

Clustering, K-Means, and K-Nearest Neighbors

CMSC 478

UMBC

Outline

Clustering basics

K-means: basic algorithm & extensions

Cluster evaluation

Non-parametric mode finding: density estimation

Graph & spectral clustering

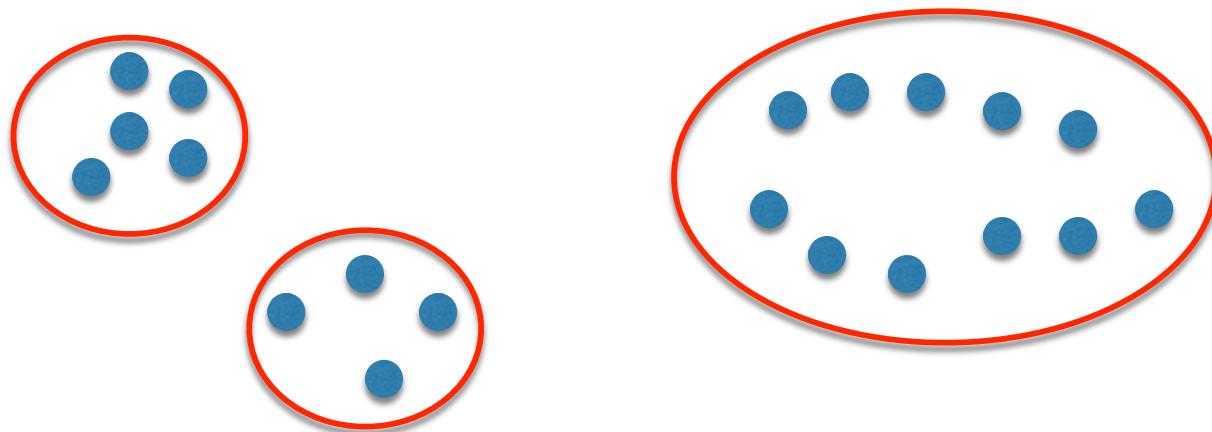
Hierarchical clustering

K-Nearest Neighbor

Clustering

Basic idea: group together **similar** instances

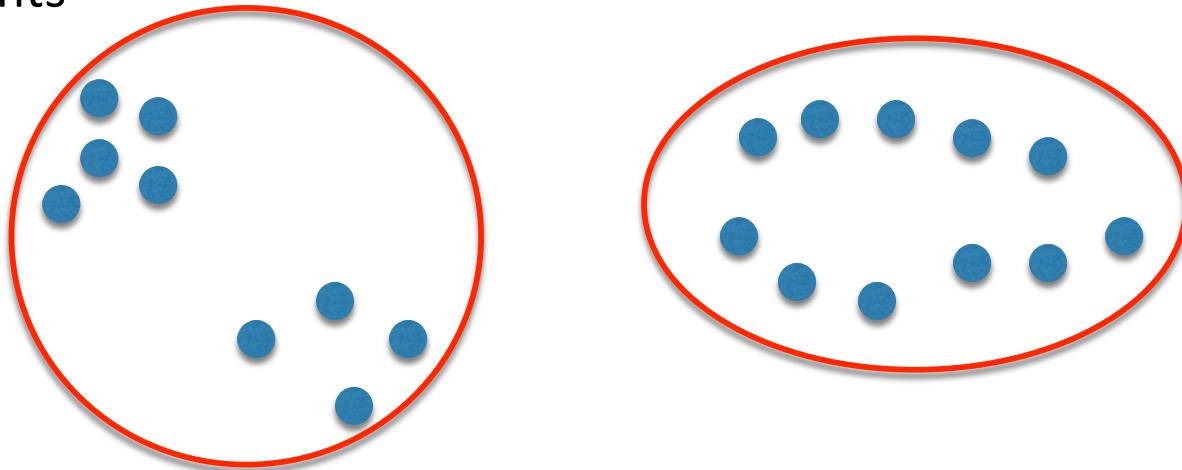
Example: 2D points



Clustering

Basic idea: group together **similar** instances

Example: 2D points



One option: small **Euclidean distance** (squared)

$$\text{dist}(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2^2$$

Clustering results are crucially dependent on the measure of **similarity** (or **distance**) between points to be clustered

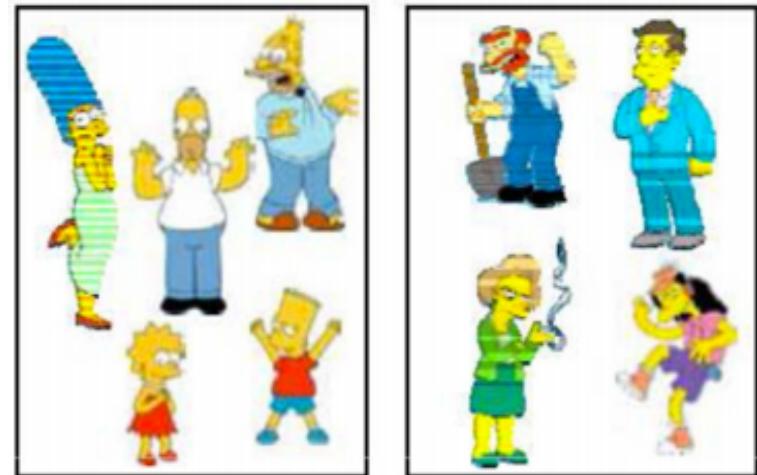
Clustering algorithms

Simple clustering: organize elements into k groups

K-means

Mean shift

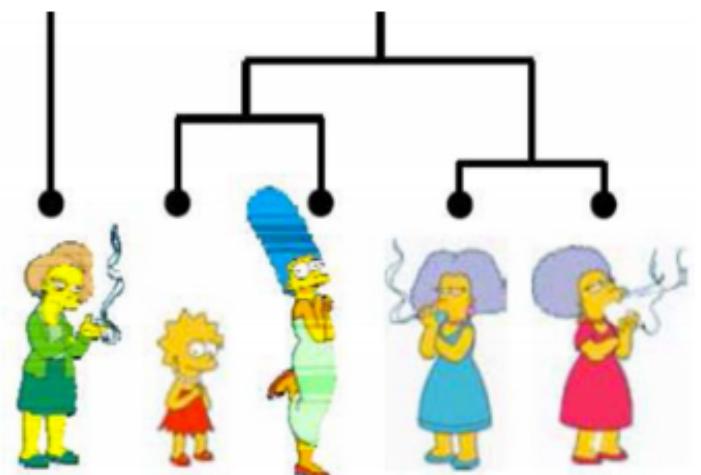
Spectral clustering



Hierarchical clustering: organize elements into a hierarchy

Bottom up - agglomerative

Top down - divisive



Clustering examples: Image Segmentation

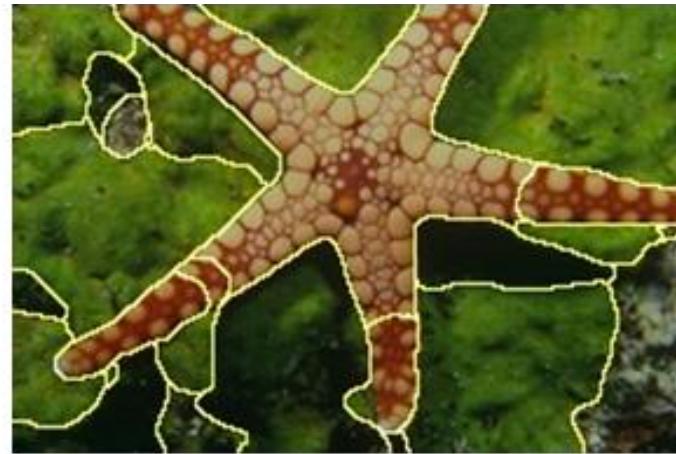


image credit: Berkeley segmentation benchmark

Clustering examples: News Feed

Google +Subhransu

News U.S. edition Modern Personalize

Top Stories

- Indiana
- Iran
- Nigeria
- Yemen
- Trevor Noah
- Germanwings
- Joni Mitchell
- Streaming media
- Google
- J. Paul Getty
- Springfield-Holyoke

Suggested for you

- World
- U.S.
- Business
- Technology
- Entertainment
- Sports
- Health
- Spotlight
- Science

Top Stories

Nuclear deal within reach, vows Iran and Russia
The Australian - 2 hours ago
Russia and Iran claimed a breakthrough in talks on a framework deal cutting back Tehran's nuclear program, but the US denied everything had been agreed as discussions were due to resume overnight.

Related Iran »

Religious Freedom Act: Are businesses becoming more socially activist? (+video)
Christian Science Monitor - 10 minutes ago
The companies castigating Indiana's RFRA law are not promoting liberal idealism over profits: Their response is a recognition that - at least when it comes to the issue of gay marriage - social activism is also good business.

ISIS' legacy in Tikrit: booby traps, IEDs and fear
CNN - 1 hour ago
Tikrit, Iraq (CNN) ISIS is gone, but the fear remains. As Iraqi forces, aided by Shiite militiamen, took control Wednesday of the northern city of Tikrit, they found vehicles laden with explosives and buildings that might be booby-trapped.

Germanwings Crash: Video May Show Plane's Final Moments
ABC News - 1 hour ago
Two magazines have reported details of a disturbing video taken from inside the doomed Germanwings plane moments before it crashed into the French Alps, but investigators have denied its existence.

Get Google News on the go.
Try the free app for your phone or tablet.

GET IT ON **Download on the**

Recent

ISIS Seizes Yarmouk Refugee Camp in Damascus, Syria: Witnesses
NBCNews.com - 24 minutes ago

Obama Praises Goodluck Jonathan For Conceding Elections
Forbes - 6 minutes ago

Oil rallies as Iran nuclear talks drag on, overshadowing supply concerns
Reuters - 6 minutes ago

Weather for Amherst, Massachusetts

Today	Thu	Fri	Sat
46° 28°	59° 45°	64° 47°	48° 30°

The Weather Channel - Weather Underground - AccuWeather

[Create account](#)

Clustering examples: Image Search

Google jaguars

Web News **Images** Videos Maps More Search tools

+Subhransu SafeSearch

Seattle Seahawks Players Cars Logo Baby Football Nfl

Cards showing clusters of images for "jaguars":

- Seattle Seahawks Players: Two images of football players.
- Cars: Two images of cars.
- Logo: Three images of the Jacksonville Jaguars logo.
- Baby: Two images of baby jaguars.
- Football: Two images of footballs.
- Nfl: Two images of NFL logos.

Three rows of jaguar images:

- Row 1: Five images of jaguars in various poses.
- Row 2: Six images of jaguars, including close-ups and a logo.
- Row 3: Seven images of jaguars in various environments.

Outline

Clustering basics

K-means: basic algorithm & extensions

Cluster evaluation

Non-parametric mode finding: density estimation

Graph & spectral clustering

Hierarchical clustering

K-Nearest Neighbor

Clustering using k-means

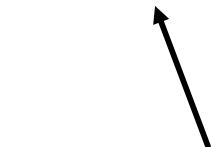
Data: D-dimensional observations (x_1, x_2, \dots, x_n)

Goal: partition the n observations into k ($\leq n$) sets

$S = \{S_1, S_2, \dots, S_k\}$ so as to minimize the within-cluster sum of squared distances

$$\arg \min_S \sum_{i=1}^k \sum_{x \in S_i} \|x - \mu_i\|^2$$

cluster center



Lloyd's algorithm for k-means

Initialize k centers by picking k points randomly among all the points

Repeat till convergence (or max iterations)

Assign each point to the nearest center (assignment step)

Estimate the mean of each group (update step)

$$\arg \min_{\mathbf{S}} \sum_{i=1}^k \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \mu_i\|^2$$

Properties of the Lloyd's algorithm

Guaranteed to converge in a finite number of iterations
objective decreases monotonically
local minima if the partitions don't change.
finitely many partitions → k-means algorithm must converge

Running time per iteration
Assignment step: $O(NKD)$
Computing cluster mean: $O(ND)$

Issues with the algorithm:
Worst case running time is super-polynomial in input size
No guarantees about global optimality
Optimal clustering even for 2 clusters is NP-hard [Aloise et al., 09]

k-means++ algorithm

A way to pick the good initial centers

Intuition: spread out the k initial cluster centers

The algorithm proceeds normally once the centers are initialized

[Arthur and Vassilvitskii'07] The approximation quality is $O(\log k)$ in expectation

k-means++ algorithm for initialization:

1. Choose one center uniformly at random among all the points
2. For each point \mathbf{x} , compute $D(\mathbf{x})$, the distance between \mathbf{x} and the nearest center that has already been chosen
3. Choose one new data point at random as a new center, using a weighted probability distribution where a point \mathbf{x} is chosen with a probability proportional to $D(\mathbf{x})^2$
4. Repeat Steps 2 and 3 until k centers have been chosen

Fast kmeans

- Intuition: If a data point is close to center i and far from center j, and center j has not moved much since the last iteration, we don't need to recalculate the distance for center j.
- Use triangle inequality to prune the number of distances that you should recalculate.

k-means for image segmentation



K=2



Grouping pixels based
on intensity similarity

K=3



feature space: intensity value (1D)



Outline

Clustering basics

K-means: basic algorithm & extensions

Cluster evaluation

Non-parametric mode finding: density estimation

Graph & spectral clustering

Hierarchical clustering

K-Nearest Neighbor

Outline

Clustering basics

K-means: basic algorithm & extensions

Cluster evaluation

Non-parametric mode finding: density estimation

Graph & spectral clustering

Hierarchical clustering

K-Nearest Neighbor

Clustering using density estimation

One issue with k-means is that it is sometimes hard to pick k

The [mean shift algorithm](#) seeks **modes** or **local maxima** of density in the feature space

Mean shift automatically determines the number of clusters

$$K(\mathbf{x}) = \frac{1}{Z} \sum_i \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}_i\|^2}{h}\right)$$

Kernel density estimator



Small h implies more modes (bumpy distribution)

Mean shift algorithm

For each point x_i :

 find m_i , the amount to
 shift each point x_i to its
 centroid

return $\{m_i\}$

Mean shift algorithm

For each point x_i :

set $m_i = x_i$

while not converged:

 compute *weighted average of neighboring
 point*

return $\{m_i\}$

Mean shift algorithm

For each point x_i :

set $m_i = x_i$

while not converged:

compute $m_i = \frac{\sum_{x_j \in N(x_i)} x_j K(m_i, x_j)}{\sum_{x_j \in N(x_i)} K(m_i, x_j)}$

return $\{m_i\}$

Neighbors of x_i



*self-clustering to based on
kernel (similarity to other
points)*

Pros:

Does not assume shape on
clusters

Generic technique

Finds multiple modes

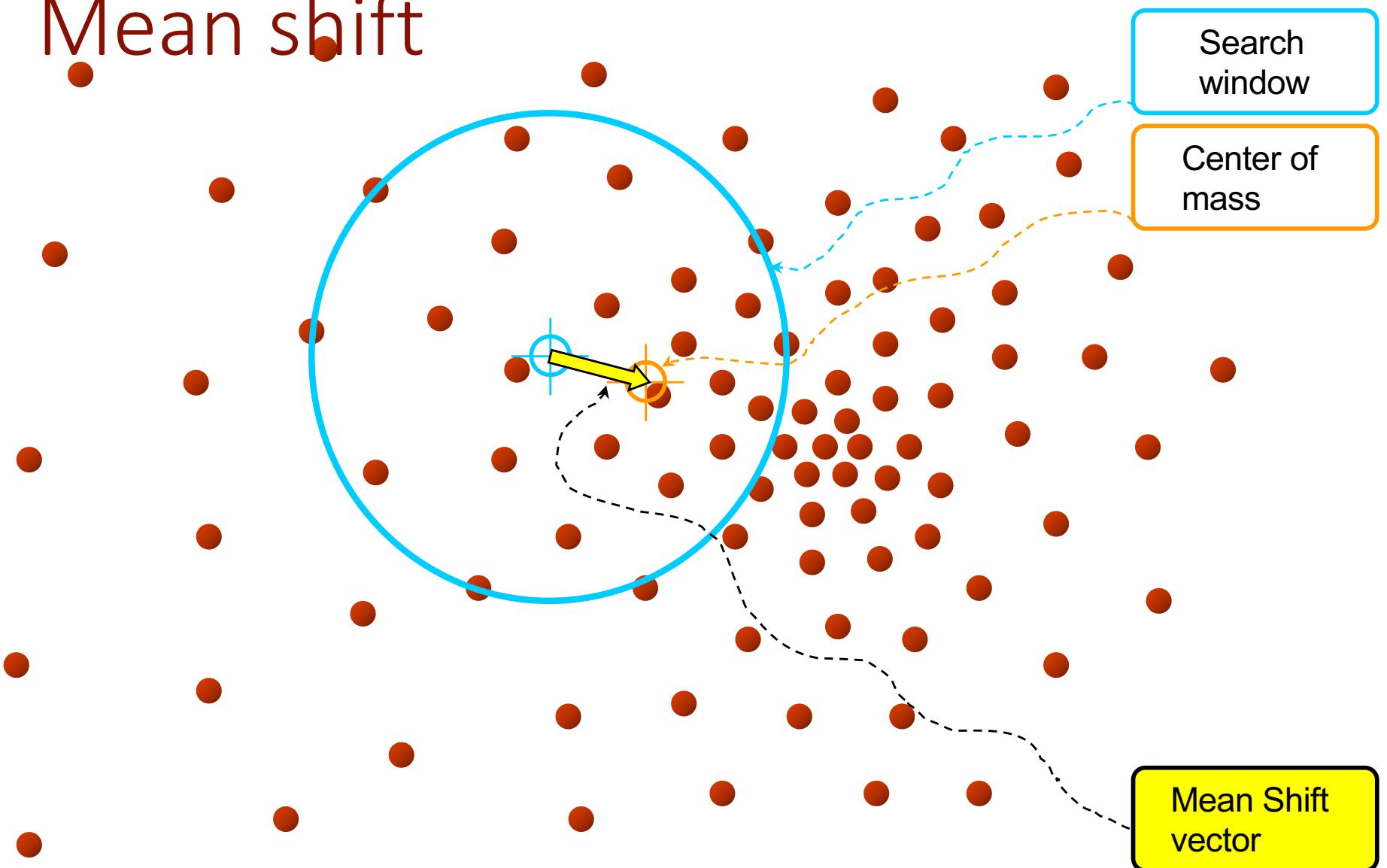
Parallelizable

Cons:

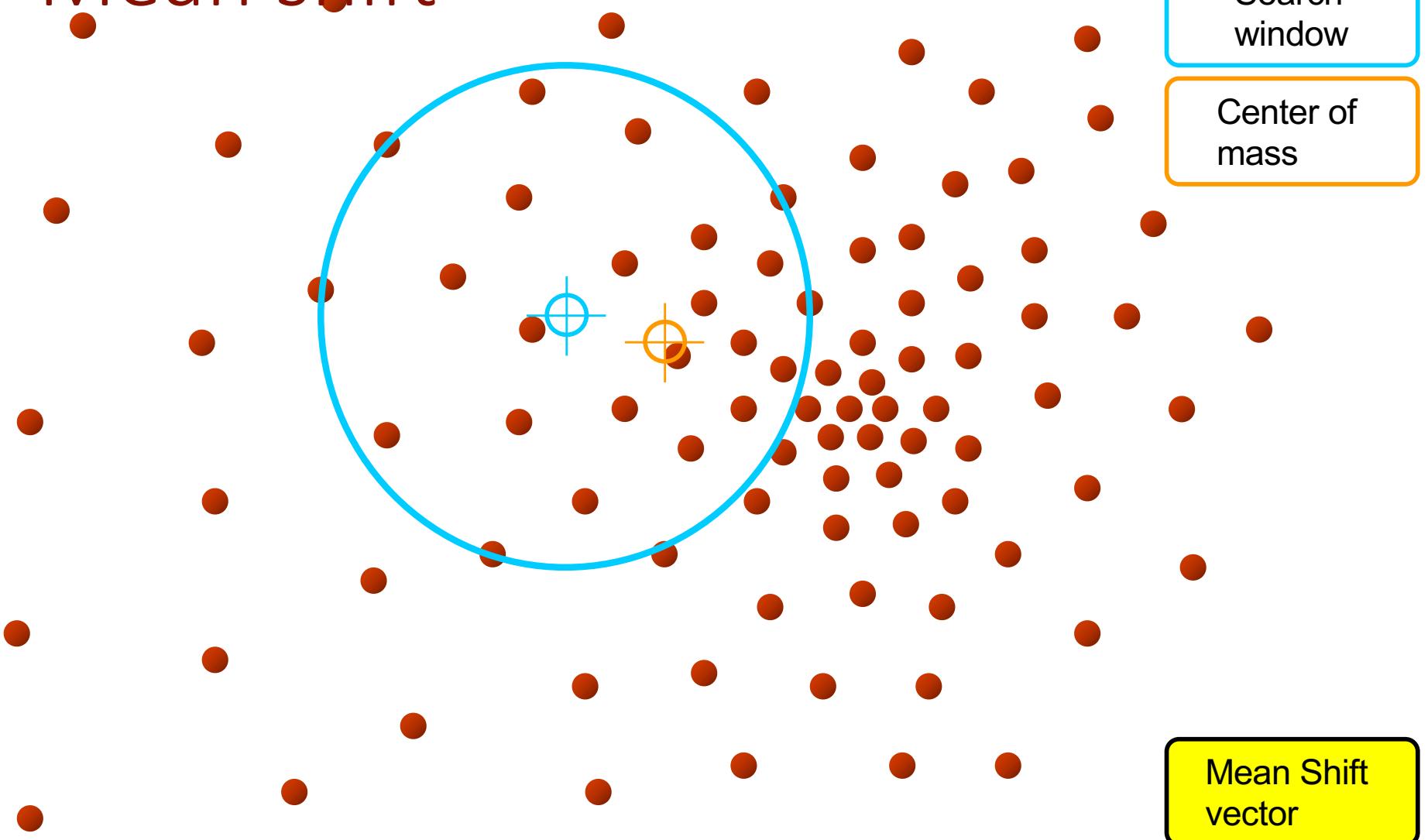
Slow: $O(DN^2)$ per iteration

Does not work well for
high-dimensional
features

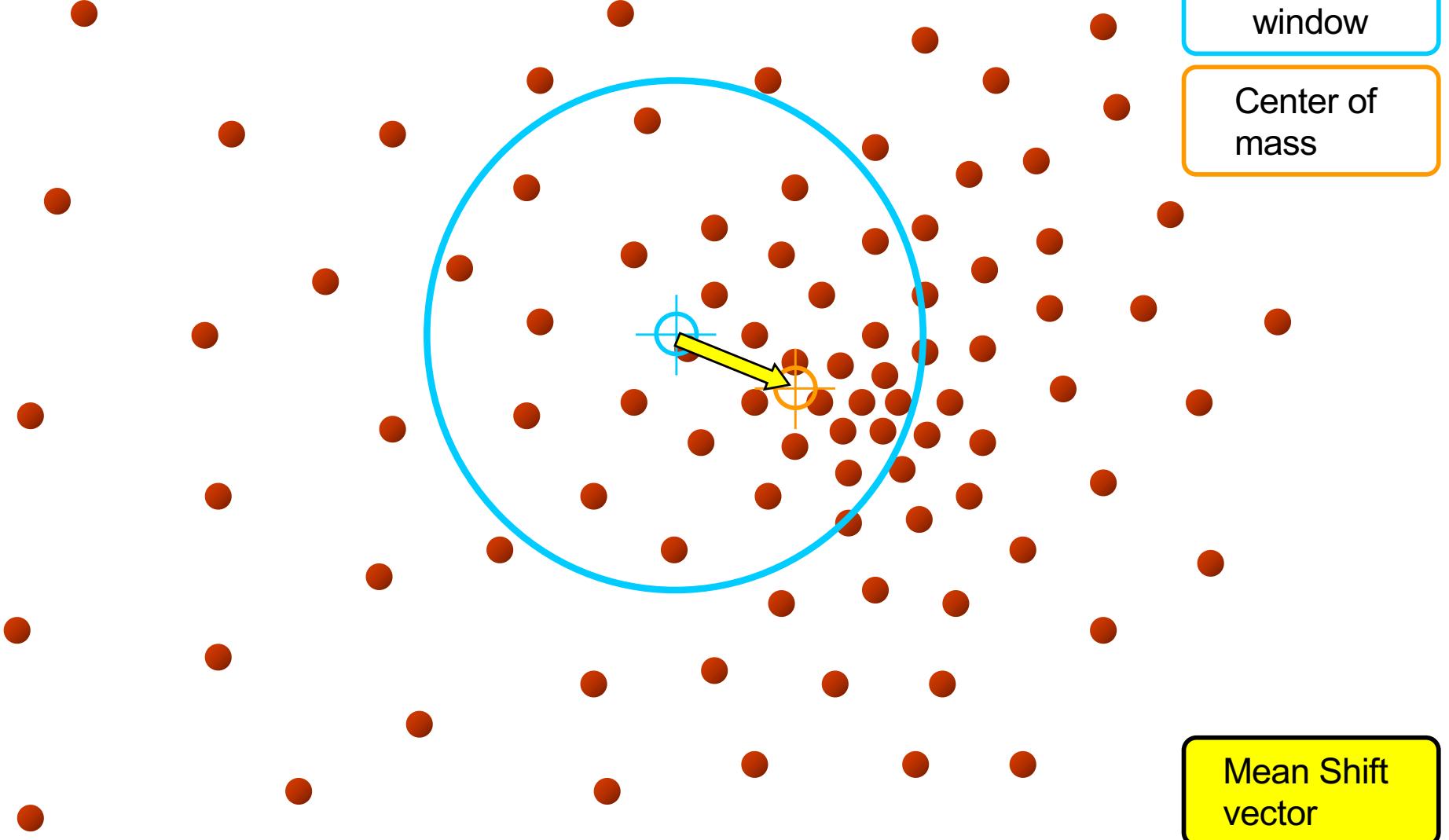
Mean shift



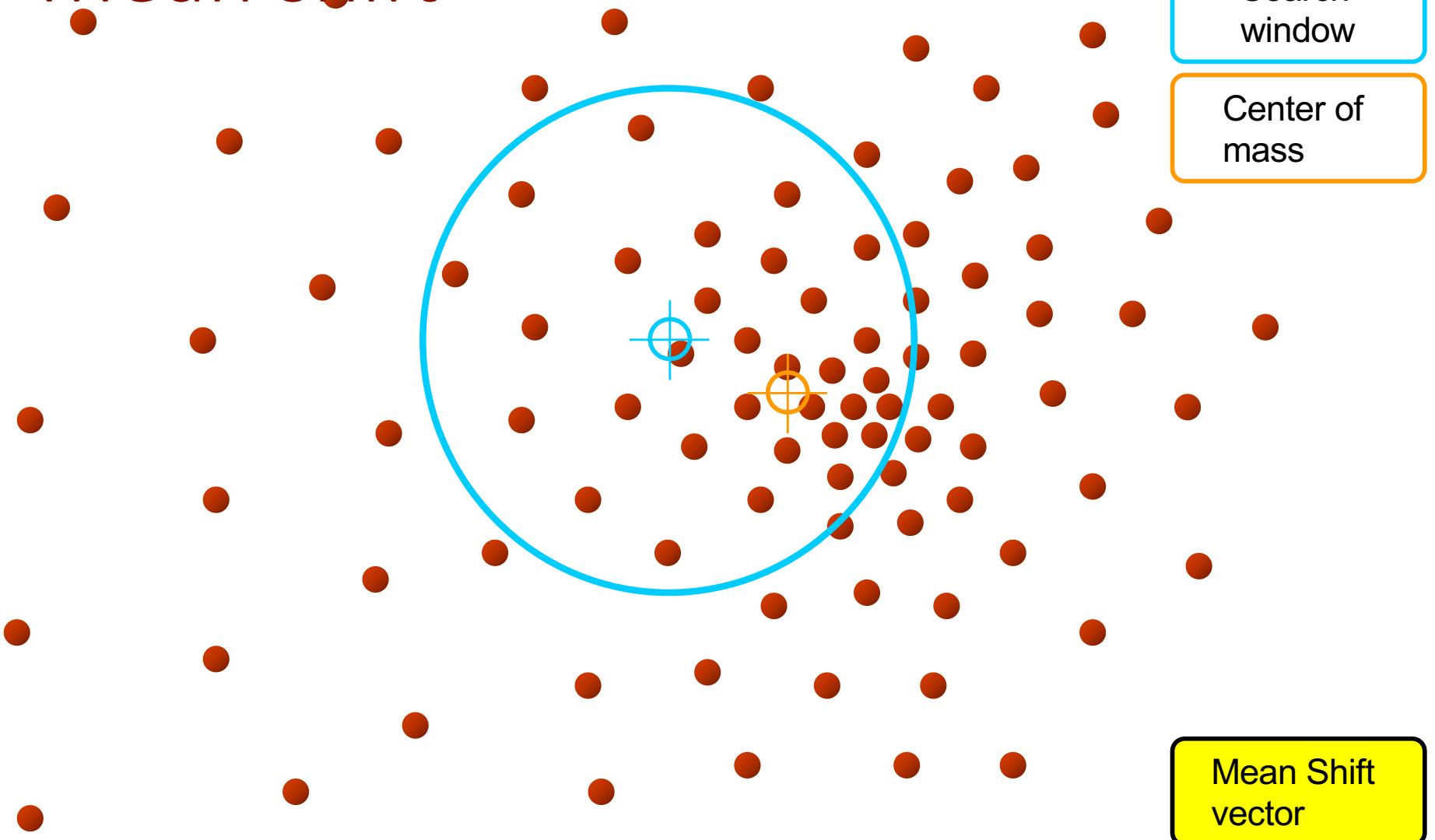
Mean shift



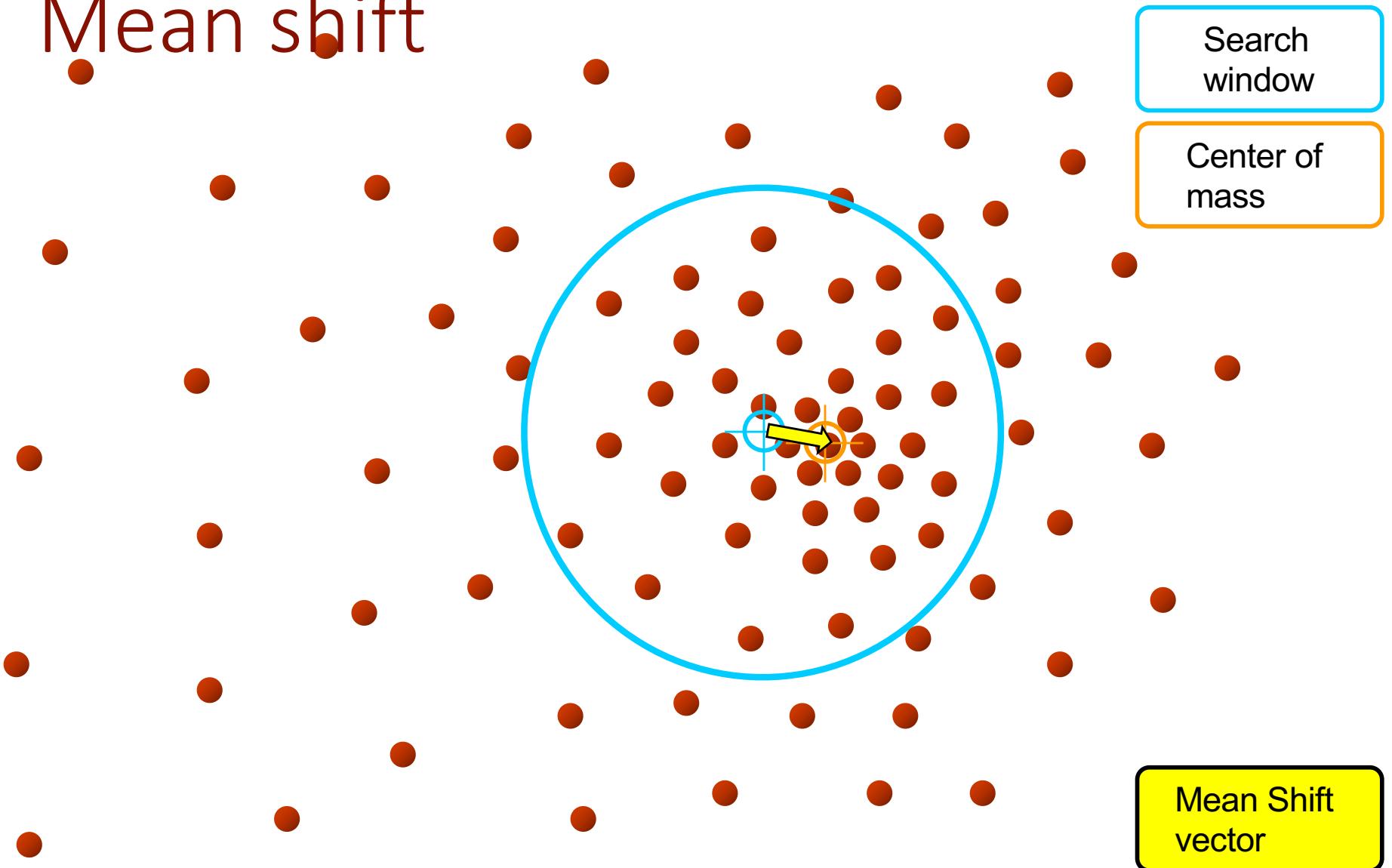
Mean shift



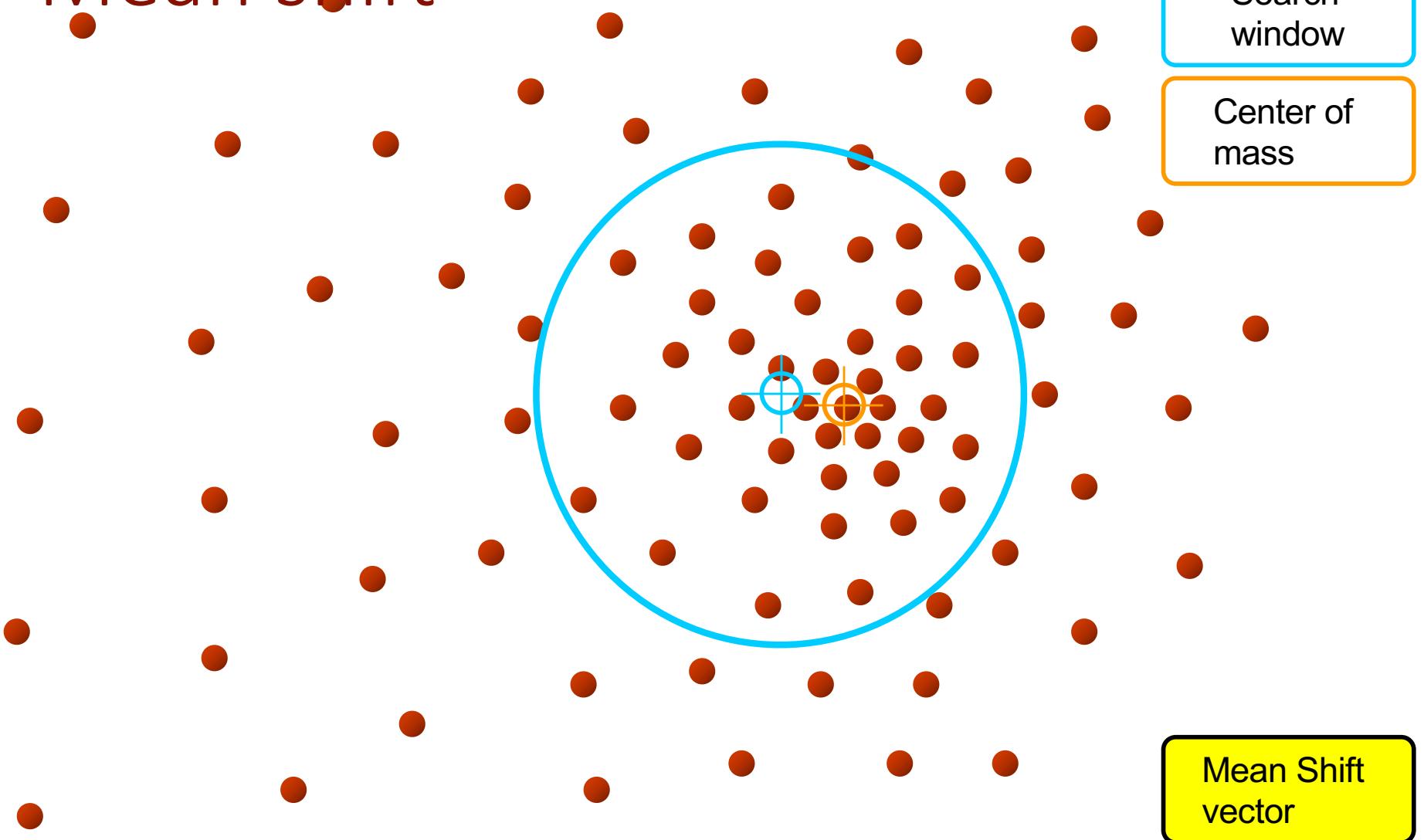
Mean shift



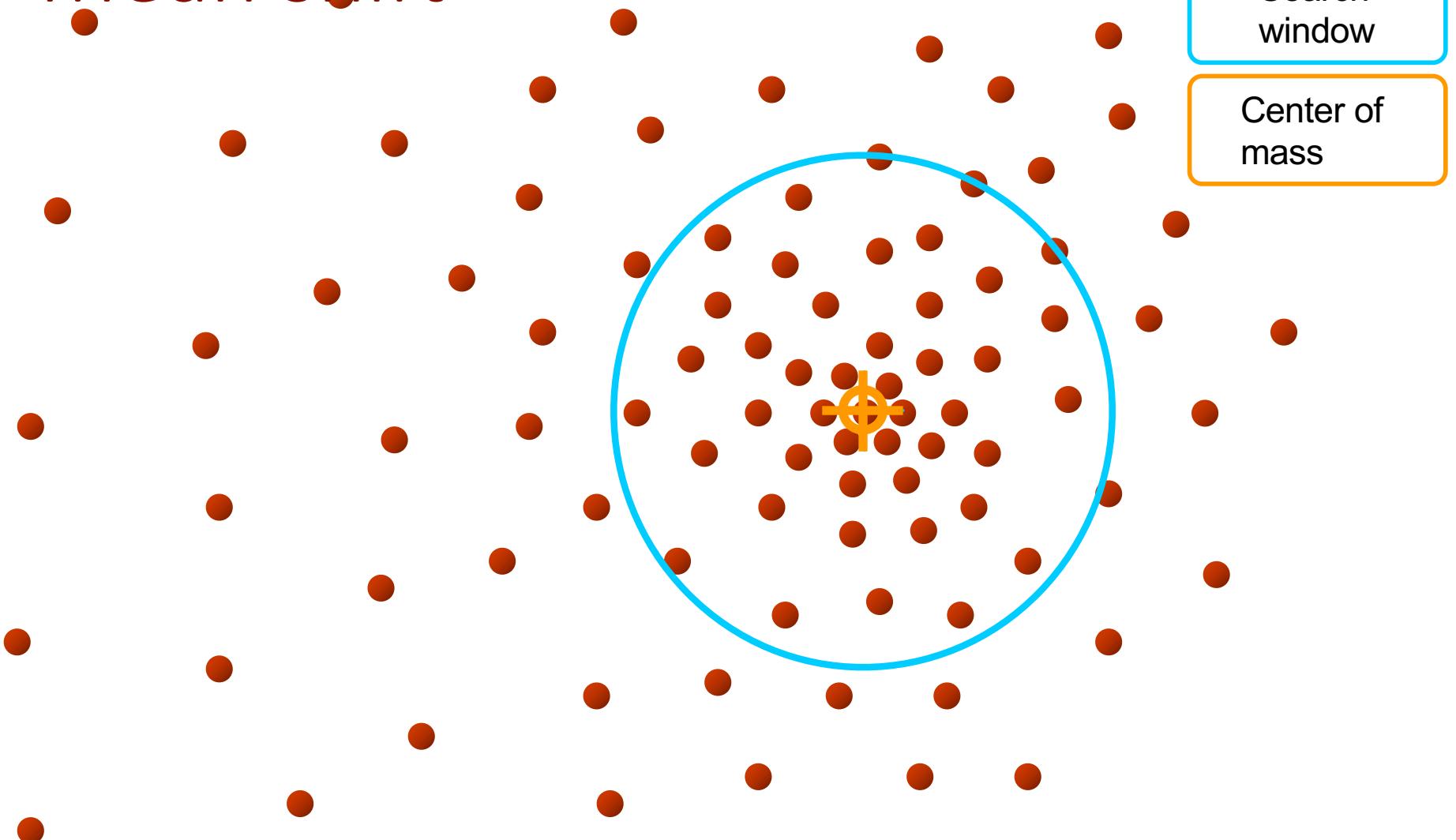
Mean shift



Mean shift

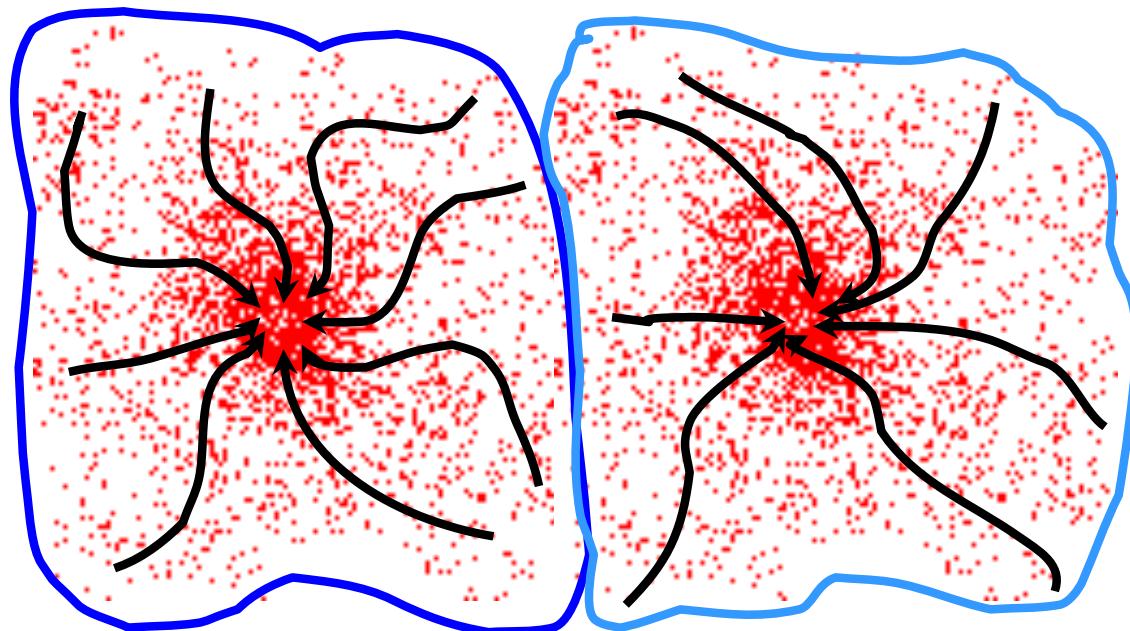


Mean shift



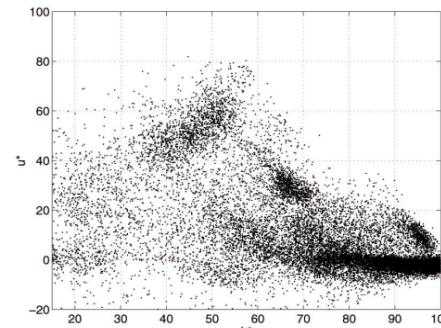
Mean shift clustering

- ◆ Cluster all data points in the **attraction basin** of a mode
- ◆ **Attraction basin** is the region for which all trajectories lead to the same mode — correspond to clusters

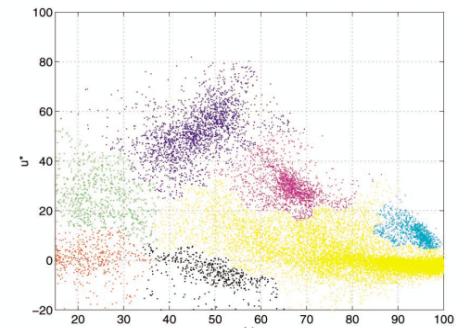


Mean shift for image segmentation

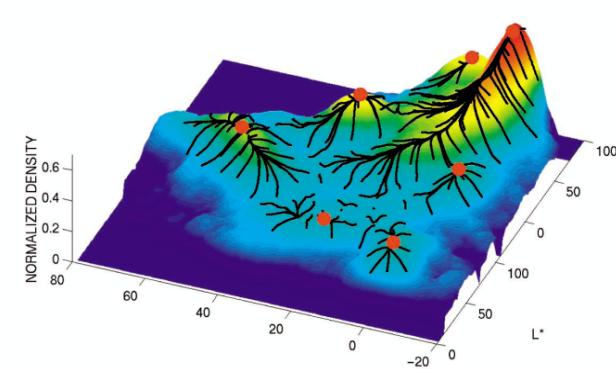
- ◆ Feature: $L^*u^*v^*$ color values
- ◆ Initialize windows at individual feature points
- ◆ Perform mean shift for each window until convergence
- ◆ Merge windows that end up near the same “peak” or mode



(a)



(b)



Mean shift clustering results



<http://www.caip.rutgers.edu/~comanici/MSPAMI/msPamiResults.html>

Outline

Clustering basics

K-means: basic algorithm & extensions

Cluster evaluation

Non-parametric mode finding: density estimation

Graph & spectral clustering

Hierarchical clustering

K-Nearest Neighbor

Hierarchical clustering

Agglomerative: a “bottom up” approach where elements start as individual clusters and clusters are merged as one moves up the hierarchy

Divisive: a “top down” approach where elements start as a single cluster and clusters are split as one moves down the hierarchy

Agglomerative clustering

Agglomerative clustering:

- First merge very similar instances

- Incrementally build larger clusters out of smaller clusters

Algorithm:

- Maintain a set of clusters

- Initially, each instance in its own cluster

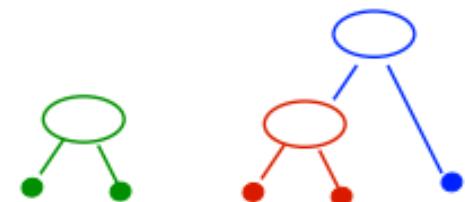
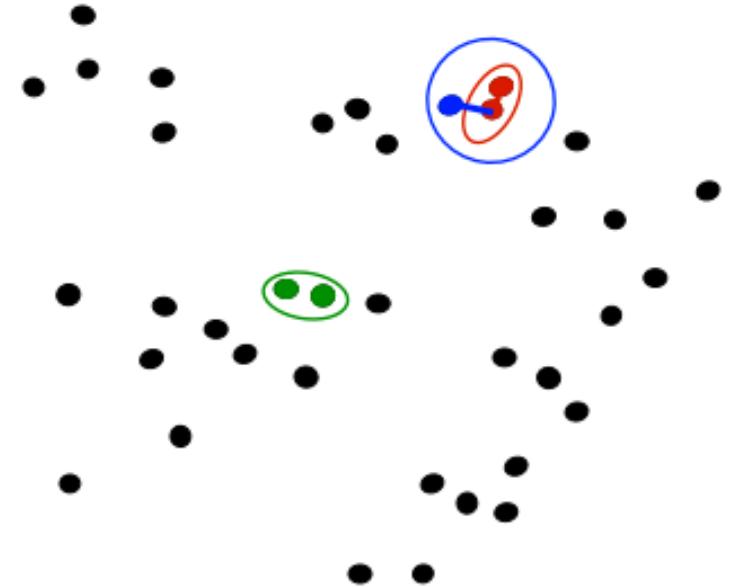
Repeat:

- Pick the two “closest” clusters

- Merge them into a new cluster

- Stop when there’s only one cluster left

Produces not one clustering, but a family of clusterings represented by a [dendrogram](#)



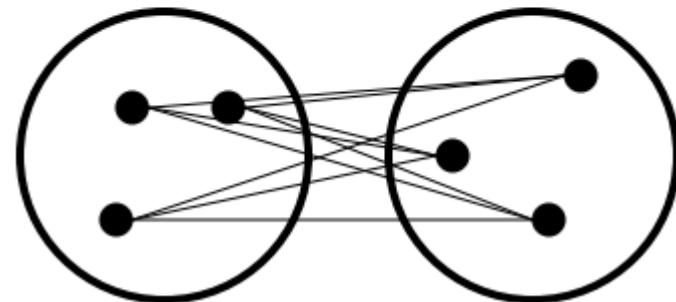
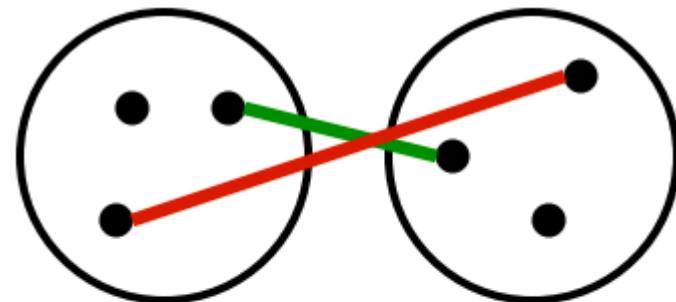
Agglomerative clustering

How should we define “closest” for clusters with multiple elements?

Closest pair: single-link clustering

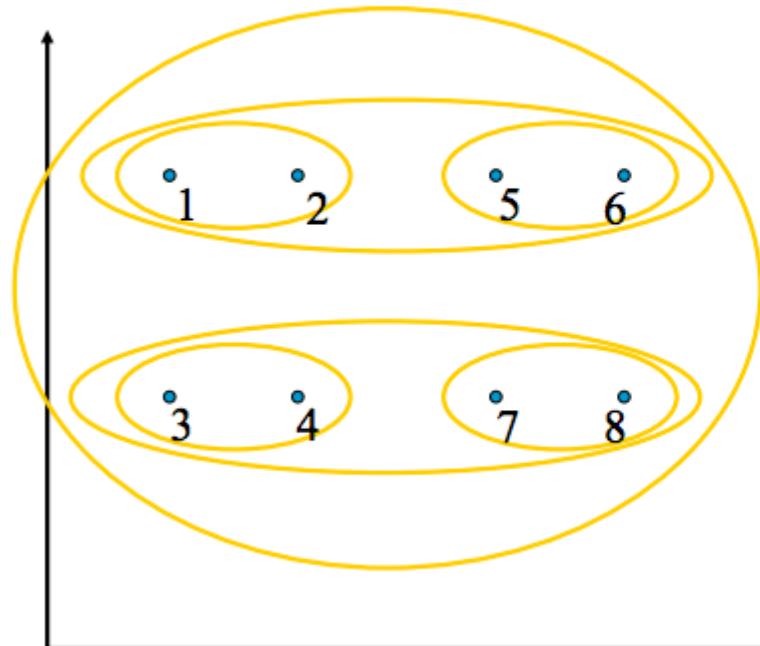
Farthest pair: complete-link clustering

Average of all pairs

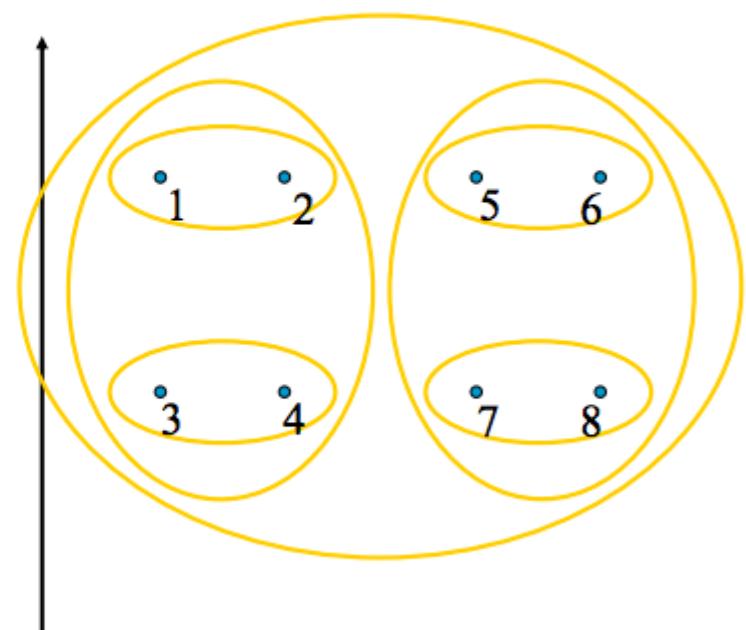


Agglomerative clustering

Closest pair
(single-link clustering)



Farthest pair
(complete-link clustering)



[Pictures from Thorsten Joachims]

Summary

- ◆ Clustering is an example of unsupervised learning
- ◆ Partitions or hierarchy
- ◆ Several partitioning algorithms:
 - k-means: simple, efficient and often works in practice
 - k-means++ for better initialization
 - mean shift: modes of density
 - slow but suited for problems with unknown number of clusters with varying shapes and sizes
 - spectral clustering: clustering as graph partitions
 - solve $(\mathbf{D} - \mathbf{W})x = \lambda \mathbf{D}x$ followed by k-means
- ◆ Hierarchical clustering methods:
 - Agglomerative or divisive
 - single-link, complete-link and average-link

Outline

Clustering basics

K-means: basic algorithm & extensions

Cluster evaluation

Non-parametric mode finding: density estimation

Graph & spectral clustering

Hierarchical clustering

K-Nearest Neighbor

Nearest neighbor classifier

Will Alice like the movie?

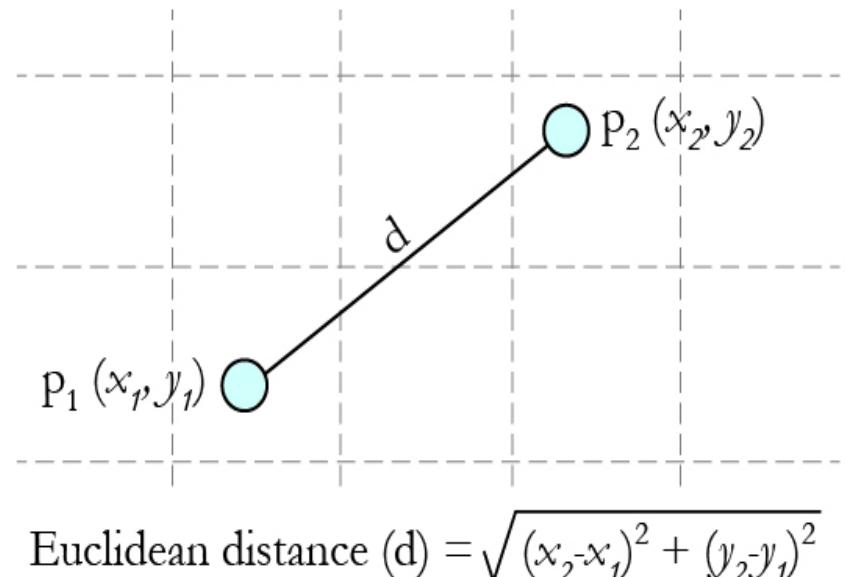
Alice and James are **similar**

James likes the movie →

Alice must/might also like the movie

Represent data as vectors of feature values

Find closest (Euclidean norm) points



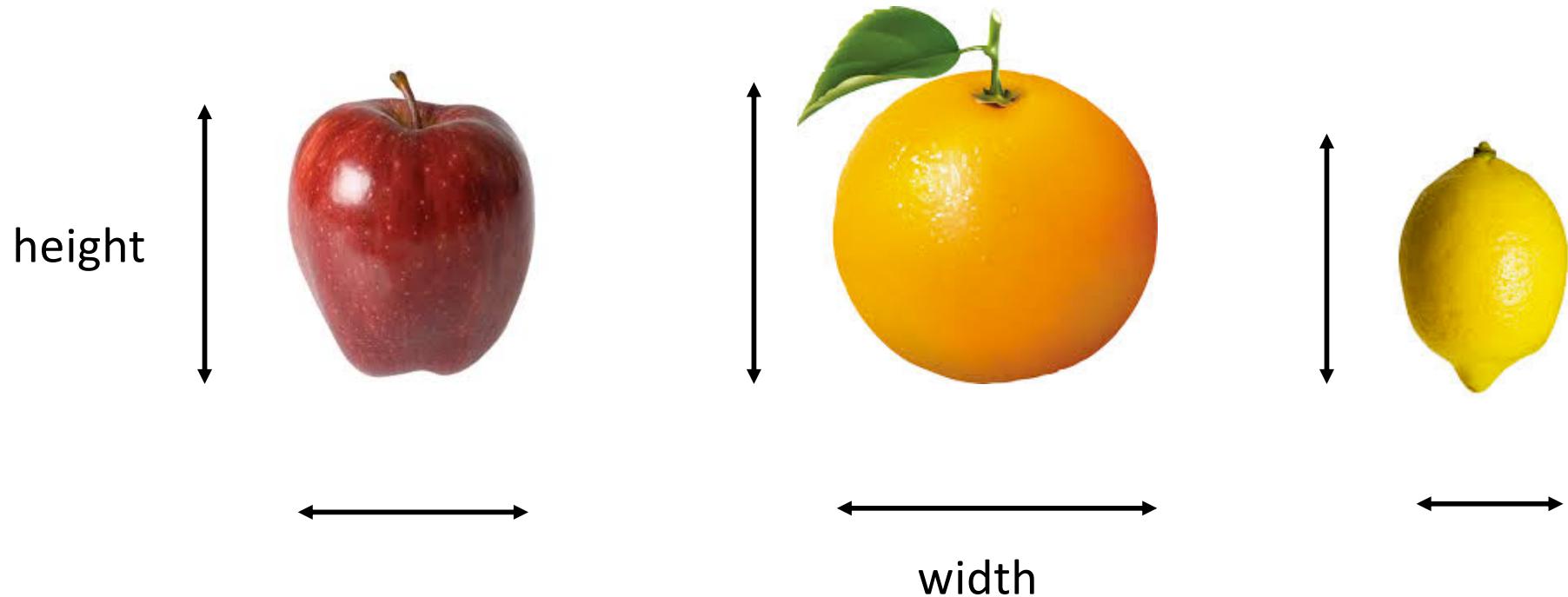
Nearest neighbor classifier

Training data is in the form of $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)$

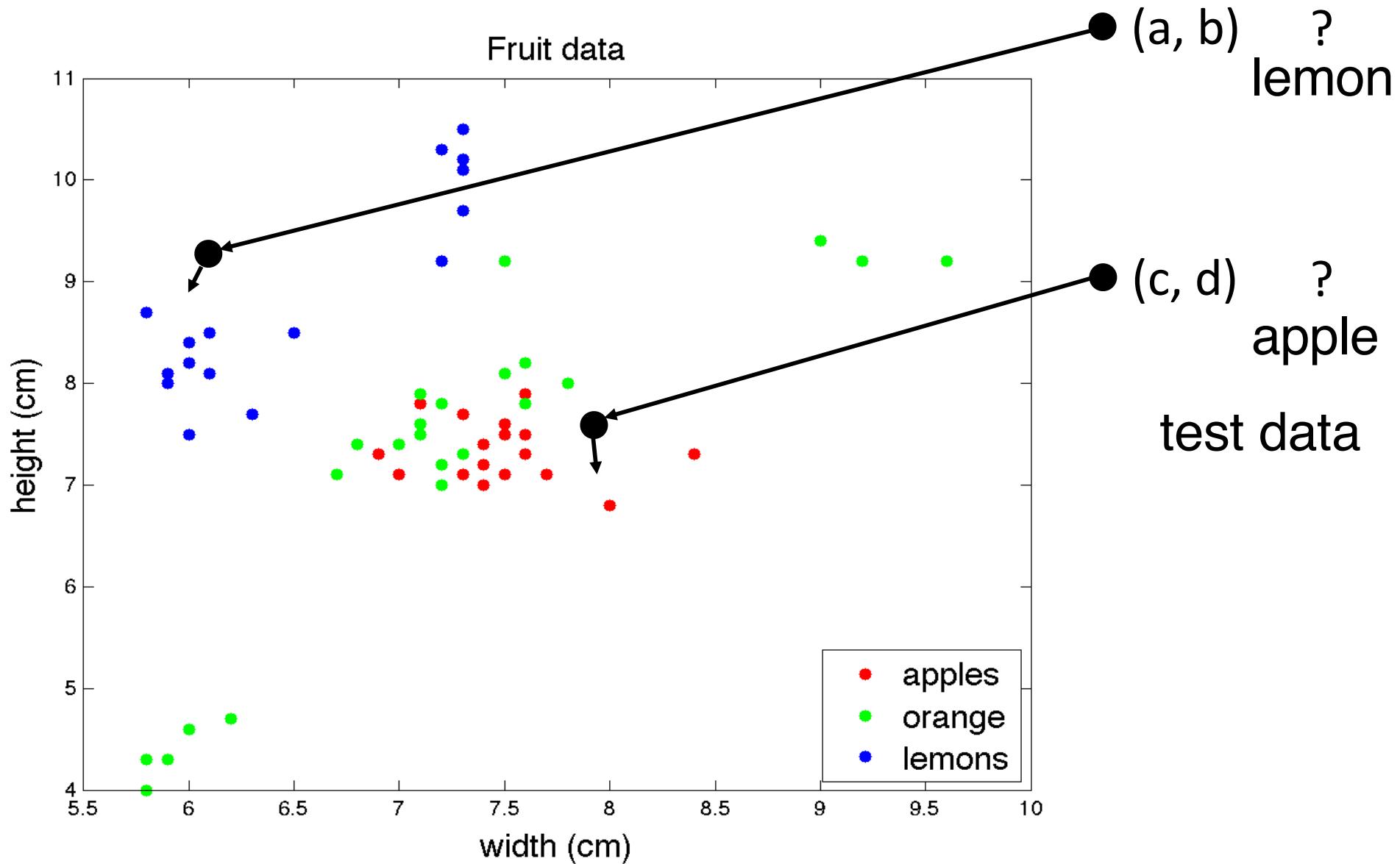
Fruit data:

label: {apples, oranges, lemons}

attributes: {width, height}

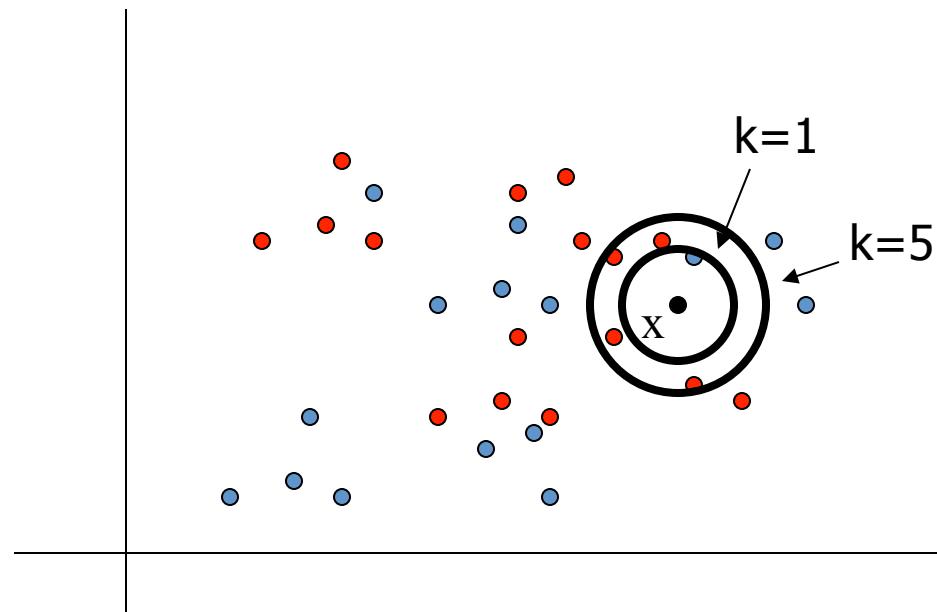


Nearest neighbor classifier



K-Nearest Neighbor Methods

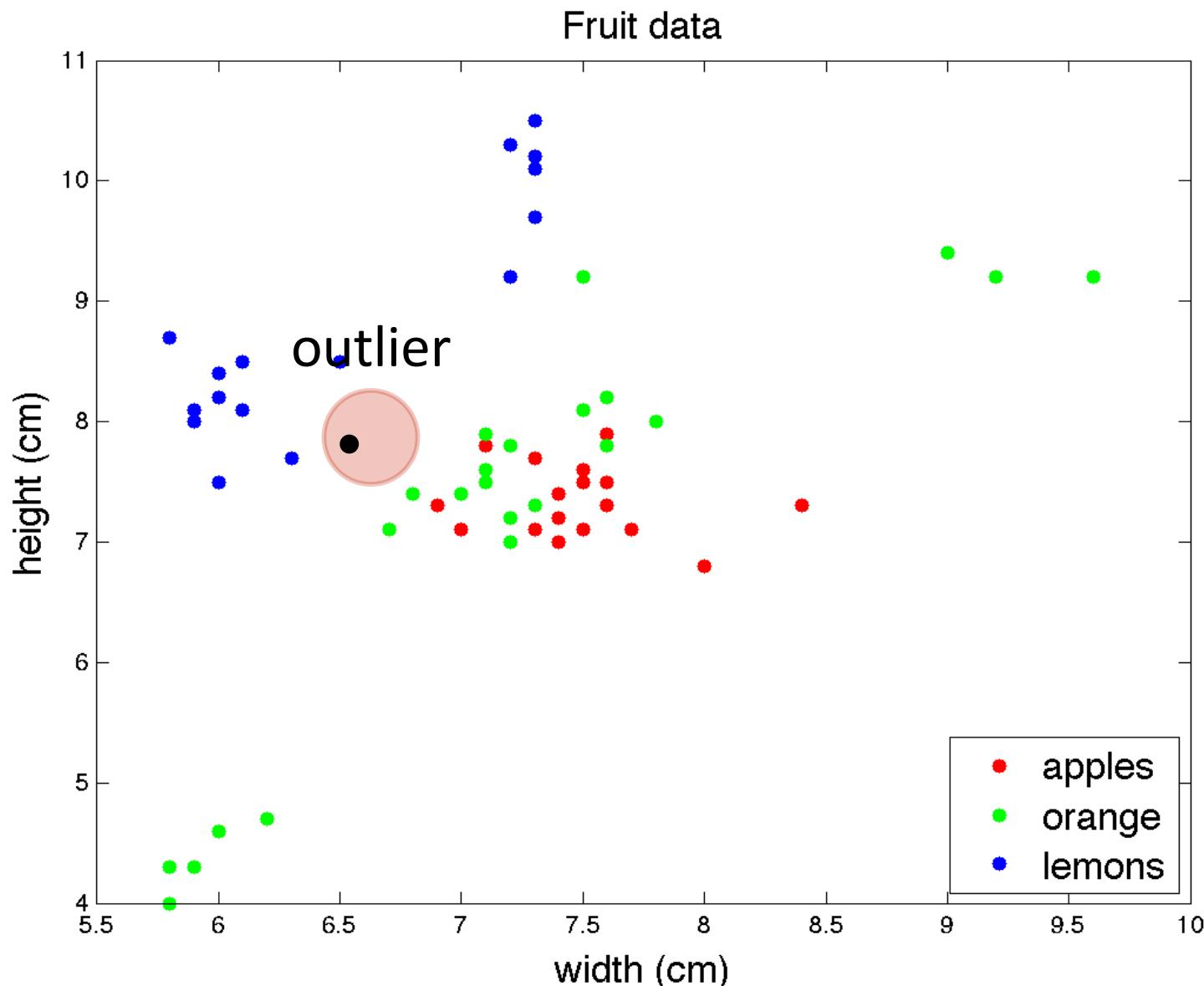
- To classify a new input vector x , examine the k -closest training data points to x and assign the object to the most frequently occurring class



common values for k : 3, 5

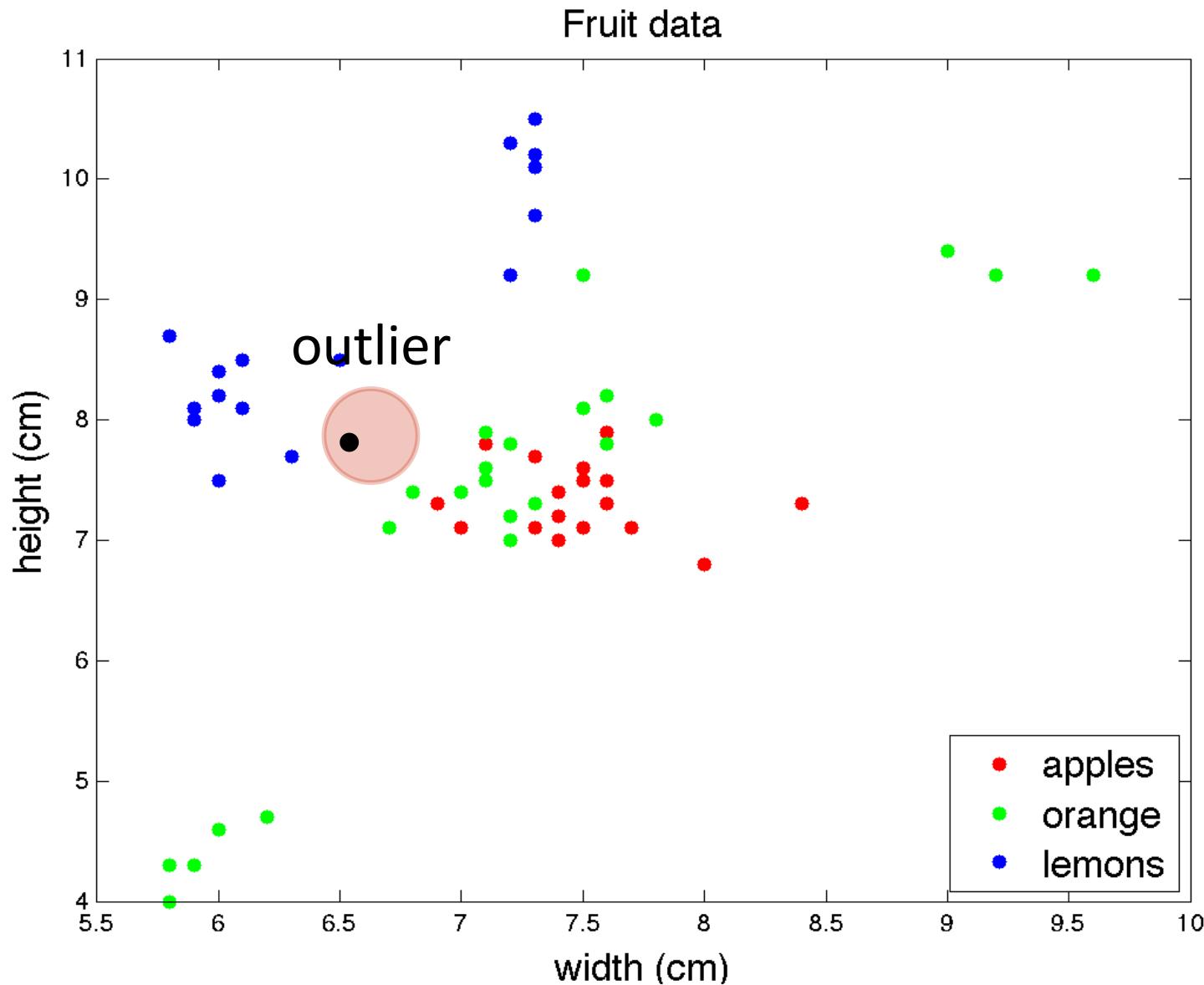
k-Nearest neighbor classifier

Take majority vote among the k nearest neighbors



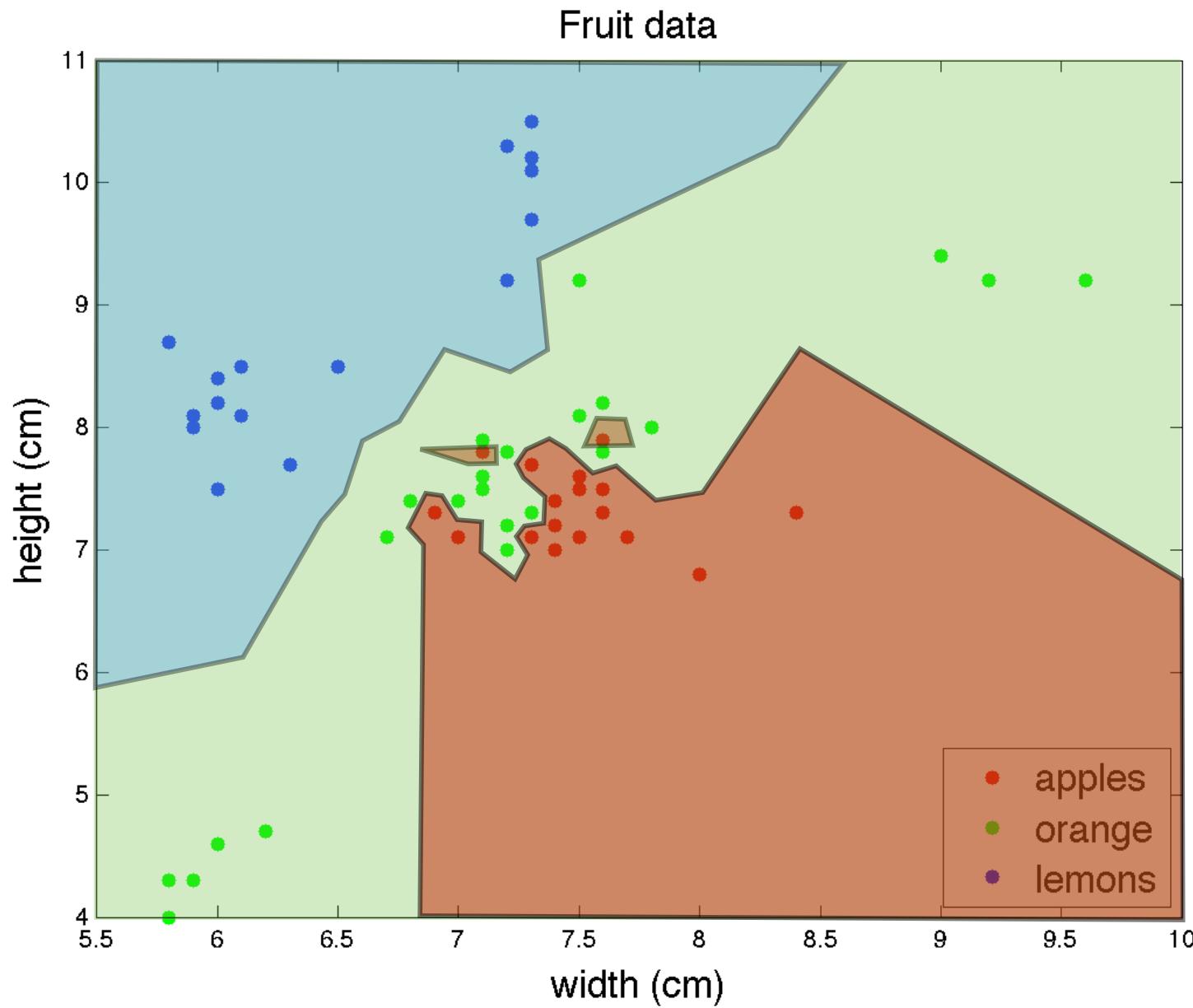
k-Nearest neighbor classifier

Take majority vote among the k nearest neighbors



What is
the
effect
of k?

Decision boundaries: 1NN



Inductive bias of the kNN classifier

Choice of features

We are assuming that all features are equally important

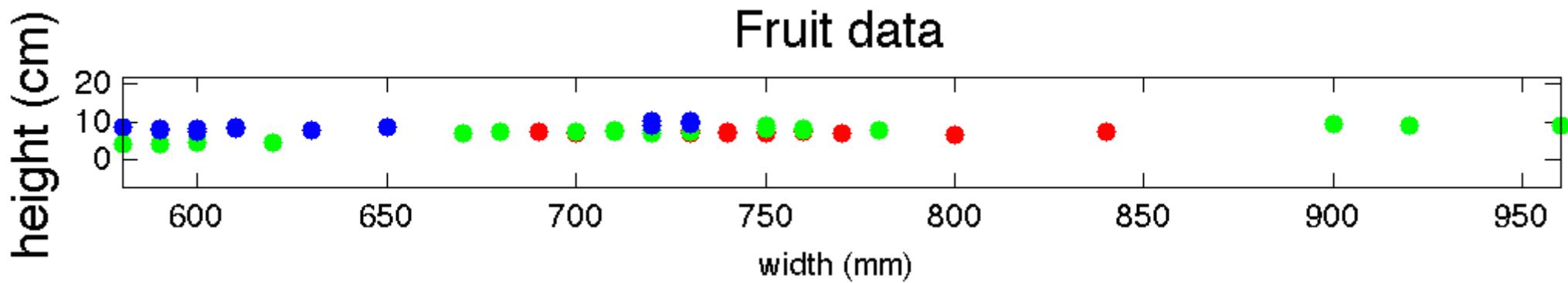
What happens if we scale one of the features by a factor of 100?

Choice of distance function

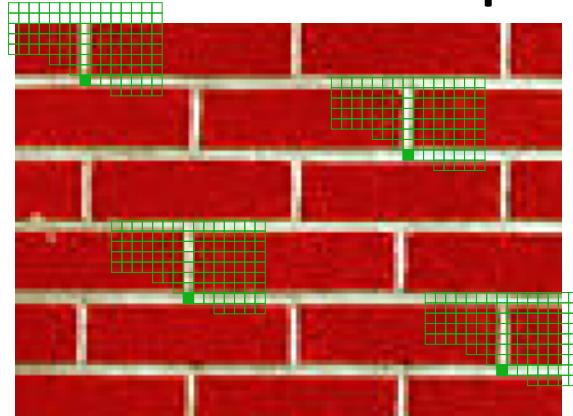
Euclidean, cosine similarity (angle), Gaussian, etc ...

Should the coordinates be independent?

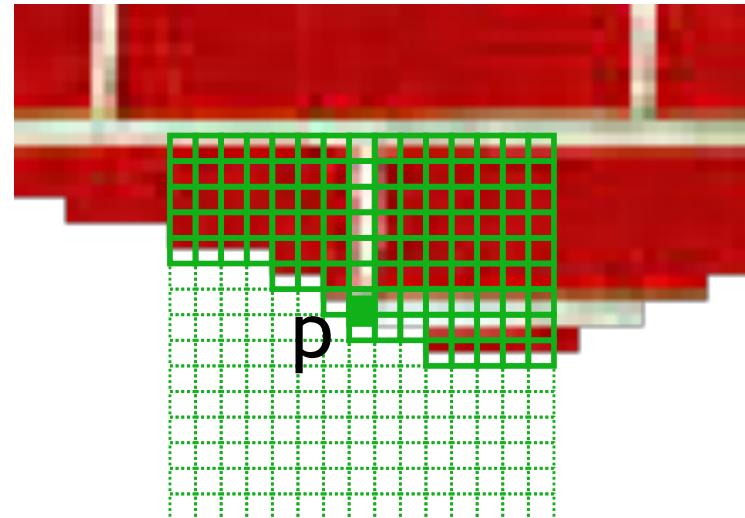
Choice of k



An example: Synthesizing one pixel



input image



synthesized image

What is $P(x|\text{neighborhood of pixels around } x)$

Find all the windows in the image that match the neighborhood

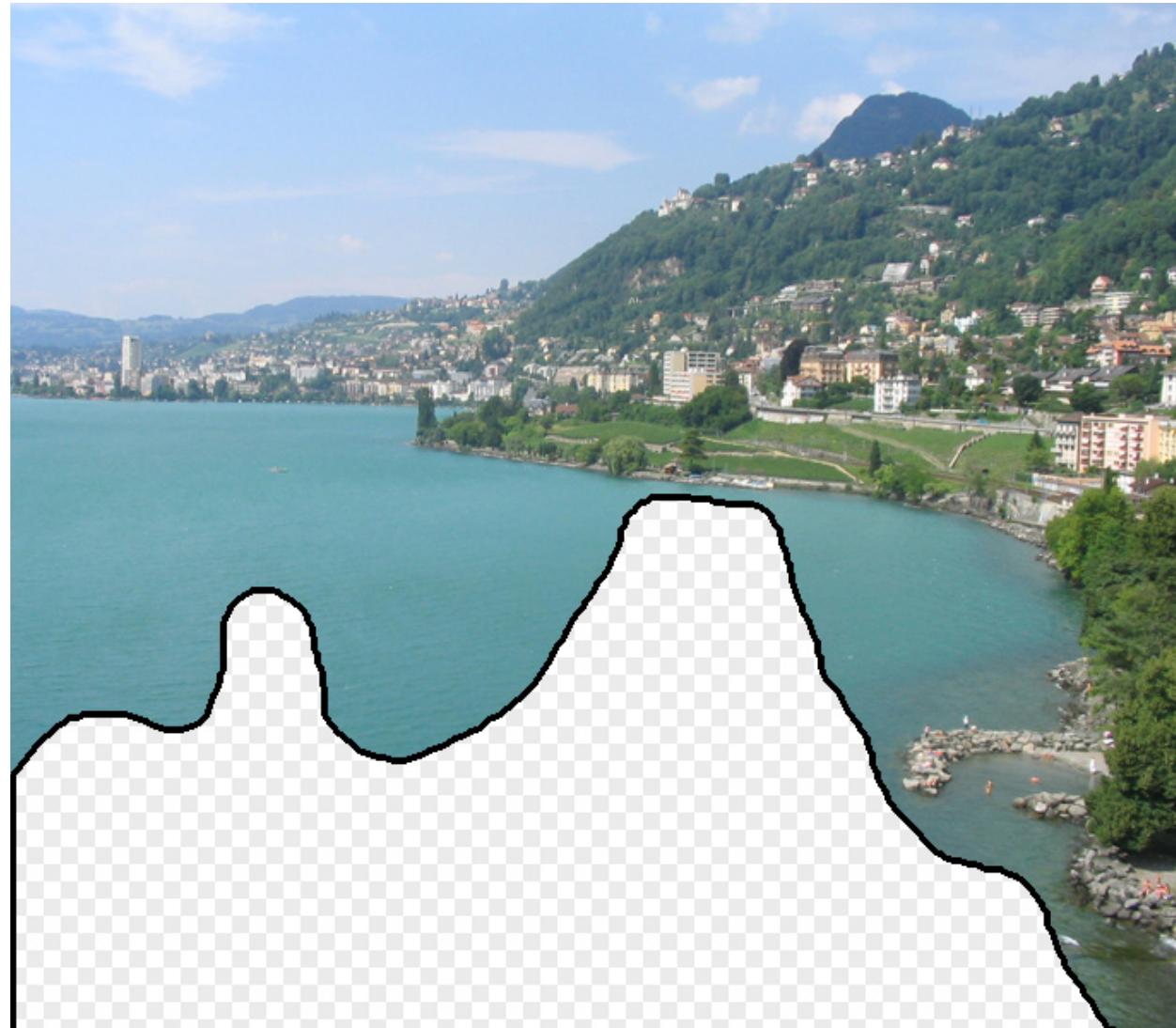
To synthesize x

pick one matching window at random

assign x to be the center pixel of that window

An **exact** match might not be present, so find the **best** matches using **Euclidean distance** and randomly choose between them, preferring better matches with higher probability

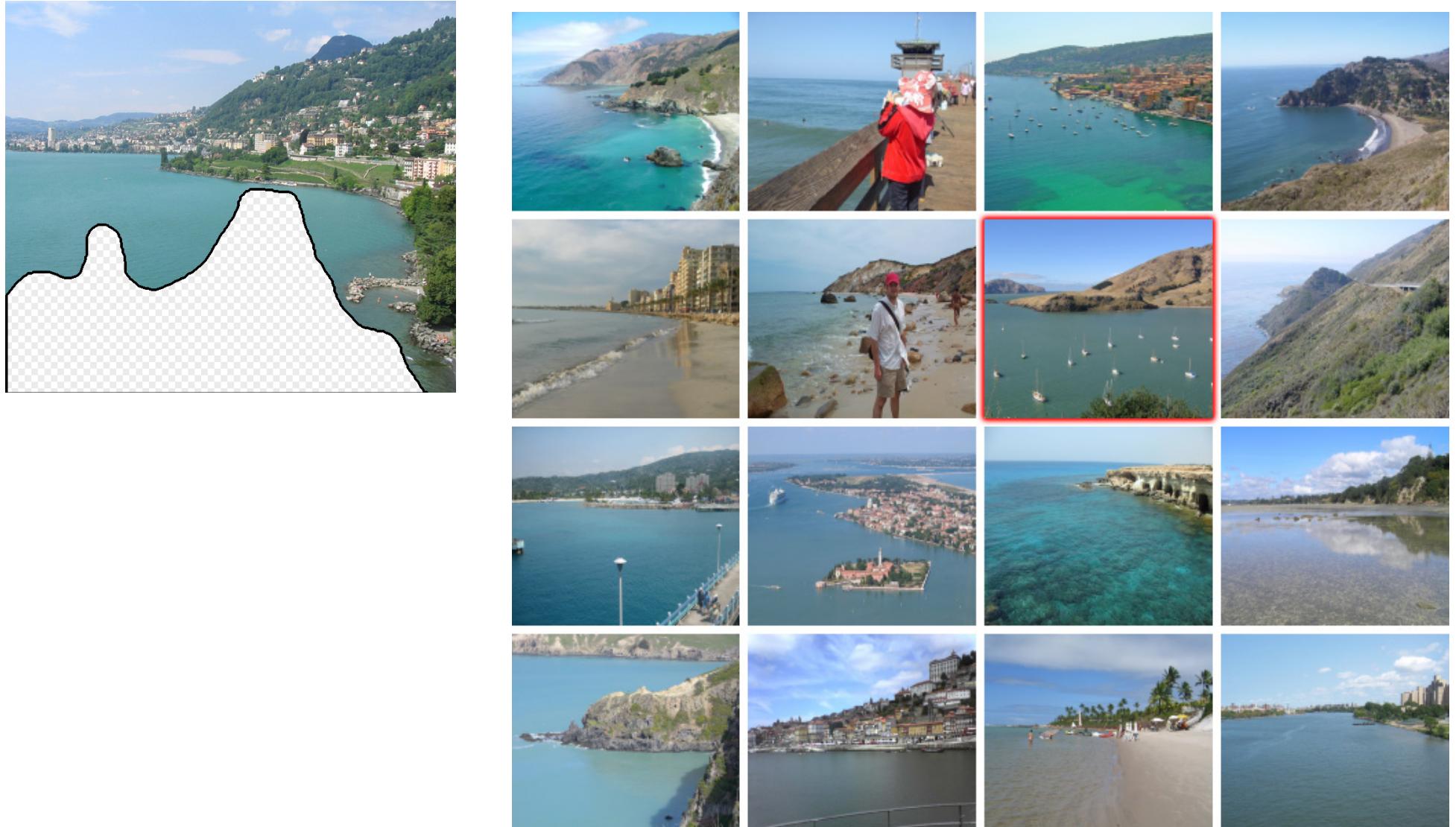
kNN: Scene Completion



“Scene completion using millions of photographs”, Hayes and Efros, TOG 2007

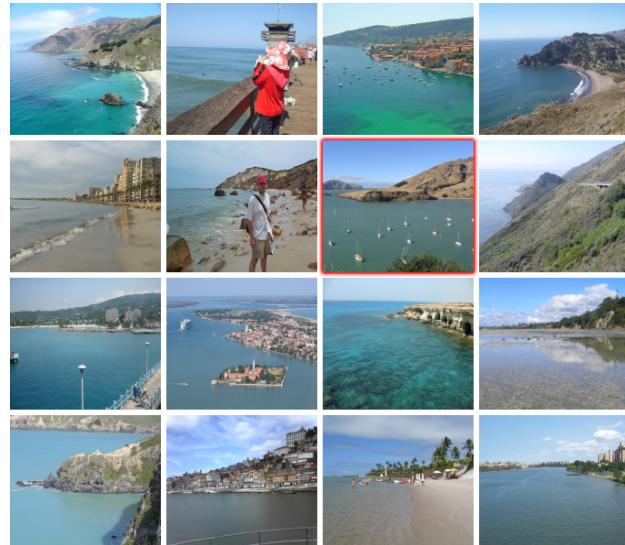
kNN: Scene Completion

Nearest neighbors



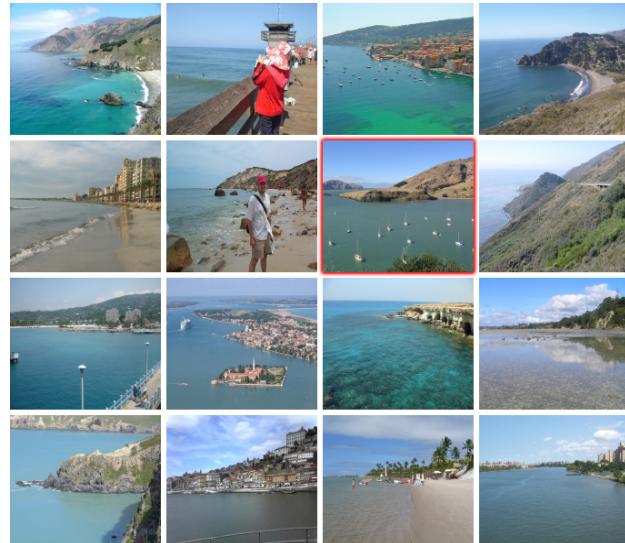
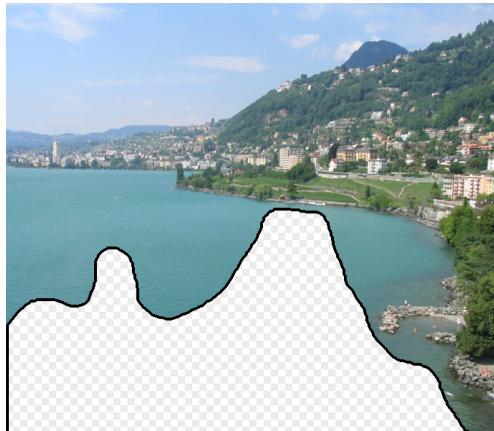
“Scene completion using millions of photographs”, Hayes and Efros, TOG 2007

kNN: Scene Completion



“Scene completion using millions of photographs”, Hayes and Efros, TOG 2007

kNN: Scene Completion



“Scene completion using millions of photographs”, Hayes and Efros, TOG 2007

Practical issue when using kNN: speed

Time taken by kNN for N points of D dimensions

time to compute distances: $O(ND)$

time to find the k nearest neighbor

$O(k N)$: repeated minima

$O(N \log N)$: sorting

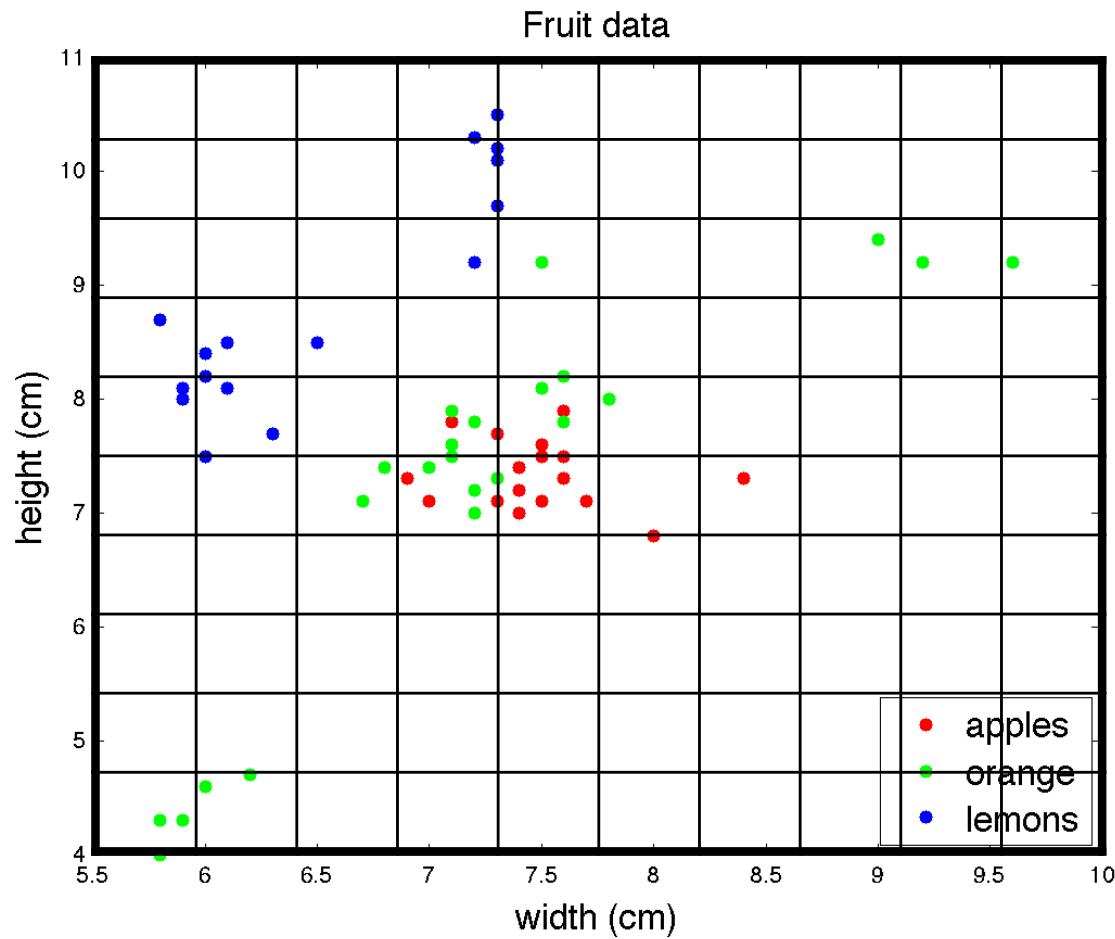
$O(N + k \log N)$: min heap

$O(N + k \log k)$: fast median

Total time is dominated by distance computation

We can be faster if we are willing to sacrifice exactness

Practical issue when using kNN: Curse of dimensionality



#bins = 10×10
d = 2

#bins = 10^d
d = 1000

Atoms in the universe:
 $\sim 10^{80}$

How many neighborhoods are there?

Outline

Clustering basics

K-means: basic algorithm & extensions

Cluster evaluation

Non-parametric mode finding: density estimation

Graph & spectral clustering

Hierarchical clustering

K-Nearest Neighbor

Slides credit

Slides are closely following and adapted from Hal Daume's book and Subranshu Maji's course.

The fruit classification dataset is from Iain Murray at University of Edinburgh

http://homepages.inf.ed.ac.uk/imurray2/teaching/oranges_and_lemons/.

The slides on texture synthesis are from Efros and Leung's ICCV 2009 presentation.

Many images are from the Berkeley segmentation benchmark

<http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds>

Normalized cuts image segmentation:

<http://www.timotheecour.com/research.html>