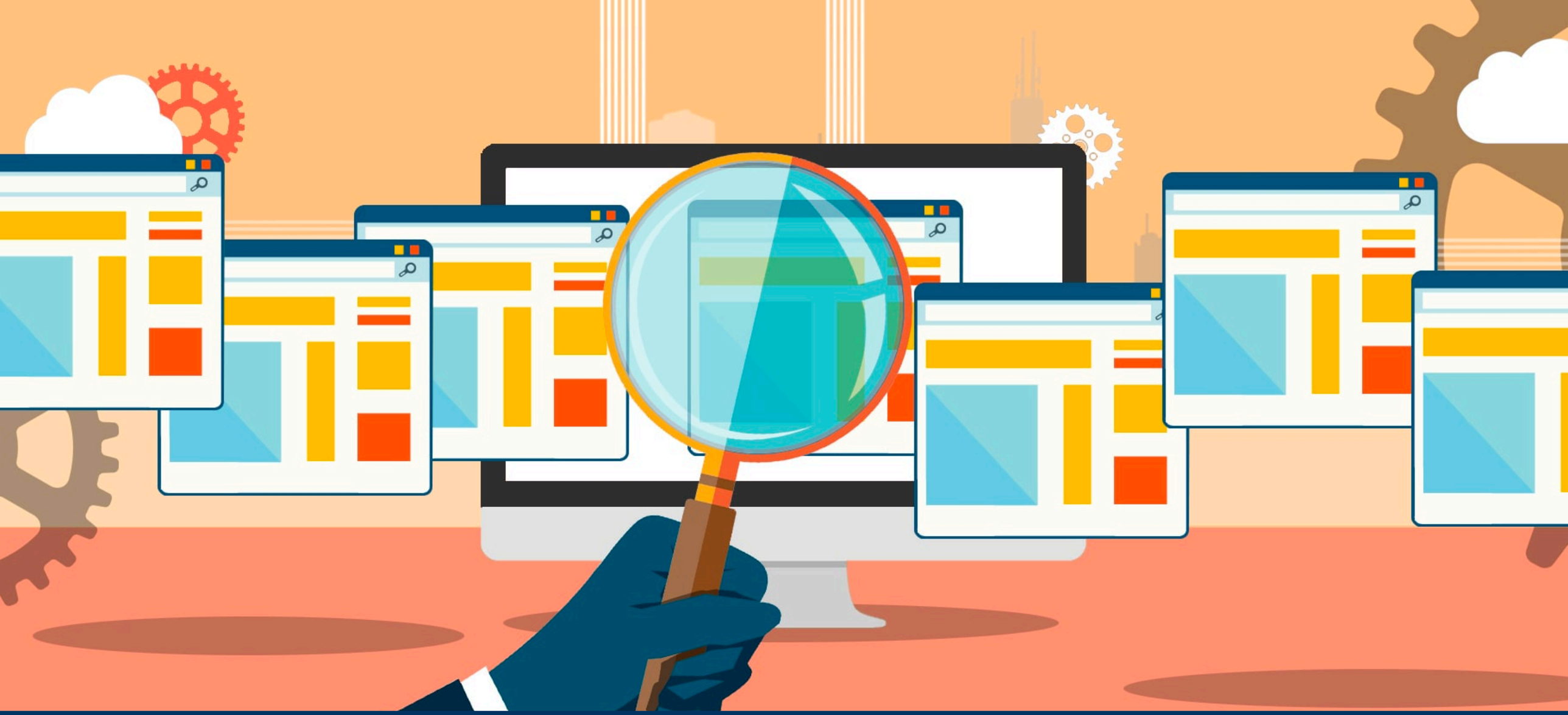


(Re)Introducing Clustering

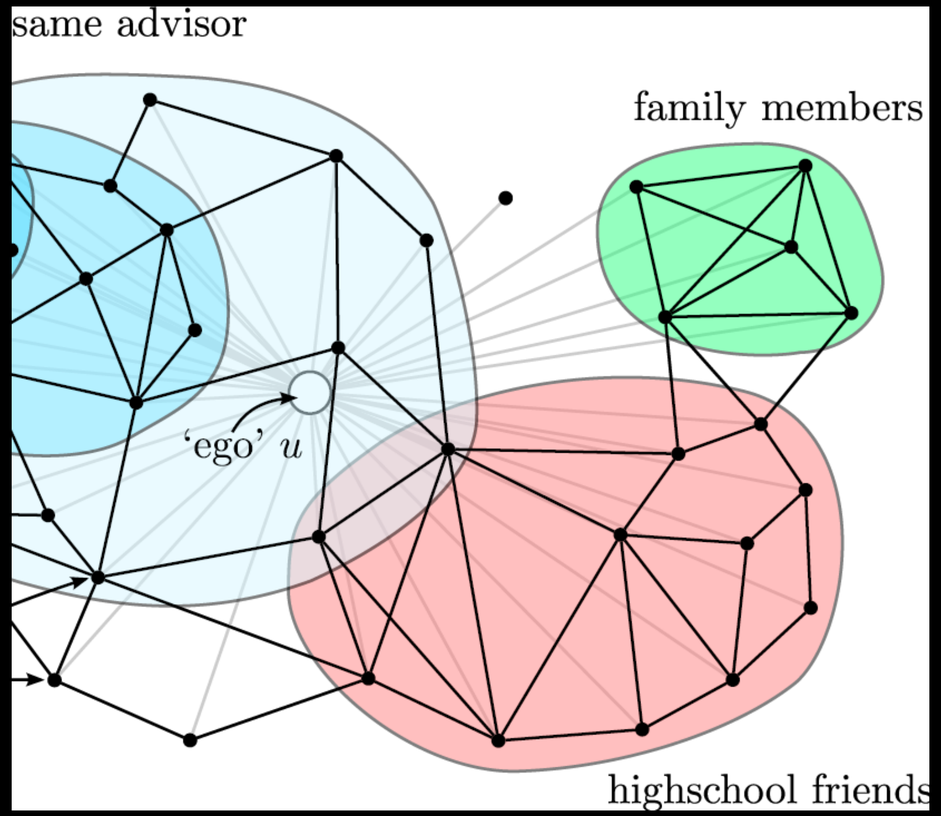
INST414 - Data Science Techniques

Six Core Learning Objectives

1. Collect and clean large-scale datasets
2. Articulate the math behind supervised and unsupervised techniques
3. Execute supervised and unsupervised machine learning techniques
4. Select and evaluate various types of machine learning techniques
5. Explain the results coming out of the models
6. Critically evaluate the accuracy of different algorithms and the appropriateness of a given approach

Where are we?

2. Articulate the math behind supervised and unsupervised techniques



How do we generate clusters?

"Supervised" vs. "unsupervised"?

Where are we?

3. Execute supervised and unsupervised machine learning techniques

Apply Clustering

We'll use k-Means from Sklearn to cluster this

```
from sklearn.cluster import KMeans

k = 4

model = KMeans(n_clusters=k)
```

```
model.fit(matrix_reduced)
```

```
KMeans(n_clusters=4)
```

Applying clustering algos to data

Identifying reasonable cluster counts

This Module's Learning Objectives

Week 1

Differentiate between unsupervised and supervised machine learning

Formally define "clustering"

Describe how one represents a cluster of multiple points

Explain how one can calculate distances between clusters

This Module's Learning Objectives

Week 1

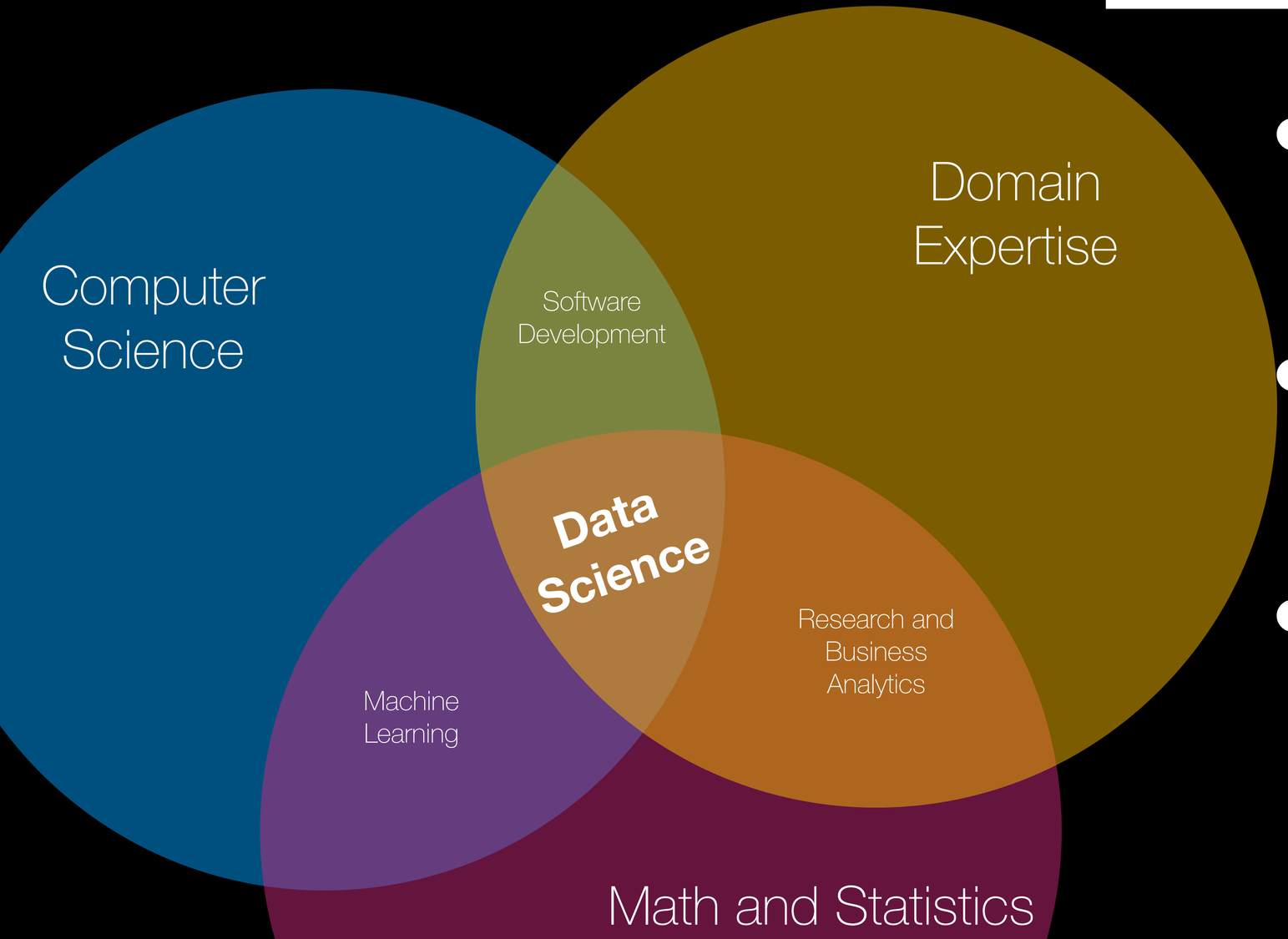
Differentiate between unsupervised and supervised machine learning

Formally define "clustering"

Describe how one represents a cluster of multiple points

Explain how one can calculate distances between clusters

Data Science ≠ Machine Learning



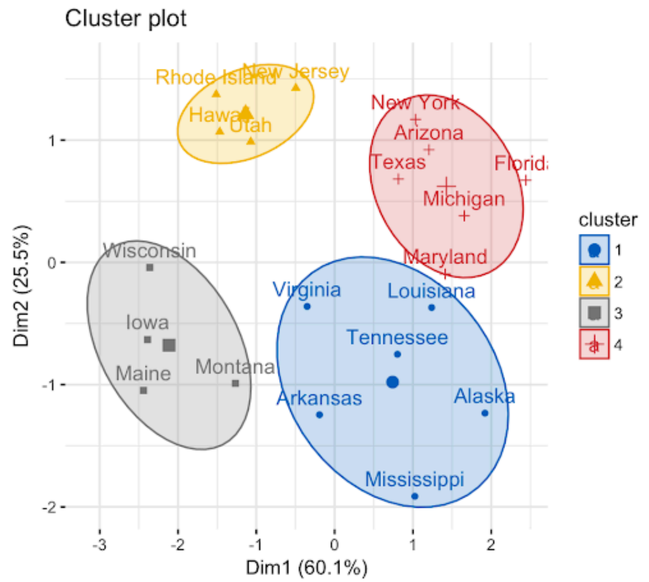
- Data science *includes* machine learning
- But also extracting and visualizing insights
- Not all insights require sophisticated models

So what *is* “Machine Learning”?

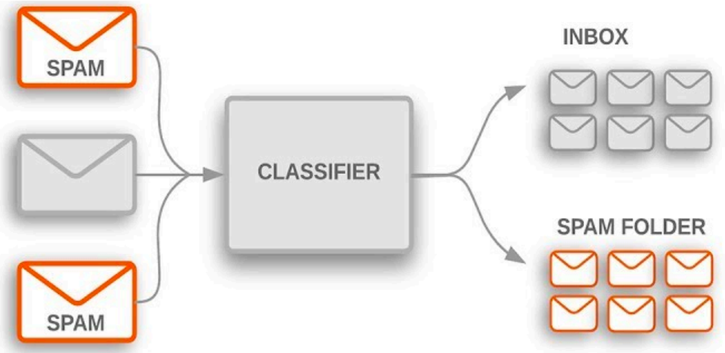
Machine learning is…

An overloaded term

The study of algorithms that “learn” from data



“Learn” some structure in the data



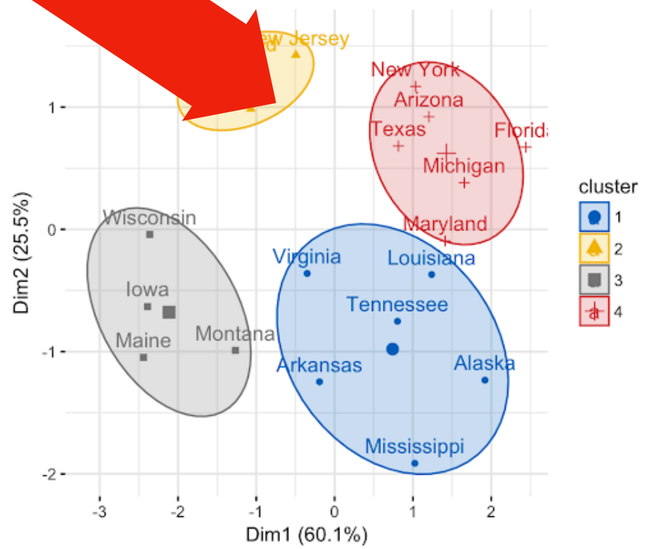
“Learn” to generalize from examples of a task

Two main types of machine learning problems…

Unsupervised learning

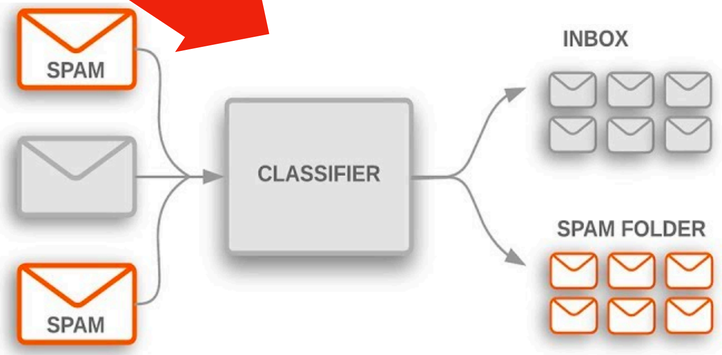
Supervised learning

Unsupervised



"Learn" some structure in the data

Supervised



"Learn" to generalize from examples of a task

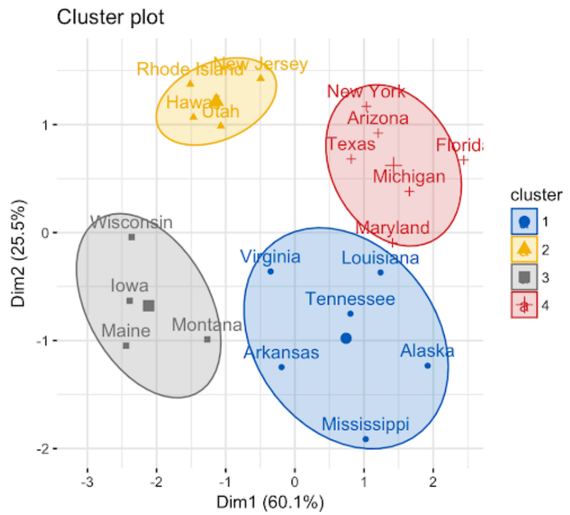
Unsupervised:
You want to find structure in your data, but you don't have examples of this structure

How do I know if the problem is "unsupervised" or "supervised"?

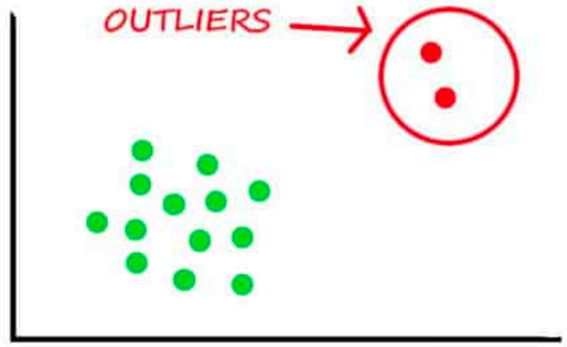
Supervised:
You want to recover (predict) <u>known relationships</u> in your data, and you have examples (i.e., labels) of these relations

Do you have "Labels"?

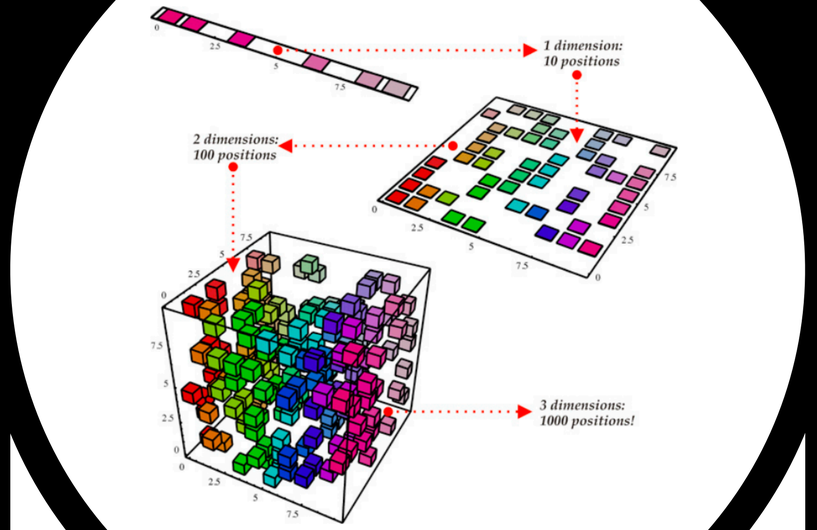
Unsupervised Learning



Extracting “similar” elements (clustering)

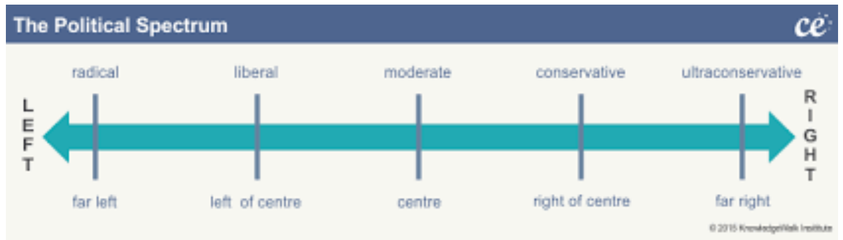


Identifying “dissimilar” elements (outliers)

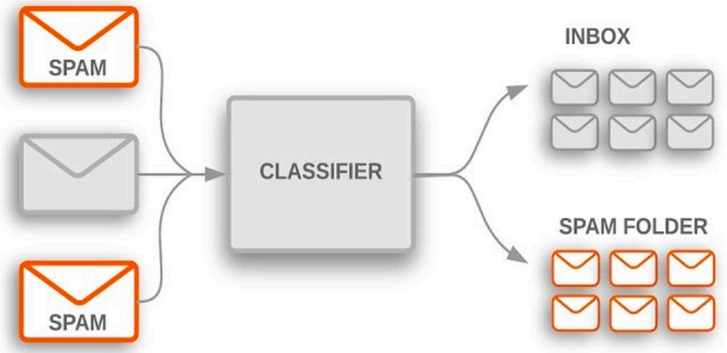


Extracting latent structure (e.g., dim. reduction)

Supervised Learning



Predicting scores based on known, continuous labels (regression)



Classifying elements based on known, categorical labels (classification)

This Module's Learning Objectives

Part 1

- Differentiate between unsupervised and supervised machine learning
- Compare and contrast the two approaches for hierarchical clustering
- Describe how one represents a cluster of multiple points
- Explain how one can calculate distances between clusters

This Module's Learning Objectives

Week 1

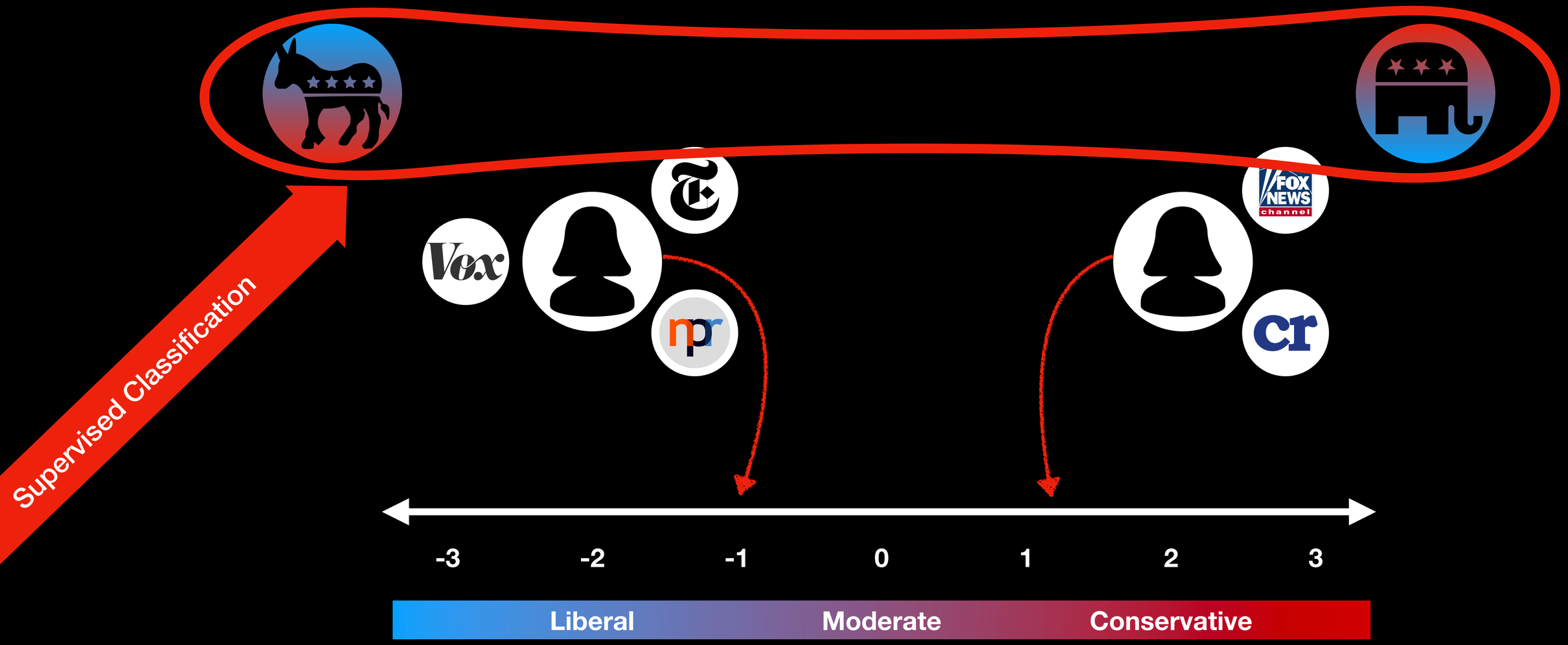
Differentiate between unsupervised and supervised machine learning

Formally define "clustering"

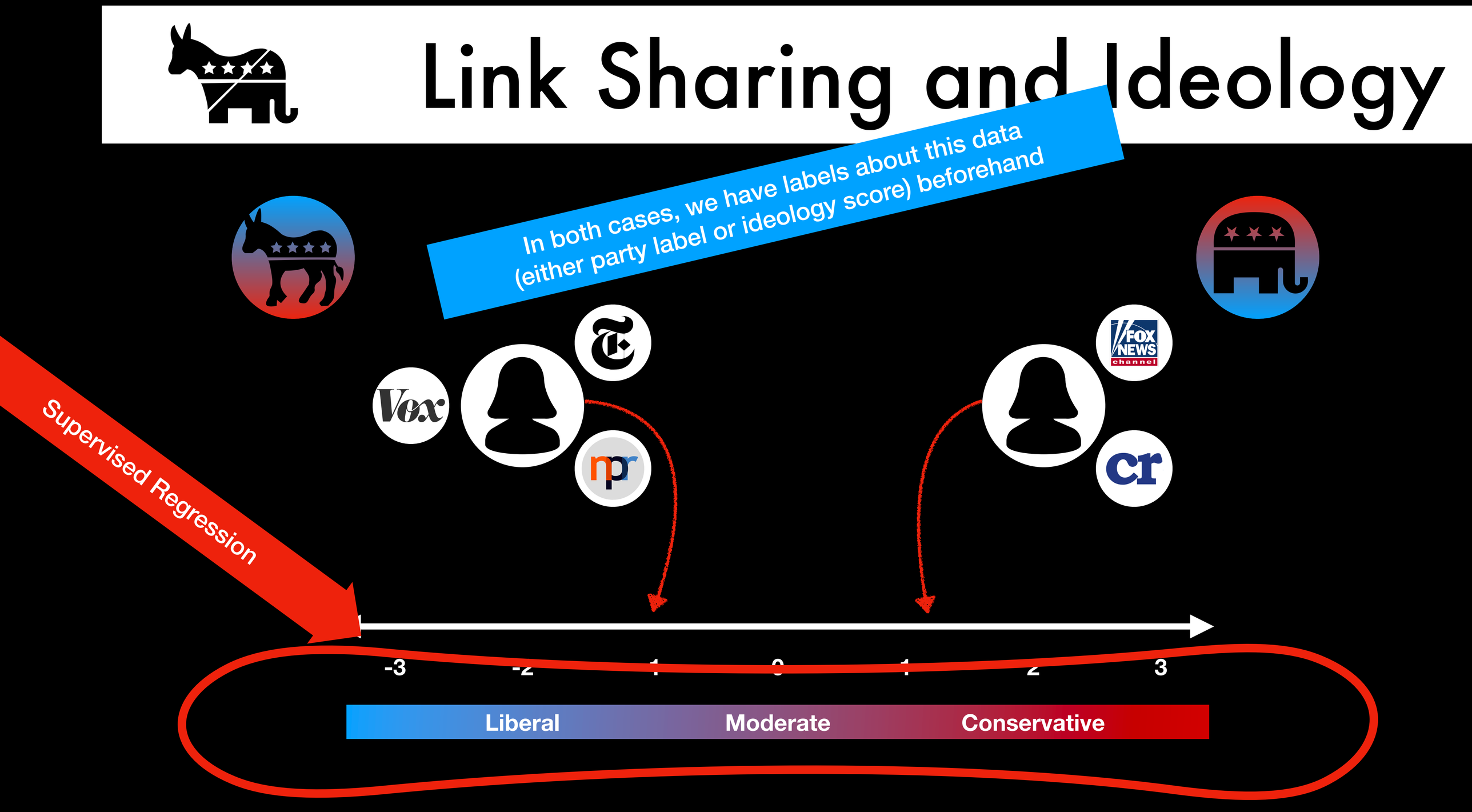
Describe how one represents a cluster of multiple points

Explain how one can calculate distances between clusters

Link Sharing and Ideology

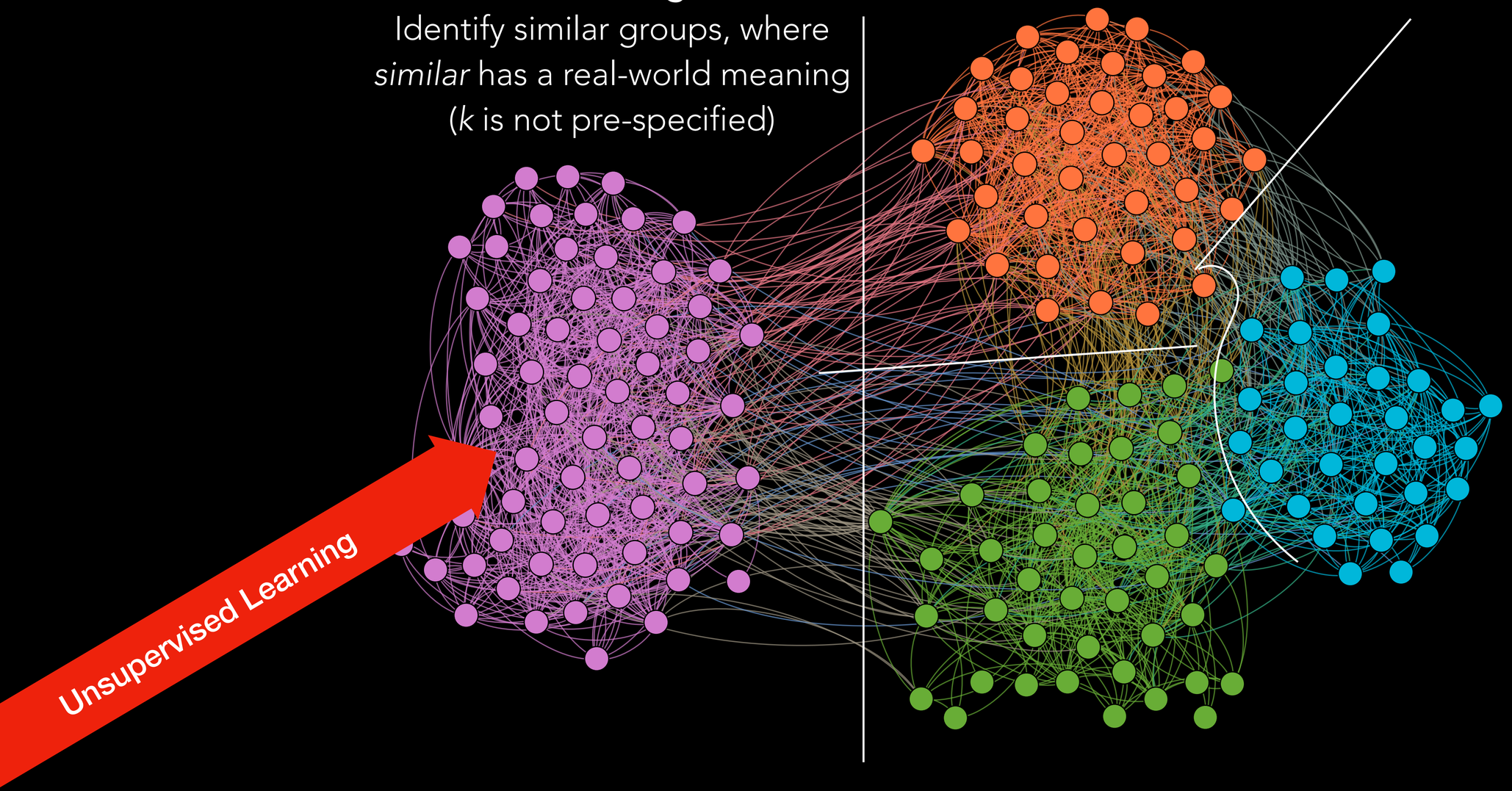


Link Sharing and Ideology



If we don't know what these clusters represent beforehand…

Clustering:
Identify similar groups, where *similar* has a real-world meaning
(*k* is not pre-specified)

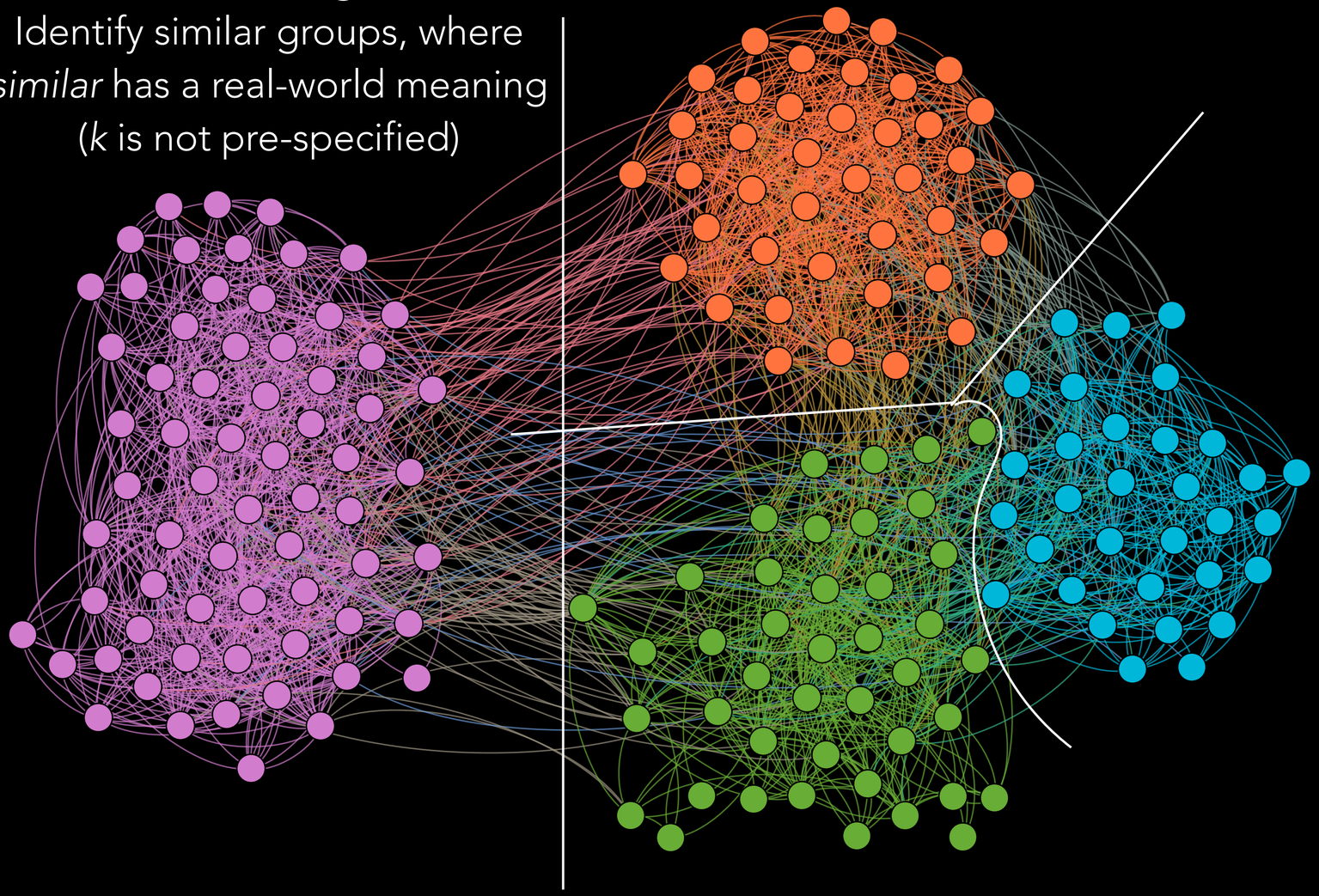


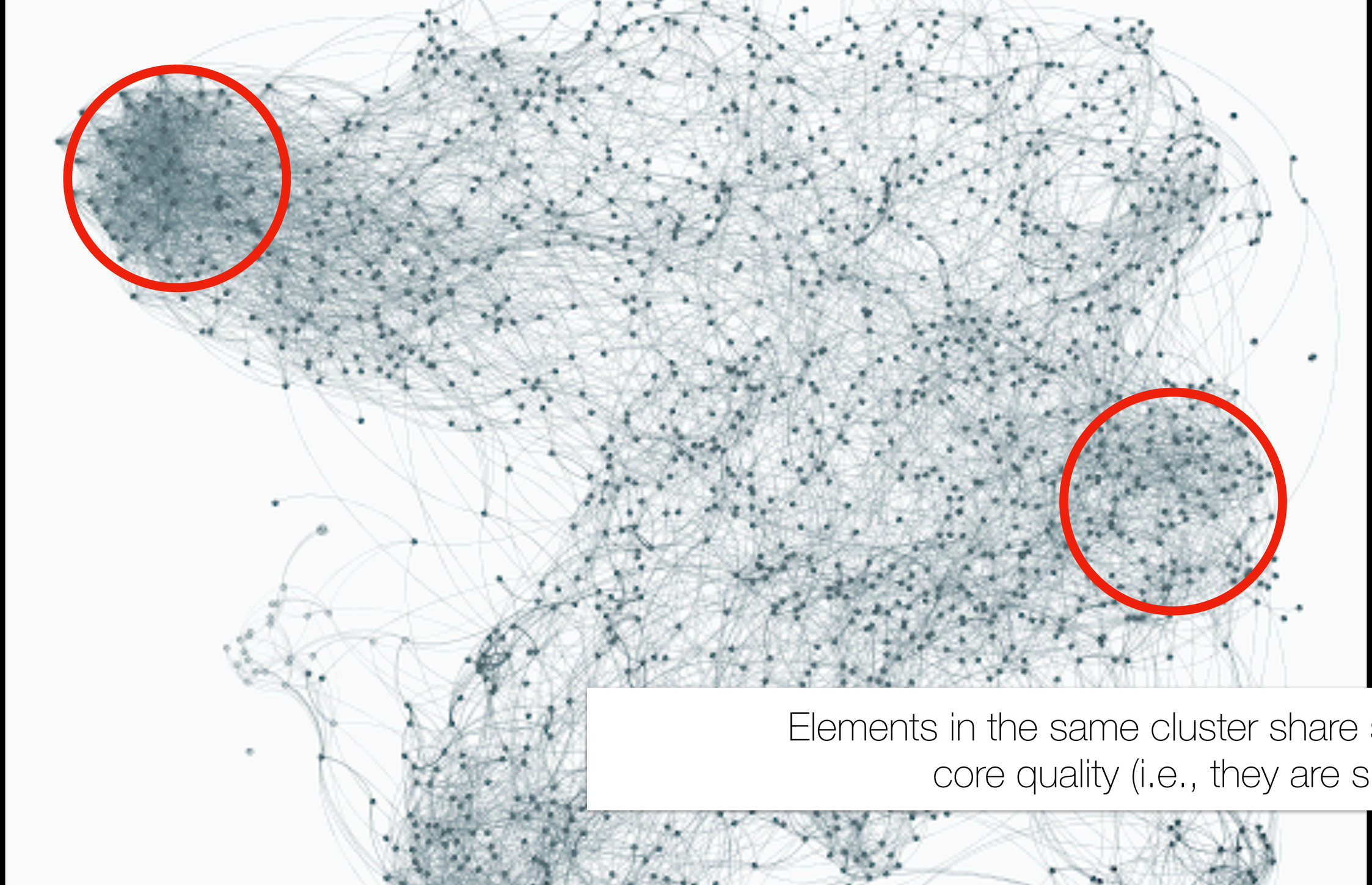
Core Motivation: Finding actionable insights from data

One solution is to group content

Clustering:

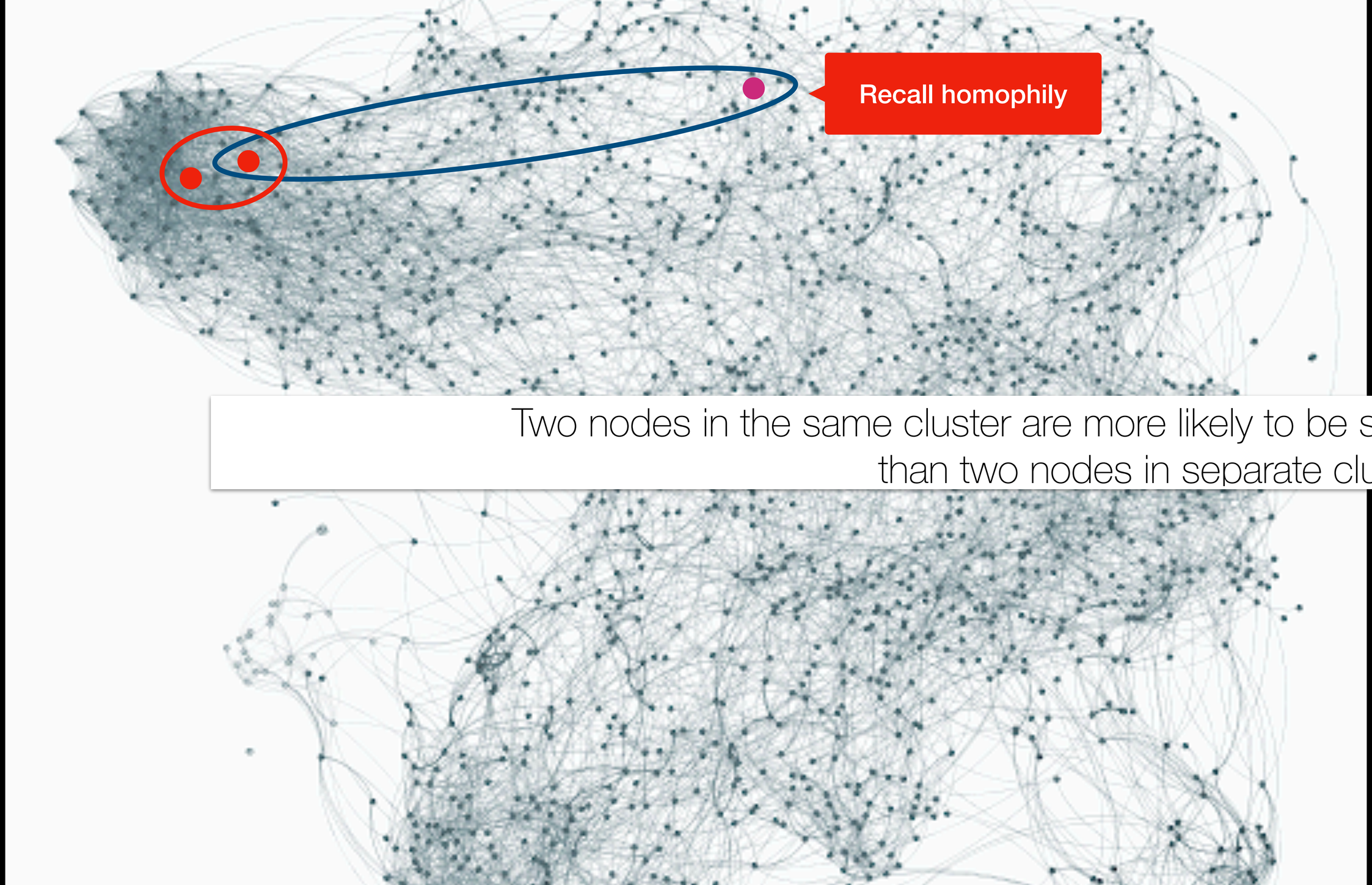
Identify similar groups, where *similar* has a real-world meaning (*k* is not pre-specified)



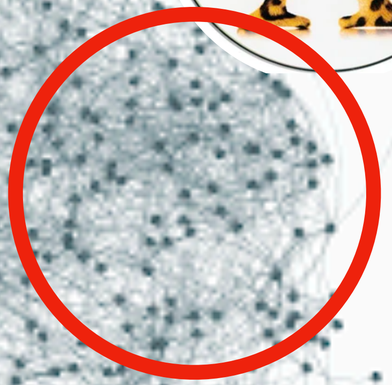


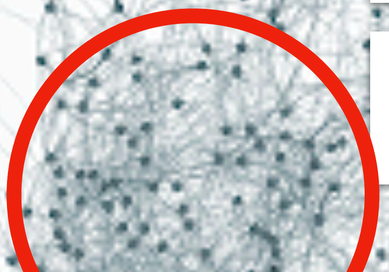
Elements in the same cluster share some core quality (i.e., they are similar)


JAGUAR
Mac OS X v10.2
Install Disc 1



Two nodes in the same cluster are more likely to be similar than two nodes in separate clusters





Now know how to calculate similarity

How do we build these clusters?

Core Motivation: Finding actionable insights from data

Clusters of entities should be similar

Corollary: Can build clusters from similar entities

We have multiple metrics for measuring similarity

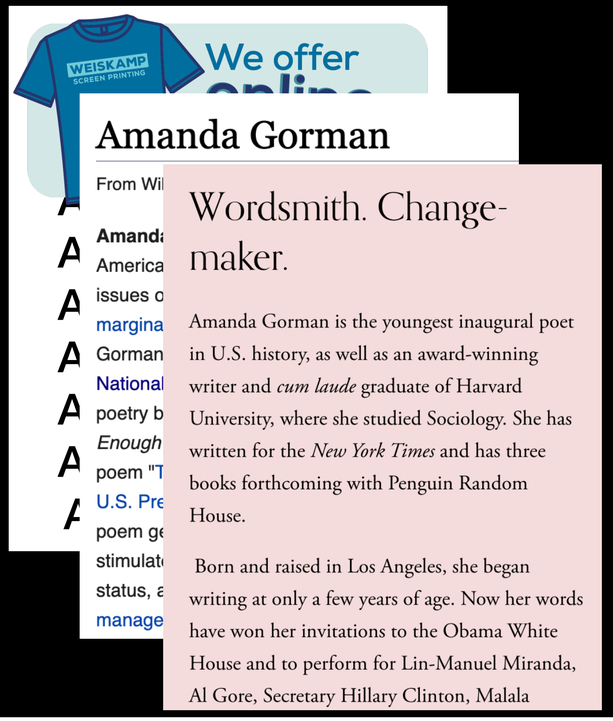
Alternate Definitions of Similarity

Adjacency Matrix *A*

	A	B	C	D	E
A	0	0	1	0	1
B	1	0	0	0	0
C	0	1	0	0	0
D	1	0	0	0	1
E	0	0	0	1	0

Feature Matrix
N x M

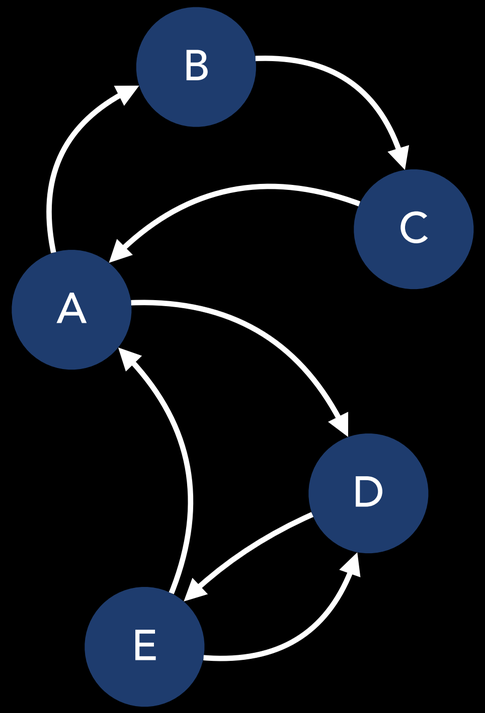
$$\begin{bmatrix} r_A, r_B, \ldots r_E \end{bmatrix}$$



Nodes with similar sets of neighbors

Nodes in with similar features

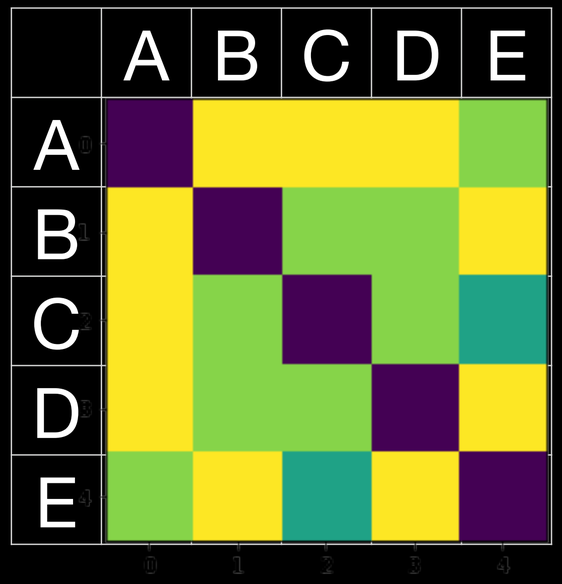
Pages about the same topic



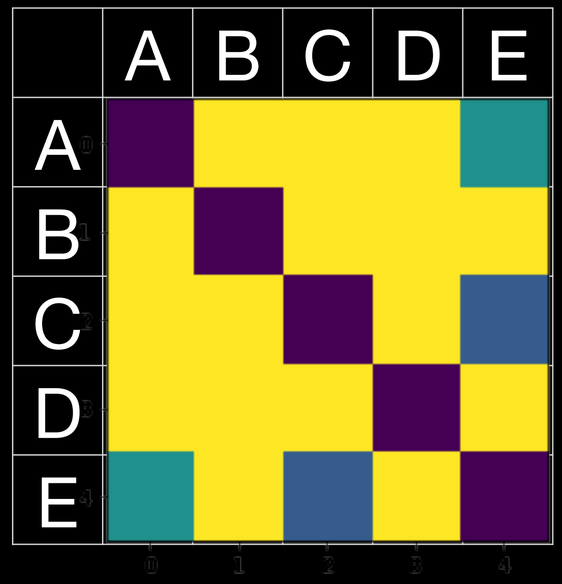
Adjacency Matrix *A*

	A	B	C	D	E
A	0	0	1	0	1
B	1	0	0	0	0
C	0	1	0	0	0
D	1	0	0	0	1
E	0	0	0	1	0

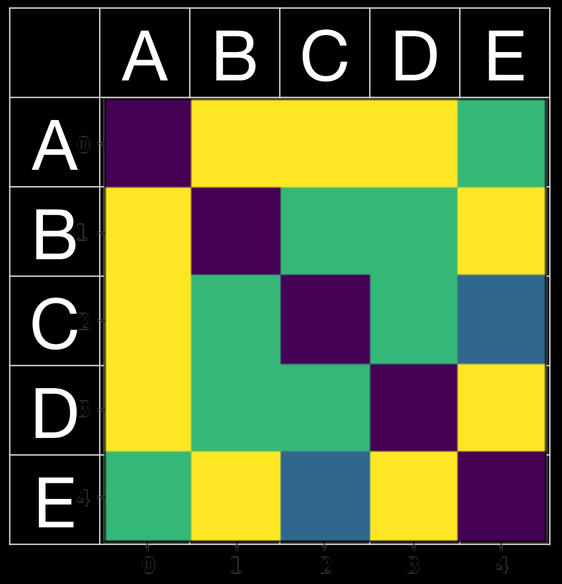
Euclidean Distance



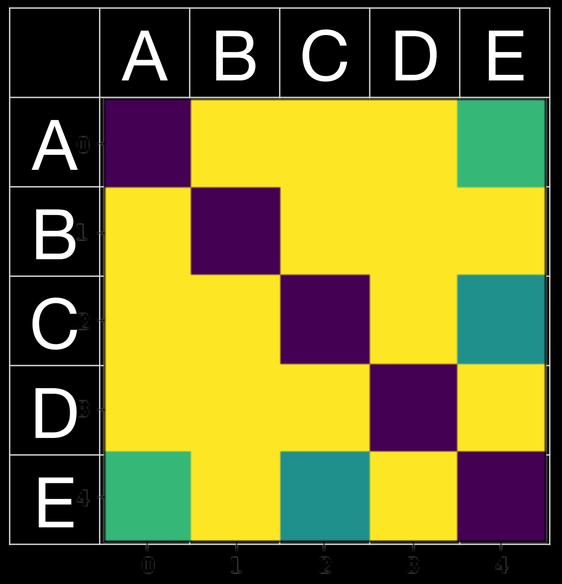
Cosine Distance



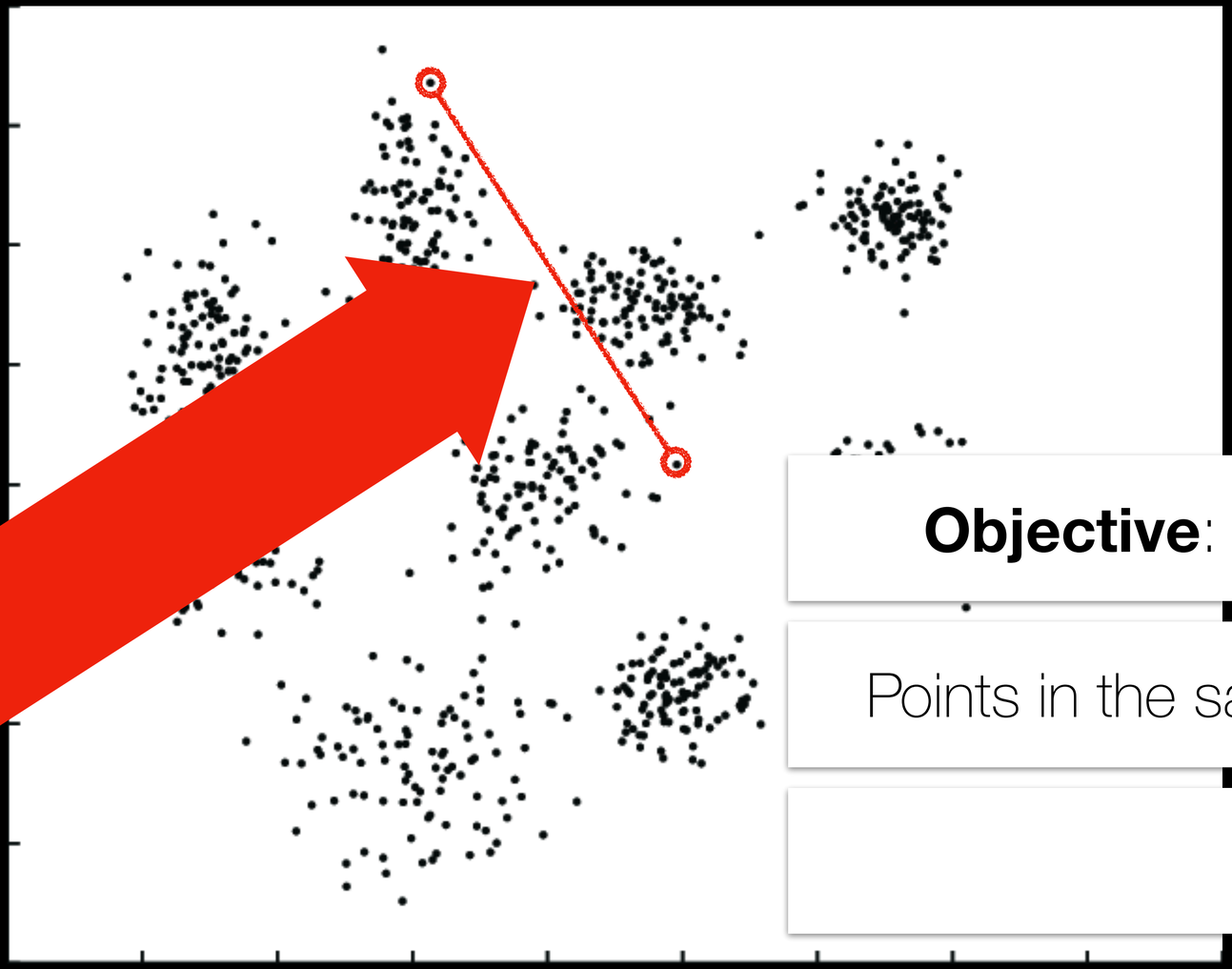
Hamming Distance



Jaccard Distance



A Formal Definition for Clustering



Given a **set of points**…

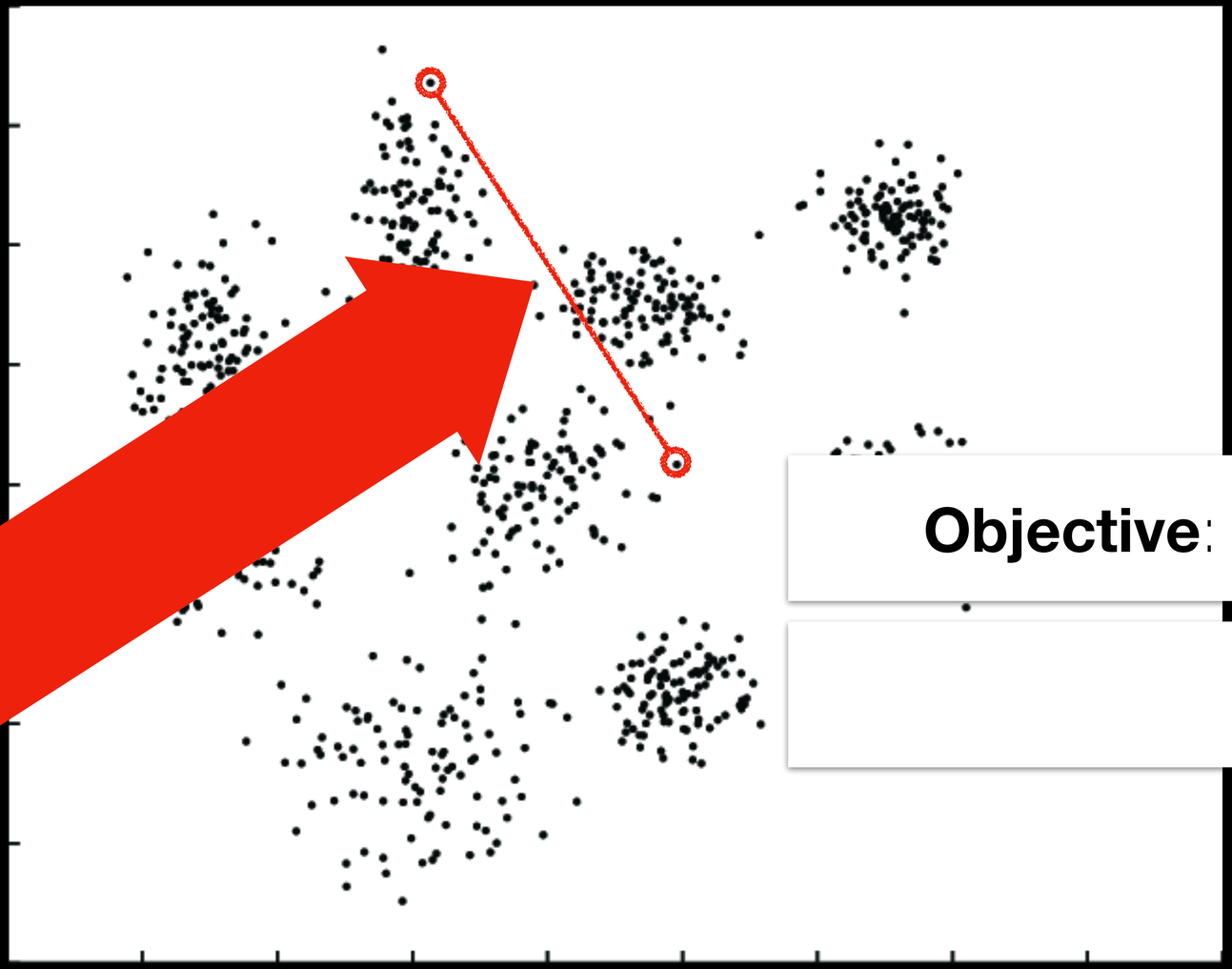
And a "**distance**" among them

Objective: Group points into ***k*** clusters, such that…

Points in the same cluster are <u>similar</u> to each other, and

Points in <u>different</u> clusters are dissimilar

A Formal Definition for Clustering



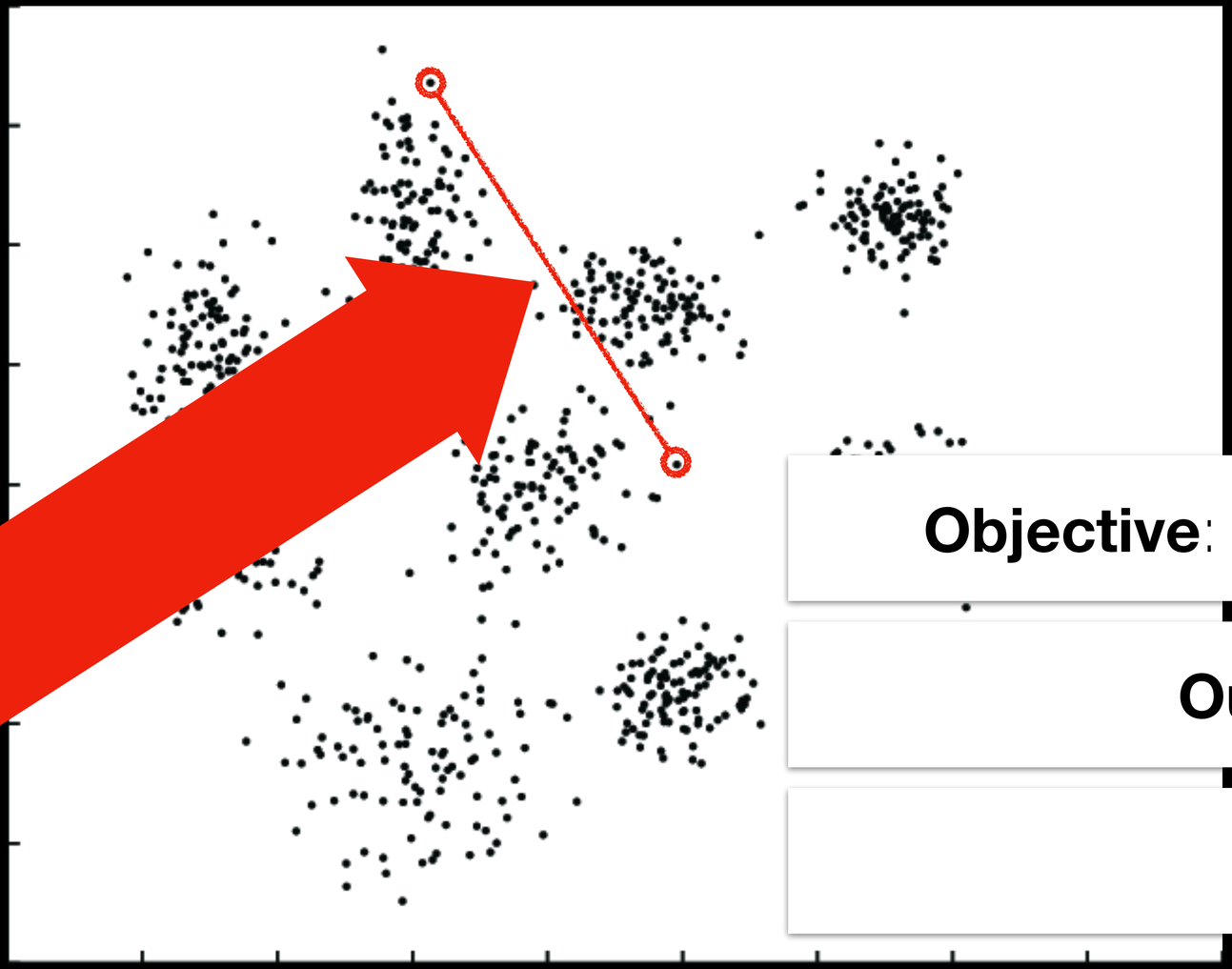
Given a **set of points**…

And a "**distance**" among them

Objective: Group points into ***k*** clusters, such that…

Output: ???

A Formal Definition for Clustering



Given a **set of points**…

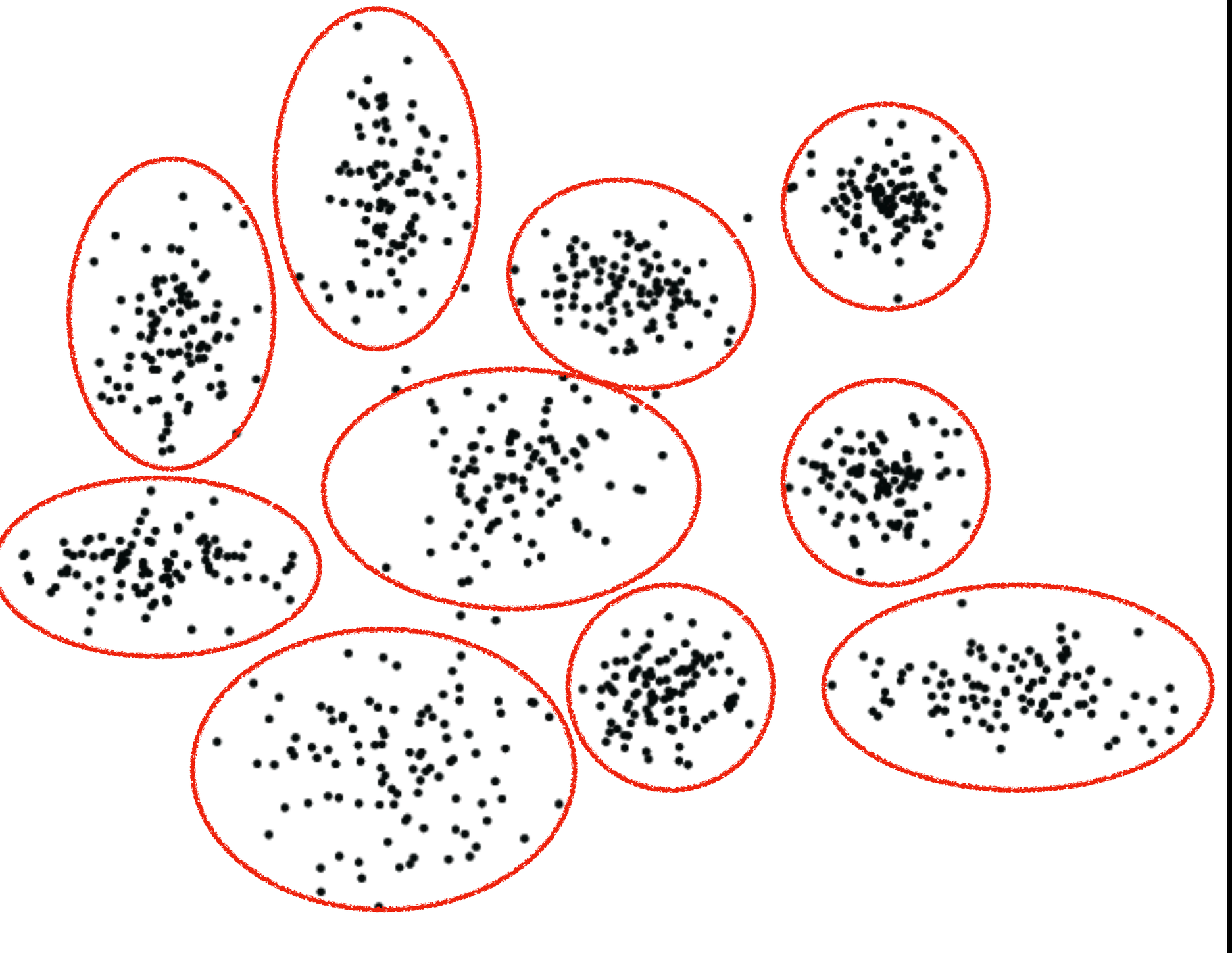
And a “**distance**” among them

Objective: Group points into ***k*** clusters, such that…

Output: List of assignments for each point

Of the form <data point, cluster id>

How many clusters are there in this image?



What’s should clustering output be here?

Cluster Assignments

Point	Cluster
d1	0
d2	0
d3	0
…	…
dn	10

What's should clustering output be here?

This Module's Learning Objectives

Week 1

Differentiate between unsupervised and supervised machine learning

Formally define "clustering"

Describe how one represents a cluster of multiple points

Explain how one can calculate distances between clusters

This Module's Learning Objectives

Week 1

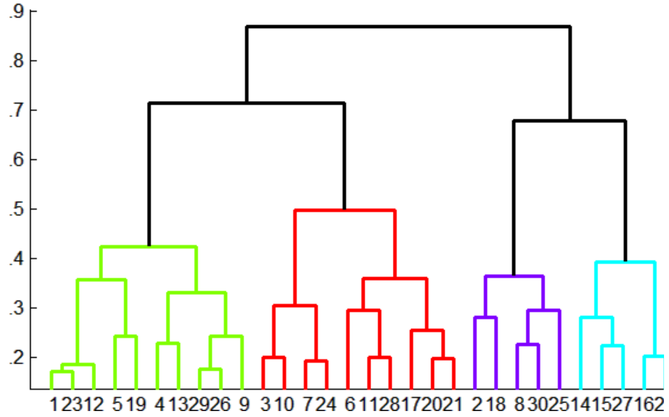
Differentiate between unsupervised and supervised machine learning

Formally define "clustering"

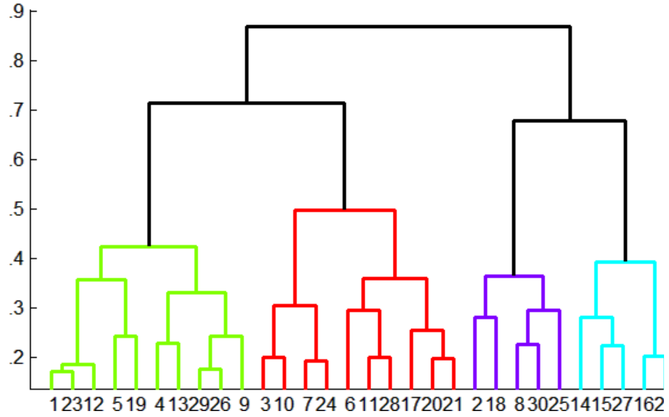
Describe how one represents a cluster of multiple points

Explain how one can calculate distances between clusters

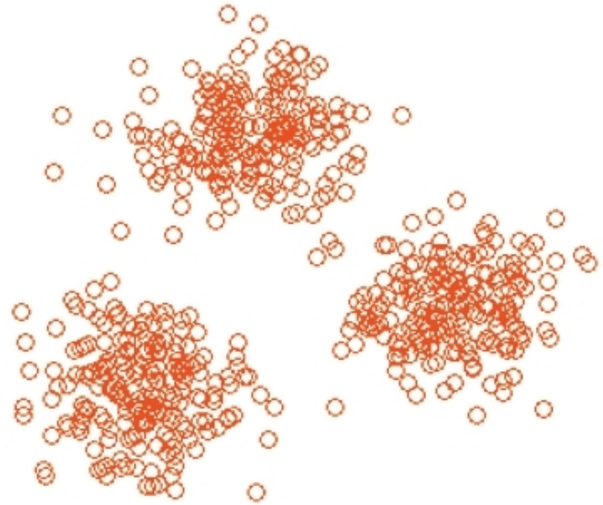
So how might we cluster in given space?



Hierarchical Clustering: Group or divide clusters



Hierarchical Clustering:
Group or divide clusters

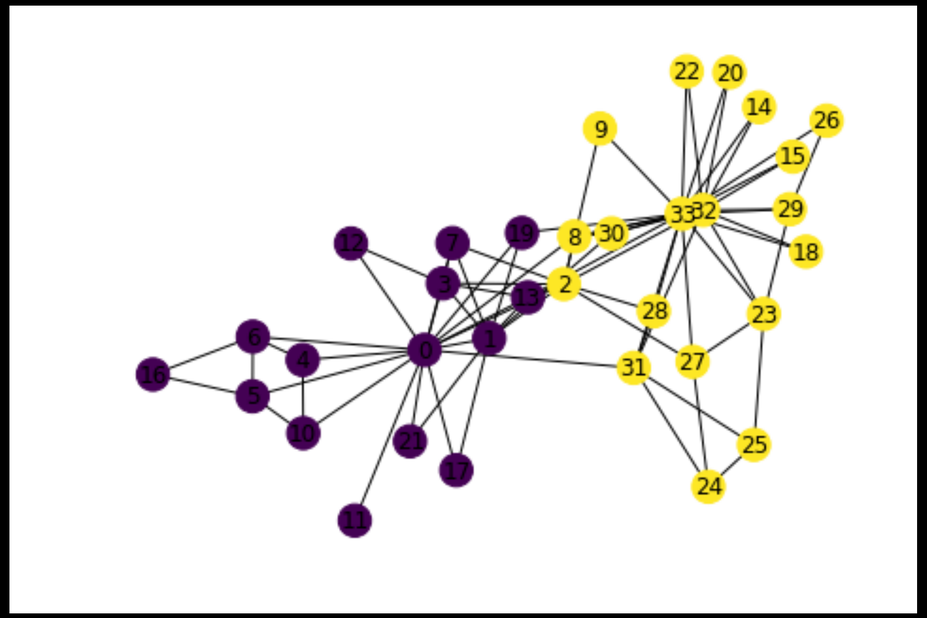


Point-Assignment Clustering:
Directly assign each
point to a single cluster

How do you think hierarchical and point-assignment clustering different?

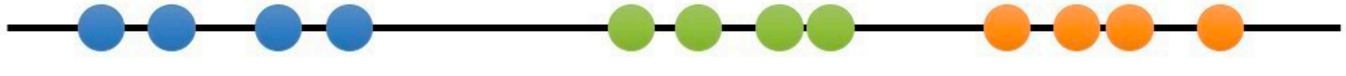
In hierarchical clustering, cluster # changes over time

Girvan-Newman Clustering

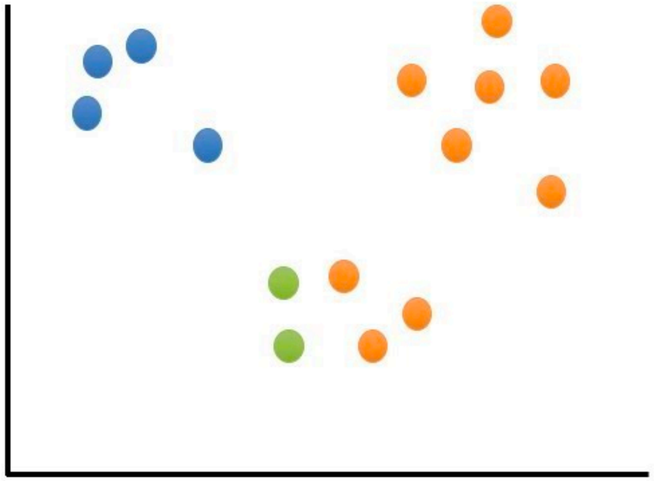


We've seen divisive clustering before…

Several community detection algorithms are point-assignment method



K-Means Clustering…

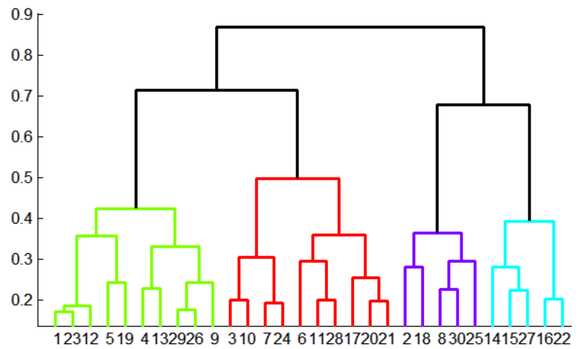


Next time…

K-Means is also a popular point-assignment method

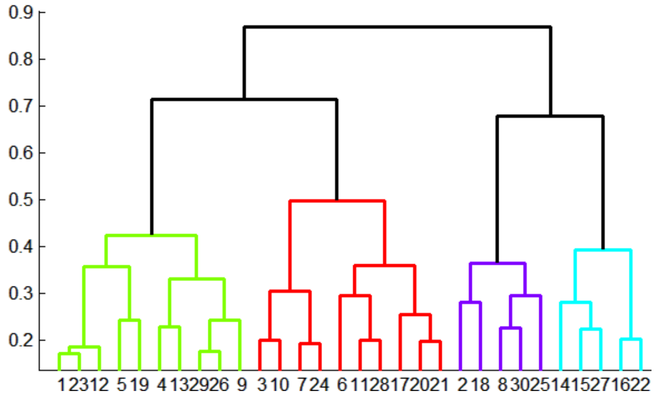
Overview: Methods of Clustering

- **Hierarchical:**
  - **Agglomerative** (bottom up):
    - Initially, each point is a cluster
    - Repeatedly combine the two "nearest" clusters into one



Agglomerative Clustering

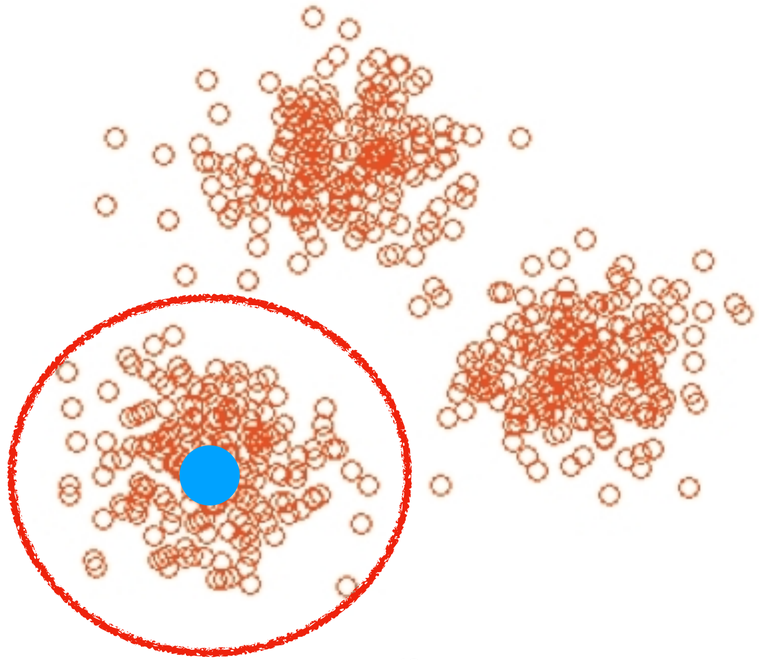
- **Key operation:**
  Repeatedly combine two nearest clusters



- **Three important questions:**
  - **1)** How do you represent a cluster of more than one point?
  - **2)** How do you determine the "nearness" of clusters?
  - **3)** When to stop combining clusters?

Agglomerative Clustering

- **Key operation: Repeatedly combine two nearest clusters**
- **(1) How to represent a cluster of many points?**
  - **Key problem:** As you merge clusters, how do you represent the "location" of each cluster, to tell which pair of clusters is closest?
- **Euclidean case:** each cluster has a ***centroid***
  - average of its (data) points

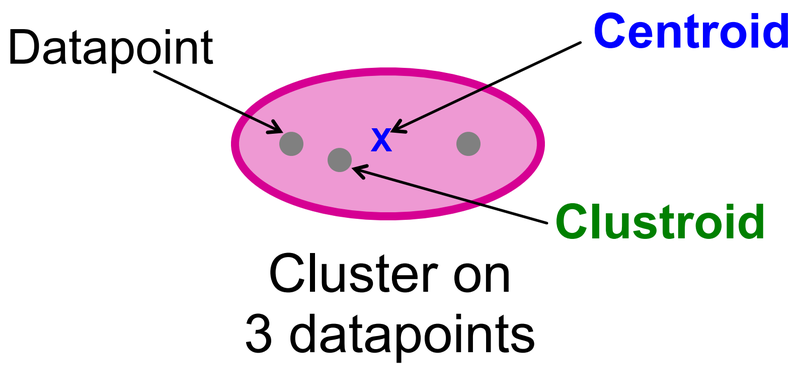


And in the Non-Euclidean Case?

What about the Non-Euclidean case?

- The only "locations" we can talk about are the points themselves
  - i.e., there is no "average" of two points

- **(1) How to represent a cluster of many points?**
  clustroid = (data)point "***closest***" to other points



"Closest" Point?

- **(1) How to represent a cluster of many points?**
  clustroid = point "***closest***" to other points
- **Possible meanings of "closest":**
  - Smallest maximum distance to other points
  - Smallest average distance to other points
  - Smallest sum of squares of distances to other points

Identifying “closest” point means setting similarity, but by what metric?

It depends

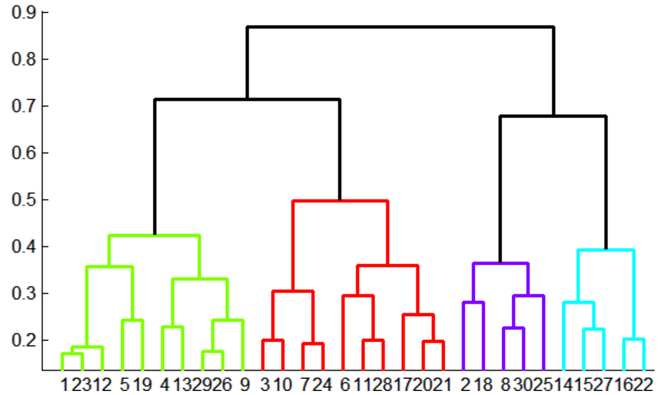
This Module's Learning Objectives

Week 1

- Differentiate between unsupervised and supervised machine learning
- Formally define "clustering"
- Describe how one represents a cluster of multiple points
- Explain how one can calculate distances between clusters

Agglomerative Clustering

- **Key operation:**
  Repeatedly combine two nearest clusters



- **Three important questions:**
  - **1)** How do you represent a cluster of more than one point?
  - **2)** How do you determine the "nearness" of clusters?
  - **3)** When to stop combining clusters?

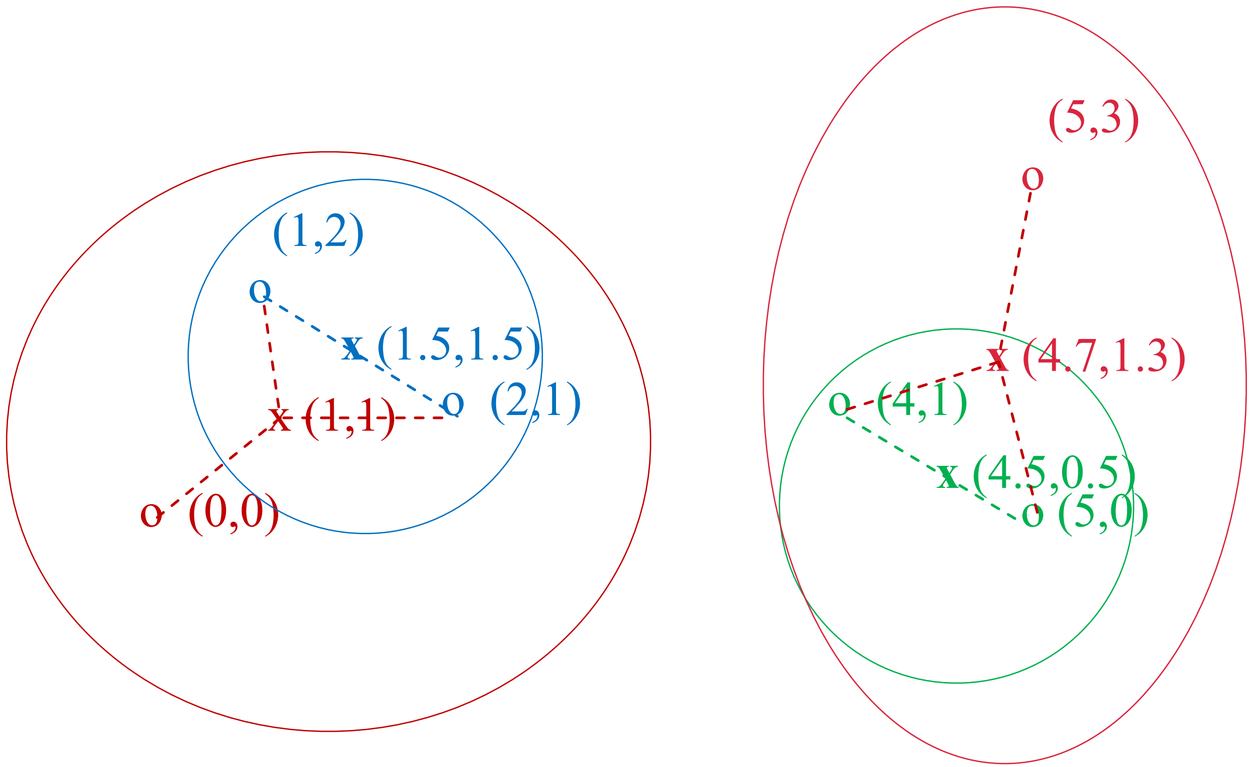
Defining "Nearness" of Clusters

- **(2) How do you determine the "nearness" of clusters?**
  - Possible approaches:
    1. Compare distances between cluster centroids/clustroids
    2. Measure minimum "intercluster" distance

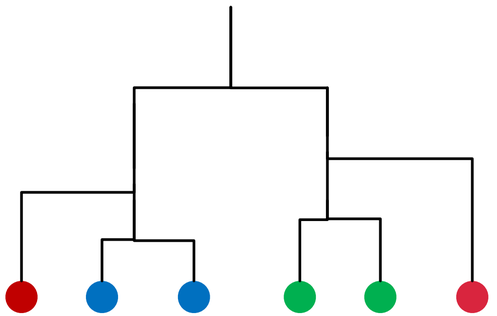
Defining "Nearness" of Clusters

- **(2) How do you determine the "nearness" of clusters?**
  - **Approach 1:**
  - Measure cluster distances by distances of exemplar point
    - Exemplar: Centroid or clustroid

Example: Agglomerative clustering

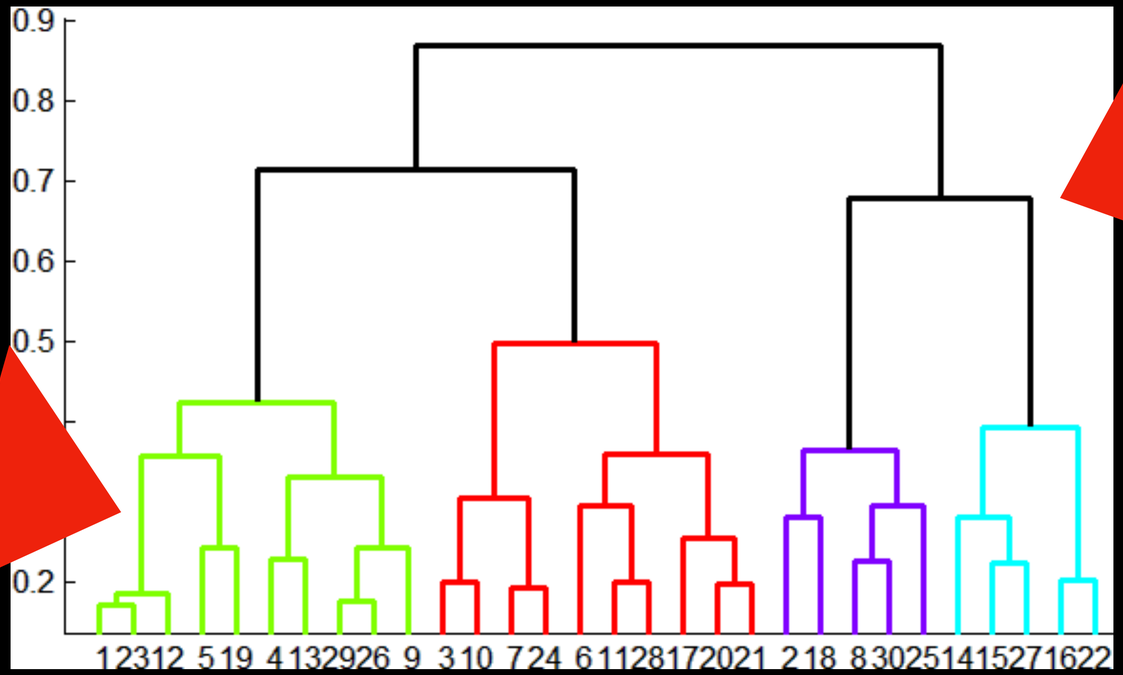


Data:
o … data point
x … centroid



Dendrogram

Nested Sub-Topics



E.g., Top-Level Topics

Dendrograms from agglomerative clusterings…

Defining "Nearness" of Clusters

- **(2) How do you determine the "nearness" of clusters?**
  - **Approach 2:**

    Intercluster distance = minimum of the distances between any two points, one from each cluster

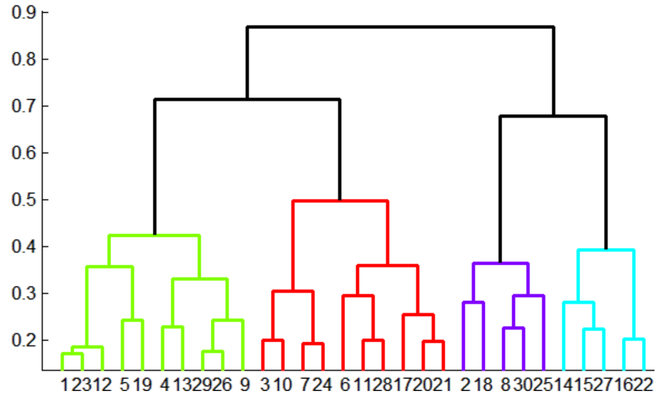
This Module's Learning Objectives

Week 1

- Differentiate between unsupervised and supervised machine learning
- Formally define "clustering"
- Describe how one represents a cluster of multiple points
- Explain how one can calculate distances between clusters

Agglomerative Clustering

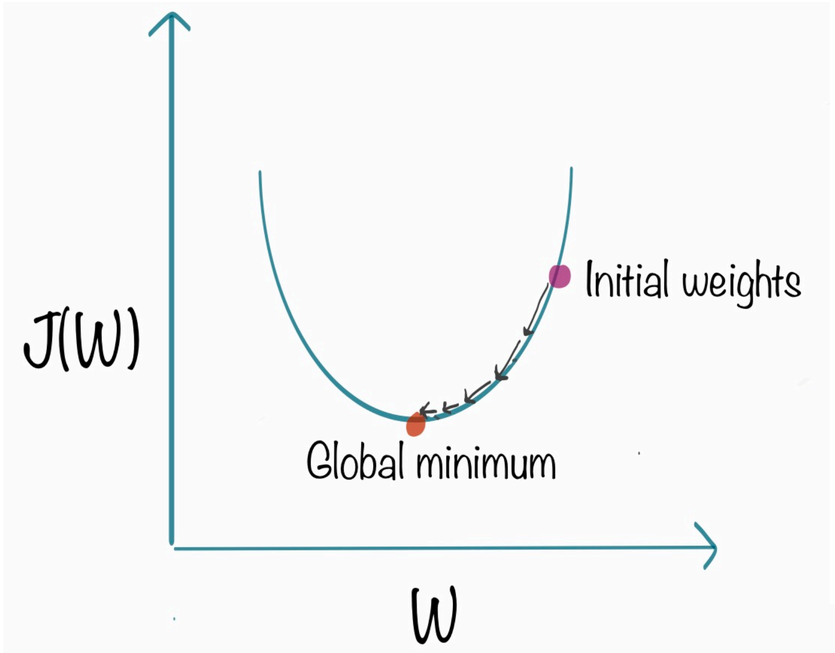
- **Key operation:**
  Repeatedly combine two nearest clusters

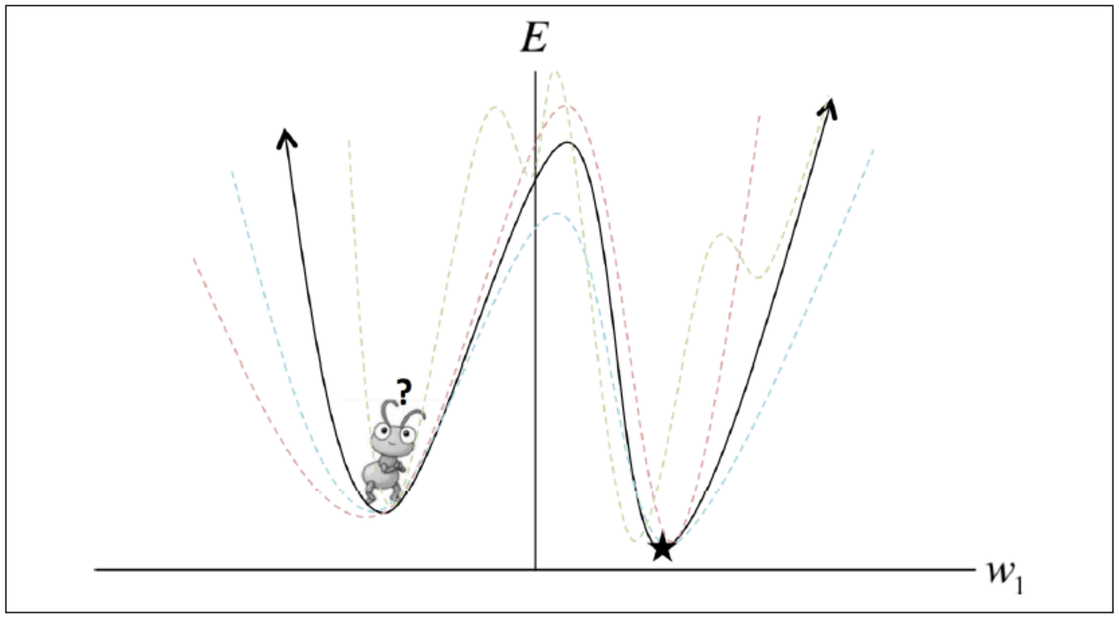


- **Three important questions:**
  - **1)** How do you represent a cluster of more than one point?
  - **2)** How do you determine the "nearness" of clusters?
  - **3)** When to stop combining clusters?

One Metric for Termination: Cohesion

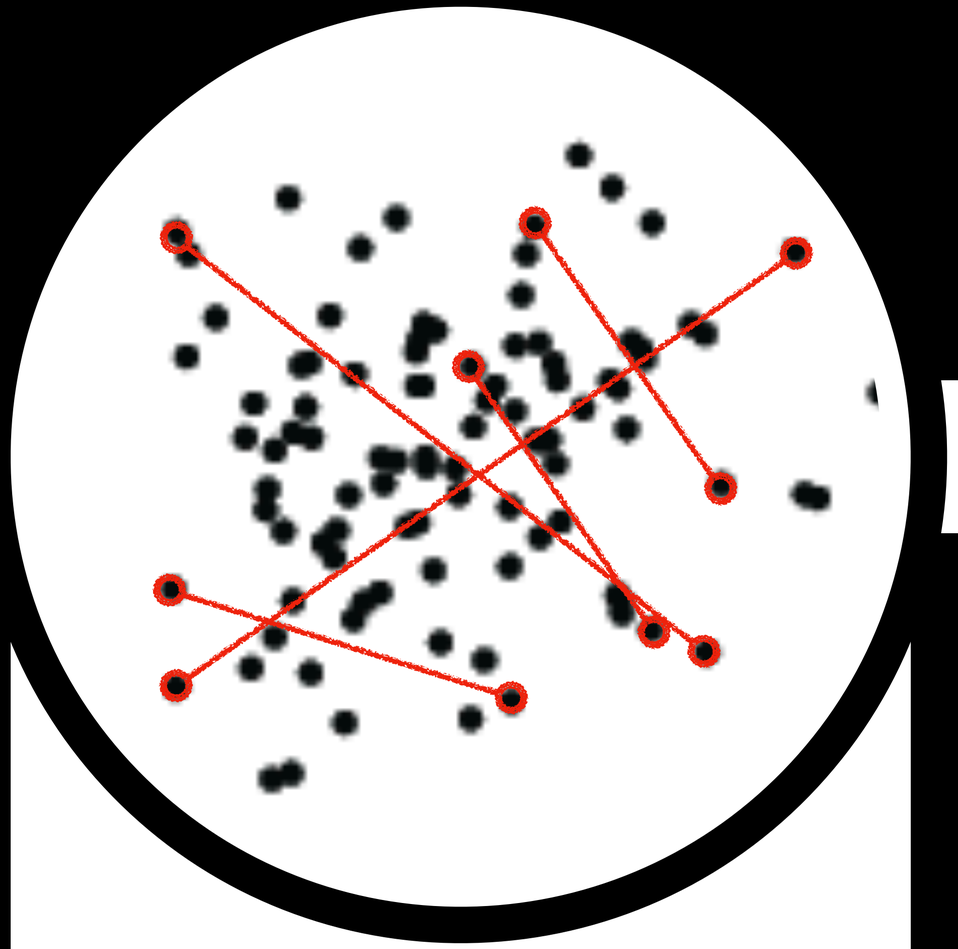
- **Optimizing for Cohesion**
  - **1.** Merge until cohesion metric maximizes/minimizes
  - **2.** Stop merging if the next cluster's cohesion is too "low"





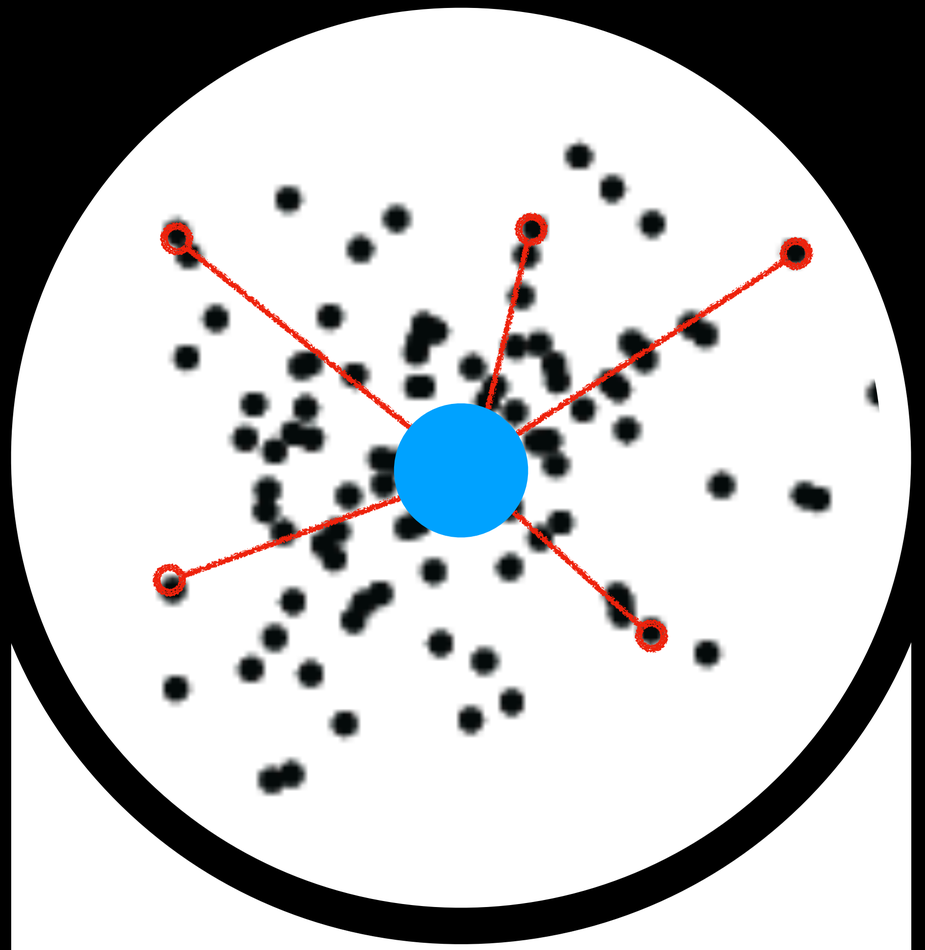
How might you measure how cohesive a cluster is?

Optimizing Cohesion via Diameter

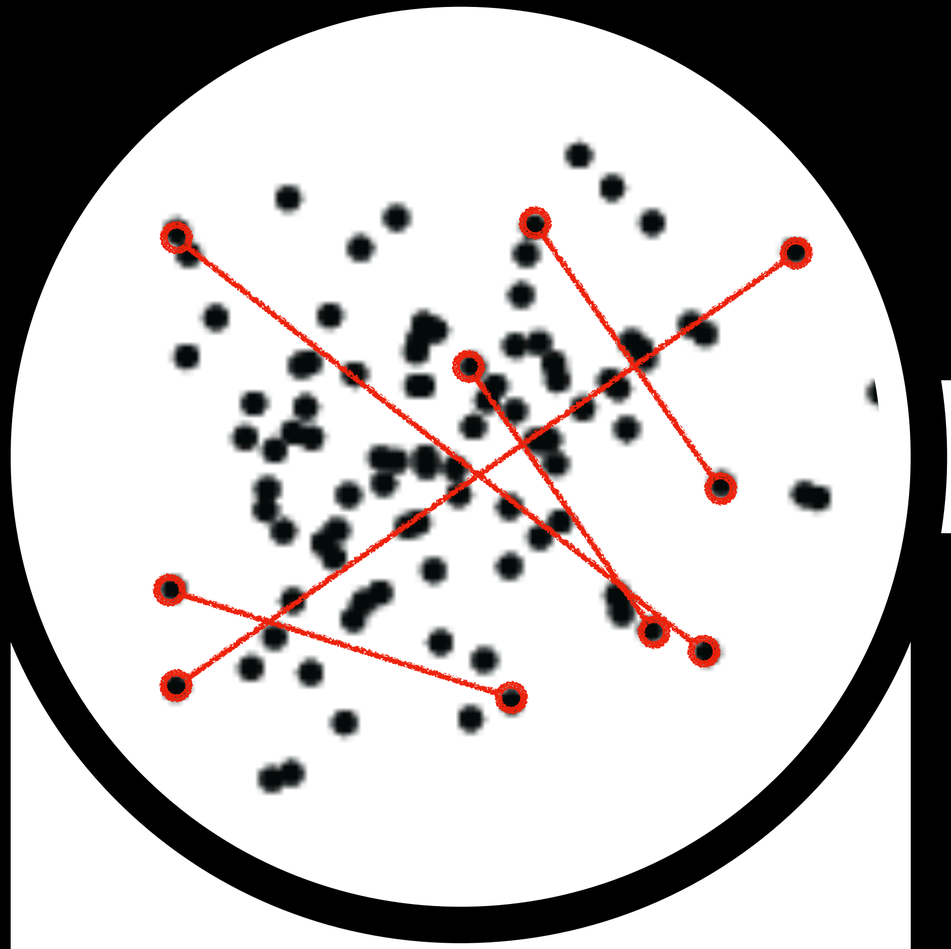


Minimize the maximum distance between any two points in the cluster

Optimizing Cohesion via Radius



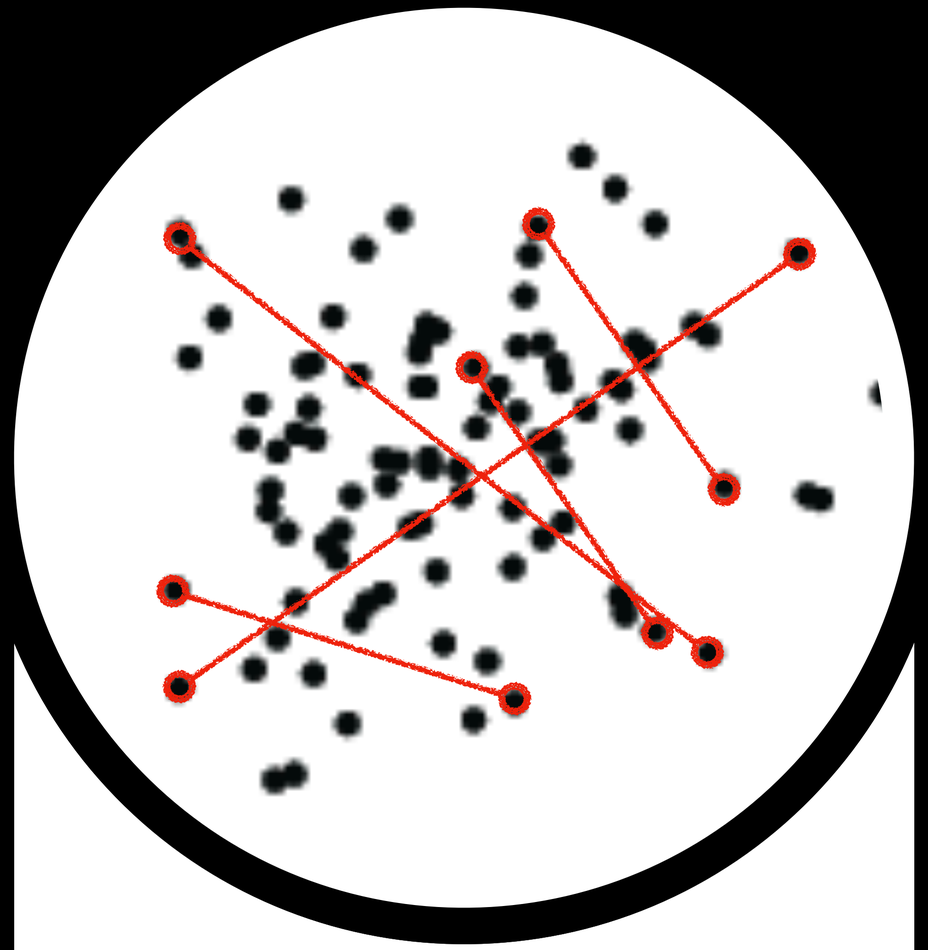
Minimize the maximum distance between centroid and all cluster points



Optimizing Cohesion via Avg. Distance

Minimize the average distance between any two points in the cluster

Optimizing Cohesion via Density



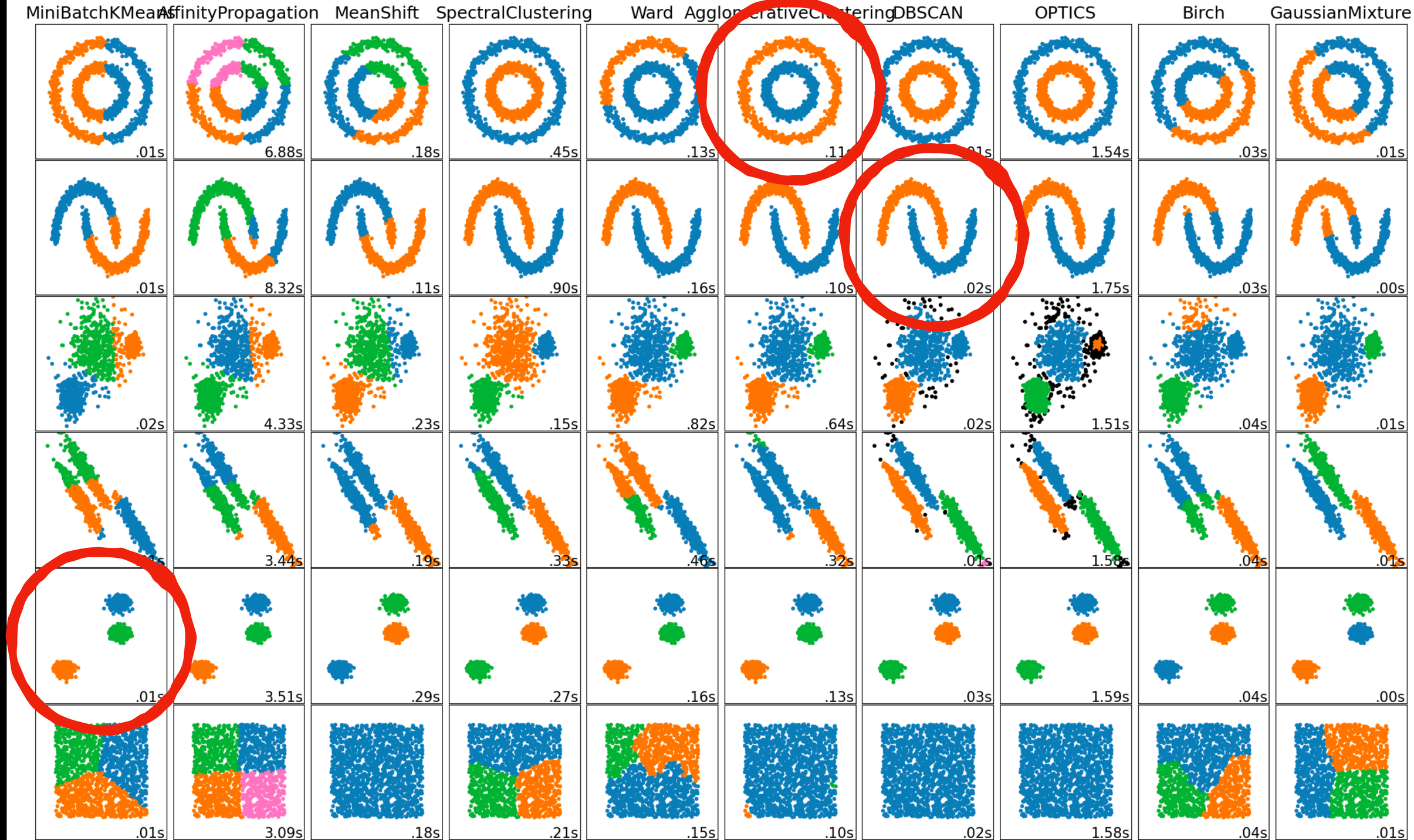
Minimize the ratio of diameter to the number of points in the cluster

Optimizing Cohesion via Diameter

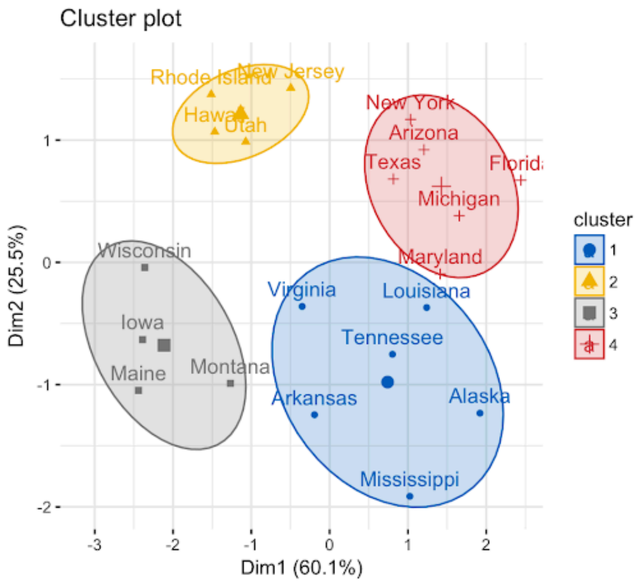
Optimizing Cohesion via Radius

Optimizing Cohesion via Avg. Distance

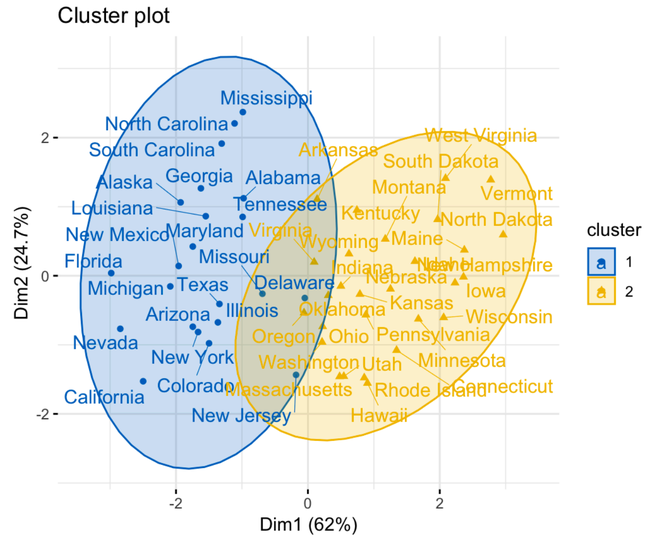
Optimizing Cohesion via Density


MiniBatchKMeans
AffinityPropagation
MeanShift
SpectralClustering
Ward
DBSCAN
OPTICS
Birch
GaussianMixture
.01s
6.88s
.18s
.45s
.13s
1.54s
.03s
.01s
.01s
8.32s
.11s
.90s
.16s
.10s
.02s
1.75s
.03s
.00s
.02s
4.33s
.23s
.15s
.82s
.64s
.02s
1.51s
.04s
.01s
3.44s
.19s
.33s
.46s
.32s
.01s
1.58s
.04s
.01s
.01s
3.51s
.29s
.27s
.16s
.13s
.03s
1.59s
.04s
.00s
.01s
3.09s
.18s
.21s
.15s
.10s
.02s
1.58s
.04s
.01s

Method name	Parameters	Scalability	Usecase	Geometry (metric used)
K-Means	number of clusters	Very large `n_samples`, medium `n_clusters` with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Affinity propagation	damping, sample preference	Not scalable with n_samples	Many clusters, uneven cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Mean-shift	bandwidth	Not scalable with `n_samples`	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium `n_samples`, small `n_clusters`	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
Ward hierarchical clustering	number of clusters or distance threshold	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints	Distances between points
Agglomerative clustering	number of clusters or distance threshold, linkage type, distance	Large `n_samples` and `n_clusters`	Many clusters, possibly connectivity constraints, non Euclidean distances	Any pairwise distance
DBSCAN	neighborhood size	Very large `n_samples`, medium `n_clusters`	Non-flat geometry, uneven cluster sizes	Distances between nearest points
OPTICS	minimum cluster membership	Very large `n_samples`, large `n_clusters`	Non-flat geometry, uneven cluster sizes, variable cluster density	Distances between points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation	Mahalanobis distances to centers
Birch	branching factor, threshold, optional global clusterer.	Large `n_clusters` and `n_samples`	Large dataset, outlier removal, data reduction.	Euclidean distance between points



Assume all points exist in <u>exactly one</u> cluster



Fuzzy Clustering: Points can exist in multiple clusters

This Module's Learning Objectives

Week 1

Differentiate between unsupervised and supervised machine learning

Formally define "clustering"

Describe how one represents a cluster of multiple points

Explain how one can calculate distances between clusters

Questions?

Prof. Cody Buntain | @codybuntain | cbuntain@umd.edu
Director, Information Ecosystems Lab