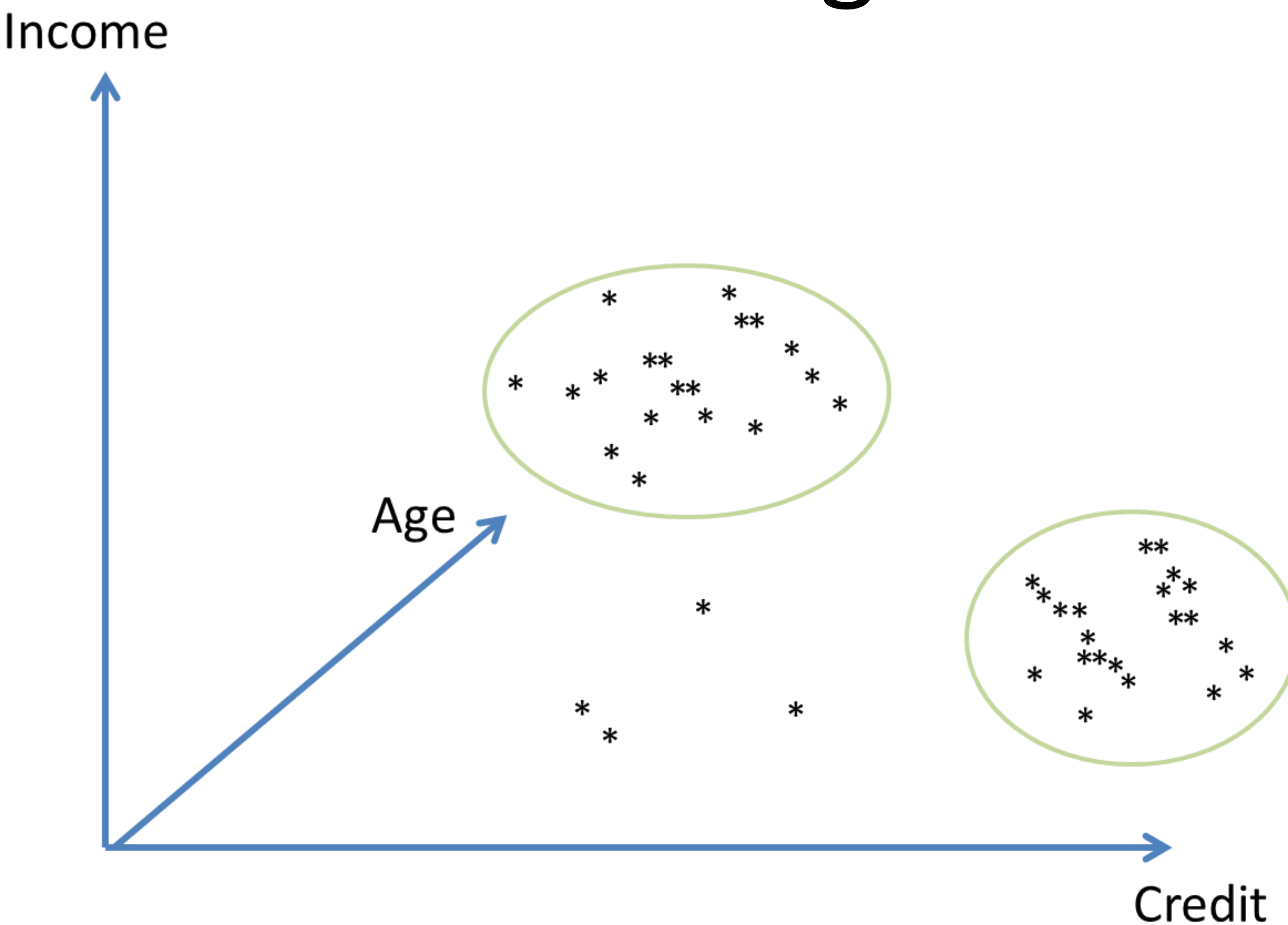
Clustering

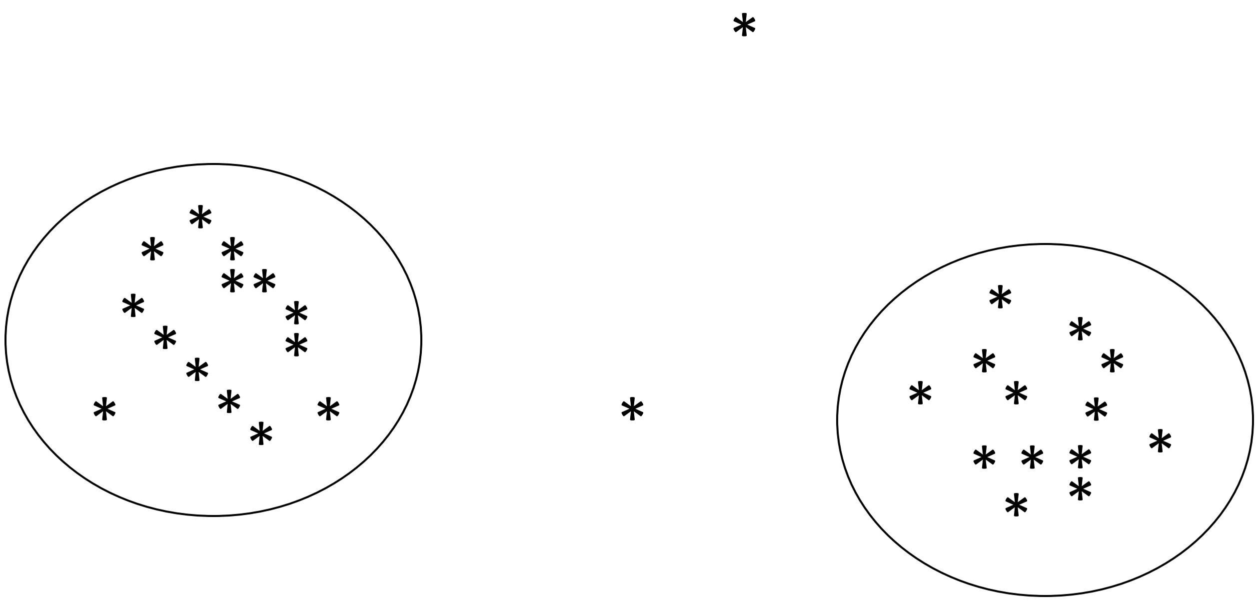


Contents

Clustering

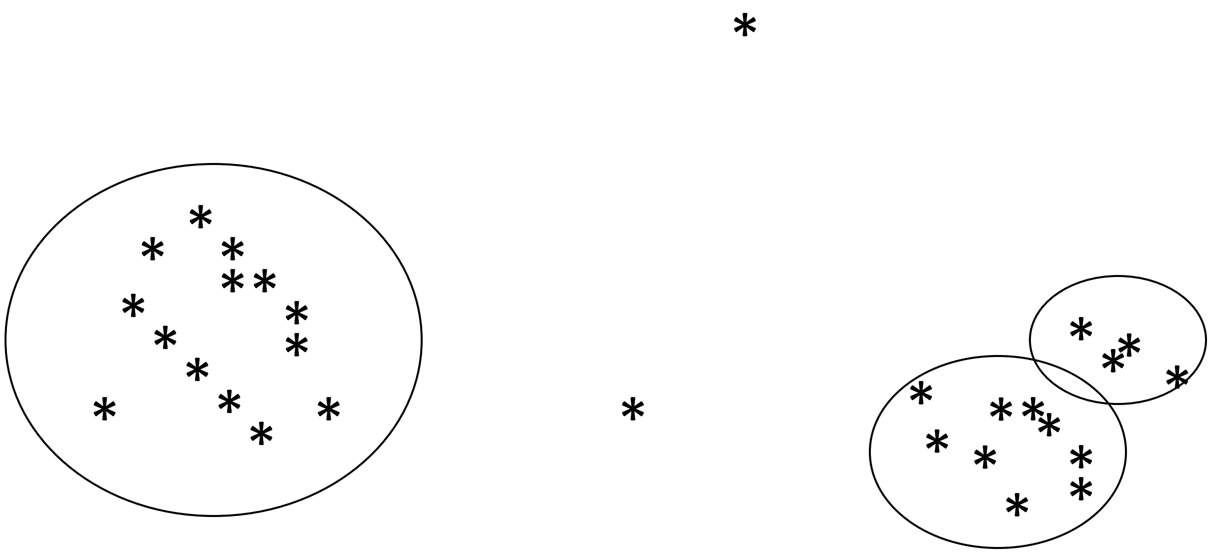
- … of individuals into groups/clusters that are
  - internally homogeneous (internal cohesion)
  - heterogeneous from group to group (external separation)
  - a clustering with fewer clusters (giving concise insights) is generally preferred
- Clustering is undirected (descriptive)
  - the groups are not defined in advance, but identified by the clustering algorithm
  - Unsupervised learning

A good clustering



- Small number of clusters, internally cohesive and externally separated

A questionable clustering



Small number of clusters, internally cohesive but external separation questionable

Too many clusters

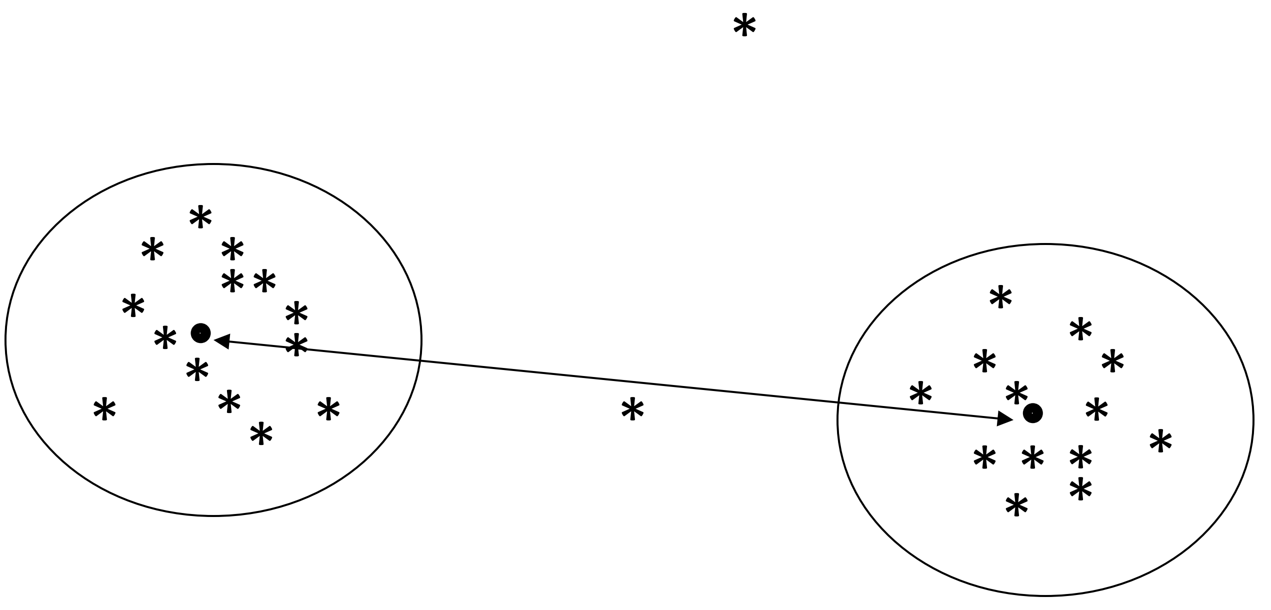
External separation also questionable

We need “density” for a cluster

Similarity measures

- Clusters are determined by a measure of similarity
- For numeric data we can use Euclidean distance (as in nearest neighbour)
  - Again implies the need for data normalisation
- For non-numeric data we need to apply some creativity

External separation



The distance between the cluster centroids gives a measure of the external separation, i.e., distance between the clusters

$$\text{distance}(C_1, C_2) = \text{distance}(\text{centroid}(C_1), \text{centroid}(C_2))$$

Cluster homogeneity

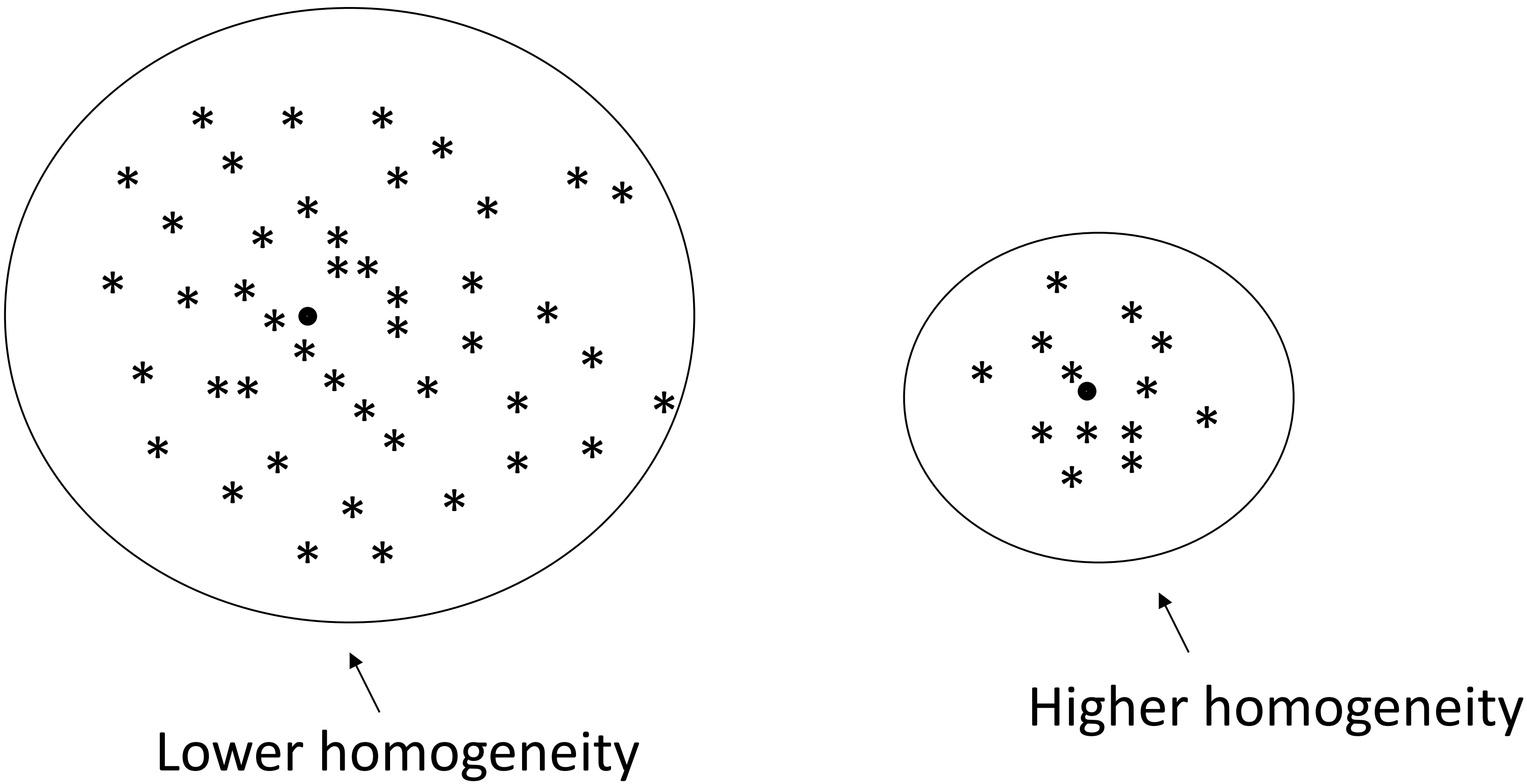
- Given a cluster C, we can measure homogeneity using

  average { distance(x,cent) | x in C }

  where cent = centroid(C)

  Obviously the lower the value, the more homogeneous

Cluster homogeneity



Similarity measures

- For qualitative data we cannot use the Euclidean distance: instead need another measure of similarity/distance
- e.g., to profile web site visitors, we can define a distance between 2 visitors via:
- CP = # pages visited by both
- CA = # pages visited by neither
- P = total # of pages at the site, then
- Similarity = (CP + CA)/P (Sokal/Michener)

E.g., Supermarket customers

Variables might be

- monthly amount in dept 1 (e.g., fresh produce),
- monthly amount in dept 2 (e.g., cereal), … etc
- monthly total value of shop
- average time/day of shop
- customer demographics … from loyalty cards
- Clustering then produces groups with similar demographics & buying patterns … we might then investigate each cluster for patterns within, and then subject them to a customised marketing campaign

Applications of clustering

- Customer profiling
- Insurance policy holder analysis
- Clustering of web visitors
- Clustering of web sources
- Genetics
- Satellite image interpretation
- Biology (plant and animal taxonomies)
- Pre-processing of data for other forms of data mining

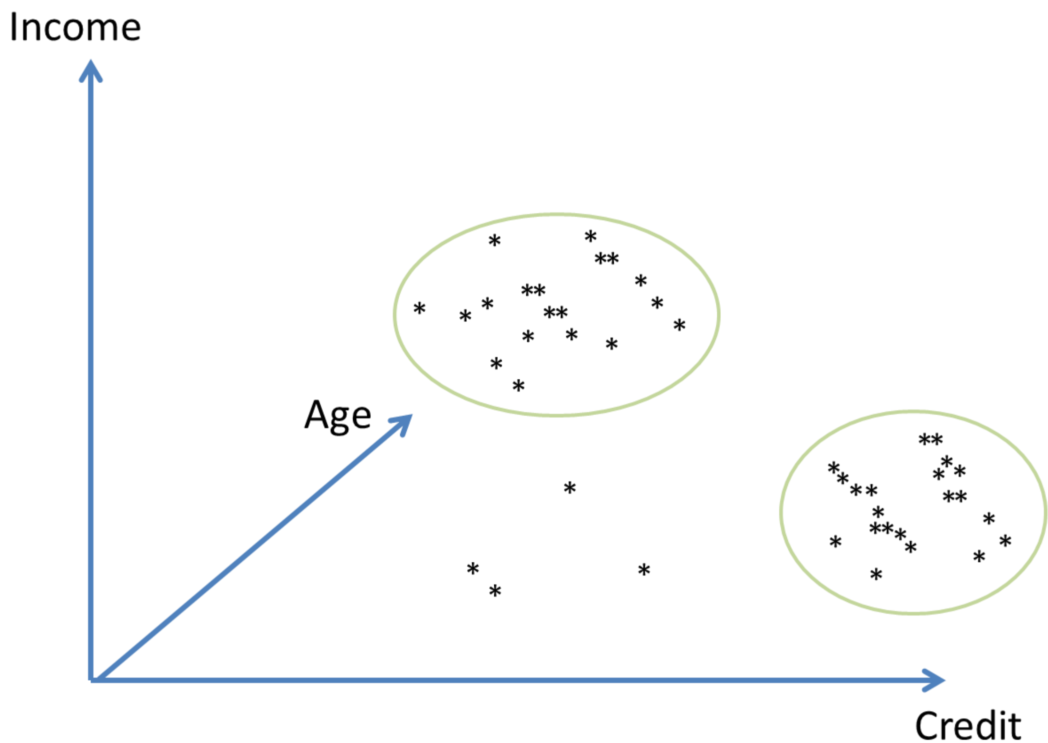
Clustering issues

- High dimensional spaces
  - Choose variables that are relevant to the stated objective
  - Reduce # variables if possible
    - also helps with visualisation and interpretation
- Outliers can adversely affect result
  - See later

Interpreting the clusters

- Visualisation can be useful …

To visualise the clusters the tool will - by necessity - pick the 3 most relevant variables for the visualisation



Interpreting the clusters

- A clustering/cluster may (or may not) have any practical significance or use
  - we can evaluate a clustering using mathematical measures, but the “best” clustering might simply be the one that throws up some business insight …

Interpreting the clusters

- Use decision trees with cluster# as the target to derive rules explaining the clustering
  - if age < 30 and monthly spend > 300 then cluster = 1
  - if age > 30 and average time of shop = Saturday am then cluster = 1

- … we can see the problem

Interpreting the clusters

- Use decision trees with cluster# as the target to
  - identify the variables most relevant to the clustering
  - evaluate the “predictability” of the clustering
    - assessed from the performance of the tree
    - can be used to compare clusterings

Interpreting the clusters

- Look at the average values of key attributes within each cluster to get a description of a typical member of the cluster
- e.g., in supermarket customer profiling we might look at:
  - Average customer profitability (e.g., low)
  - Average monthly spend (e.g., low)
  - Average monthly number of visits (e.g., high)
  - …

Interpreting the clusters

- Apply mining techniques to each cluster individually
  - do they contain useful patterns?
- Cluster# can be inserted into the records, and then used as an input variable to other techniques
  - enhancing the data

1. Agglomerative clustering

1. Put each individual in a cluster of its own
2. Pick the two clusters that are “closest” and coalesce them
3. Repeat until . . .
   – need stopping criteria based upon chosen criteria: internal cohesion, external separation, #clusters

Agglomerative clustering issues

- Agglomerative clustering is a form of hierarchical clustering (see also divisive clustering)
  - these suffer from the fact that a merger is never subsequently undone – so need to make a good choice
  - not good for large data sets

Divisive clustering

- Divisive clustering starts with a clustering containing a single group/cluster, and then partitions
- Has similarity to decision trees
- Tends not to be used in routine applications because of computational overhead

2. k-means clustering

- Clustering for numeric data
- k-*means* refers to the fact that the process is driven by the mean point within each cluster
  - For numeric data the mean point = centroid
- Divides data into a *pre-determined* (k) number of clusters

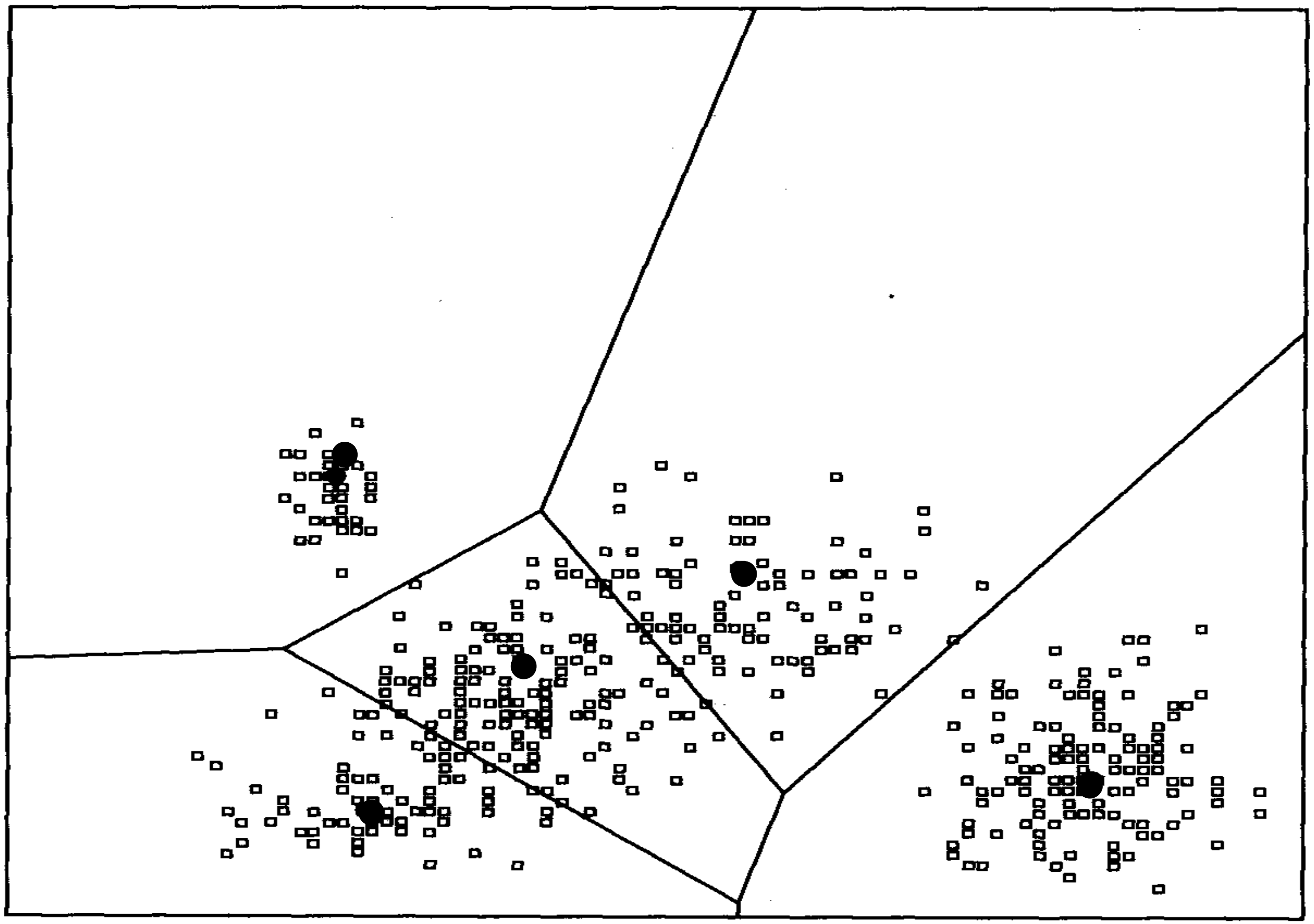
The k-means algorithm

1. Pick k seed points $s_1, s_2, s_3, \ldots, s_k$
2. Define k clusters $C_1, C_2, C_3, \ldots, C_k$ based upon these seed points using the Voronoi diagram
3. Update the seed points

$$s_i \longleftarrow \text{centroid}(C_i)$$

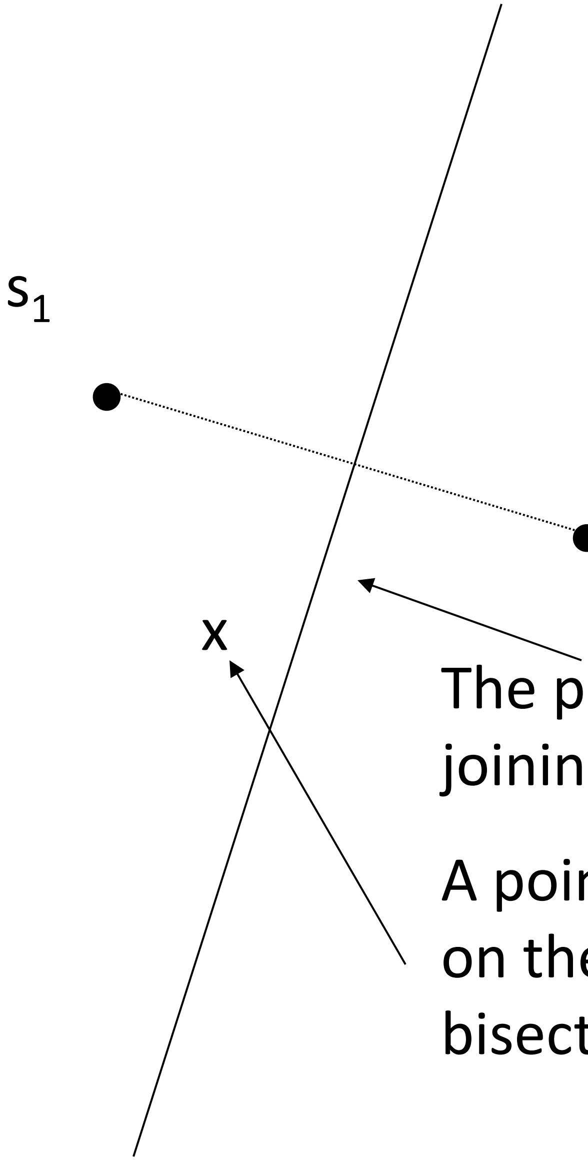
4. If seeds not stable, go to step 2

Voronoi diagram defined by 5 seeds



The Voronoi diagram

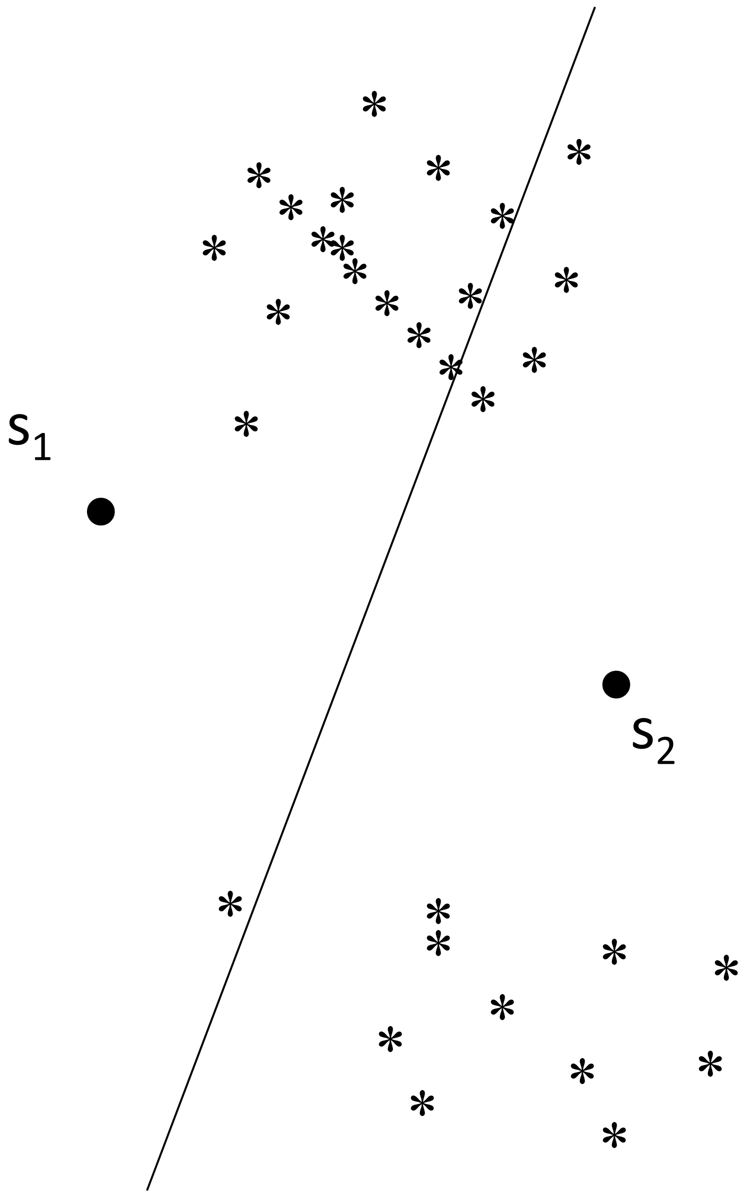
- If s_1, s_2, …, s_k are the seeds, then a point x is in cluster C_j if x is closer to s_j than to any of the other seeds
  - i.e., s_j is the seed that is closest to x
- The lines of the Voronoi diagram are constructed from segments of the relevant perpendicular bisectors:



A S I D E

The perpendicular bisector cuts the line joining s_1 and s_2 equally and at right angles

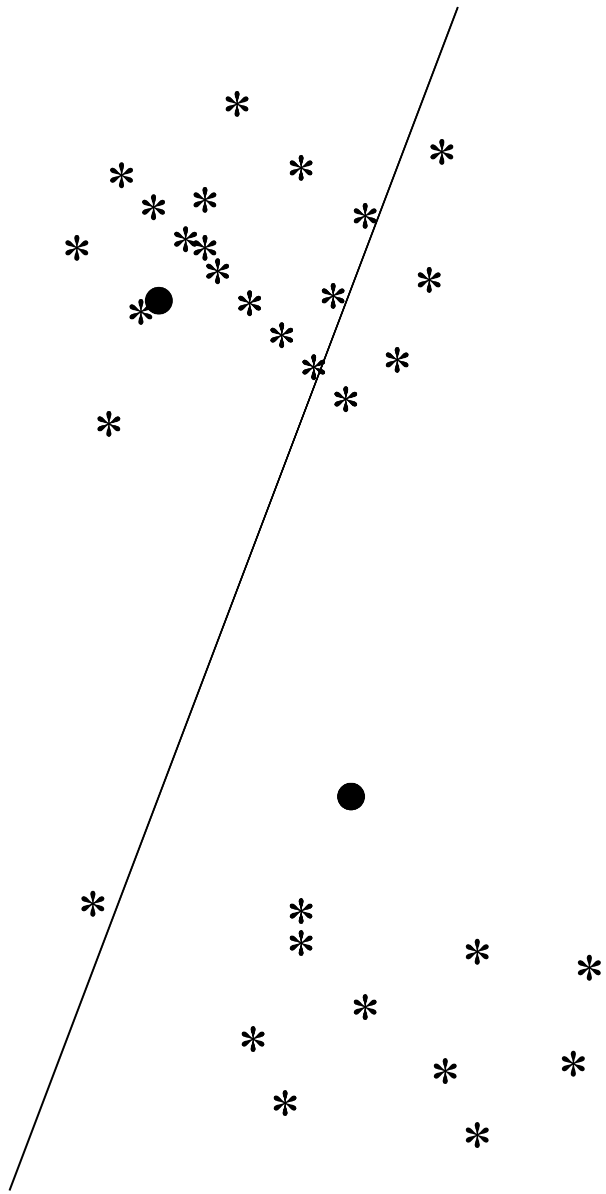
A point x is closer to s_1 than to s_2 if it lies on the same side of the perpendicular bisector as s_1



Why we need to iterate:

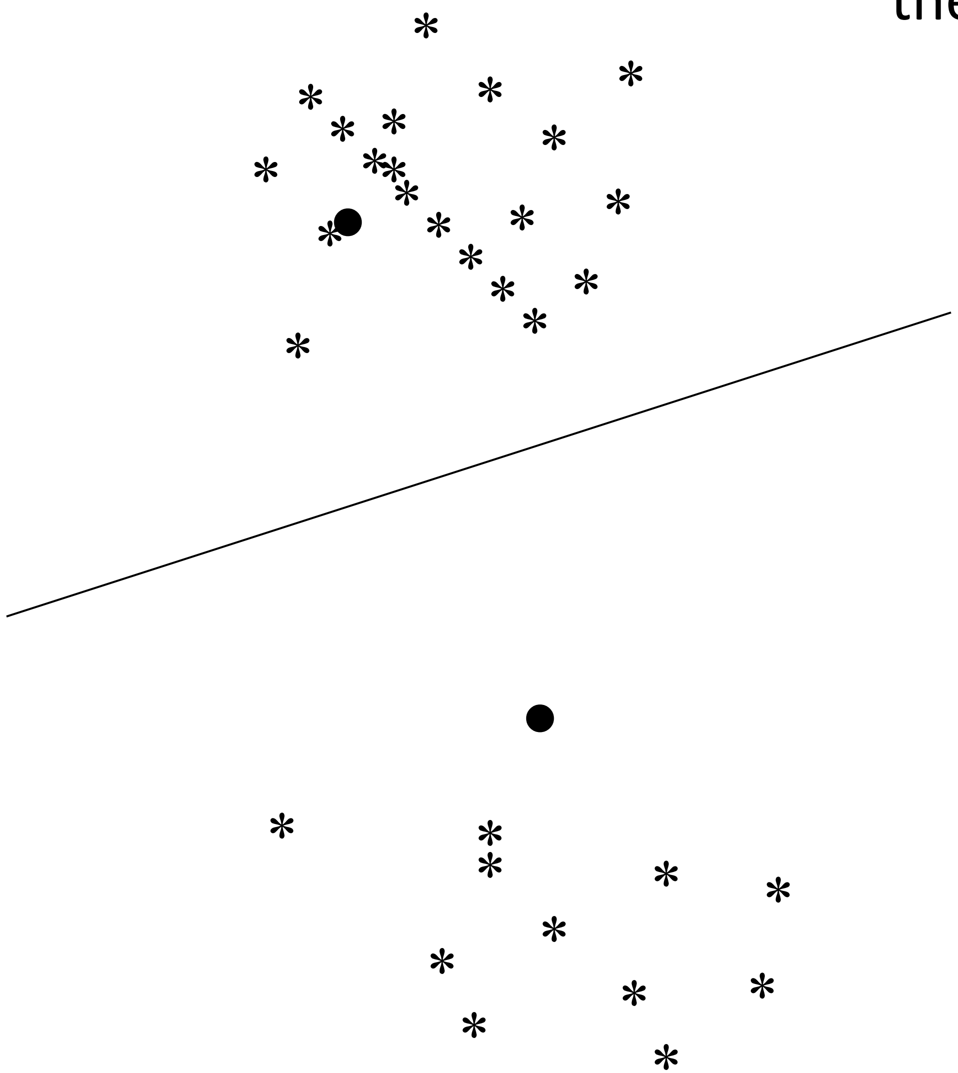
Clearly there are 2 clusters but these are not captured by the first Voronoi diagram

Moreover when we construct the first Voronoi diagram, the seeds do not in fact “represent” their cluster - they are not at or near the centroid of their cluster

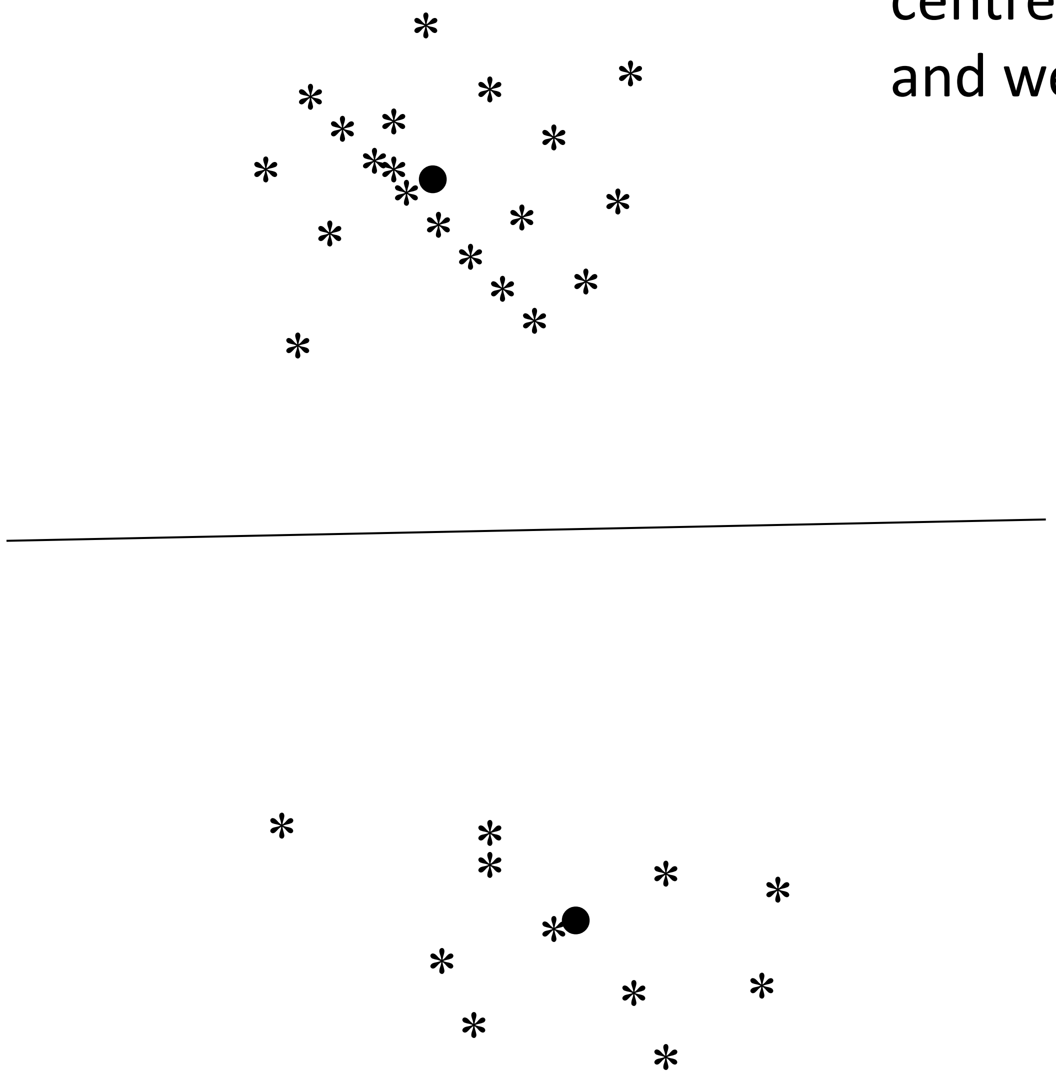


The seeds move to the centroid of the clusters (as defined by the Voronoi diagram)

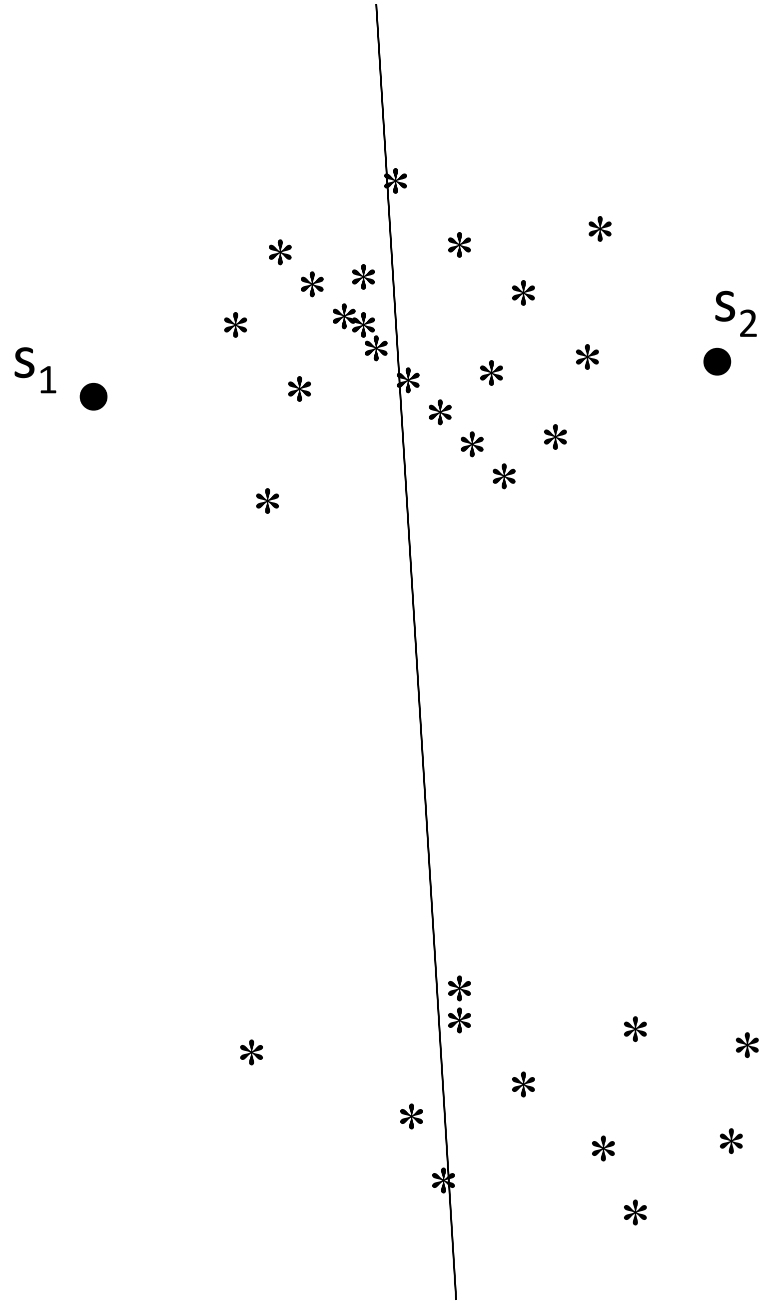
. . . and we then compute the new Voronoi diagram

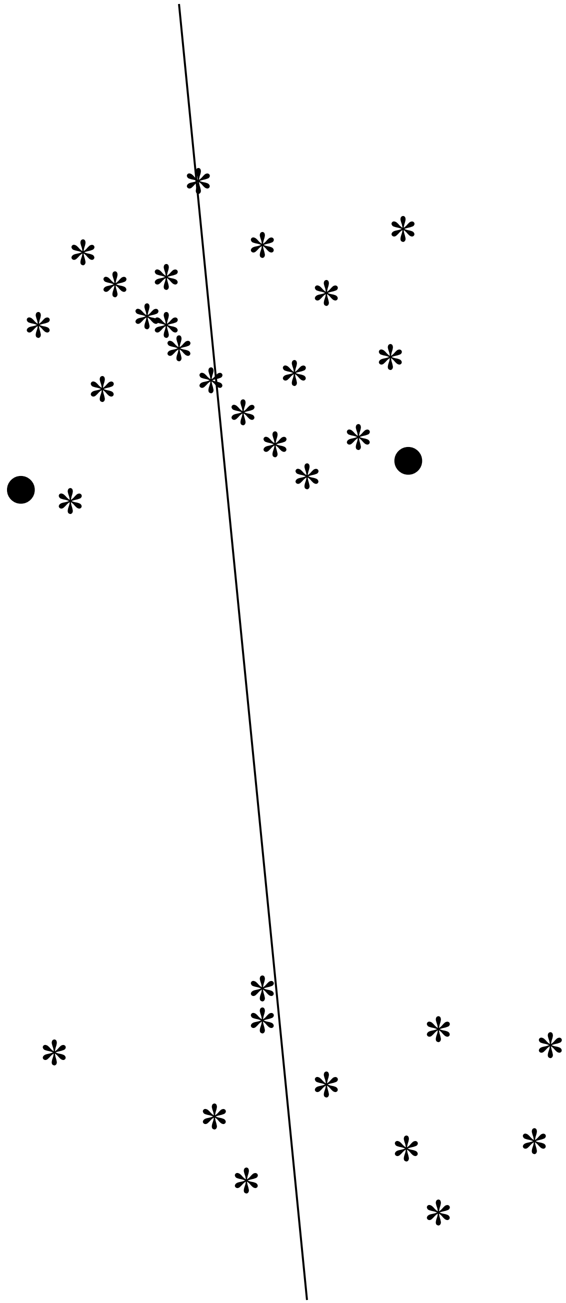


. . . the seeds move to the centre of the cluster .... and we have convergence



The initial set of seeds may be poor





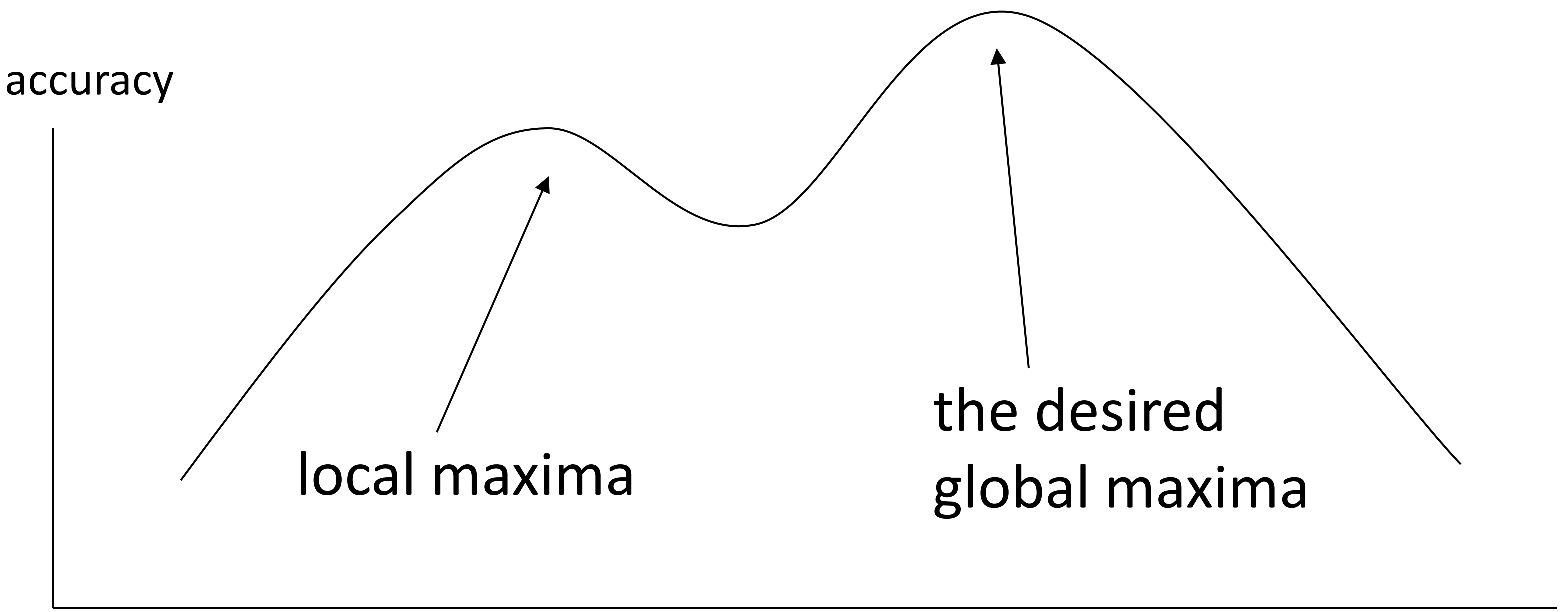
. . . we’ve not made much progress

There are techniques that can address the problem of choosing the initial set of seeds - see later

Clustering and local maxima

A
G
A
I
N

- Hill climbing approaches are susceptible to local maxima



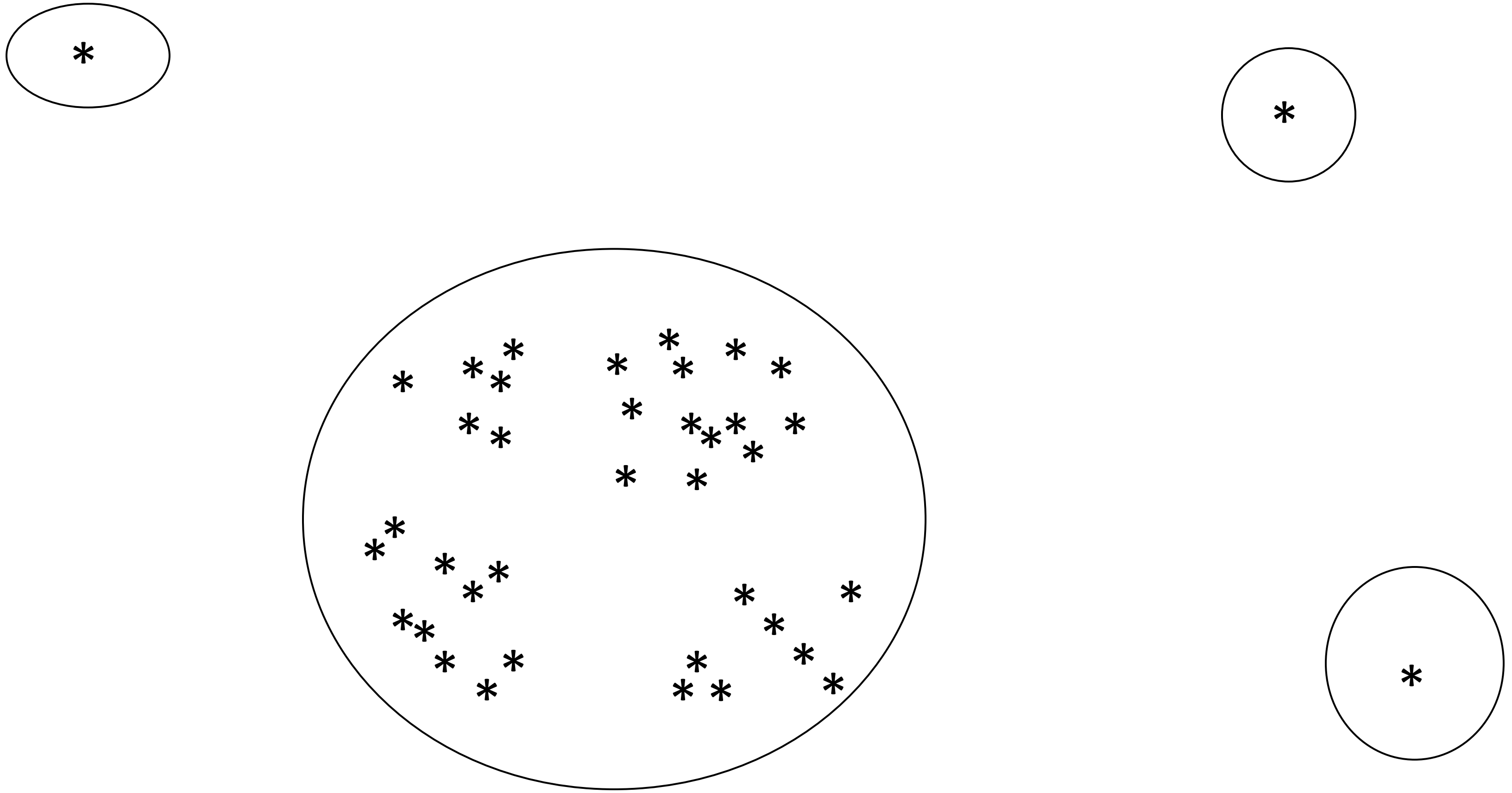
k-means issues

- Need to choose k
- We can re-run with different values for k
  - some tools will do it for you
- Susceptibility to choice of initial seeds
- Susceptibility to outliers
  - because the mean is susceptible to outliers
  - can instead use the medoid

3. Outliers

- A rare or distant value (from the norm)
  - e.g., region code will contain lots of outliers if most of your customers are local
  - might indicate a problem with data collection, e.g., DoB = 11-11-11

k-means and outliers



Not a useful clustering: all the interesting data ends up in a small number of clusters - in this case just 1

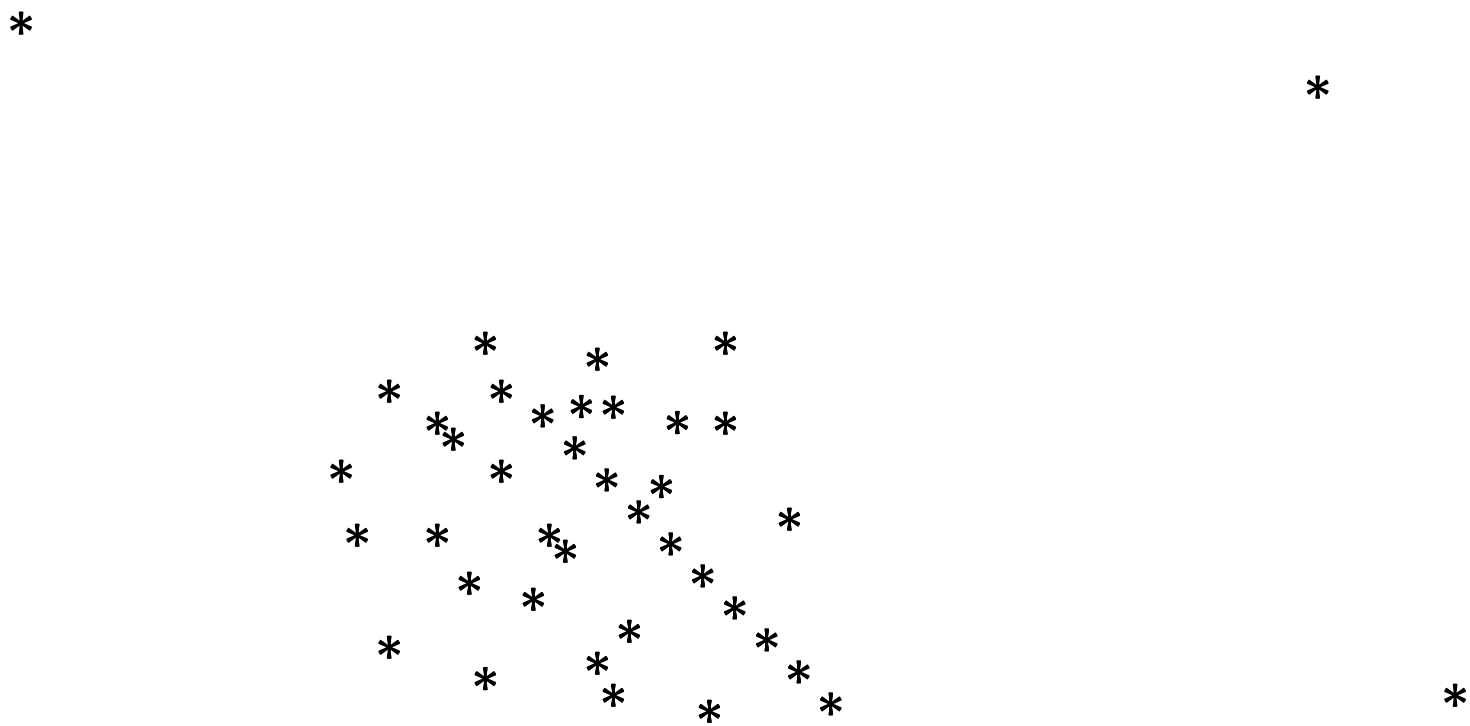
Options for handling outliers

- Do nothing
- Delete the outlying individuals (rows)
  - may introduce bias; maybe necessary

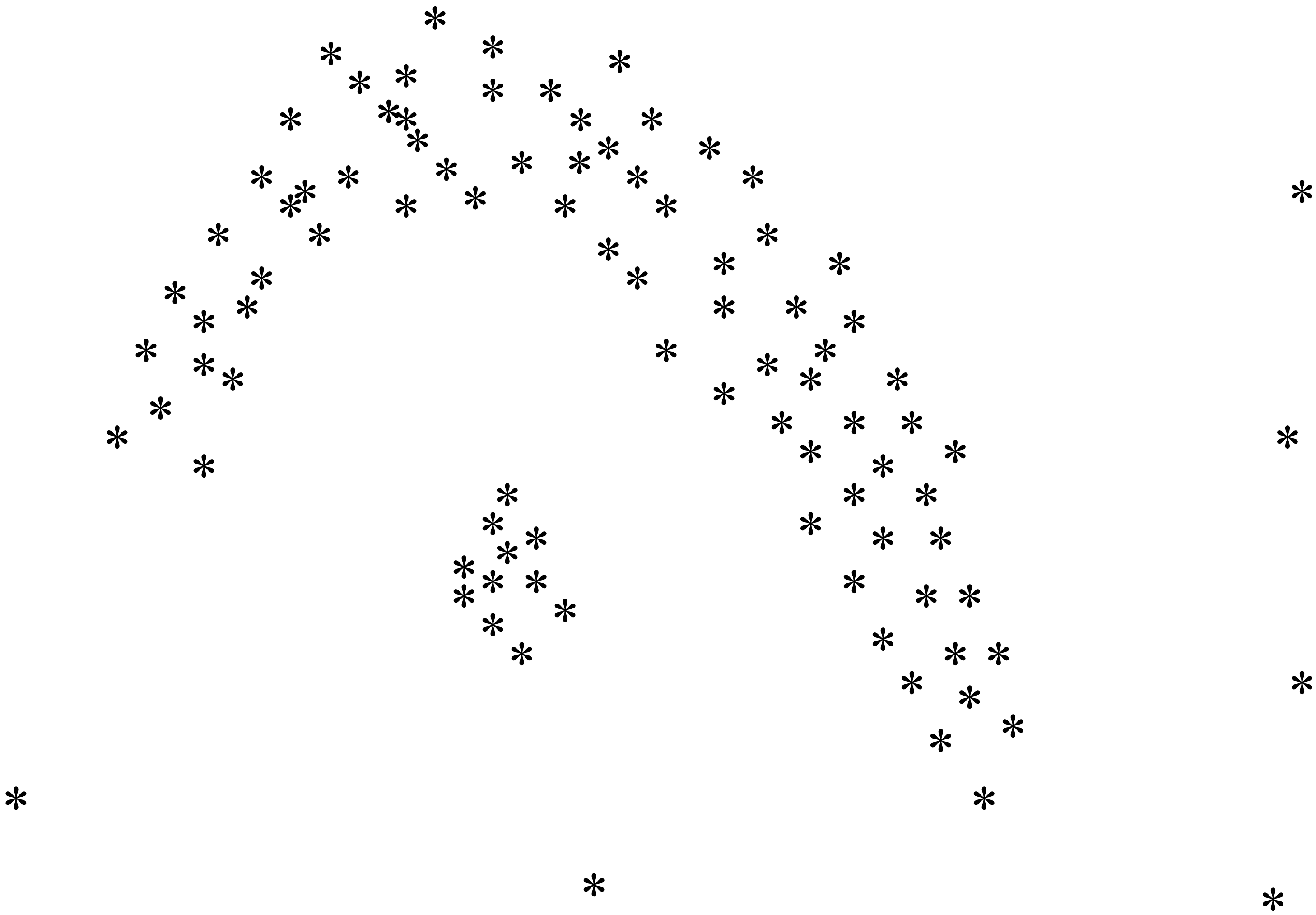
Handling outliers

- Replace the outlying column values
  - e.g., by typical/predicted value … bias
- Bin column values
  - equal height bins(outliers cease to be outliers)
  - e.g., salary ranges, aggregation of postcodes
- Delete column or replace column with derived variables
  - e.g., avg income in that postcode
  - e.g., local vs non-local postcode

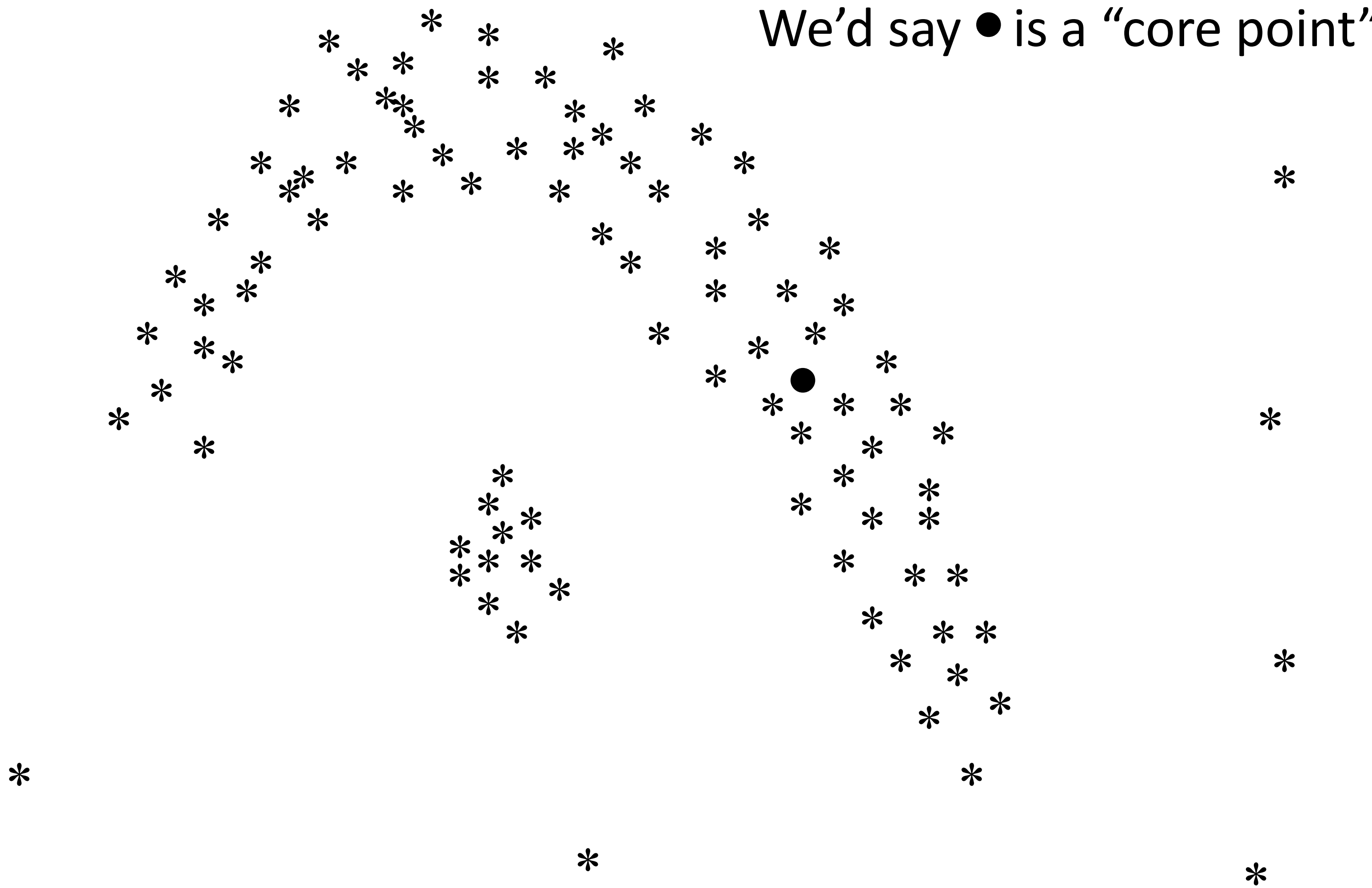
Identifying outliers: Visualisation



4. Density based clustering



Data around the point • is dense – and it is therefore in some cluster

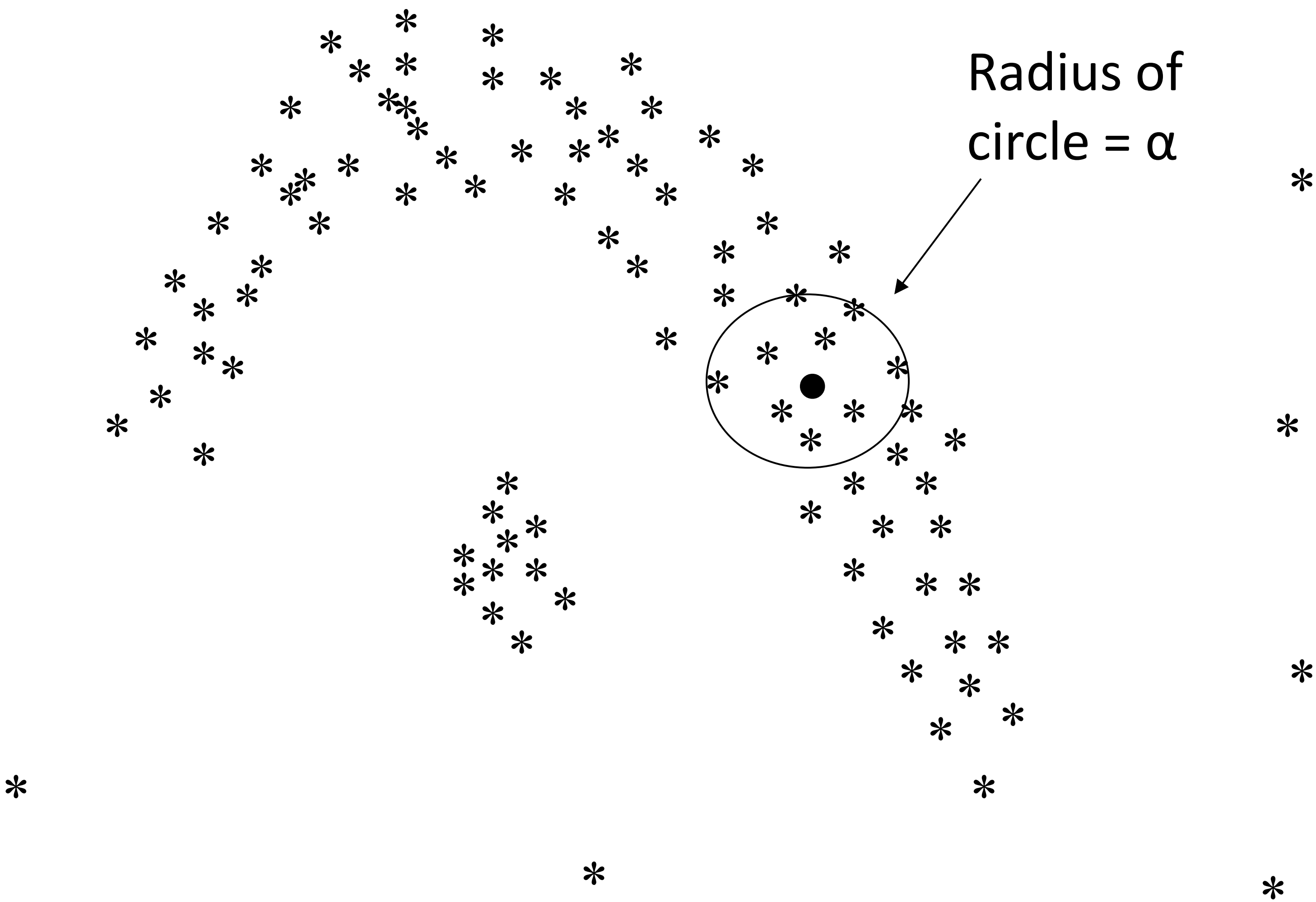


Core points

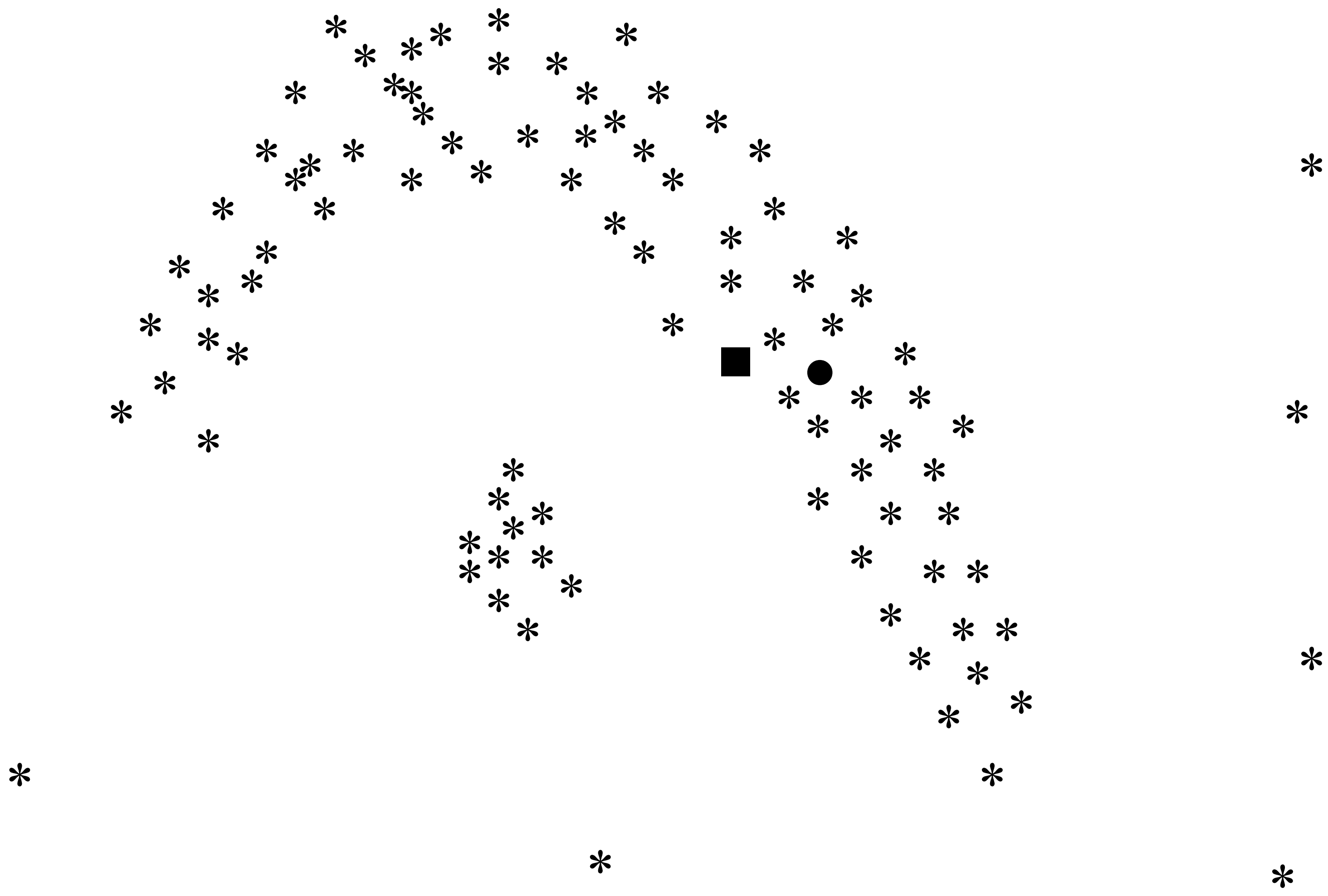
- Let D be the data set
- x is a *core point* if the number of near neighbours exceeds some threshold
- i.e., $|\{y \text{ in } D : \text{distance}(x,y) < \alpha\}| \geq \delta$

where α is typically a small number (the radius of the neighbourhood) and δ defines the minimum number of data points for the neighbourhood to be regarded as dense

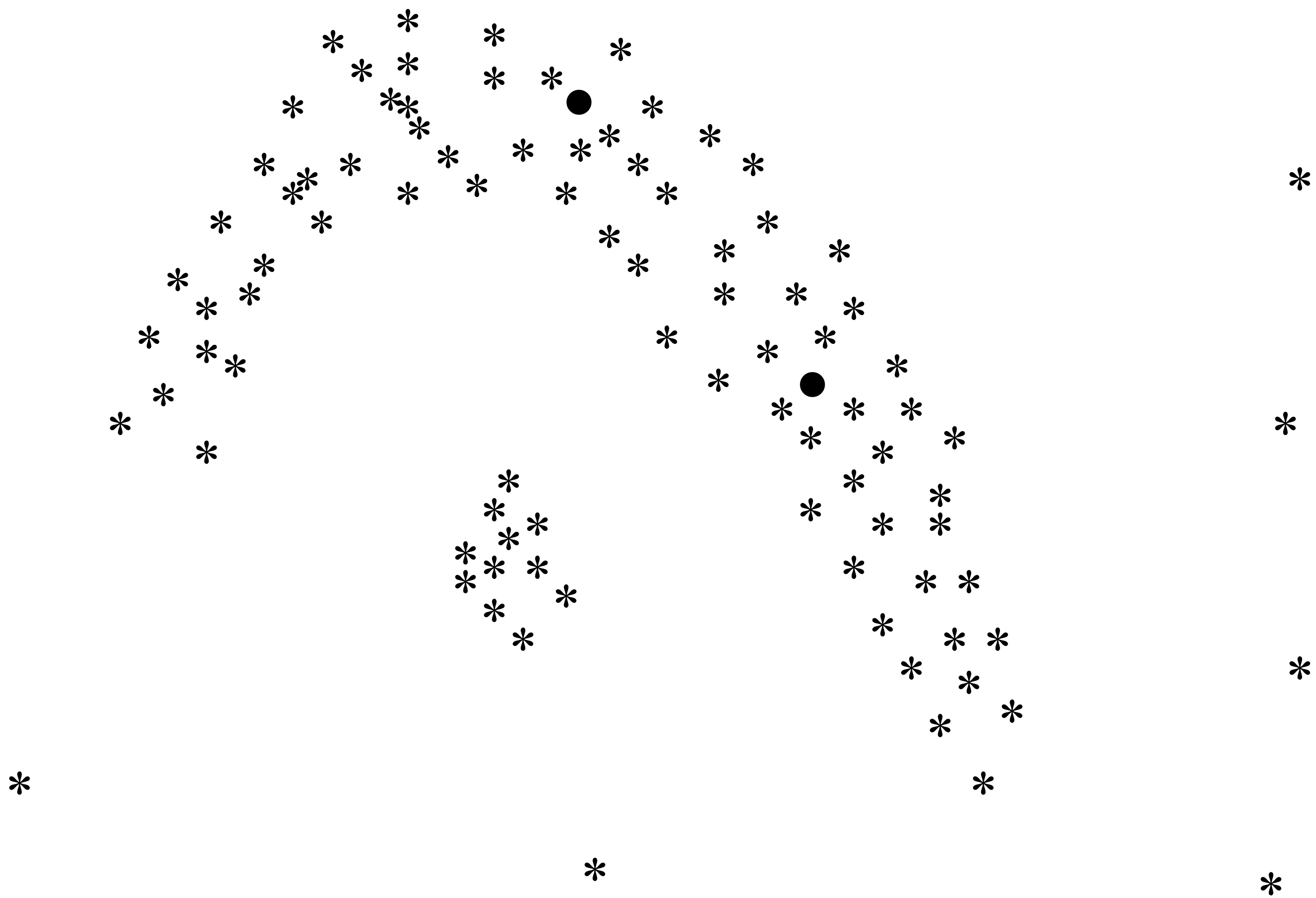
Suppose we set δ = 6, then ● is a core point … the number of close neighbours exceeds δ



■ is not a core point, but it is close to a core point … it is therefore still in the cluster



The core (interior) of the cluster is made up of a set of core point that are "connected"



Connectedness

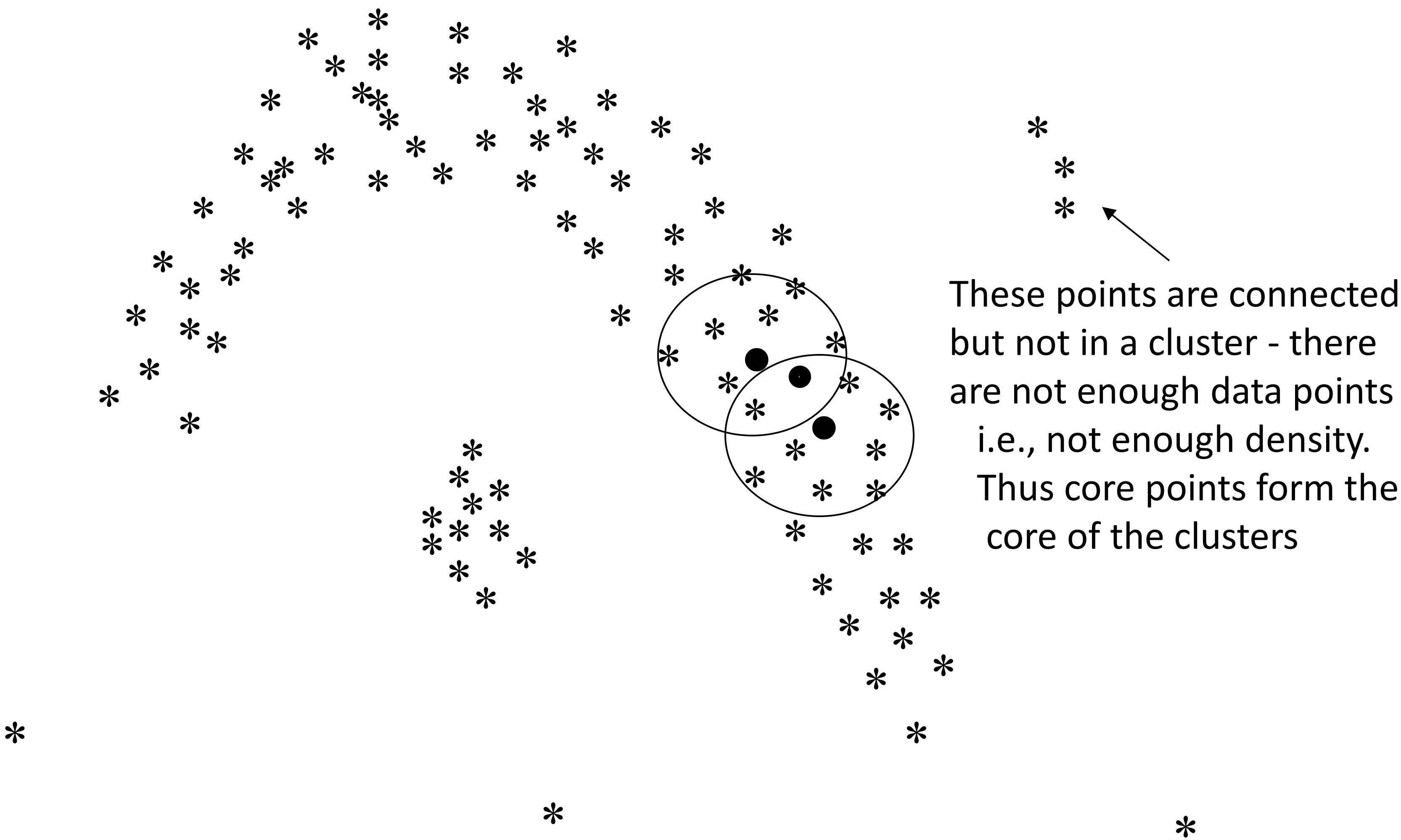
- If x and y are core points, then x and y are *connected* if there is a sequence of core points

$$x = x_0, x_1, x_2, \ldots, x_n = y$$

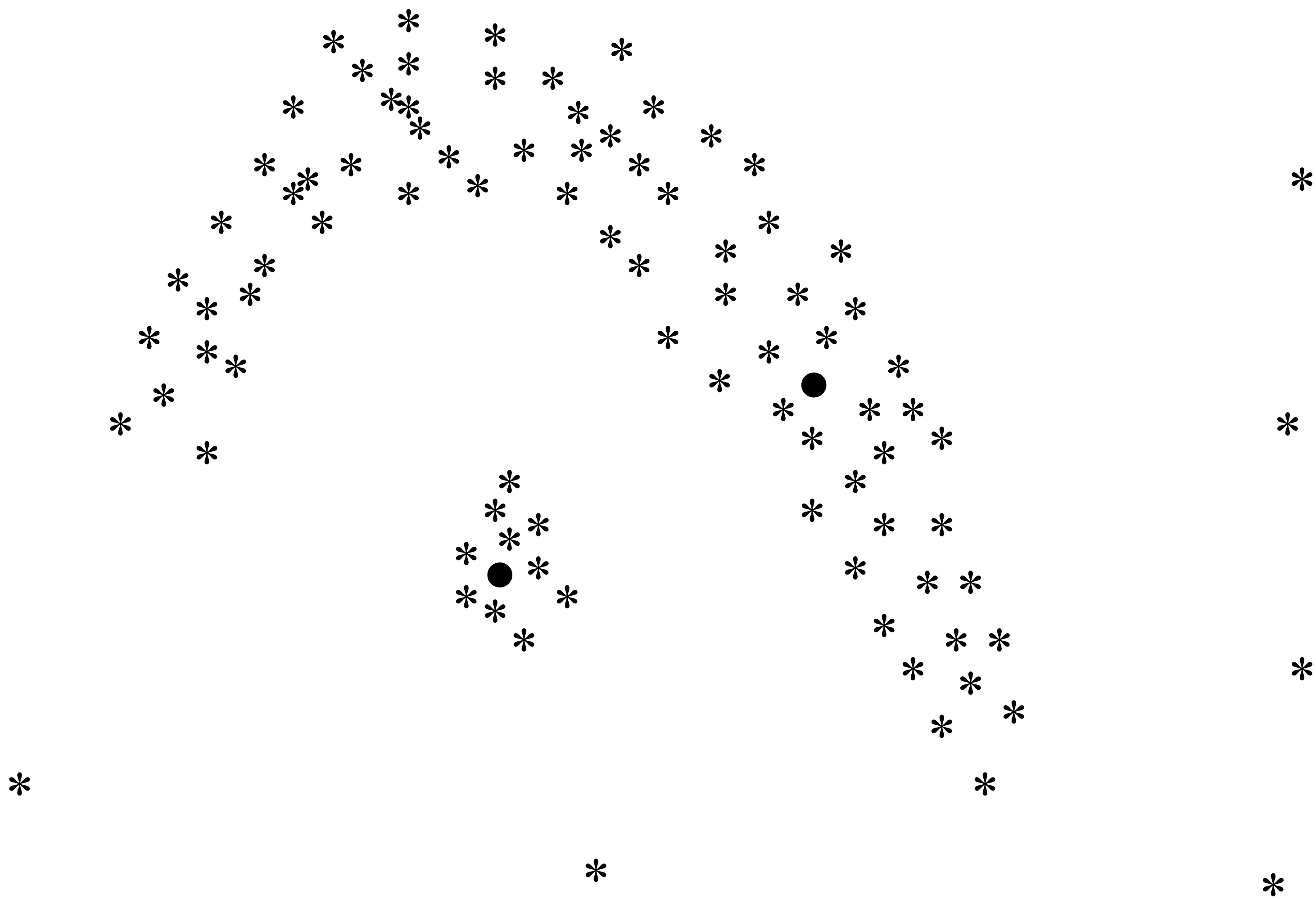
such that

$$\begin{aligned} &\text{distance}(x_0, x_1) < \alpha \\ &\text{distance}(x_1, x_2) < \alpha \\ &\text{distance}(x_2, x_3) < \alpha \\ &\ldots \\ &\text{distance}(x_{n-1}, x_n) < \alpha \end{aligned}$$

If two core points are connected then they must be in same cluster



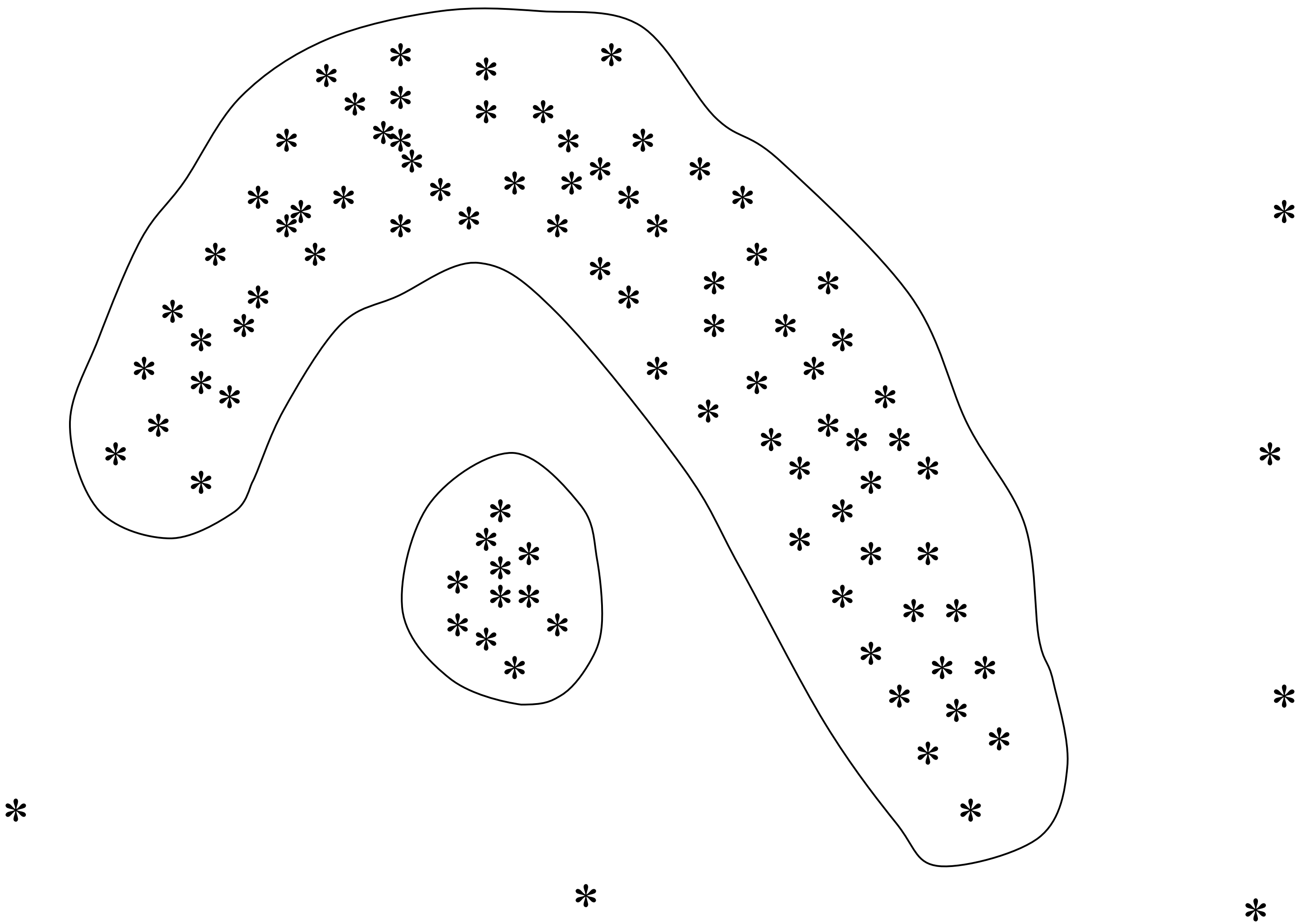
If two core points are not connected then they cannot be in the same cluster



Clusters defined by connectedness

- A cluster consists of
  - a core point x
  - the set of all core points connected to x
  - the set of all points close to such core points
    - i.e., in the α - neighbourhood of such a core point

… yielding the following clusters



5. Profiling website visitors

- To classify visitors into homogeneous subgroups with a view to identifying typical profiles. Can use for
  - marketing (aimed at subgroups)
  - monitoring subgroup evolution
  - monitoring effects of marketing and website re-structuring
  - classifying new visitors

The source data

- Effectively a log file - for each session a visitor id, and then a set of pages visited. Repeat visits by the same user are not tracked. Convert to "data matrix":

Visitor	Page1	Page2	Page3	…
1	2	9	1	…
2	3	11	7	…
3	1	4	3	…
…	…	…	…	…

Dimension reduction

- Typical websites contain many pages which are logically related/equivalent
  - e.g., several pages related to a given business product
- Computational efficiency
- High dimensional spaces
- So: Group the pages into a smaller number of page groups:

Page groups (microsoft.com)

- Initial - general access pages
- Support … help pages
- Download
- Office
- Development
- Software
- Hardware

… reduced to 13 groups

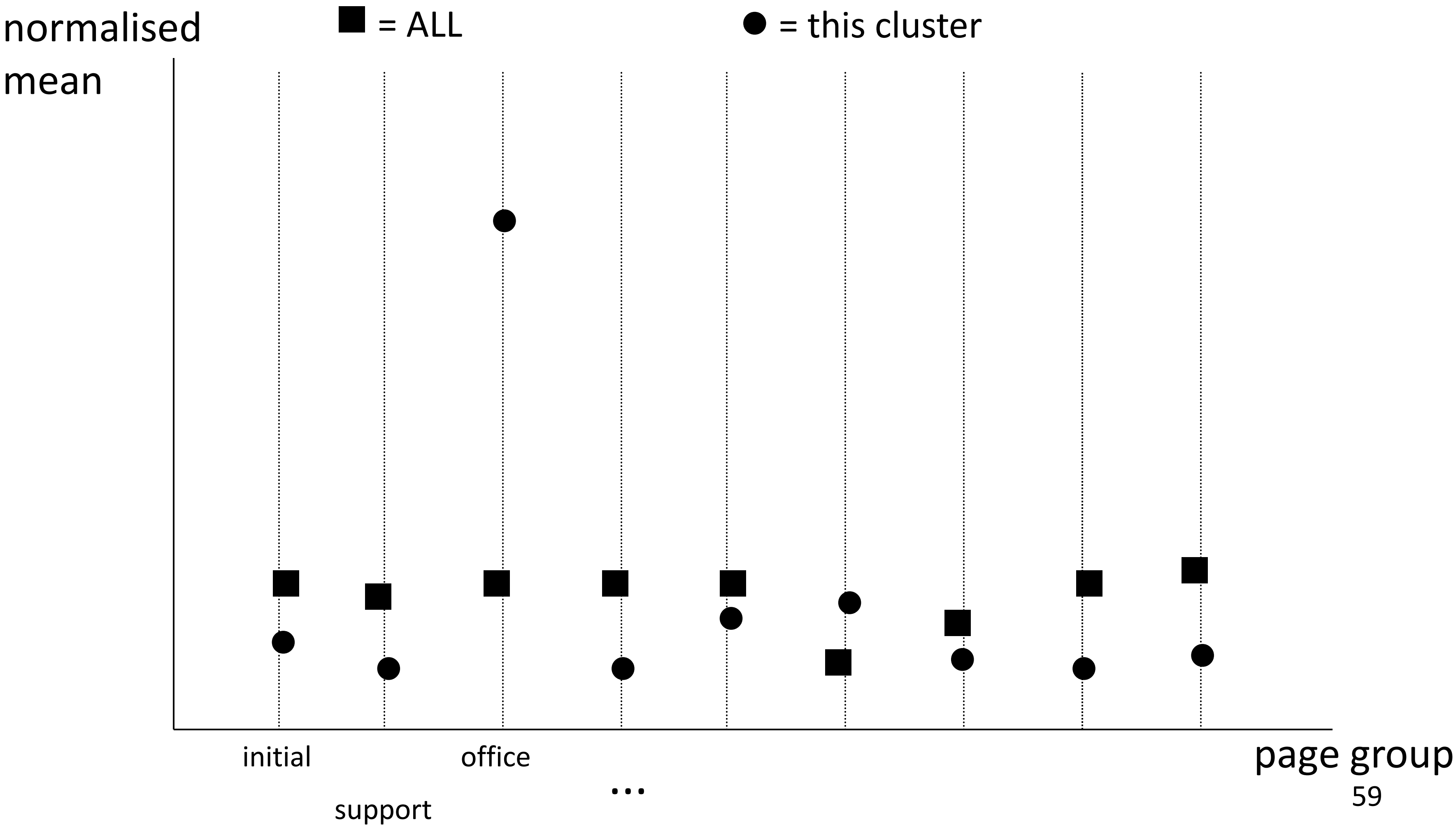
k-means clustering

- k-means clustering is fast(er) but it requires us to choose k and initial seeds. Hierarchical clustering is slow but requires no prior choices. **So**:
- Pick a representative subset of the data (in the interests of efficiency) & apply hierarchical clustering in order to get a value for k ....
- Then apply k-means to the full data set starting with initial seeds equal to the centroids of the clusters produced by the hierarchical clustering

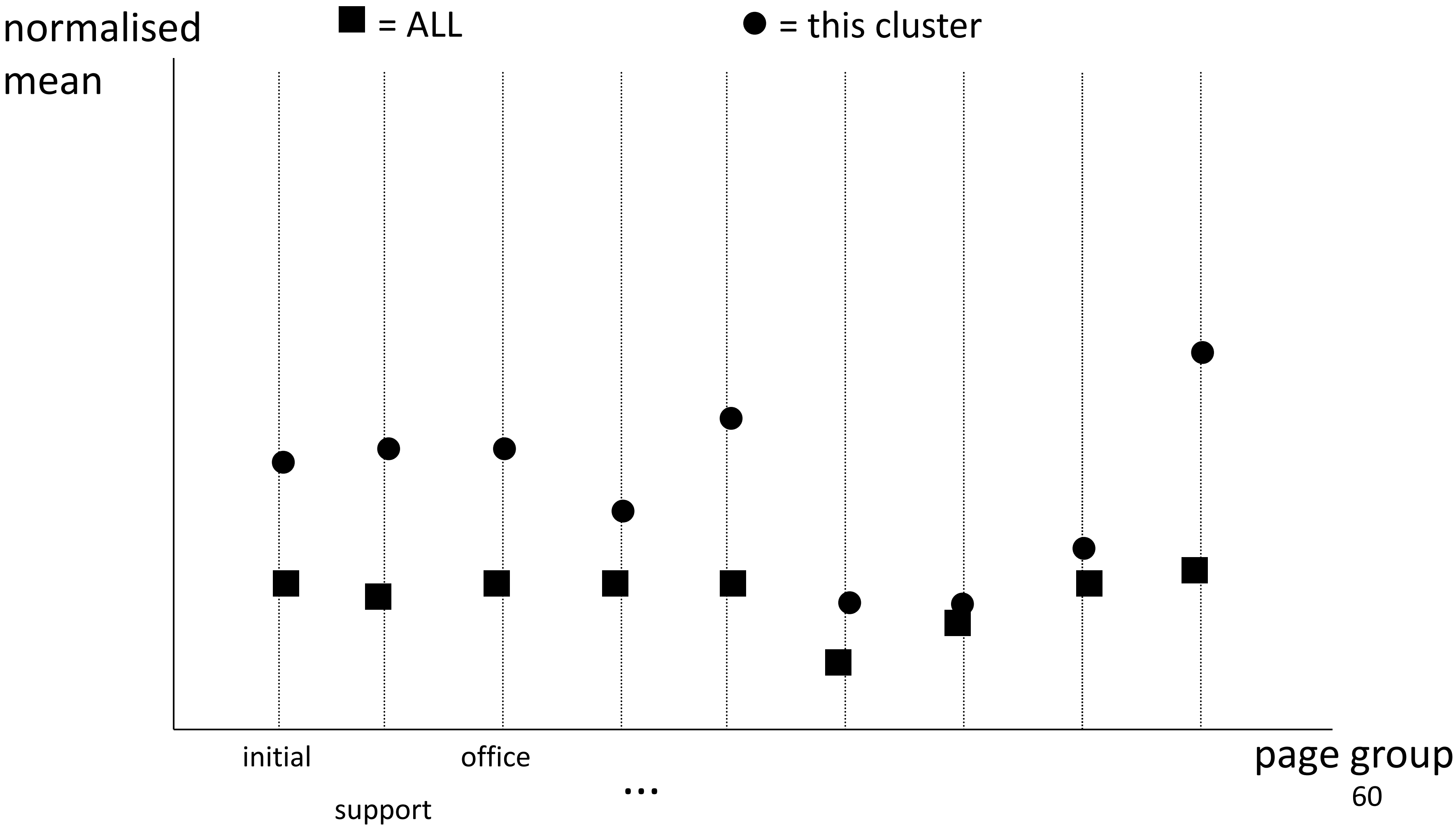
Interpreting the results: centroid coordinates

Cluster	Initial	Support	Download	…
1	1.2	1.9	6.1	…
2	3.1	1.1	0.7	…
3	2.1	0.1	0.9	…
…	…	…	…	…

Interpreting the results: a mono-thematic cluster



Interpreting the results: a poly-thematic cluster



Interpreting the results: a poly-thematic cluster