CSE 332
Introduction to Visualization

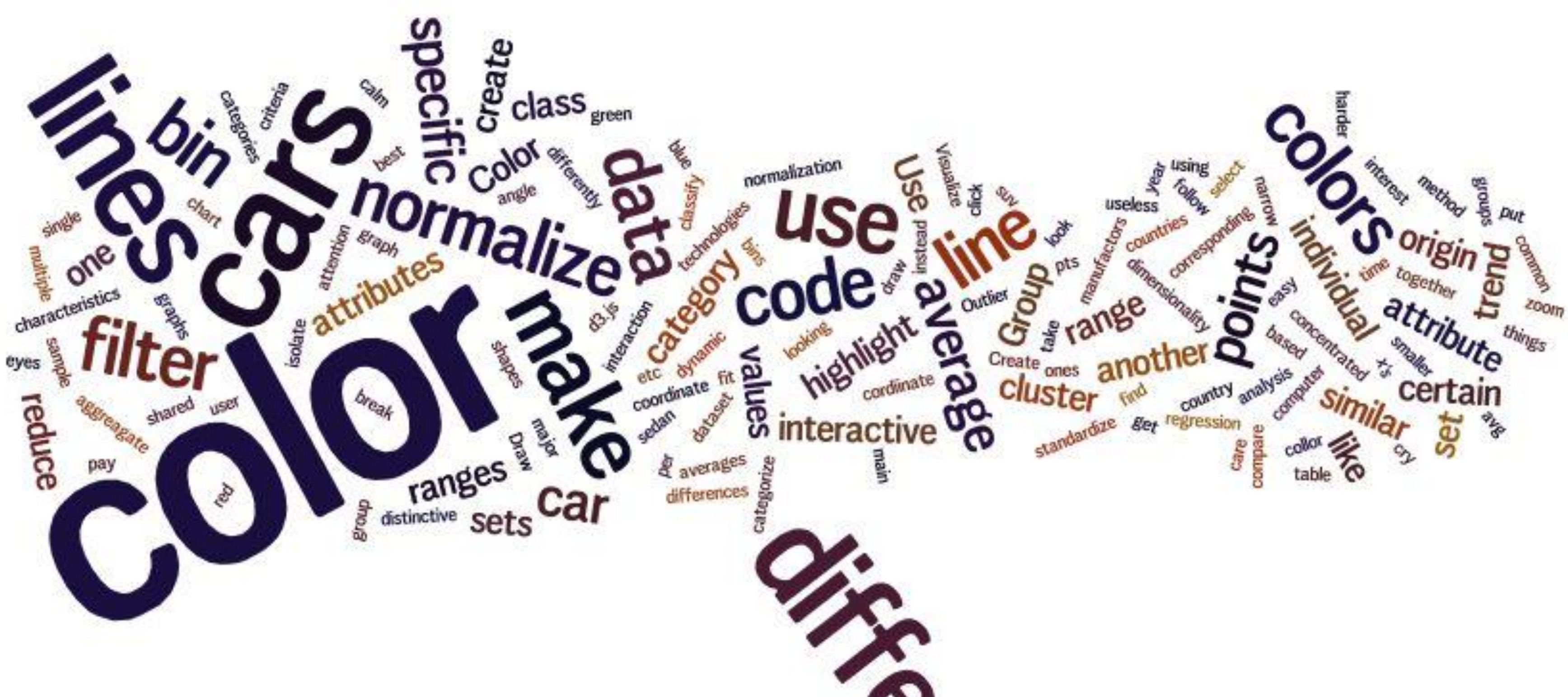
Data Reduction & Similarity metrics

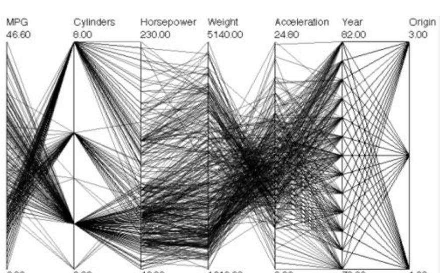
Klaus Mueller

Computer Science Department
Stony Brook University

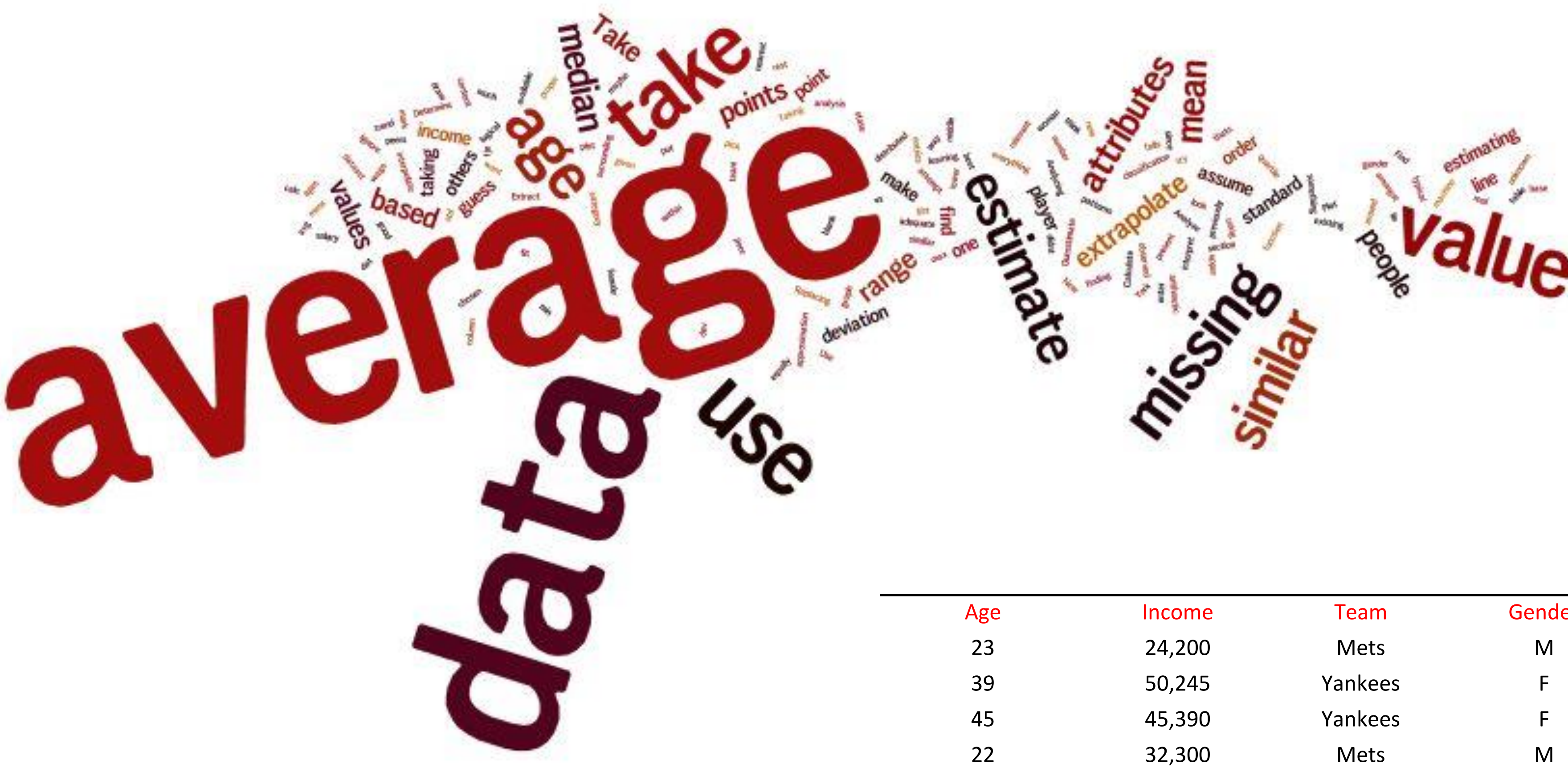
Lecture	Topic	Projects
1	Intro, schedule, and logistics	
2	Applications of visual analytics, data, and basic tasks	
3	Data preparation and representation	Project 1 out
4	Data reduction, notion of similarity and distance	
5	Introduction to D3, basic vis techniques for non-spatial data	
6	Visual perception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

What can we do to see through the mess of lines?





How would you estimate the missing value?



Age	Income	Team	Gender
23	24,200	Mets	M
39	50,245	Yankees	F
45	45,390	Yankees	F
22	32,300	Mets	M
52		Yankees	F
27	28,300	Mets	F
48	53,100	Yankees	M

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:

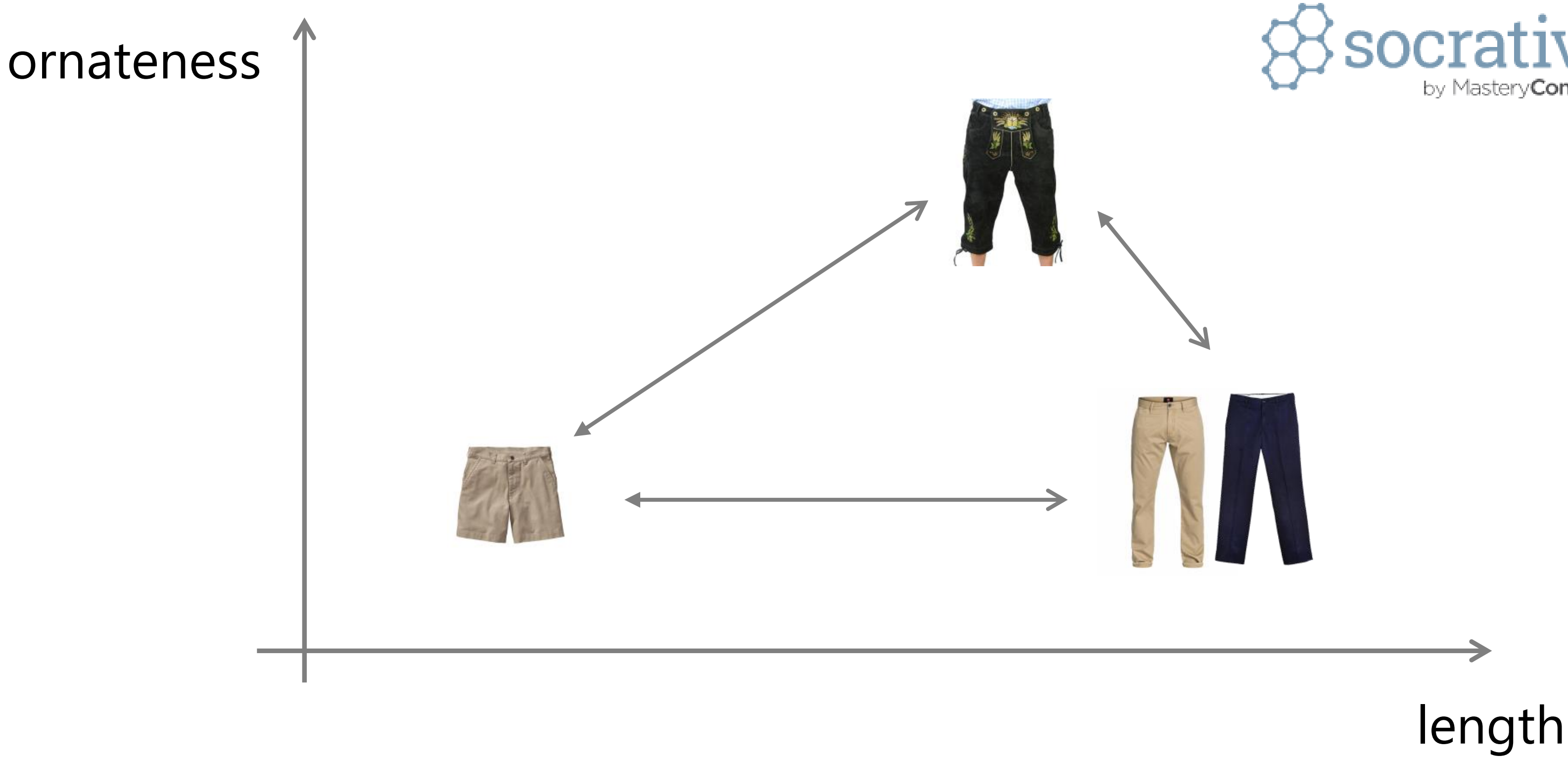
Measure of Similarity

Are all of these items pants?

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

Feature Space



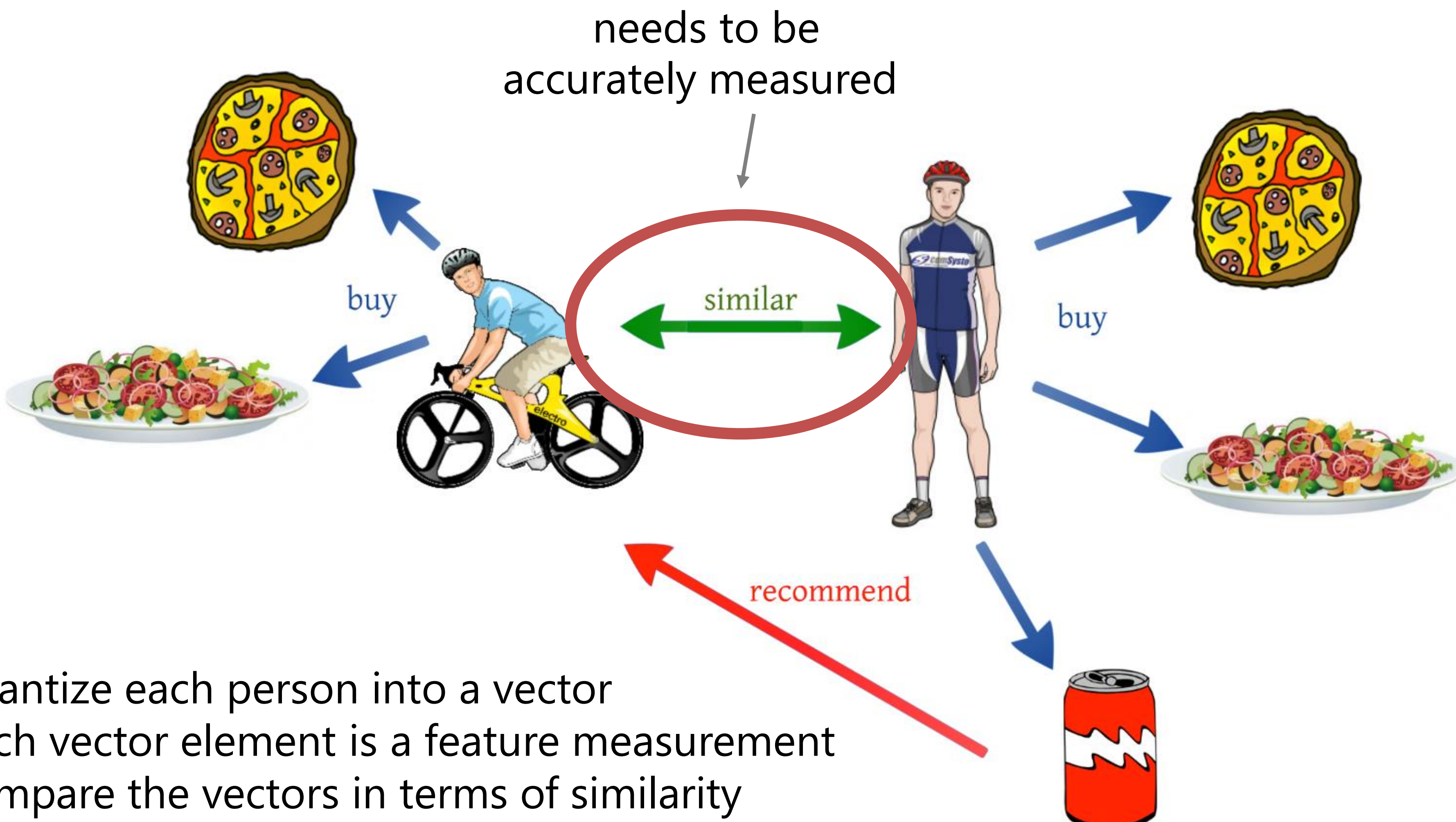
We did not consider color, texture, size, etc…

- 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



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>

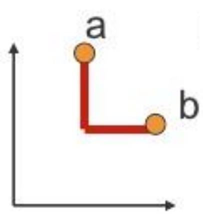
Examples:

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

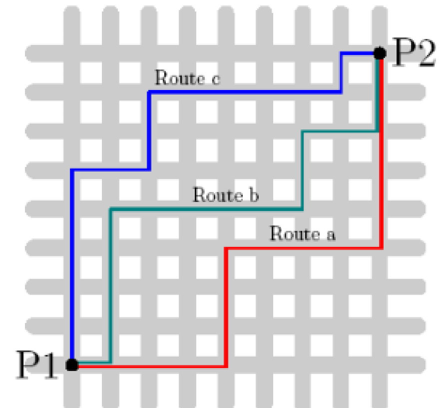
- food: <pepperoni, tossed, none>, <pepperoni, tossed, coke>, …
  expressed in numbers: <1, 1, 0>, <1, 1, 3>

Metric Distances

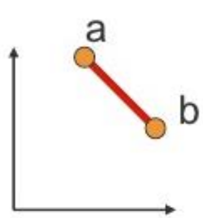
Manhattan distance



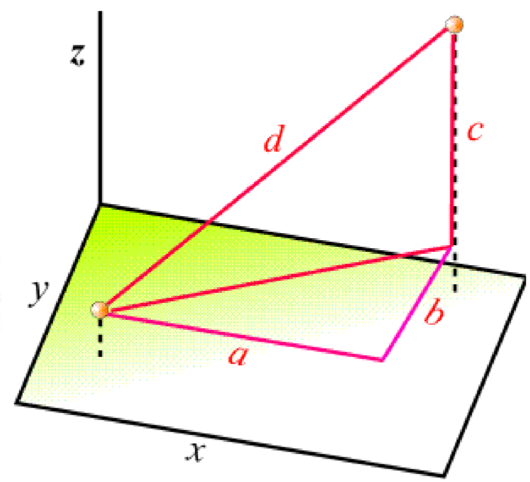
$$\text{dist}(a,b) = \|a - b\|_1 = \sum_i |a_i - b_i|$$



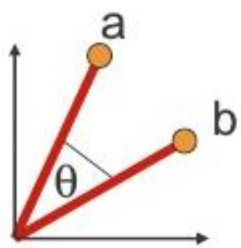
Euclidian distance



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



Cosine Similarity



$$\text{dist}(a,b) = \cos^{-1}\frac{\langle a, b\rangle}{\|a\|\|b\|}$$

how is this related to correlation?

Pearson's Correlation = correlation similarity

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

mean across all variable values for data items x, y

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

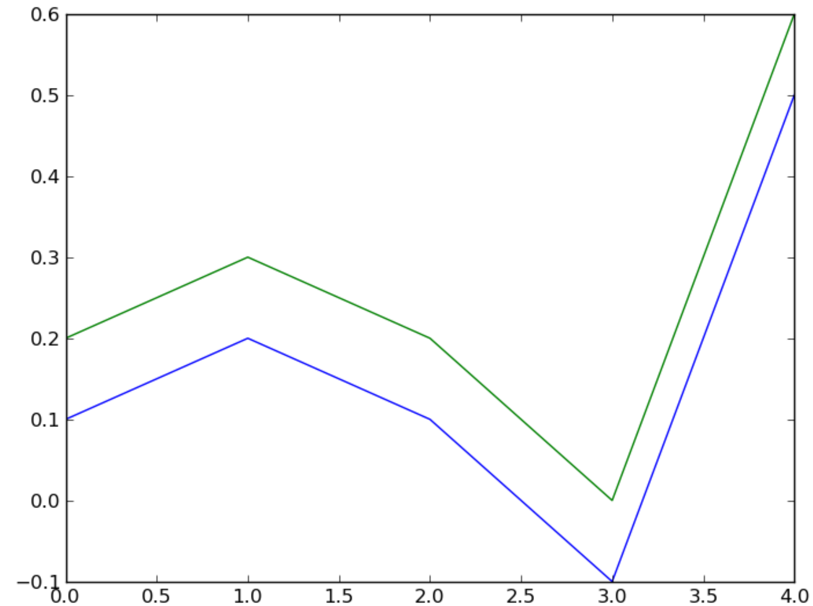
Correlation vs. Cosine Distance

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

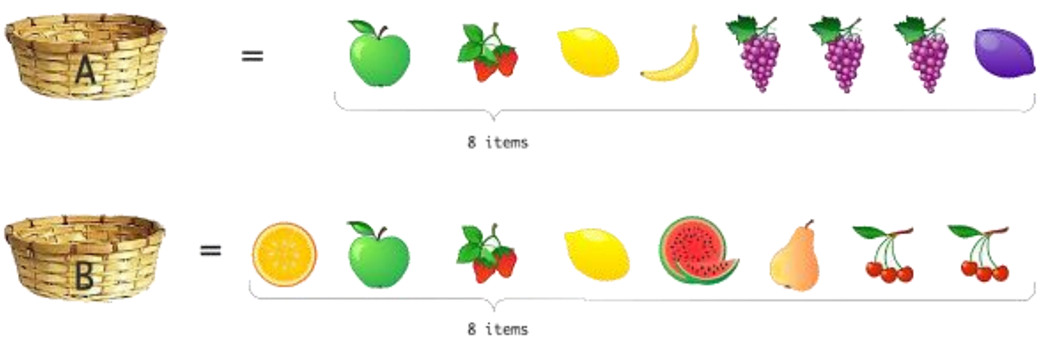
- invariant to scaling

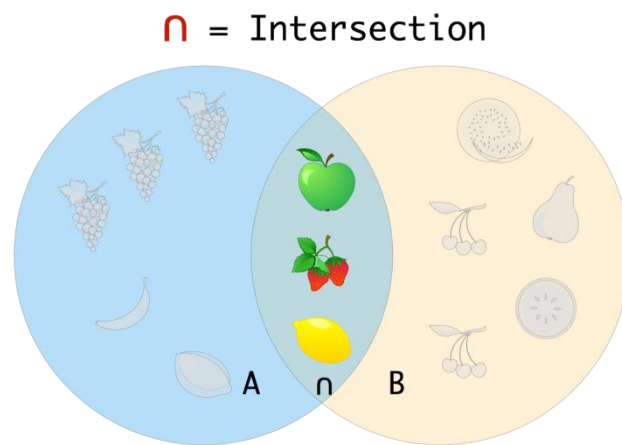


green = blue + 0.1

Jaccard Distance

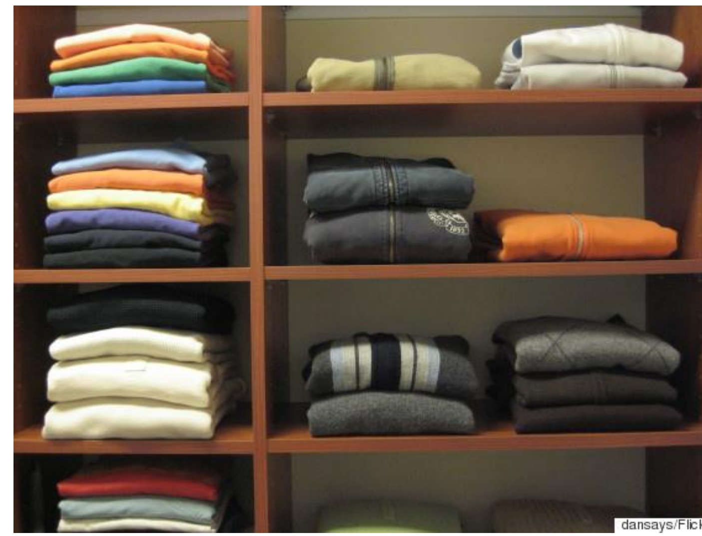
$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}.$$





What’s the Jaccard similarity of the two baskets A and B?

Organizing the Shelf



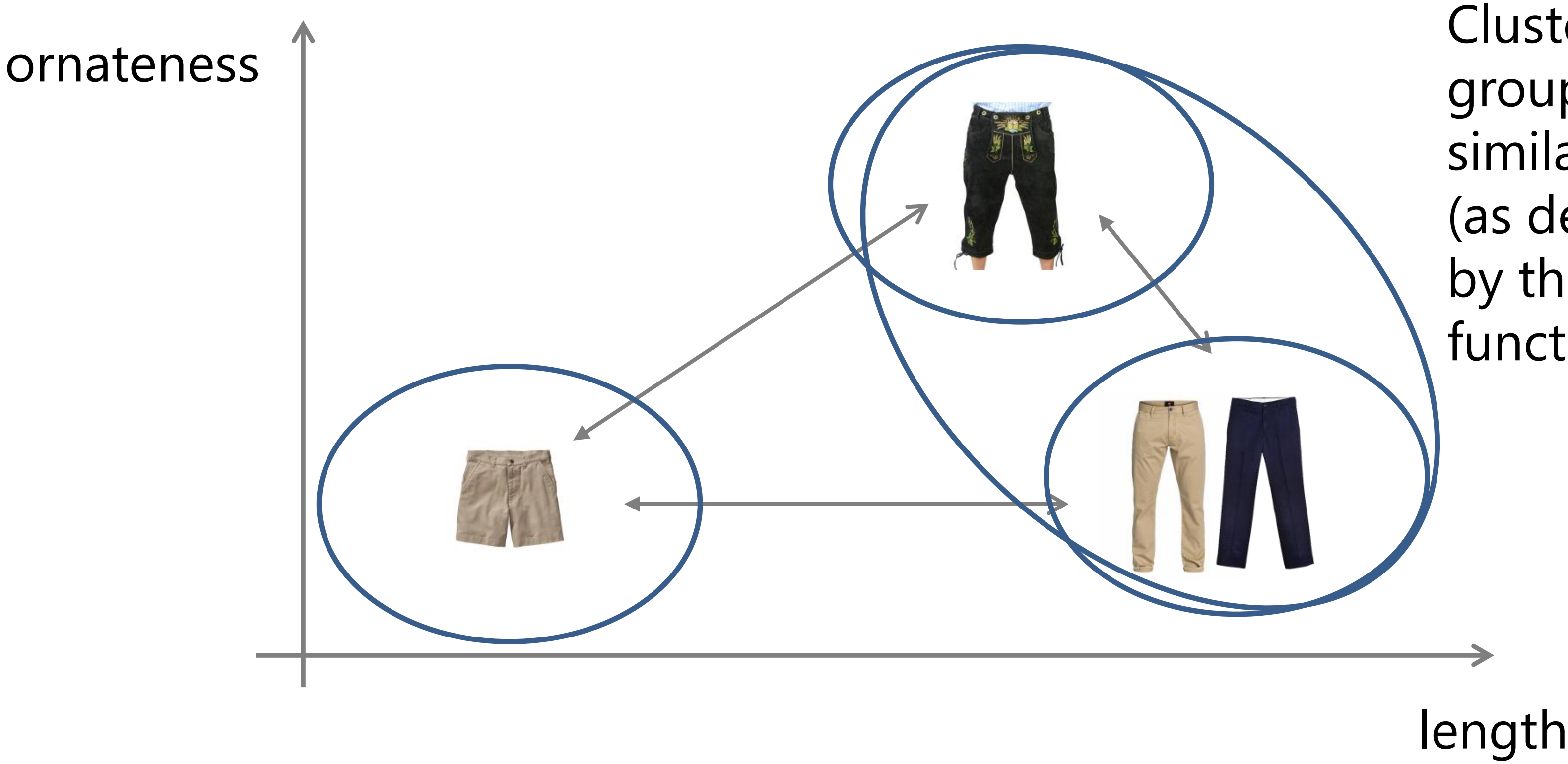
This process is called *clustering*

- 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

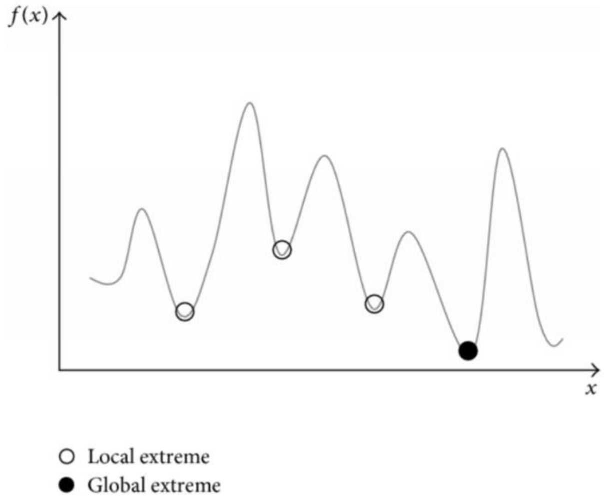
- and dissimilar from other groups of objects at the same time

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

case *i*

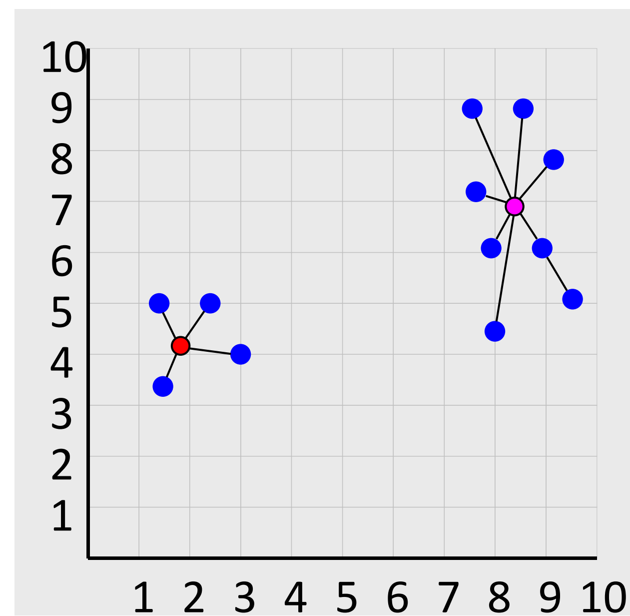
centroid for cluster *j*

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

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



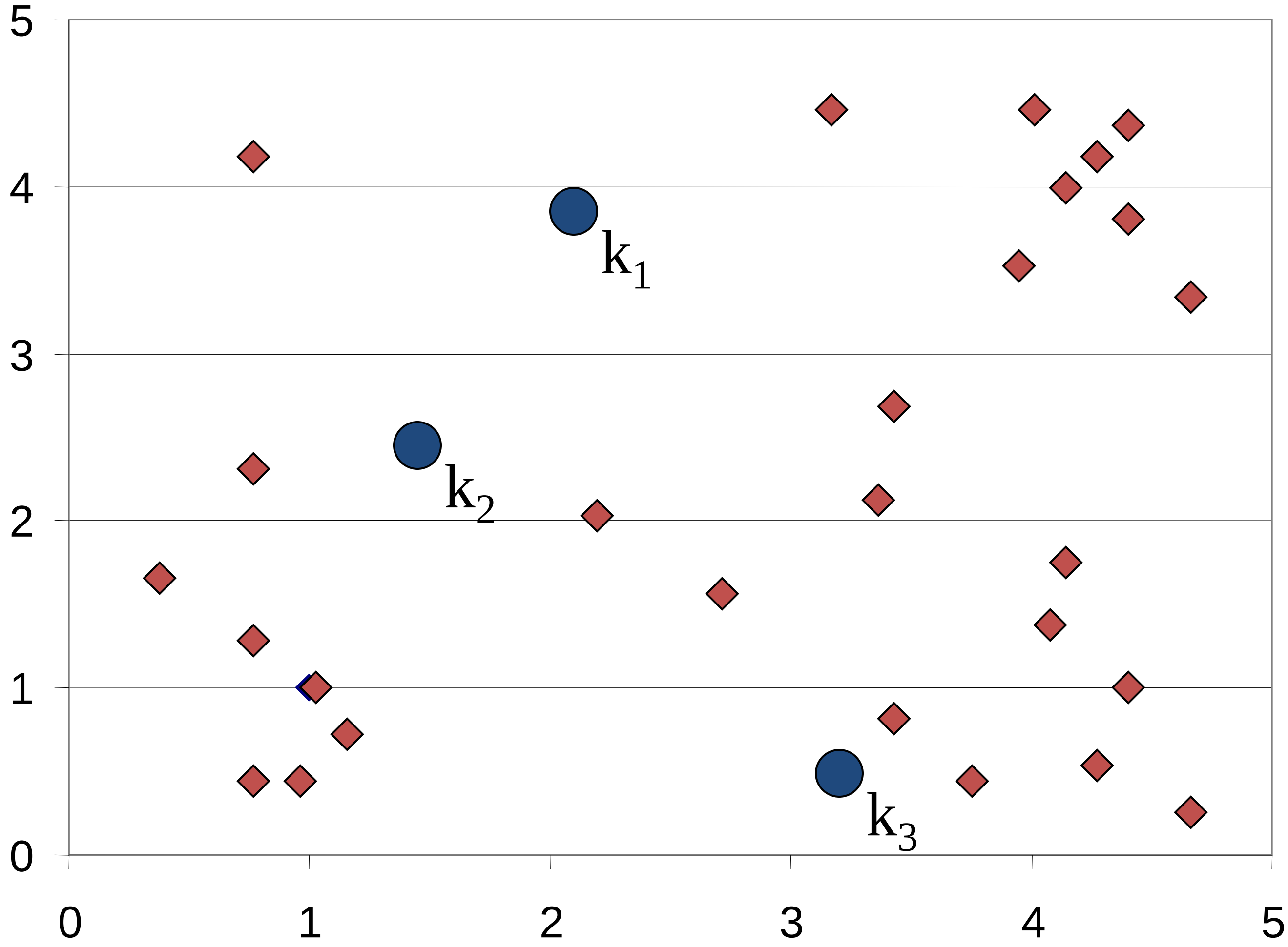
The K-Means Clustering Algorithm

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

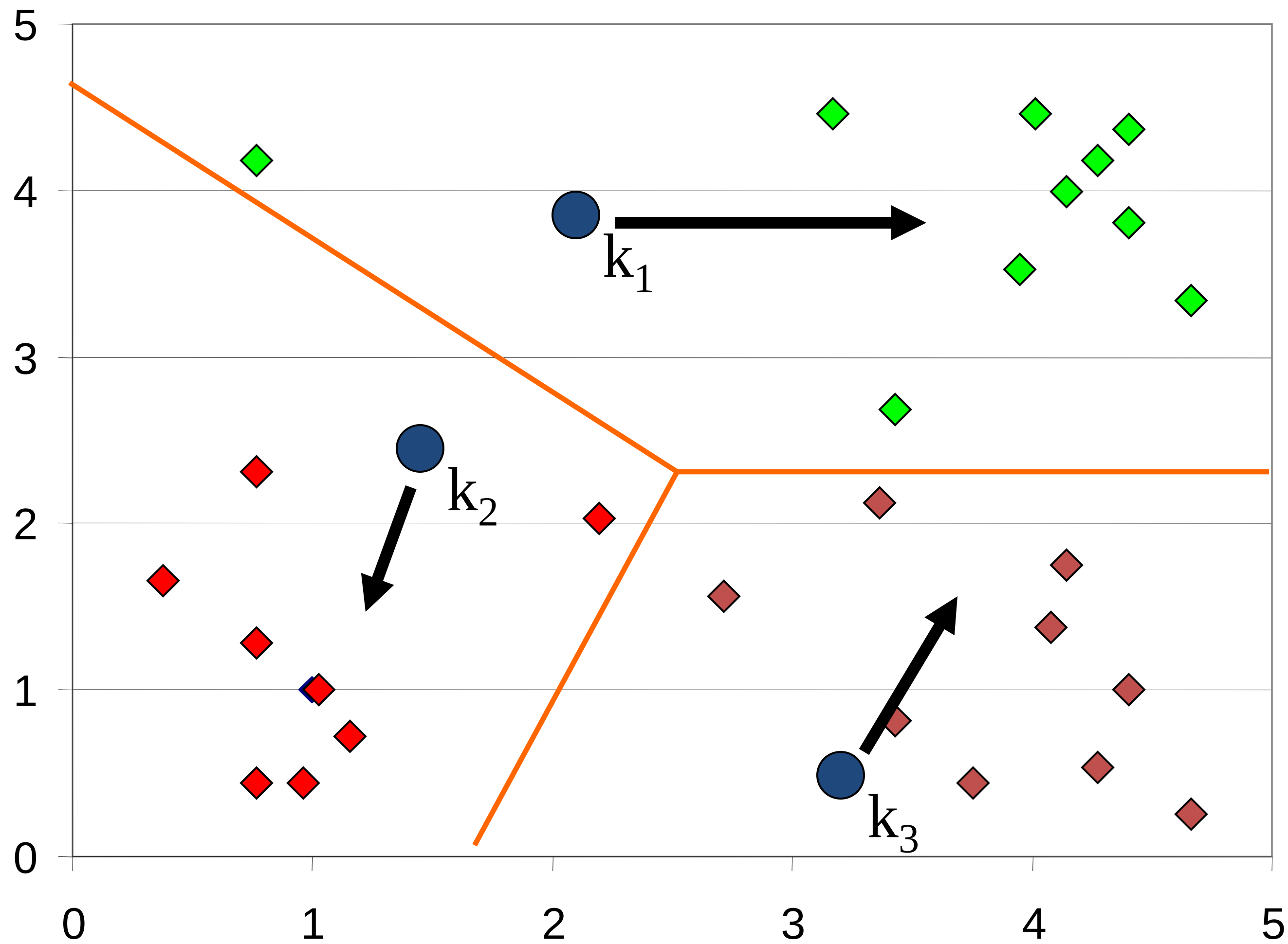
K-means Clustering: Step 1

Algorithm: k-means, Distance Metric: Euclidean Distance



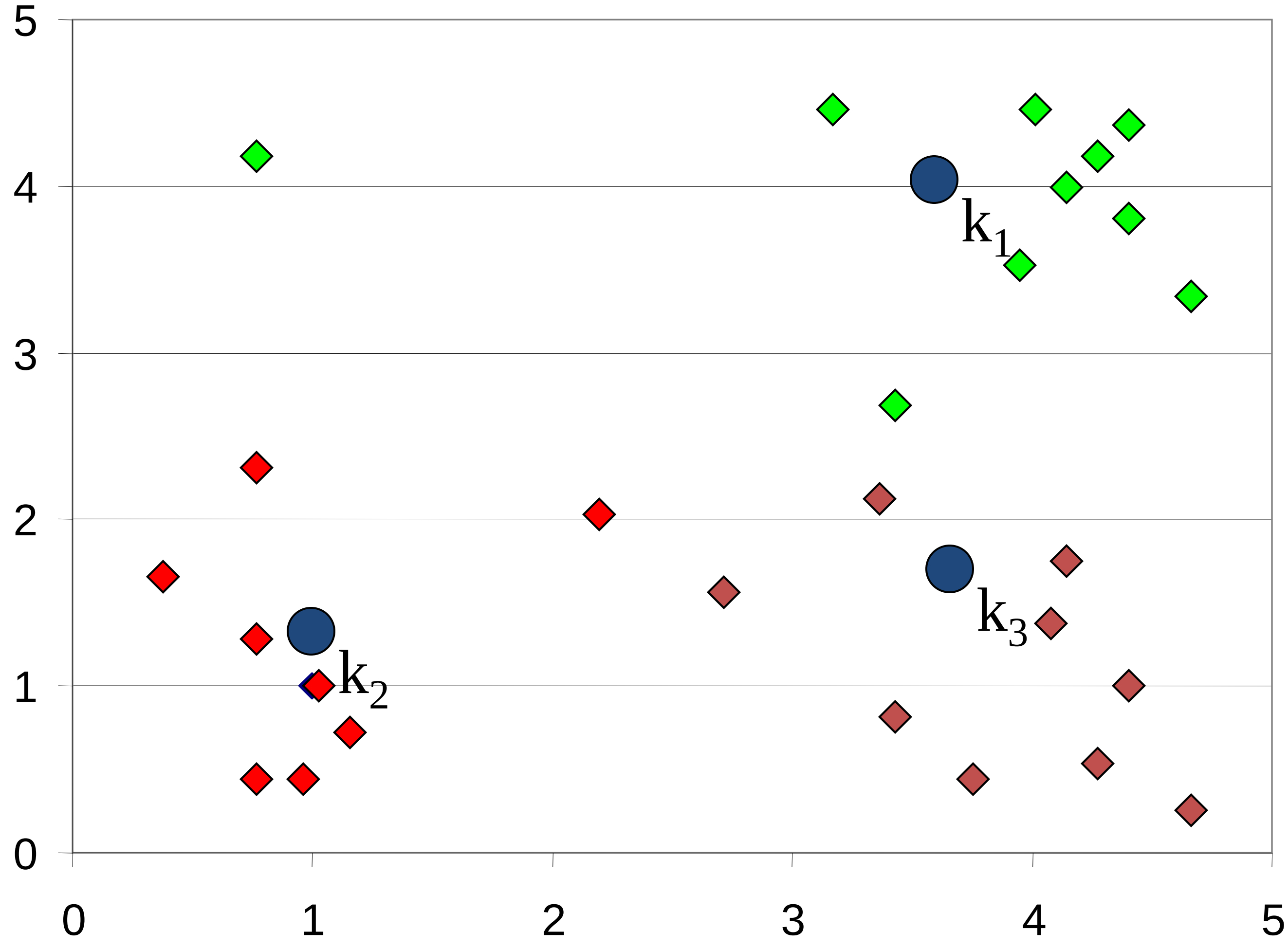
K-means Clustering: Step 2

Algorithm: k-means, Distance Metric: Euclidean Distance



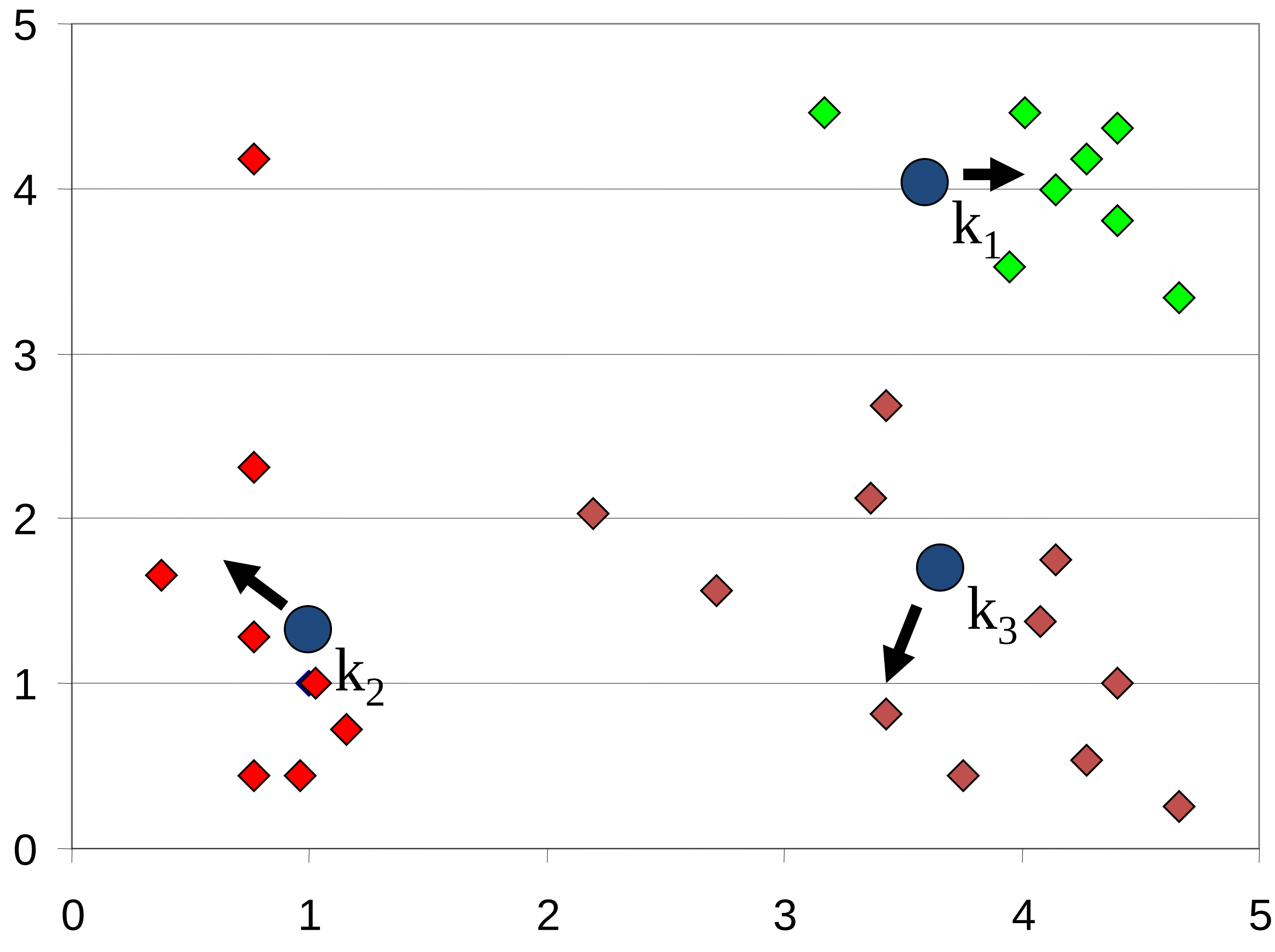
K-means Clustering: Step 3

Algorithm: k-means, Distance Metric: Euclidean Distance



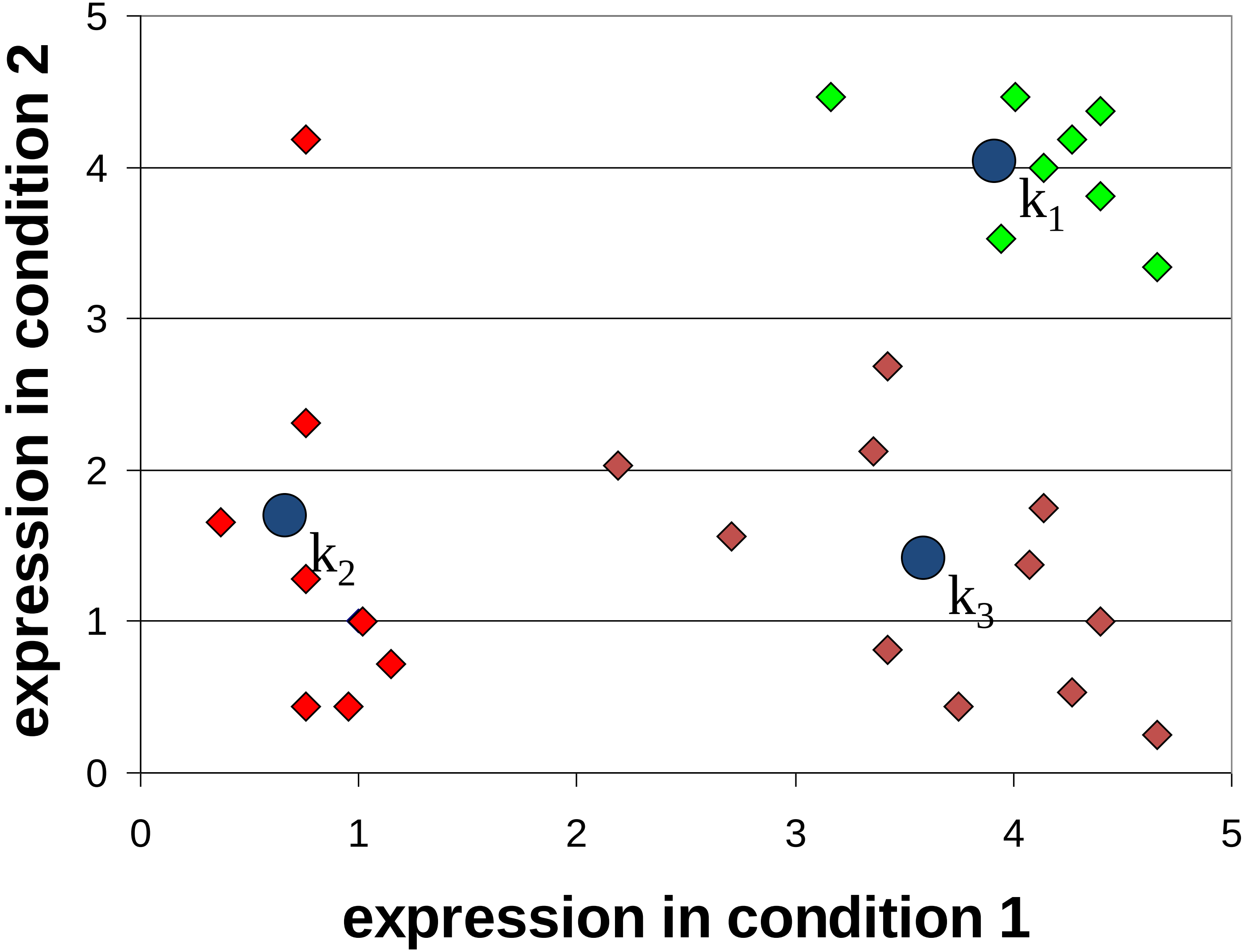
K-means Clustering: Step 4

Algorithm: k-means, Distance Metric: Euclidean Distance



K-means Clustering: Step 5

Algorithm: k-means, Distance Metric: Euclidean Distance



K-Means Algorithm – Comments

Strengths:

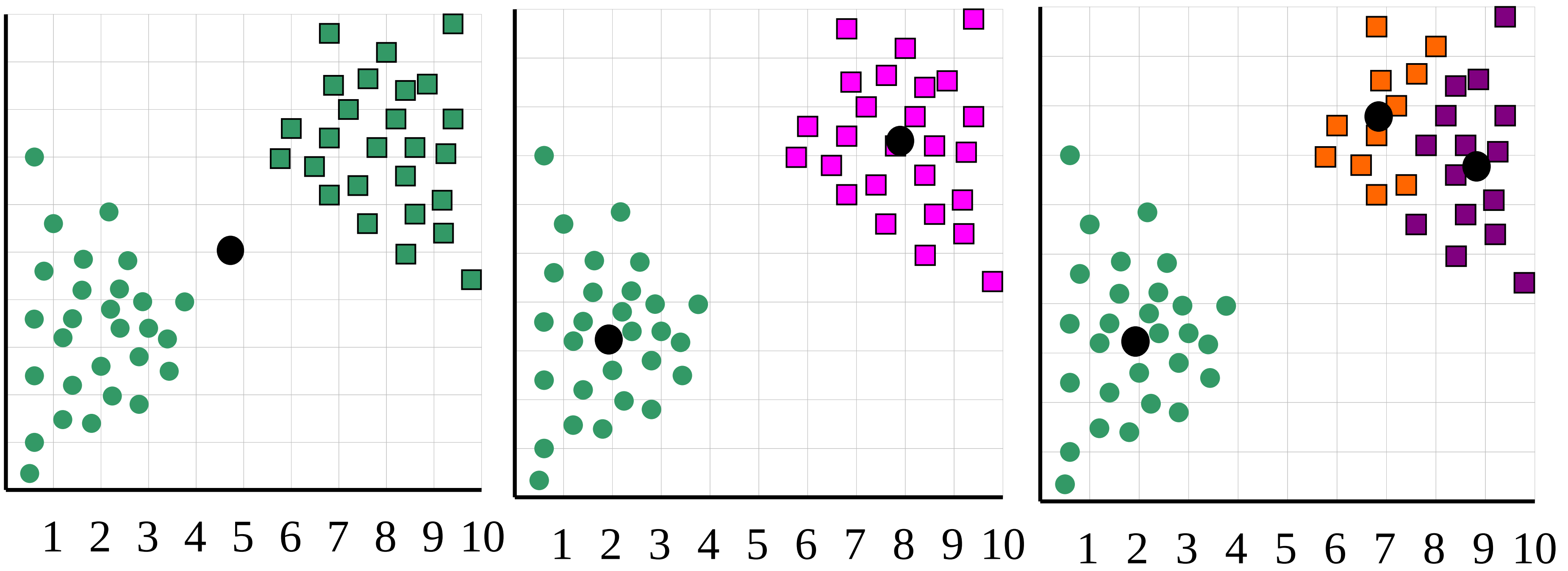
- *relatively efficient*: *O*(*tkn*), where *n* is # objects, *k* is # clusters, and *t* is # iterations. Normally, *k*, *t* << *n*.
- simple to code

socrative by MasteryConnect

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

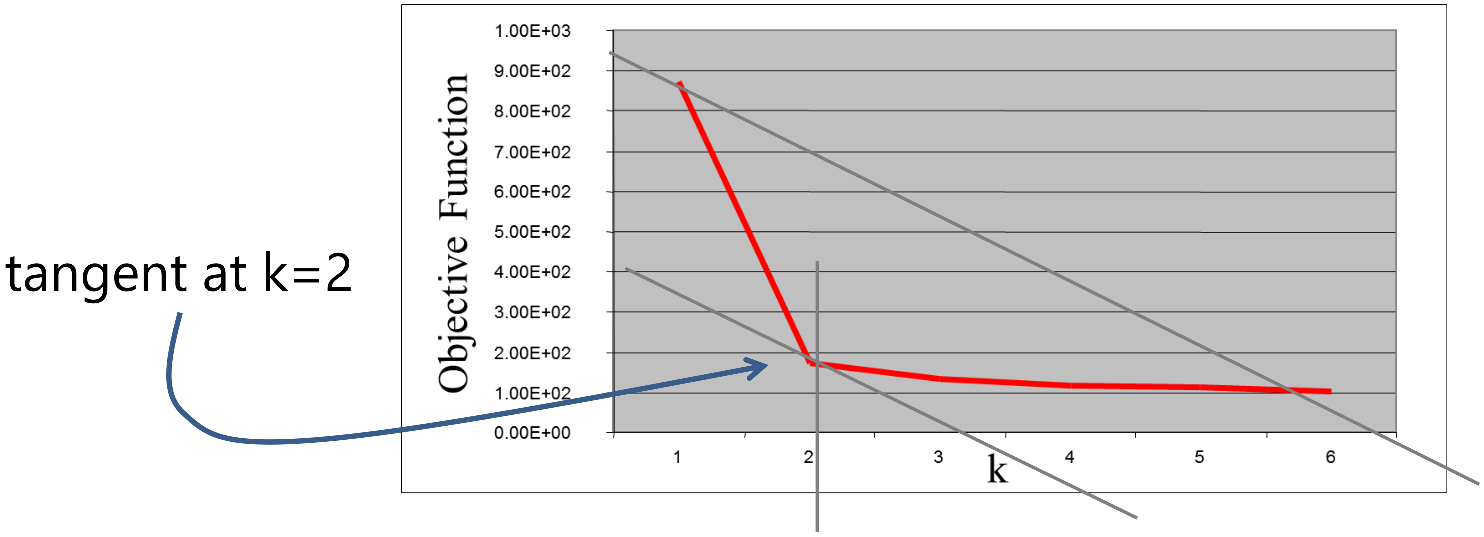
k=3, MSE=133.6

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

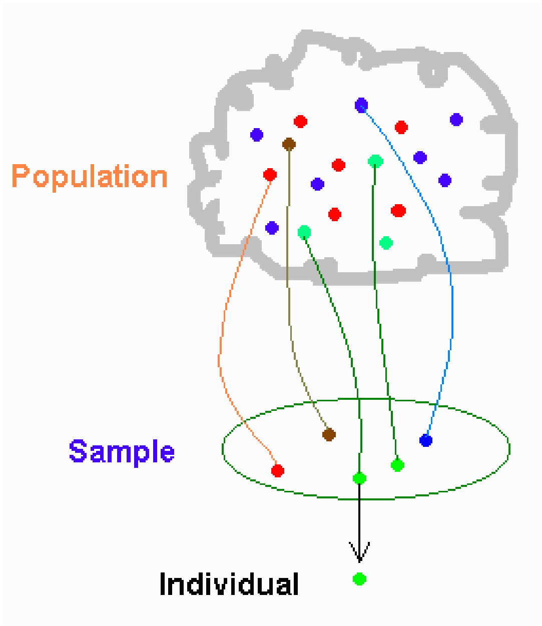


- the abrupt change at k = 2 is highly suggestive of two clusters
- this technique is known as “knee finding” or “elbow finding”

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



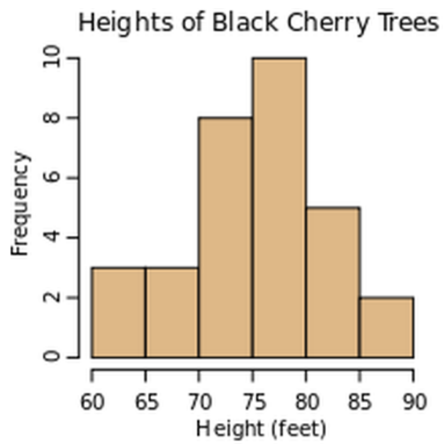
Simplest: random sampling

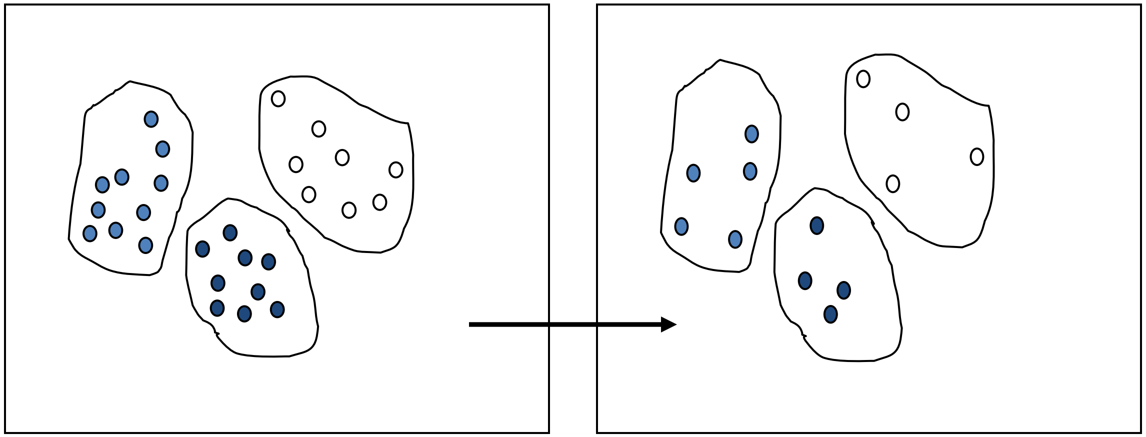
- pick sample points at random
- will work if the points are distributed uniformly
- this is usually not the case
- outliers will likely be missed
- so the sample will not be representative

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