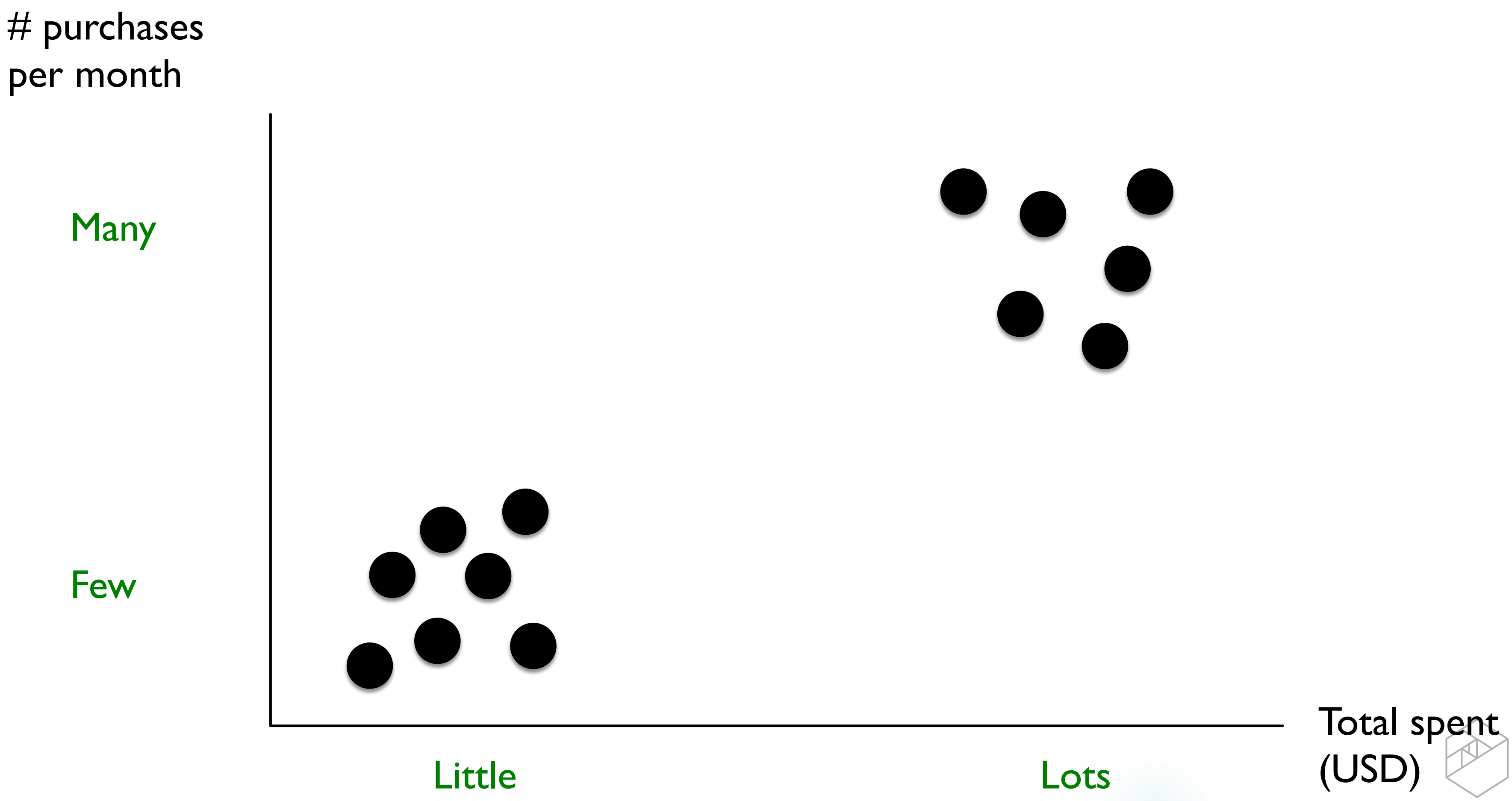
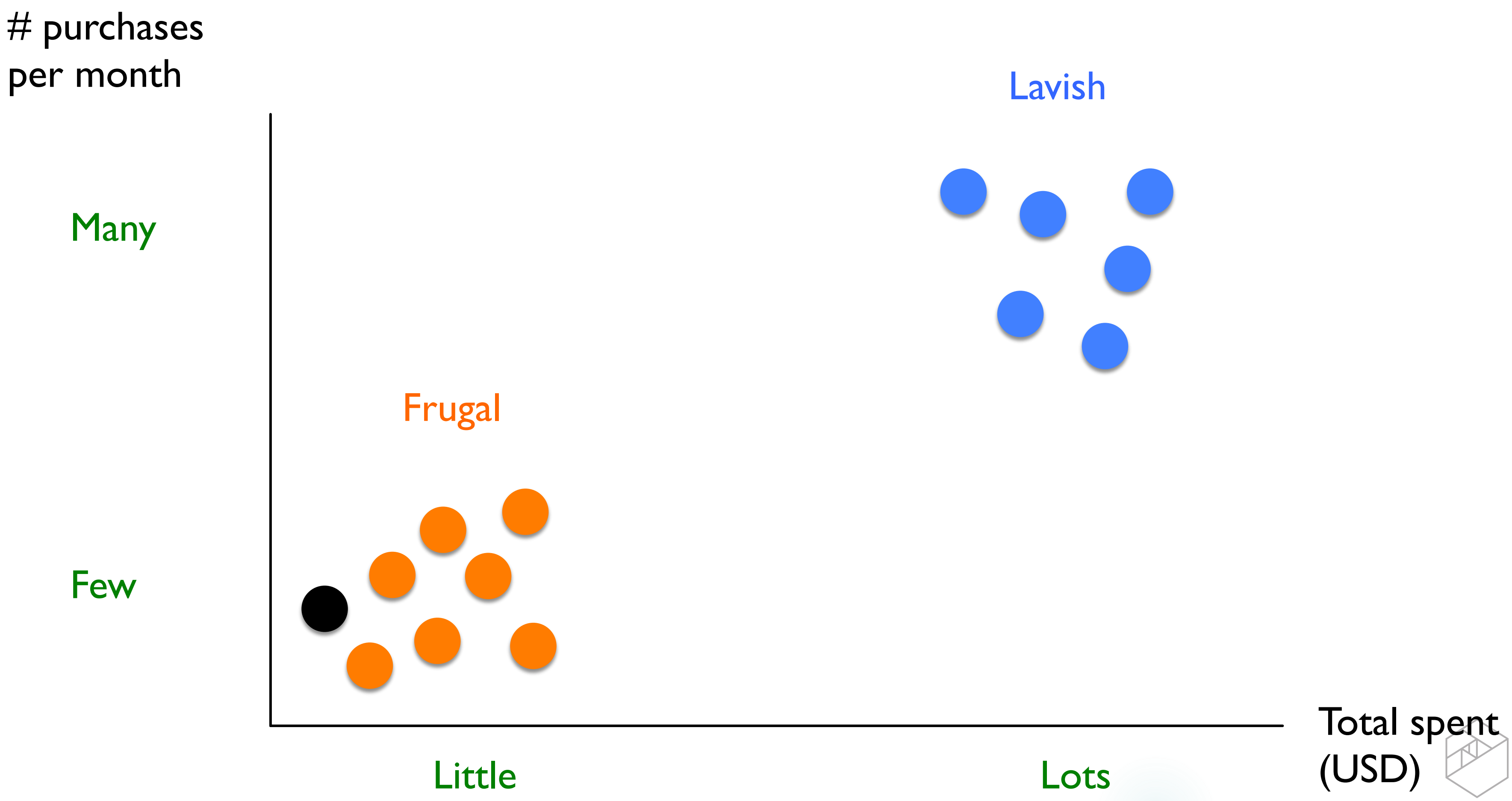
Introduction to Clustering

METIS

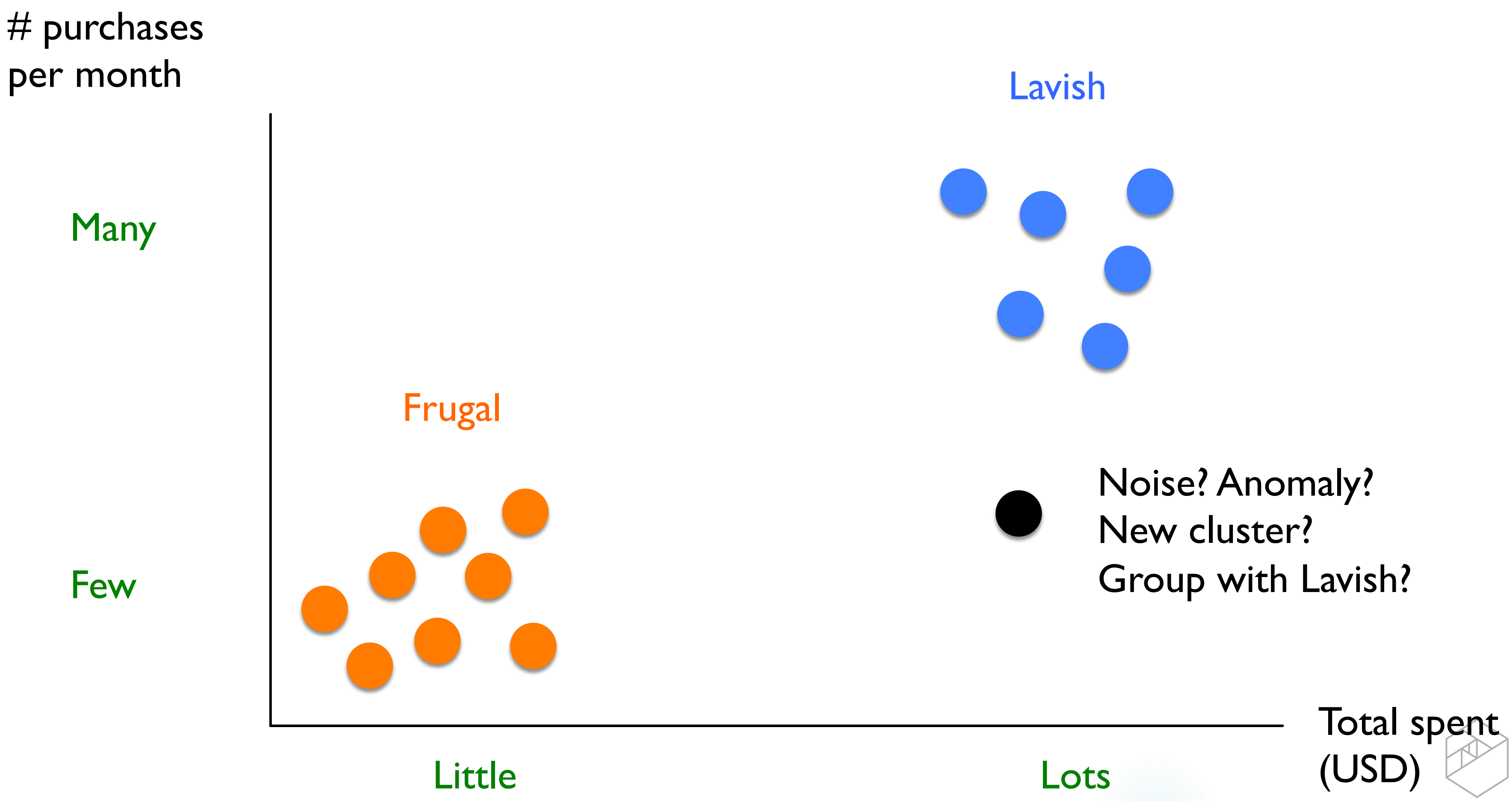
Cholera cases (black rectangles)
in London epidemic of 1854

- Physician John Snow proposed epidemic caused by contaminated water
- Identified contaminated water pump
  (other pumps too far away)


purchases
per month
Many
Few
Little
Lots
Total spent
(USD)



purchases
per month
Lavish
Many
Frugal
Few
Little
Lots
Total spent
(USD)



purchases
per month
Lavish
Many
Frugal
Noise? Anomaly?
New cluster?
Group with Lavish?
Few
Total spent
(USD)
Little
Lots

Motivation: clustering provides insight

- mechanism/motivation
- connectivity/correlations
- simplification/convenience

What's a cluster?

Intuitive definition:
group of data points that are close to each other

To make this computer friendly, need a mathematical definition of "close."

Close (most common definitions):
based on distance or density

Clustering as unsupervised learning

Unlabeled data $\xrightarrow{\text{algorithm}}$ Structured data

New data (unlabeled) $\xrightarrow{\text{assignment}}$ New data included in structure

Clustering vs. partitioning

Clustering:
points MAY be assigned to a cluster; could also be outliers

Partitioning:
points MUST be assigned to a cluster; no other categories

k-means clustering

k-means clustering*

A partitioning algorithm that divides the data into *k* clusters

Points are assigned to a cluster based on metric
(such as Euclidean distance) to nearest cluster centroid

Value of *k* is chosen by the user

*An example of clustering vs. partitioning confusion

k-means clustering: the algorithm

1. Choose *k* centroids
2. Assign points to cluster based on nearest centroid
3. Recompute centroids
4. Repeat steps (2) and (3) until algorithm converges

k-means: toy example

k-means: toy example

k-means: toy example

k-means: strengths and weaknesses

Strengths:

1. Simple—one parameter (k clusters)
2. Typically fast—for n points in d-dimensions, runtime is $O(nkdi)$ where i is number of iterations until convergence
3. Guaranteed to converge
4. Easy to implement

Weaknesses:

1. Optimal k is often not obvious
2. Can get trapped in local minima (initial conditions matter)
3. Sensitive to outliers (partitioning not clustering)
4. Scaling affects results

k-means: How to choose *k*

If you have an external constraint or domain knowledge, use it!
Example: customer segmentation study for a bank
that offers **five** types of savings account → $k = 5$

What if you don't have such knowledge?
Or you are exploring the possibility of offering more/fewer types of savings accounts?

k-means performance: inertia

Idea: good clustering → points close to cluster centroids

Quantify this idea: sum of squares of distances of points from corresponding cluster centroid should be small

Give it a name: call this sum **inertia**

k-means performance: inertia

$$I = \sum_{j=1}^{k} \sum_{x_i \in \text{cluster } j} | x_i - x_{c,j} |^2$$

sum over clusters

sum over points
in cluster *j*

position of point *i*
in cluster *j*

position of centroid
of cluster *j*

k-means performance: inertia

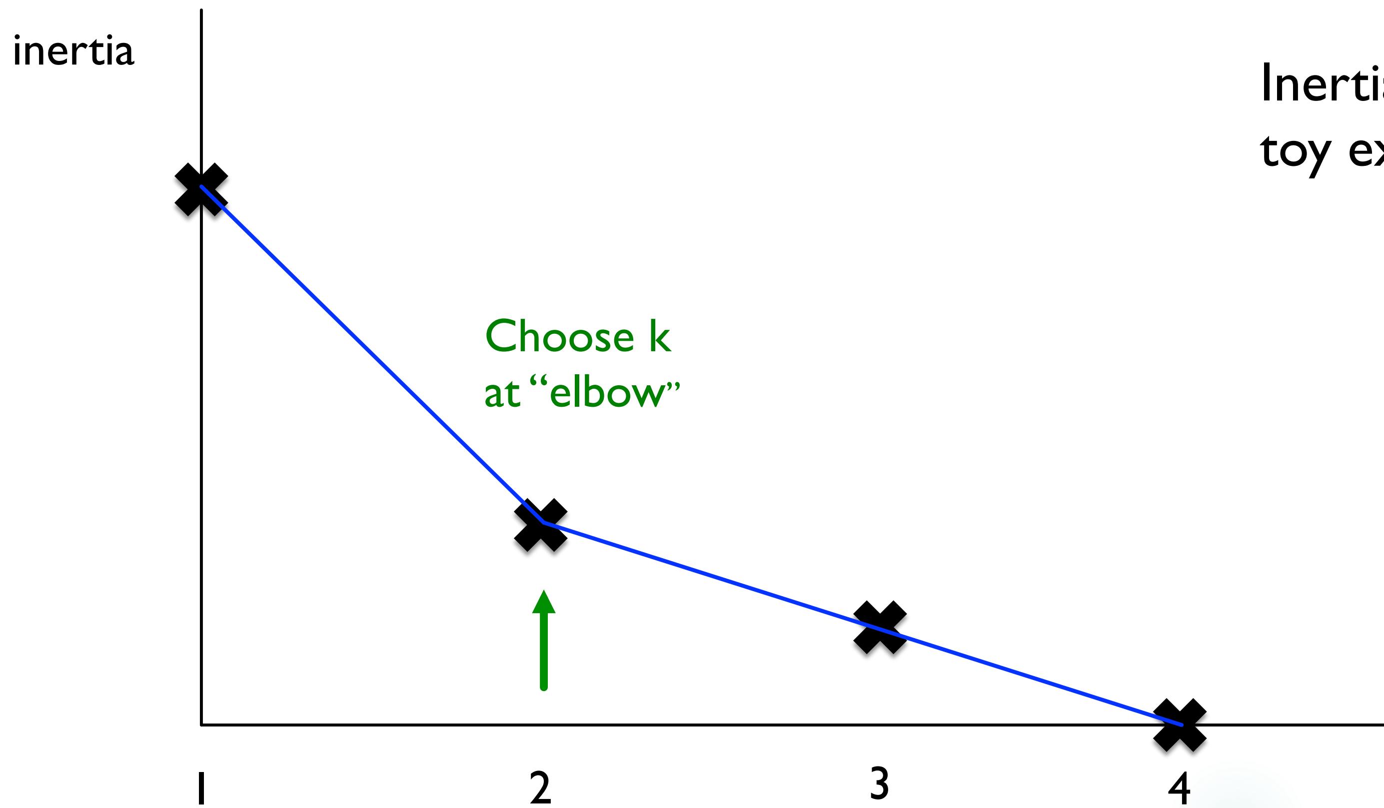
Intuition: want I as small as possible

Problem: $I \geq 0$

Minimum is zero, which occurs in two (useless) cases:

All points at same location ($I = 0$ for all k)
Number of clusters = number of points ($k = n$)

k-means performance: inertia



Inertia for
toy example

k-means performance: silhouette coefficient

Idea: good clustering → points close to cluster centroids **and** far away from other clusters

Quantify this idea: compare two distances for each point *i*

a(*i*): intra-cluster distance

b(*i*): inter-cluster distance

→ Calculate metric based on ratio

Give it a name: call this metric **silhouette coefficient**

k-means performance: silhouette coefficient

$a(i)$: mean distance between i and all other points in same cluster
$b(i)$: mean distance between i and all other points in nearest cluster that does not include i

silhouette coefficient

$$s(i) = \begin{cases} 1 - a(i)/b(i), & \text{if } a(i) < b(i) \\ 0, & \text{if } a(i) = b(i) \\ b(i)/a(i) - 1, & \text{if } a(i) > b(i) \end{cases}$$

k-means performance: silhouette coefficient

$$-1 \leq s(i) \leq 1$$

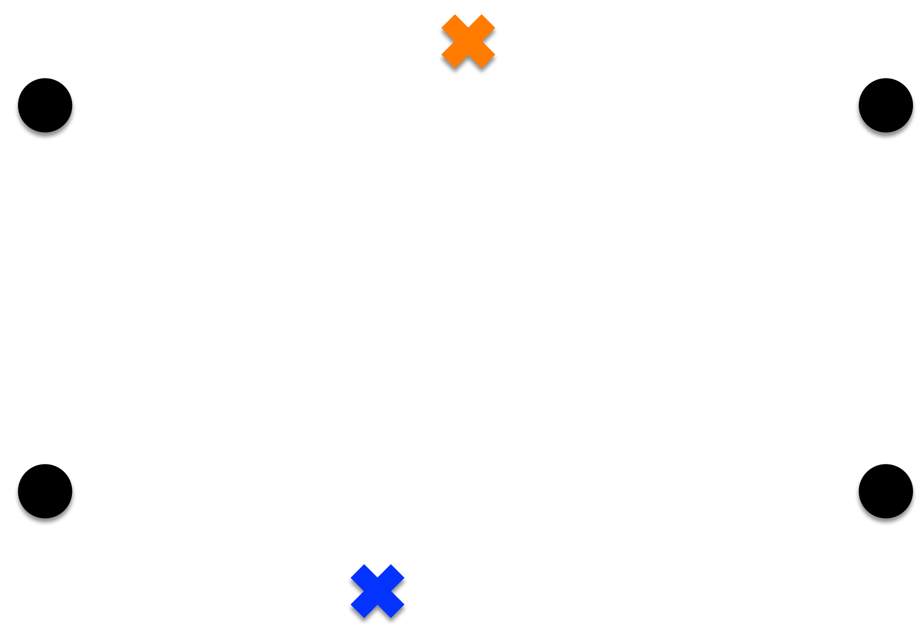
poor clustering

good clustering

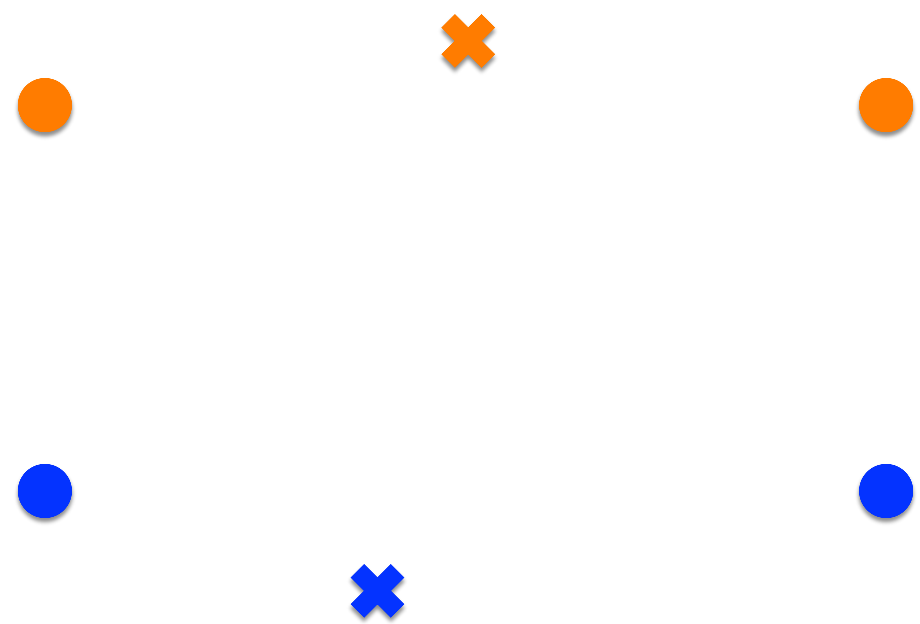
Choose k such that average silhouette coefficient over all clusters is largest

k-means: local minima

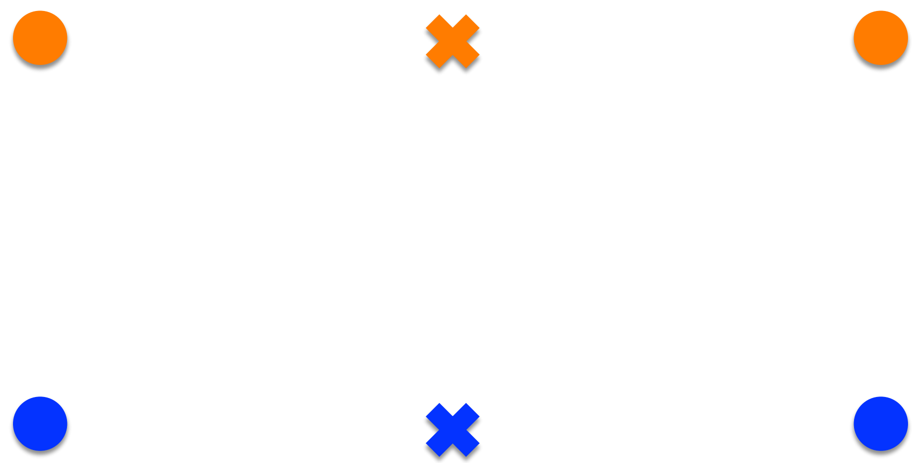
k-means: local minima



k-means: local minima

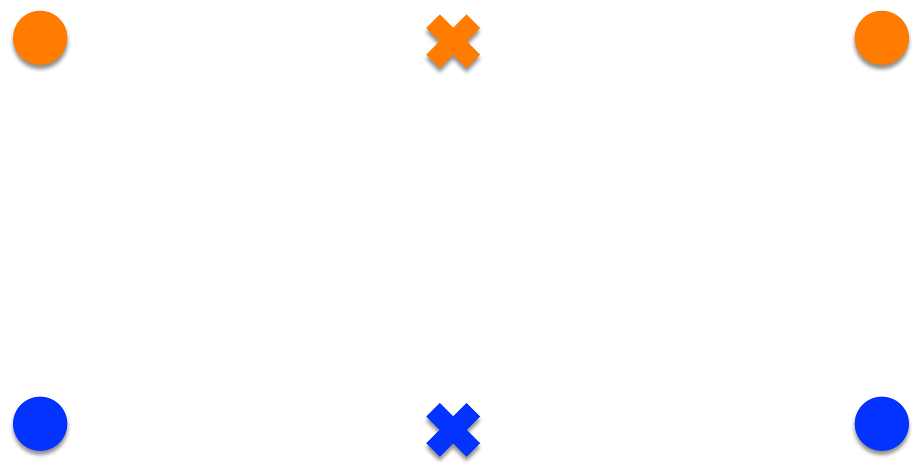


k-means: local minima



k-means: toy example

k-means: local minima



Moral: run k-means for various initial centroid guesses

k-means: adding new data

1. Add new data to nearest cluster

2. Treat clusters as labeled data
Use this data to train a classifier
Apply classifier to new data

Summary

- Clustering: unsupervised learning technique for grouping data
- k-means clustering: simple and popular partitioning algorithm
  - One parameter
  - Typically fast
  - Choice of *k* requires judgment
  - Implemented in scikit-learn
    from sklearn.cluster import Kmeans

    https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

Questions?