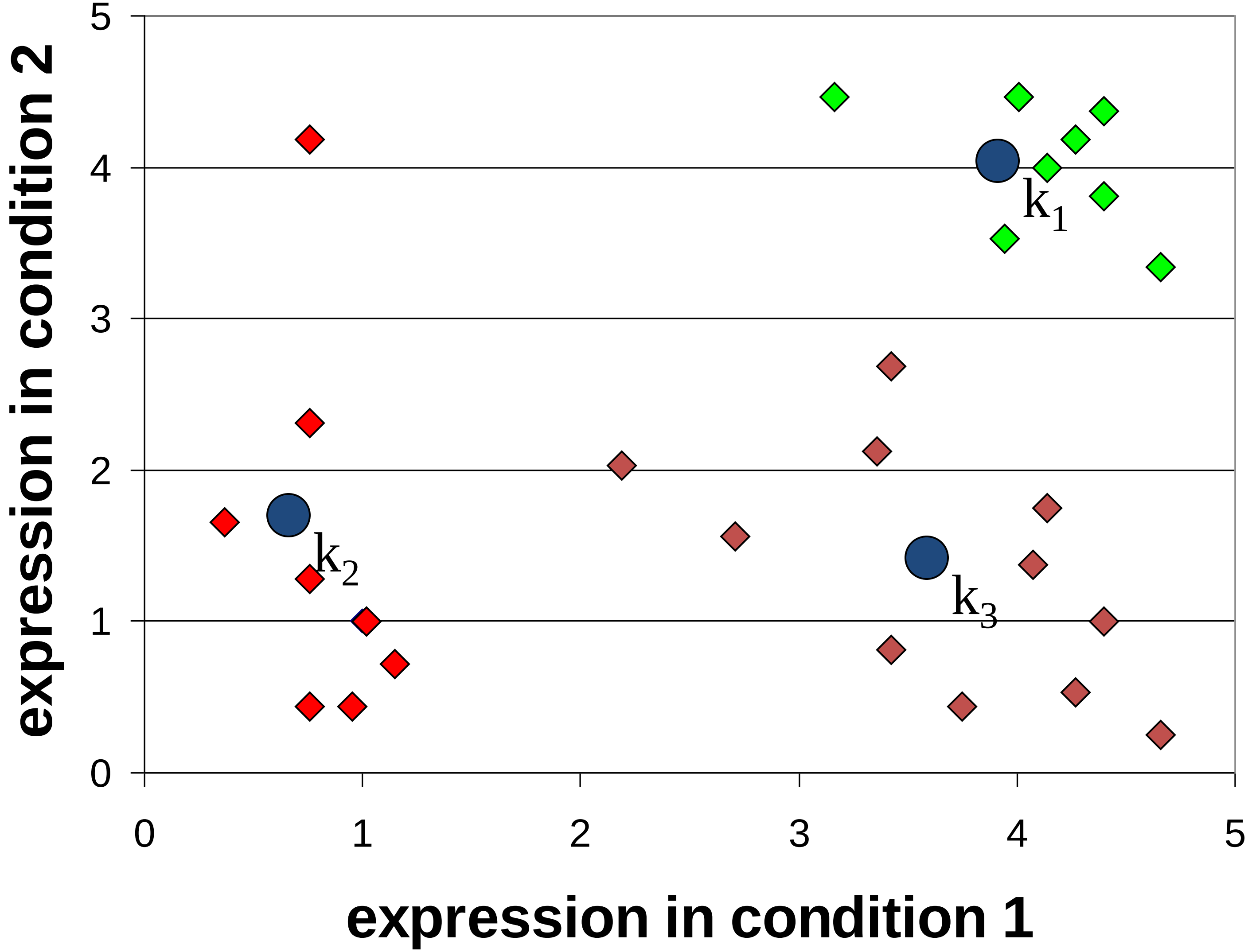
K-means Clustering: Step 4

Algorithm: k-means, Distance Metric: Euclidean Distance



K-means Clustering: Step 5

Algorithm: k-means, Distance Metric: Euclidean Distance



K-Means Algorithm – Comments

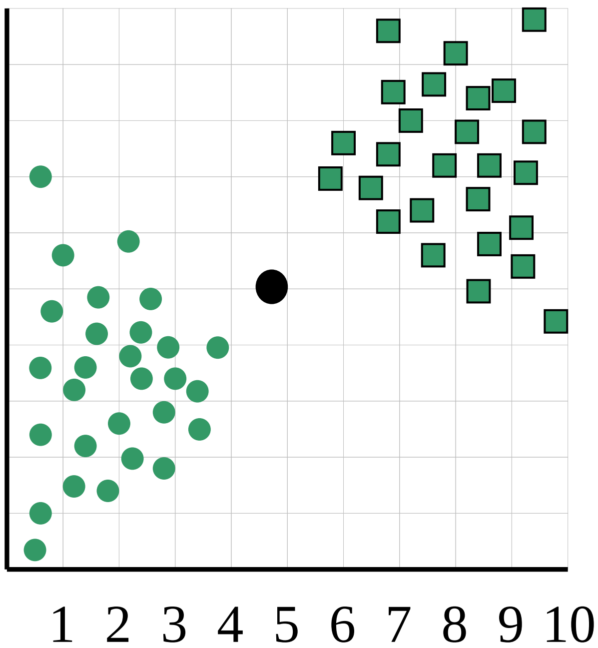
Strengths:

- *relatively efficient*: $O(tkn)$, where n is # objects, k is # clusters, and t is # iterations. Normally, k, t << n.
- simple to code

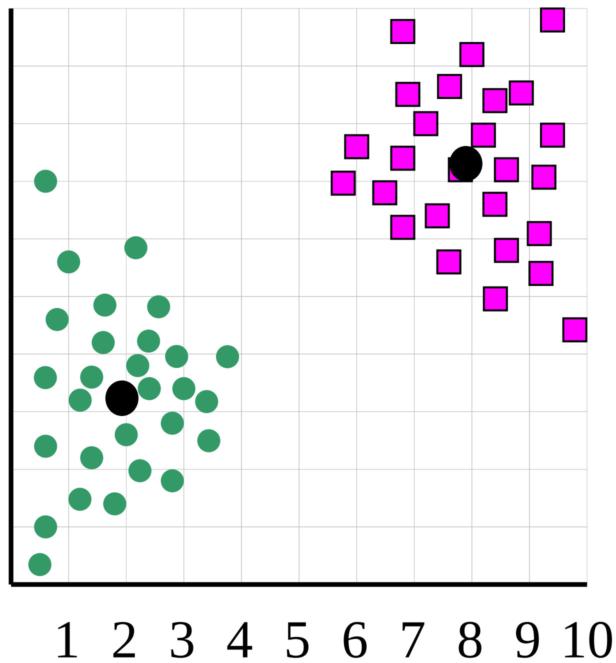
Weaknesses:

- need to specify k in advance which is often unknown
- find the best k by trying many different ones and picking the one with the lowest error
- often terminates at a *local optimum*
- the *global optimum* may be found by trying many times and using the best result

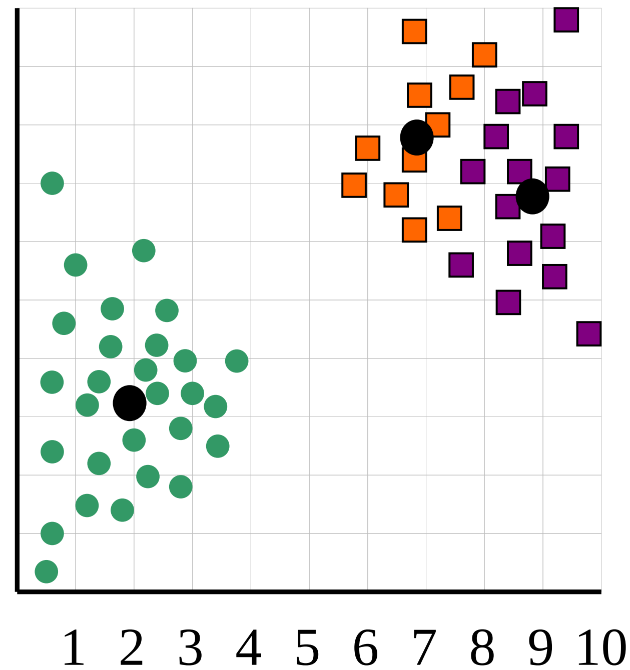
How Can We Find the Best K?



k=1, MSE=873.0



k=2, MSE=173.1



k=3, MSE=133.6

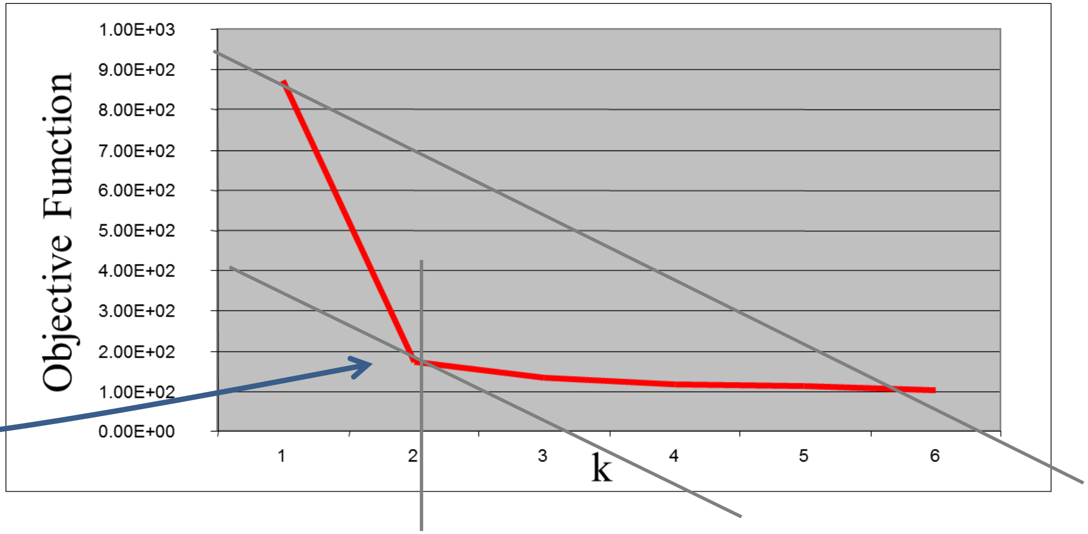
How About K=2?

Is there a principled way we can know when to stop looking?

Yes…

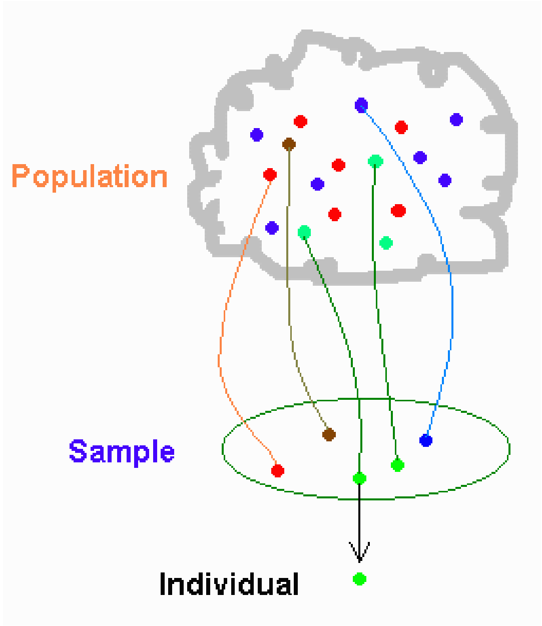
- we can plot the objective function values for k equals 1 to 6…
- then check for a flattening of the curve

tangent at k=2



- the abrupt change at k = 2 is highly suggestive of two clusters
- this technique is known as “knee finding” or “elbow finding”

Back to Data reduction



What is sampling?

- pick a <u>representative</u> subset of the data
- discard the remaining data
- pick as many you can afford to keep
- recall: once it’s gone, it’s gone
- be smart about it

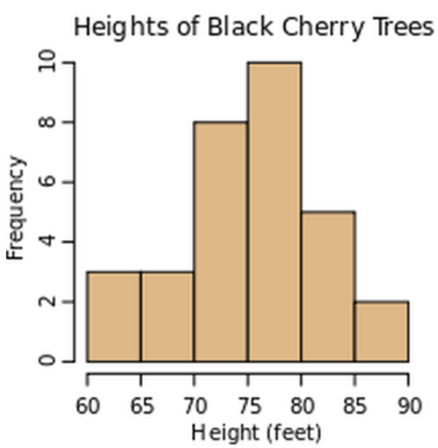
Simplest: random sampling

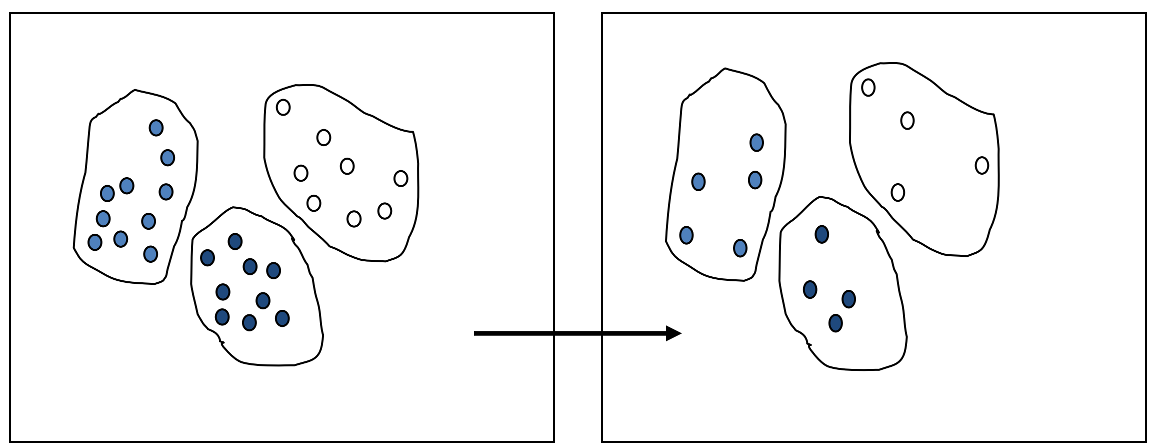
- pick sample points at random
- will work if the points are distributed uniformly
- this is usually not the case
- outliers will likely be missed
- so the sample will not be representative

Better: Adaptive Sampling

Pick the samples according to some knowledge of the data distribution

- cluster the data (outliers will form clusters as well)
- these clusters are also called *strata* (hence, stratified sampling)
- the size of each cluster represents its percentage in the population
- guides the number of samples – bigger clusters get more samples





sampling rate ~ bin height

sampling rate ~ cluster size

Redundancy Sampling

Good candidates for elimination are *redundant* data

- how many cans of ravioli will you buy?

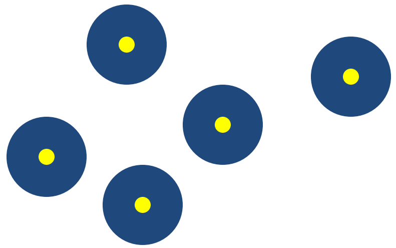
Redundancy Sampling

Eliminate redundant attributes

- eliminate correlated attributes
  - km vs. miles
  - a + b + c = d → can eliminate 'c' (or 'a' or 'b')

Eliminate redundant data

- cluster the data with small ranges
- only keep the cluster centroids
- store size of clusters along to keep importance



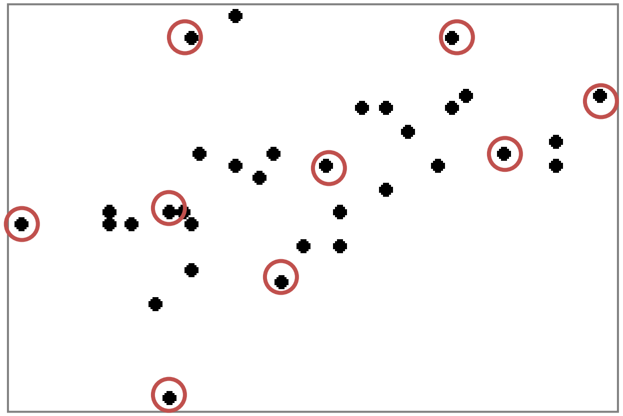
Sampling of Well-Scattered Points

Used in the CURE high-dimensional clustering algorithm

- S. Guha, R. Rajeev, and K. Shim. "CURE: an efficient clustering algorithm for large databases." *ACM SIGMOD*, 27(2): 73-84, 1998

Algorithm

- initialize the point set *S* to empty
- pick the point farthest from the mean as the first point for *S*
- then iteratively pick points that are furthest from the points in S collected so far



Complexity is O($m \cdot n^2$)

- *n* is the total number of points, *m* is the number of desired points
- can find arbitrarily shaped clusters and preserve outliers, too
- need some good data structures to run efficiently: kd-tree, heap
- can get really expensive when the dimensionality *d* is large because each pairwise distance has O(*d*)

SUMMARY

Learned about

- distance metrics to evaluate similarity among data points
- correlation, cosine, Euclidian, Jacquard, Manhattan distance
- used it for clustering that can identify groups in data
- these groups can be used for unbiased data reduction and augmentation
- the k-means algorithm as a simple yet effective clustering scheme
- the elbow method to pick a good k = number of clusters
- advanced sampling methods: well-scattered points,. Reservoir sampling for streaming data