Unsupervised Learning

Research skills: Machine Learning

Afra Alishahi

March 11, 2015

Supervised Learning

- **Classification**
  - Movie/book/restaurant reviews: good vs. bad
  - Emails: spam vs. not spam
- **Regression**
  - Predict height based on weight and gender
  - Predict income based on education, specialization and country

➡ We look for a pre-specified structure in data

- Training data: feature sets annotated with labels or numbers

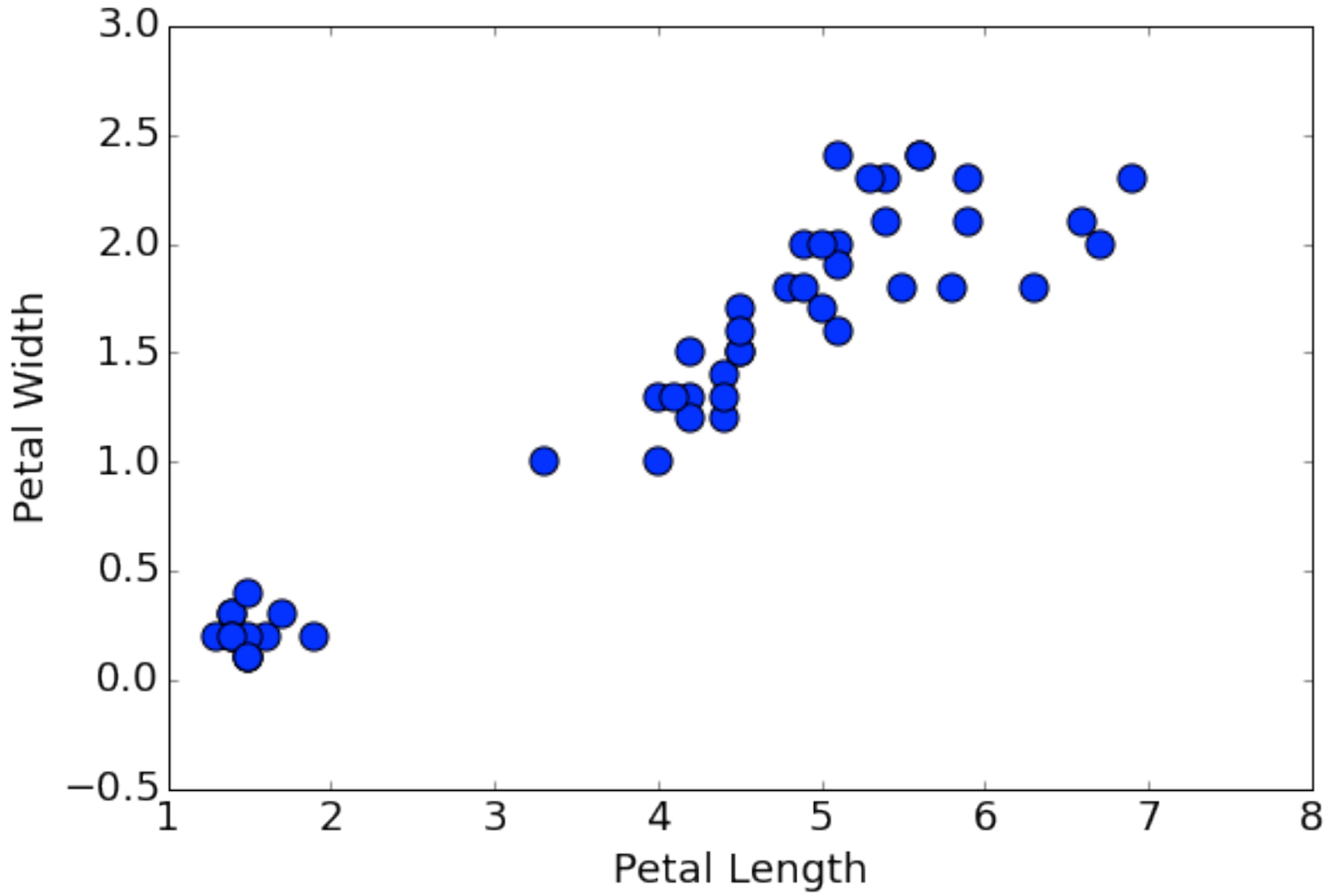
Exploratory Data Analysis

- Sometimes the structure of data is not known in advance
  - Emails: work vs. family vs. friends vs. advertisement vs. ...?
  - Shapes: square vs. circle vs. triangle vs. ...?
  - Types of questions asked in a forum

- We have a number of observation points, but no pre-defined set of labels attached to them.

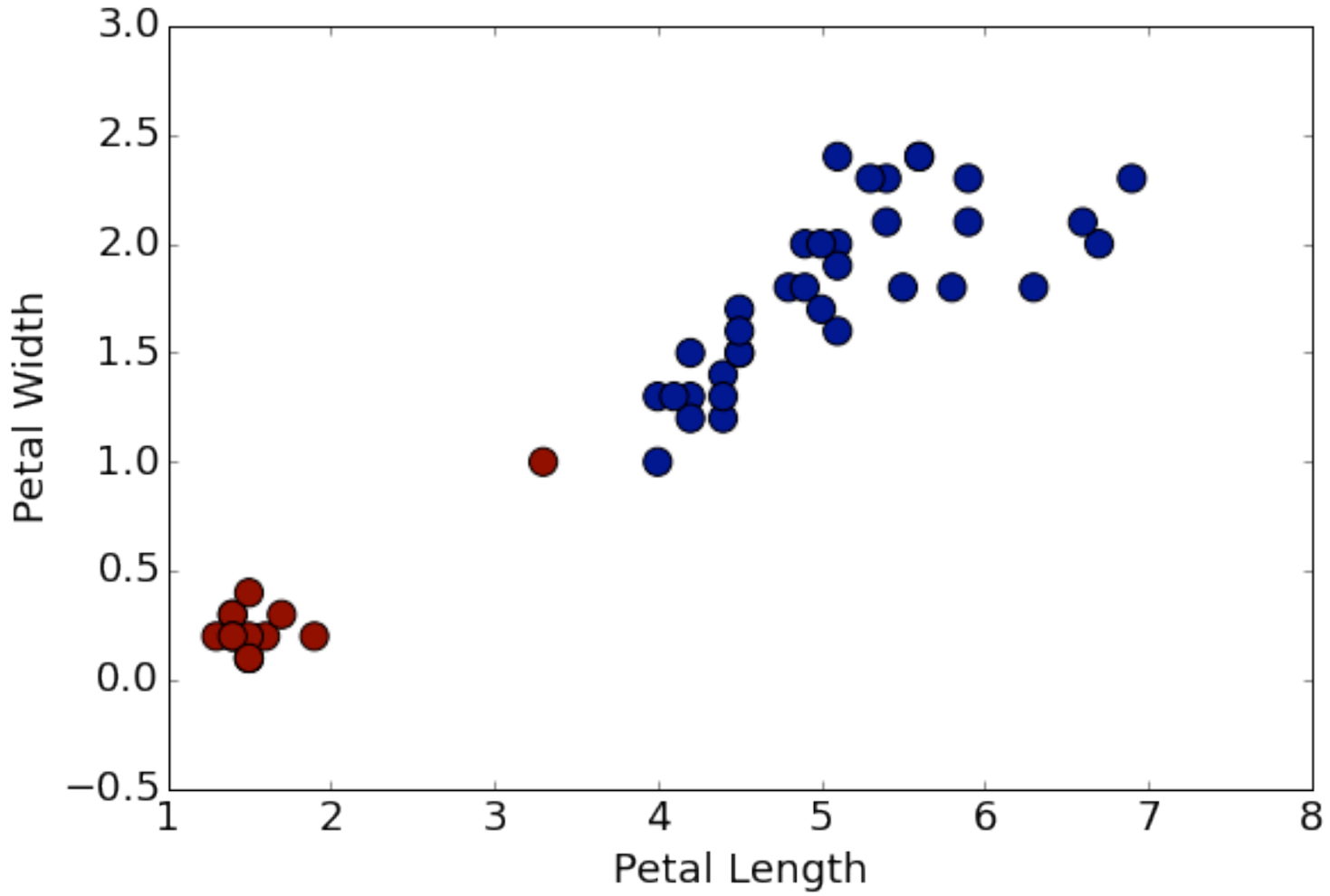
Clustering

- The Iris dataset:



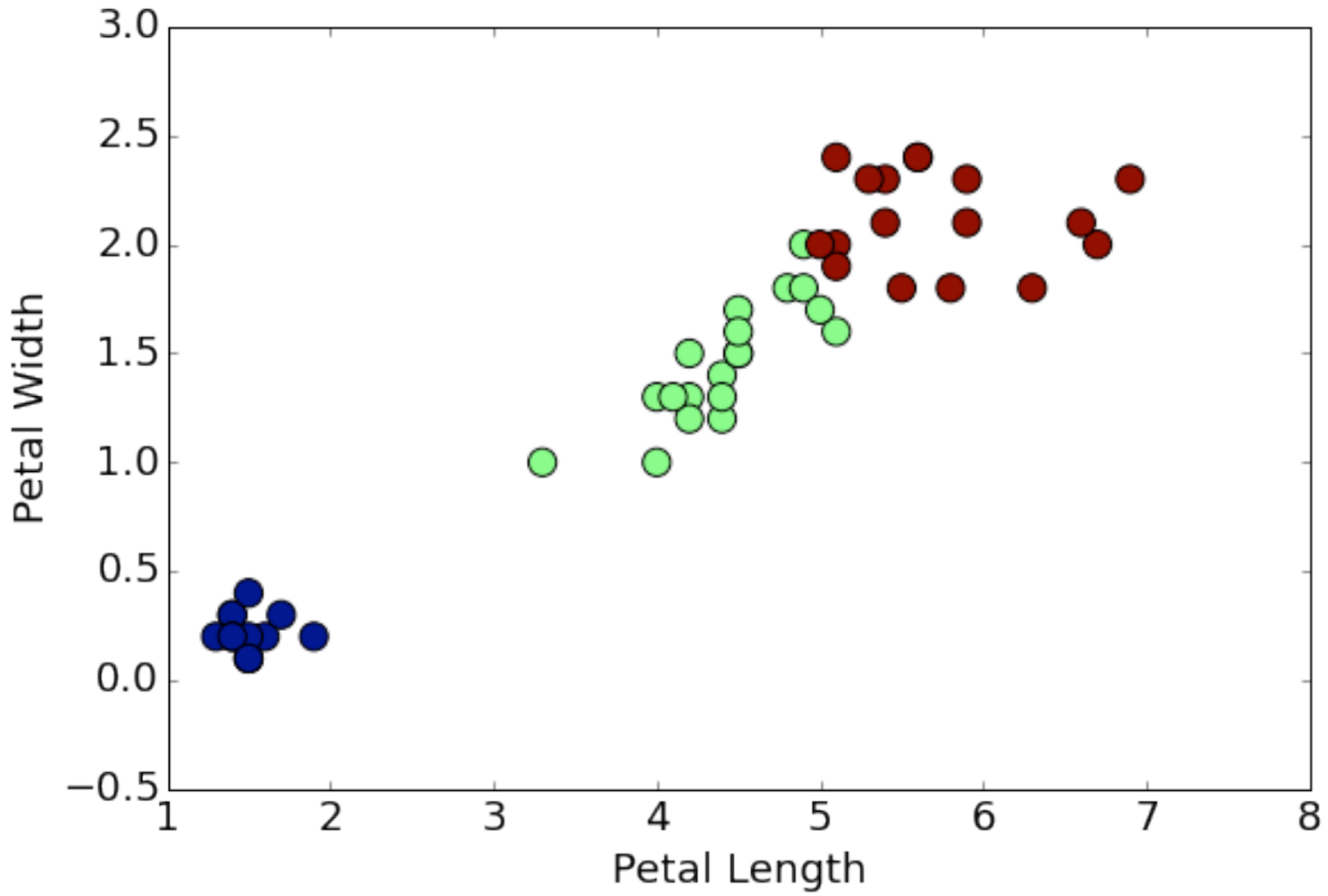
Clustering

- **Two** clusters:



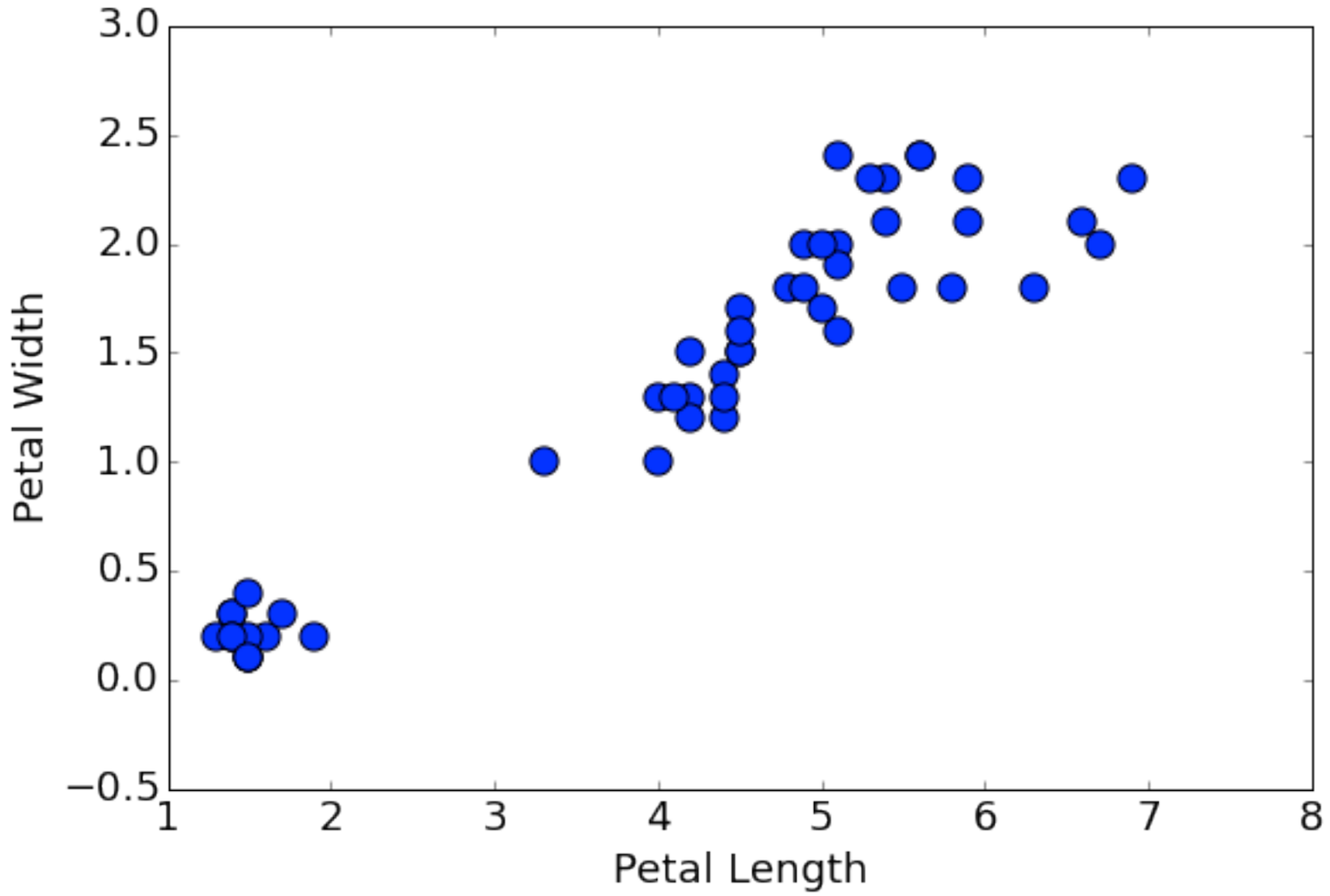
Clustering

- **Three** clusters:



Clustering

- How do we group the observation points together?

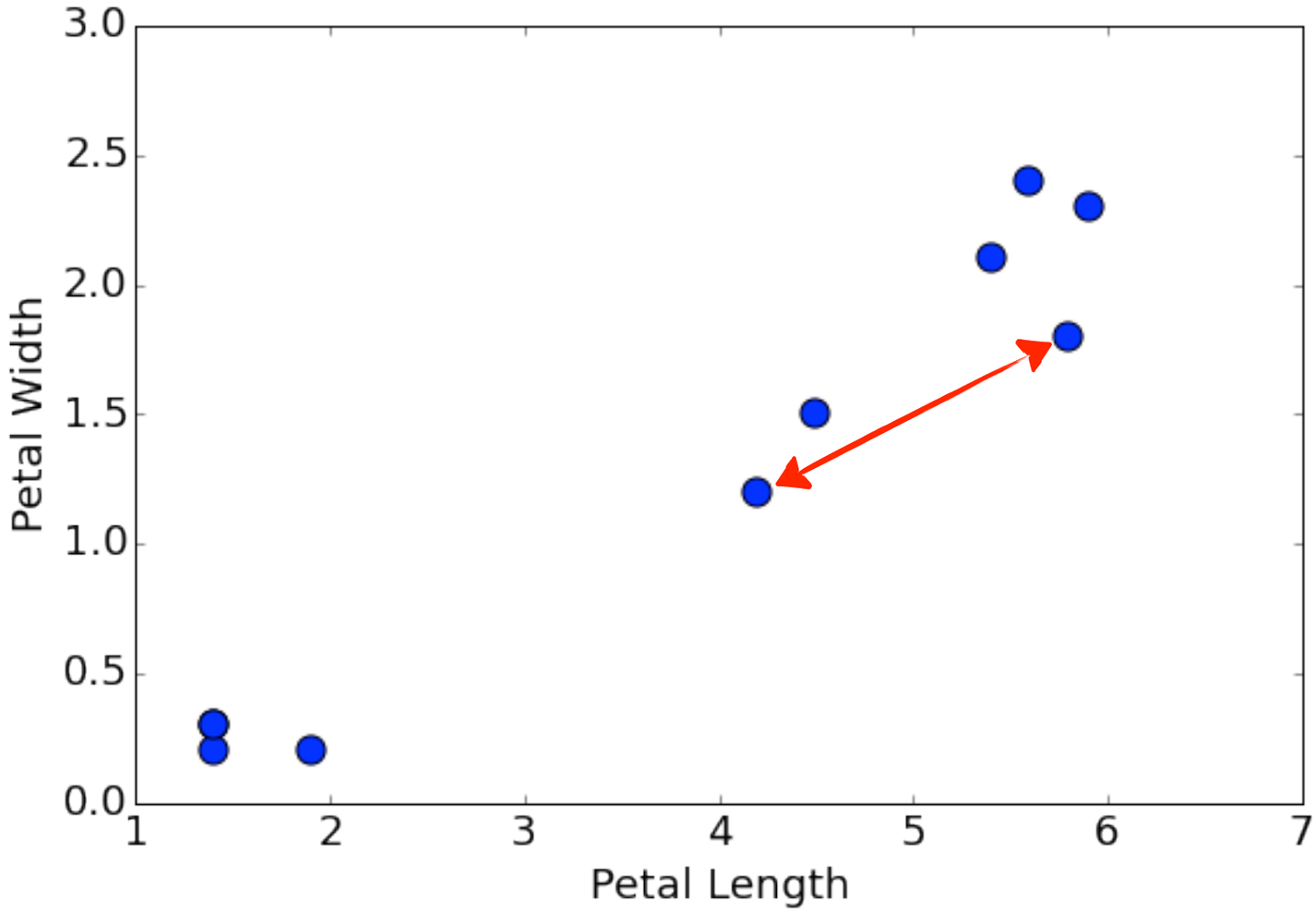


What Makes a Cluster "Good"?

- Clusters should be coherent
  - Members of the same cluster must be as close/similar to each other as possible
  - Clusters must be as distant/dissimilar from each other as possible

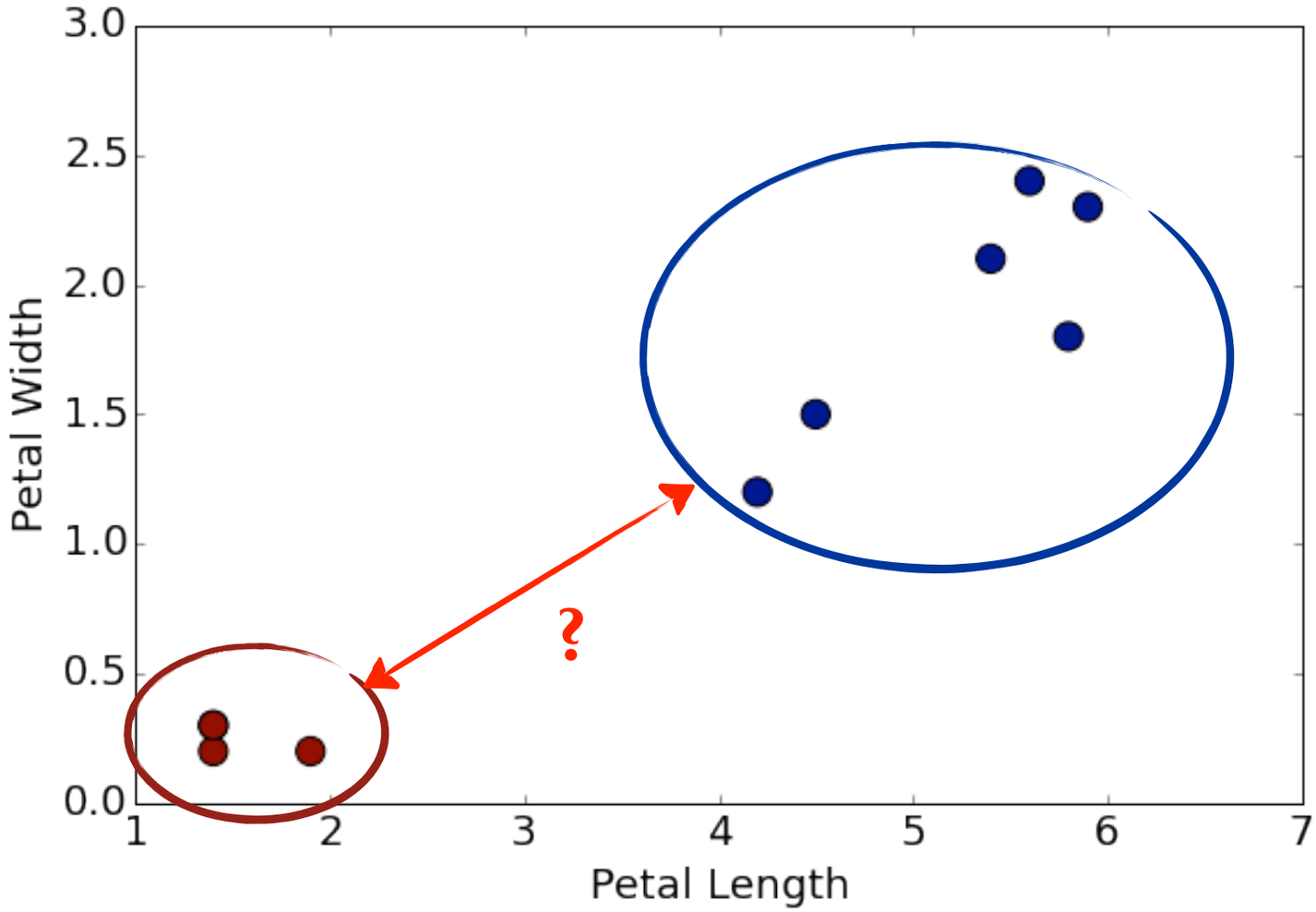
- Needed machinery:
  - Distance between two data points
  - Distance between two clusters

Distance between Data Points

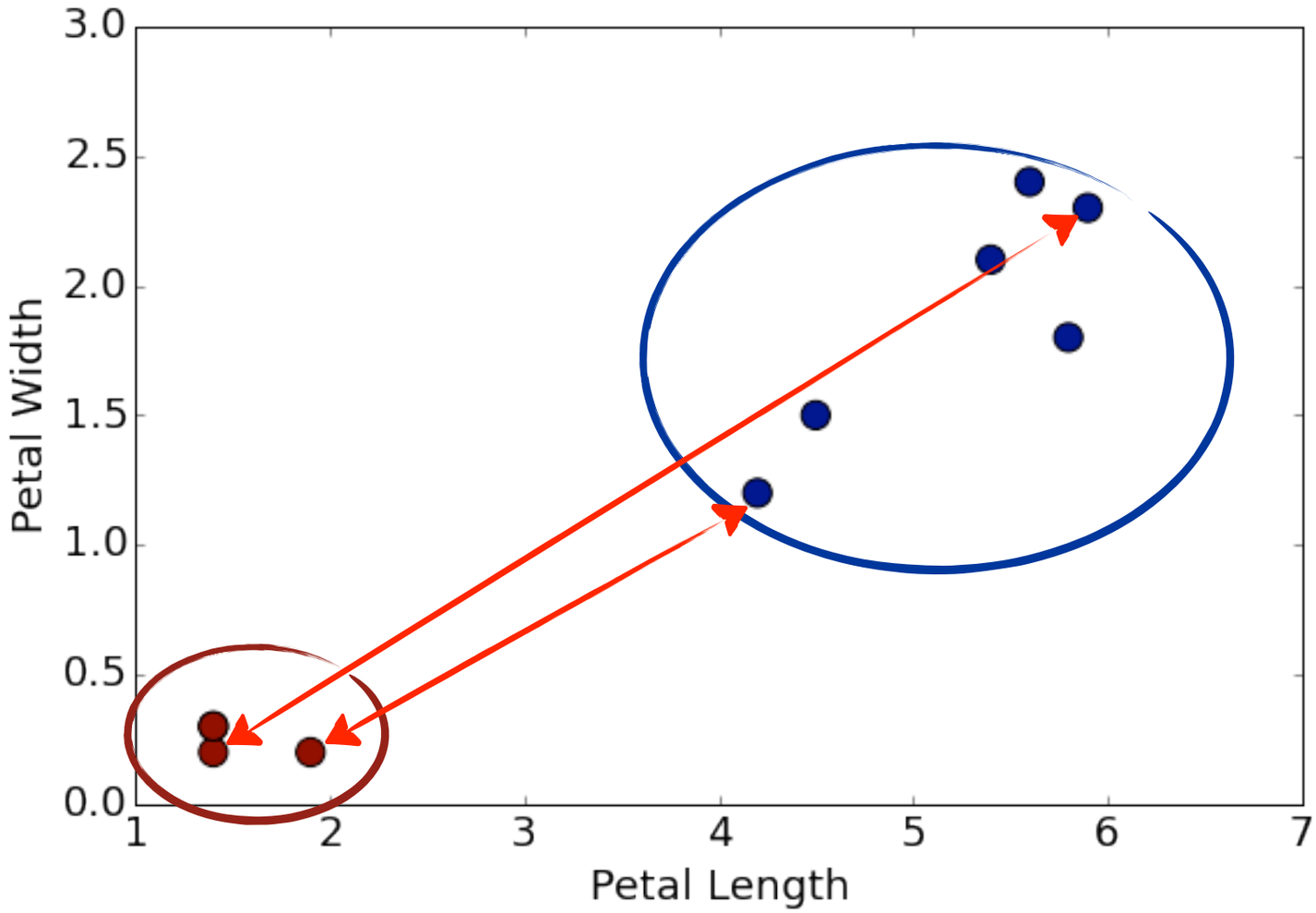


For numerical features, Euclidean distance is a good measurement.

Distance between Clusters

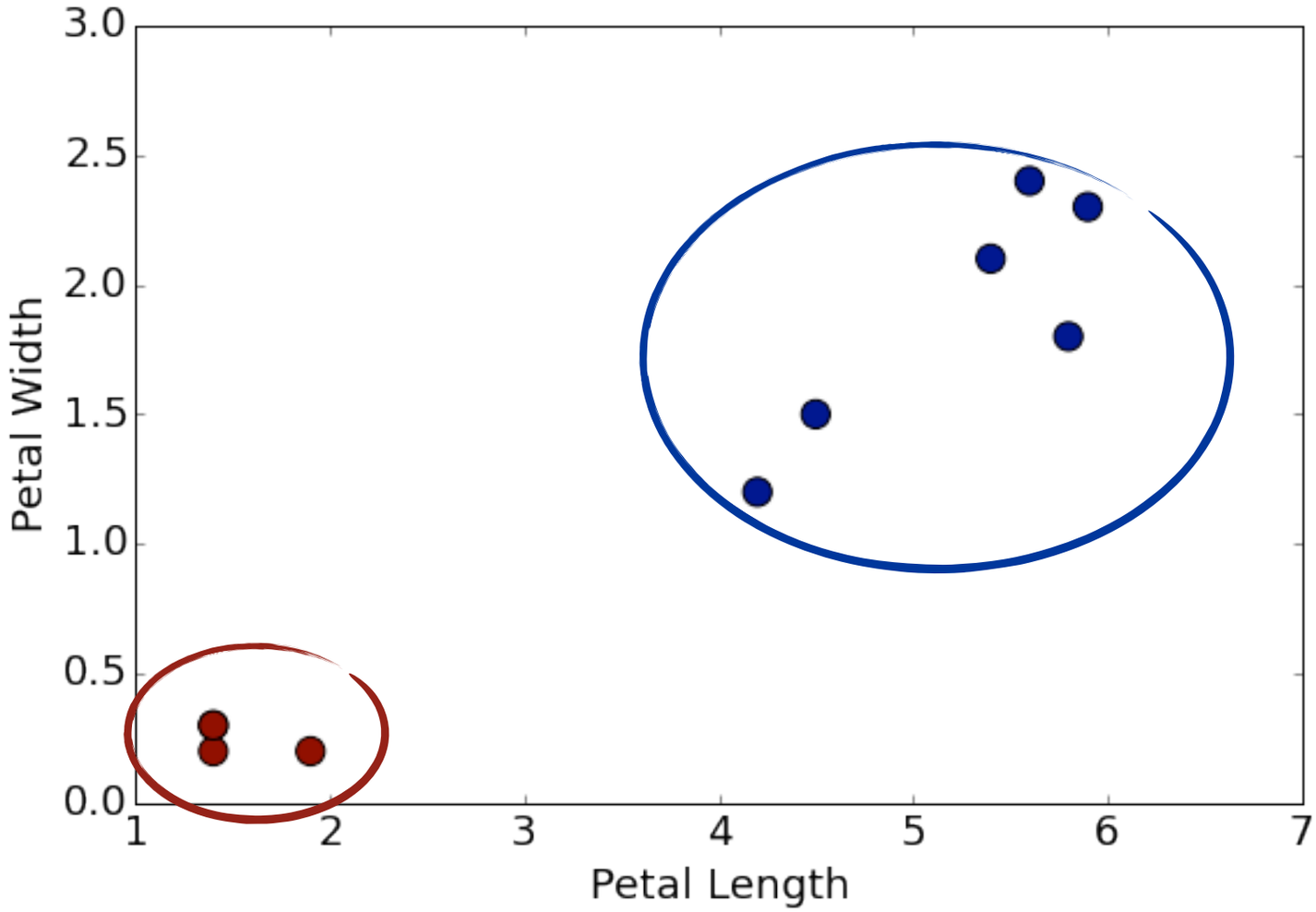


Distance between Clusters

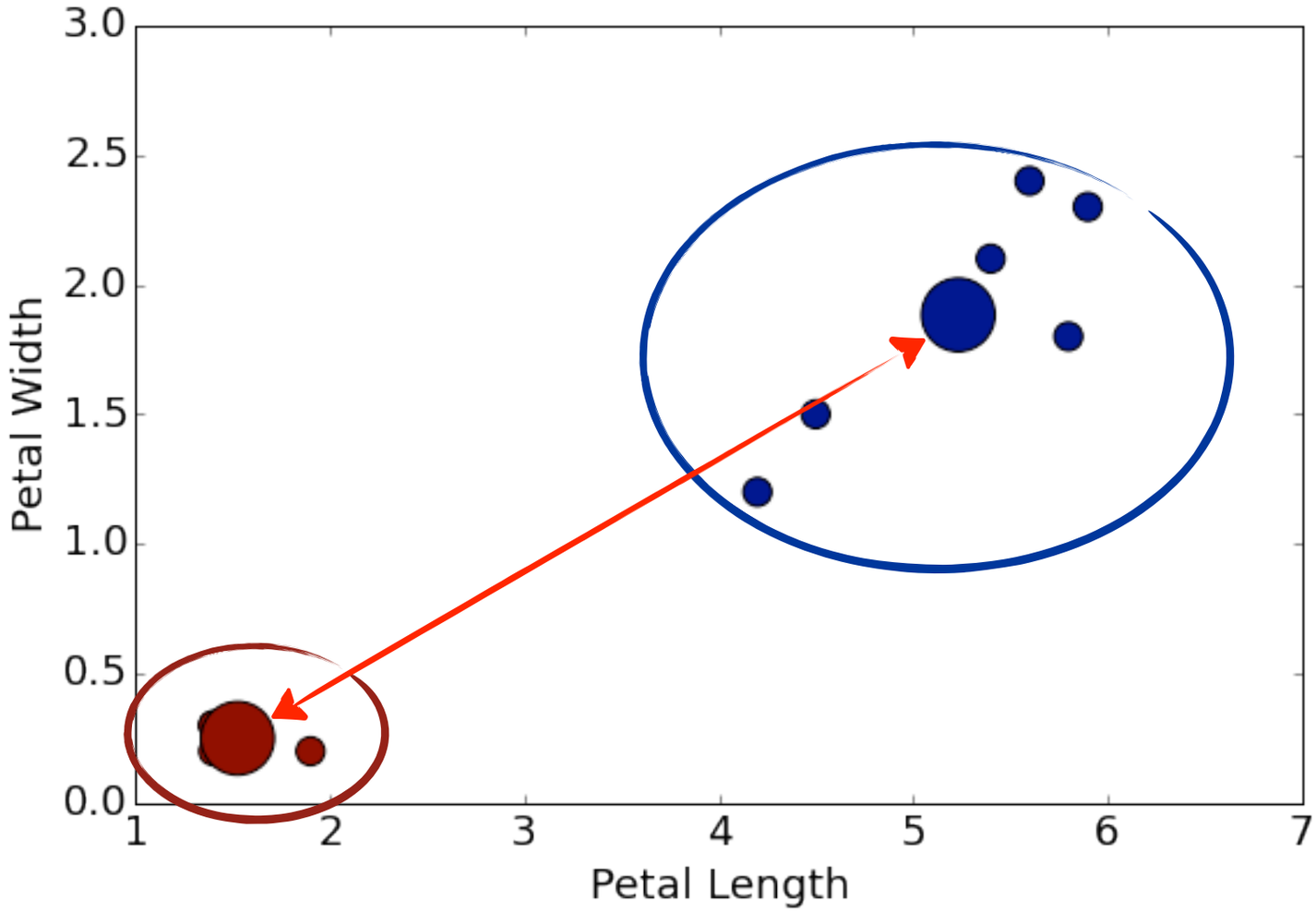


- **Single link:** distance btw of two most similar members
- **Complete link:** distance btw of two least similar members

Cluster Centroids



Cluster Centroids



Cluster centroid: $\mu_k = \frac{1}{||k||} \sum_{x \in k} x$

K-means Clustering Algorithm

- Given:
  - a dataset $X = \{x_1, \cdots, x_n\}$
  - a distance measure $d(x_i, x_j)$
- Randomly assign data points to K clusters
- repeat
  - calculate cluster centroids

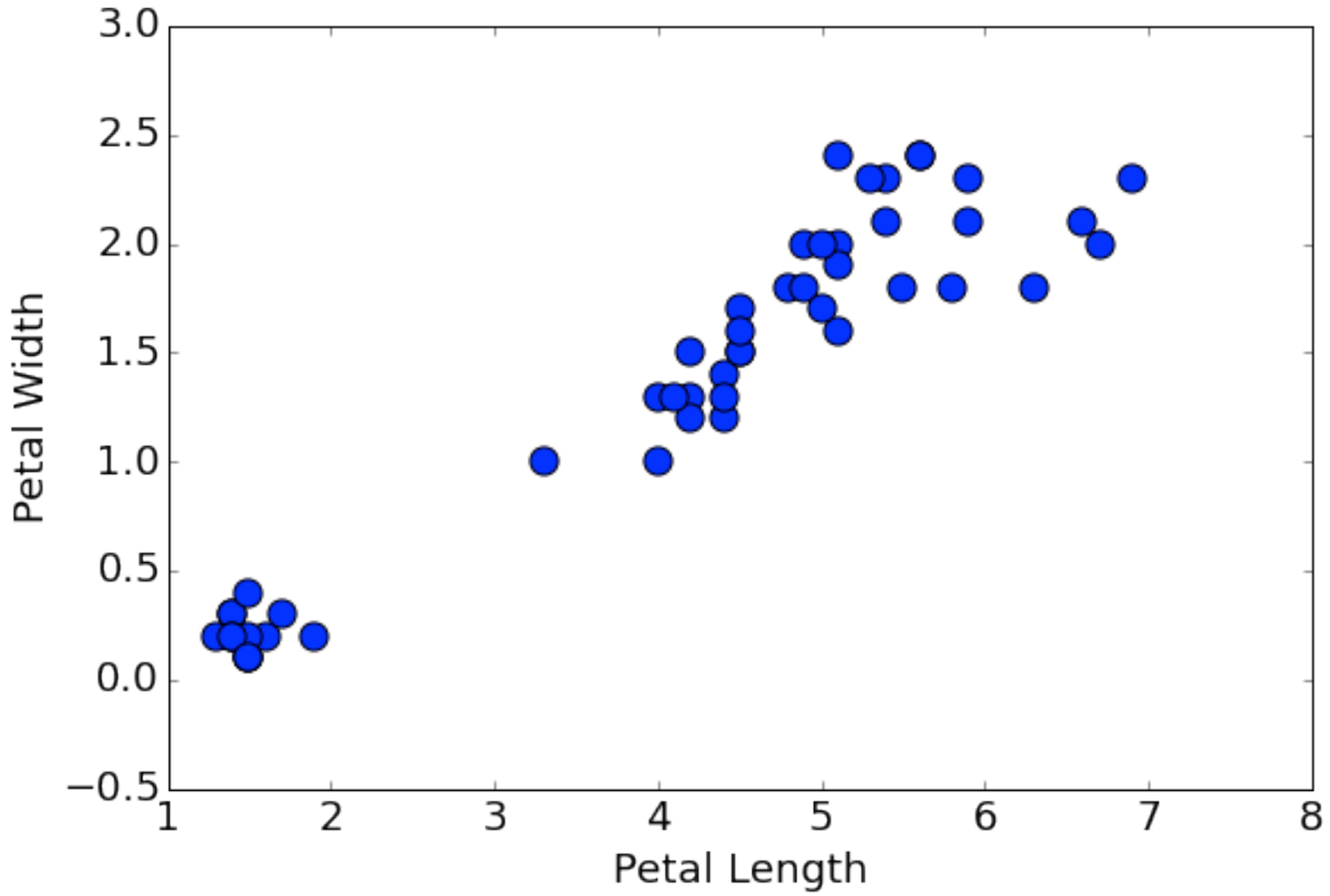
$$\mu_k = \frac{1}{||k||} \sum_{x \in k} x$$

  - assign each data point to the cluster with the closest centroid

$$k = \{x | \forall k', d(x, \mu_{k'}) \leq d(x, \mu_k)\}$$

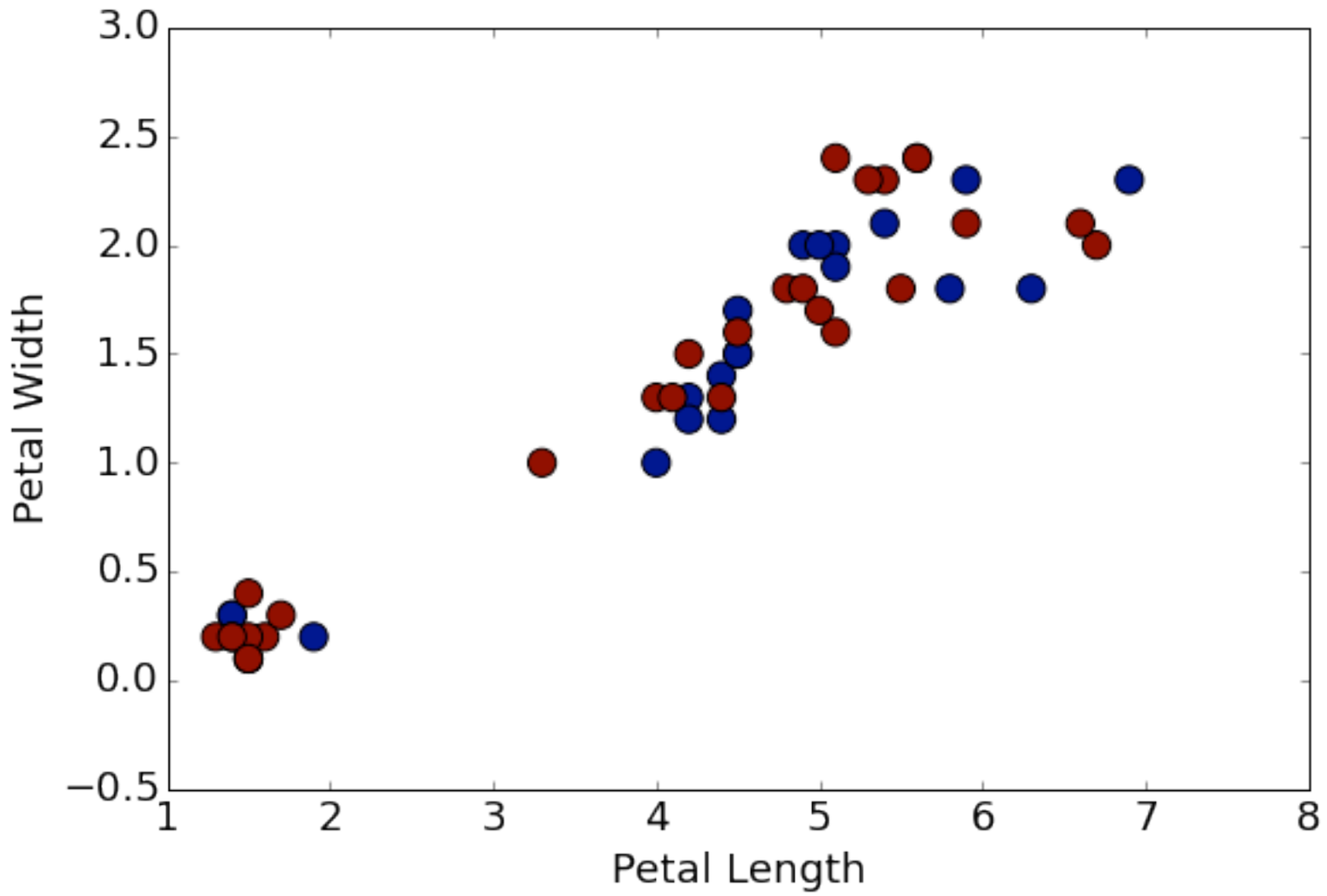
K-means Clustering Algorithm

- The Iris dataset:



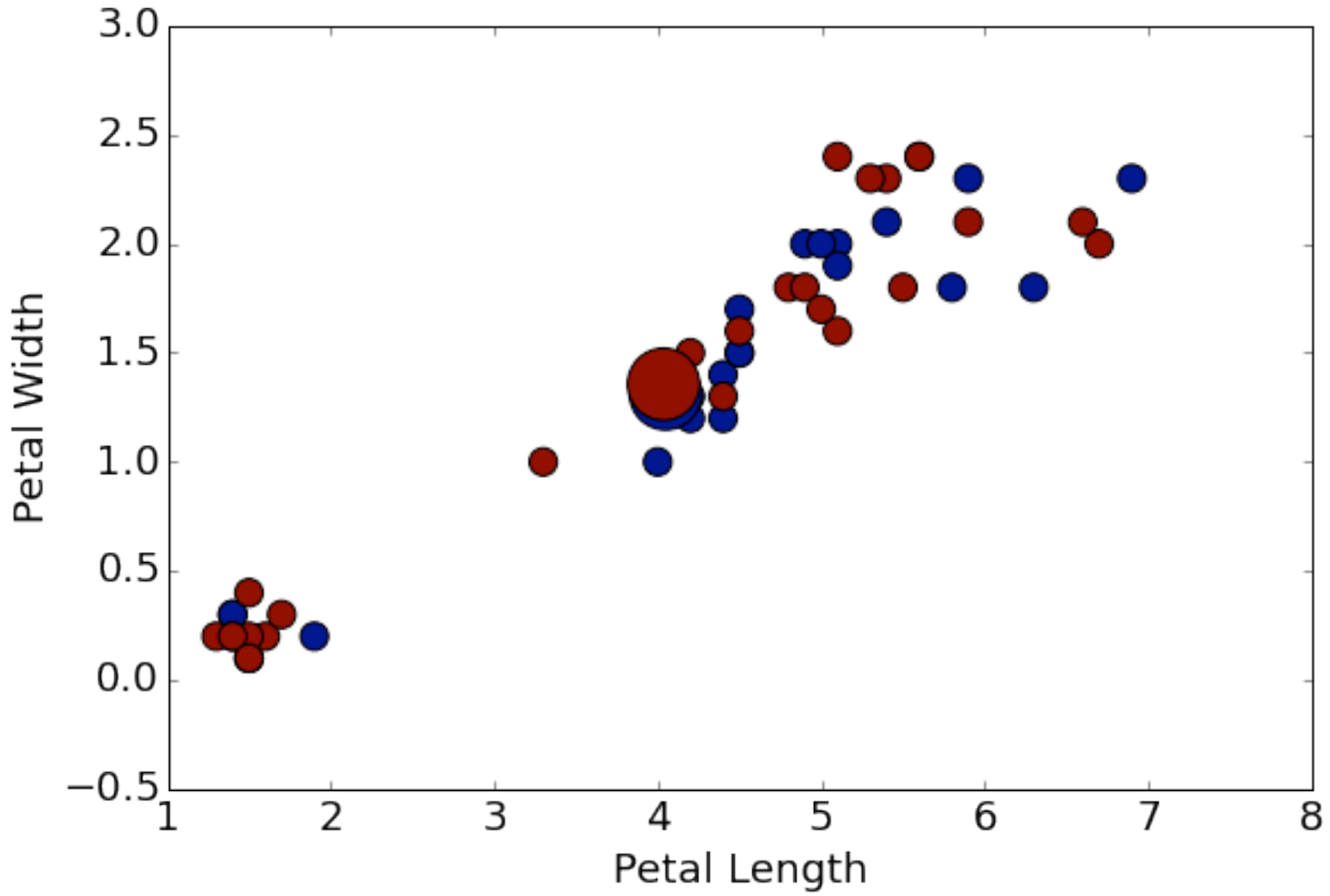
K-means Clustering Algorithm

- Randomly assign points to two clusters:



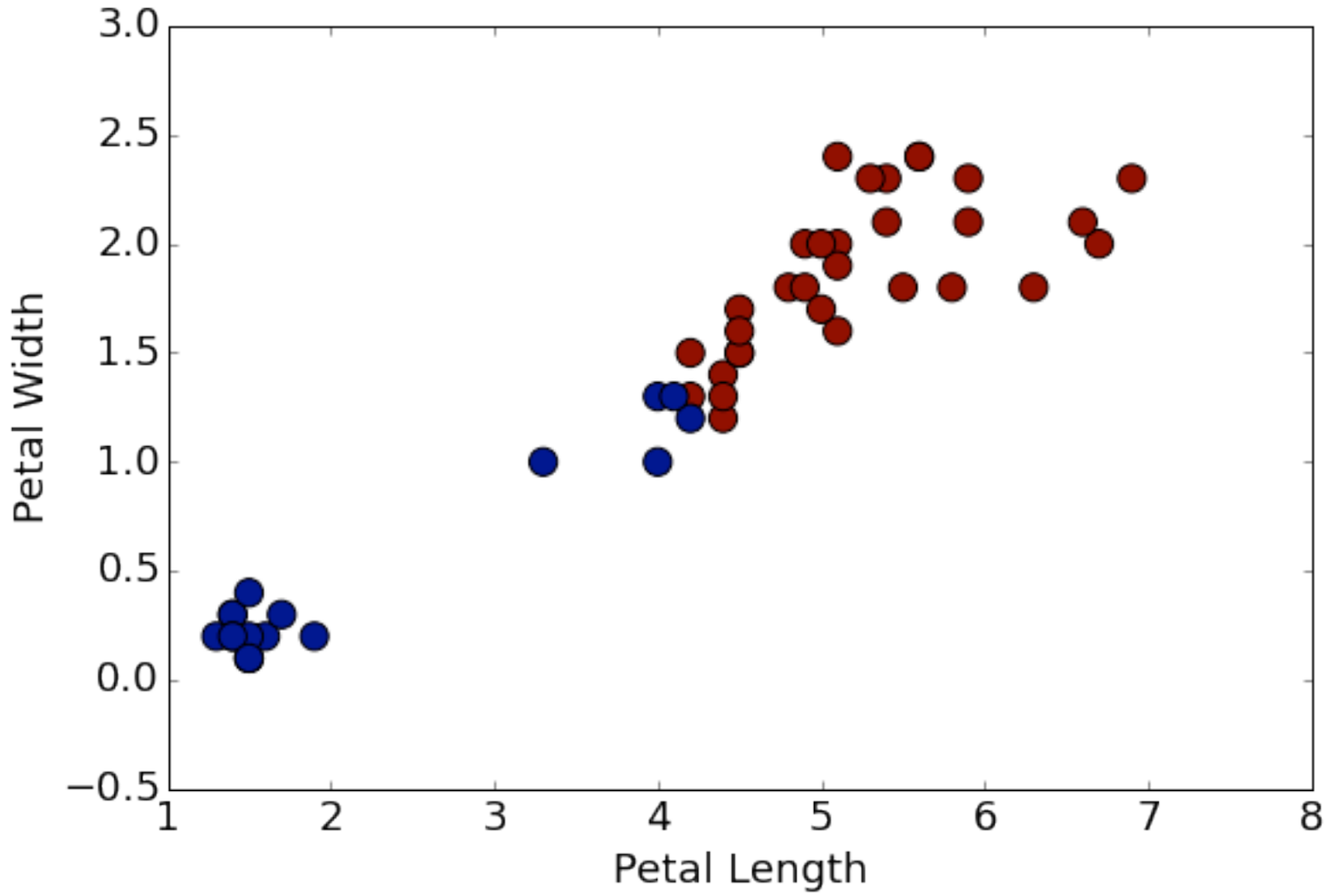
K-means Clustering Algorithm

- Calculate the centroids:



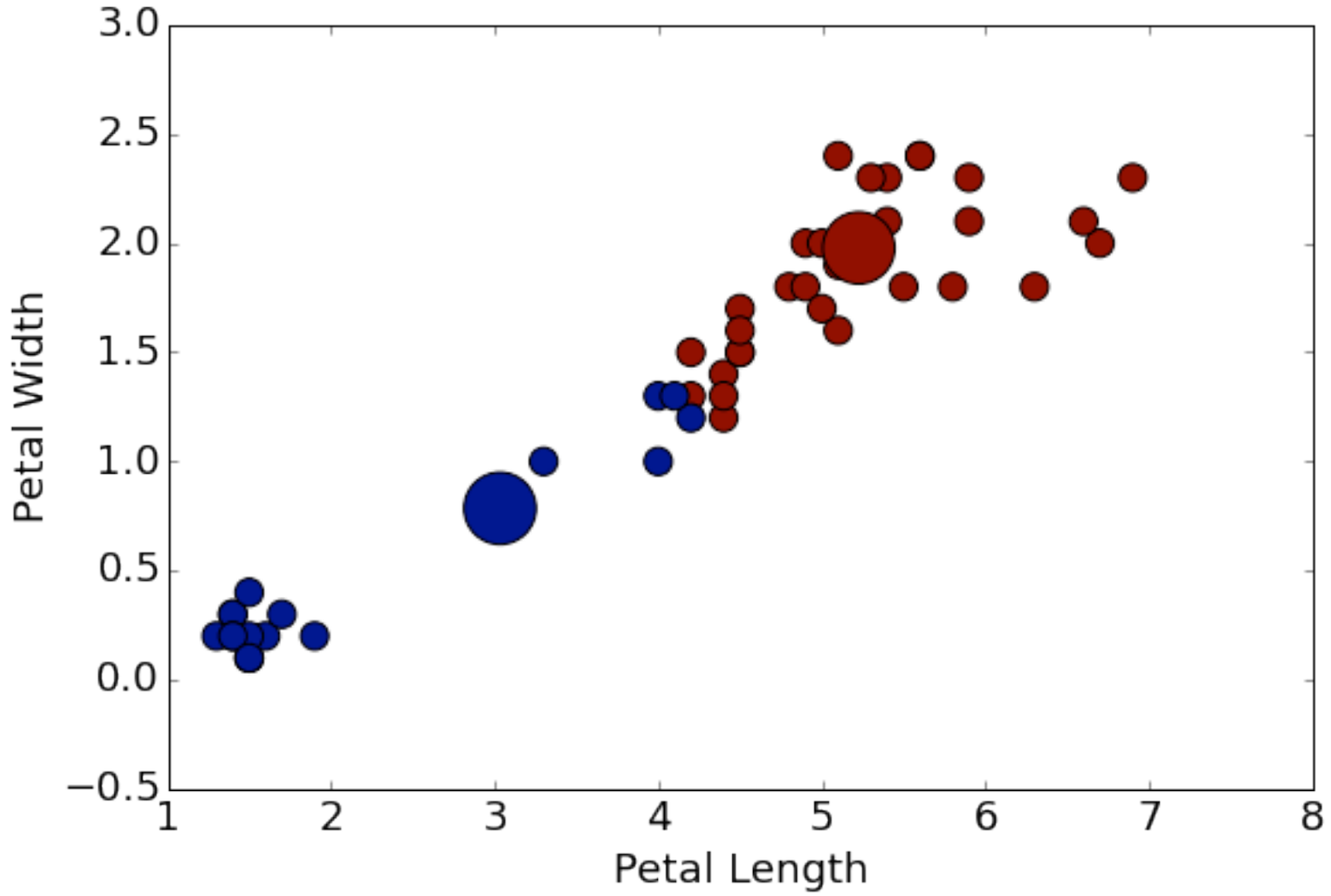
K-means Clustering Algorithm

- Re-assign the data points to the clusters:



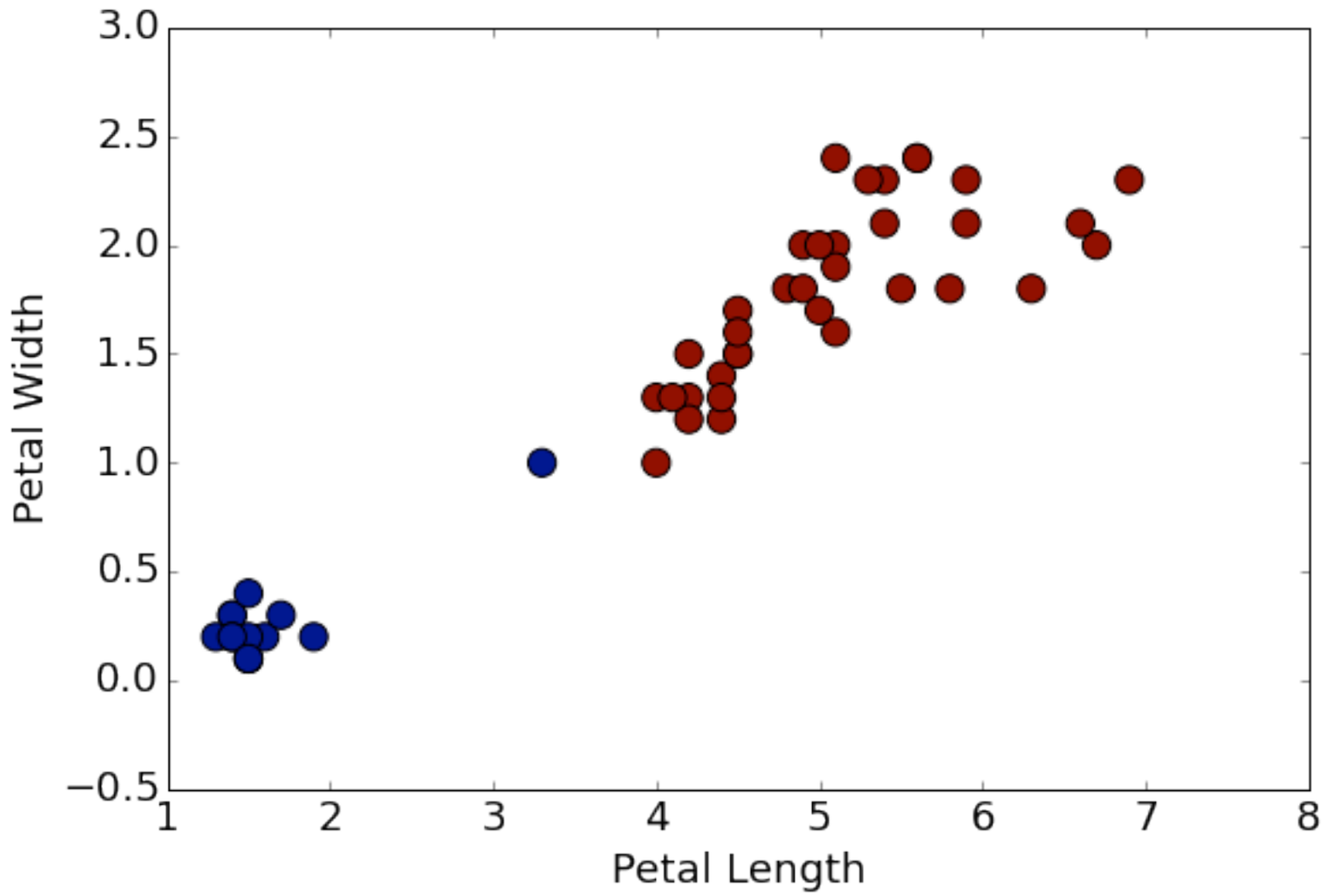
K-means Clustering Algorithm

- Re-calculate the centroids:



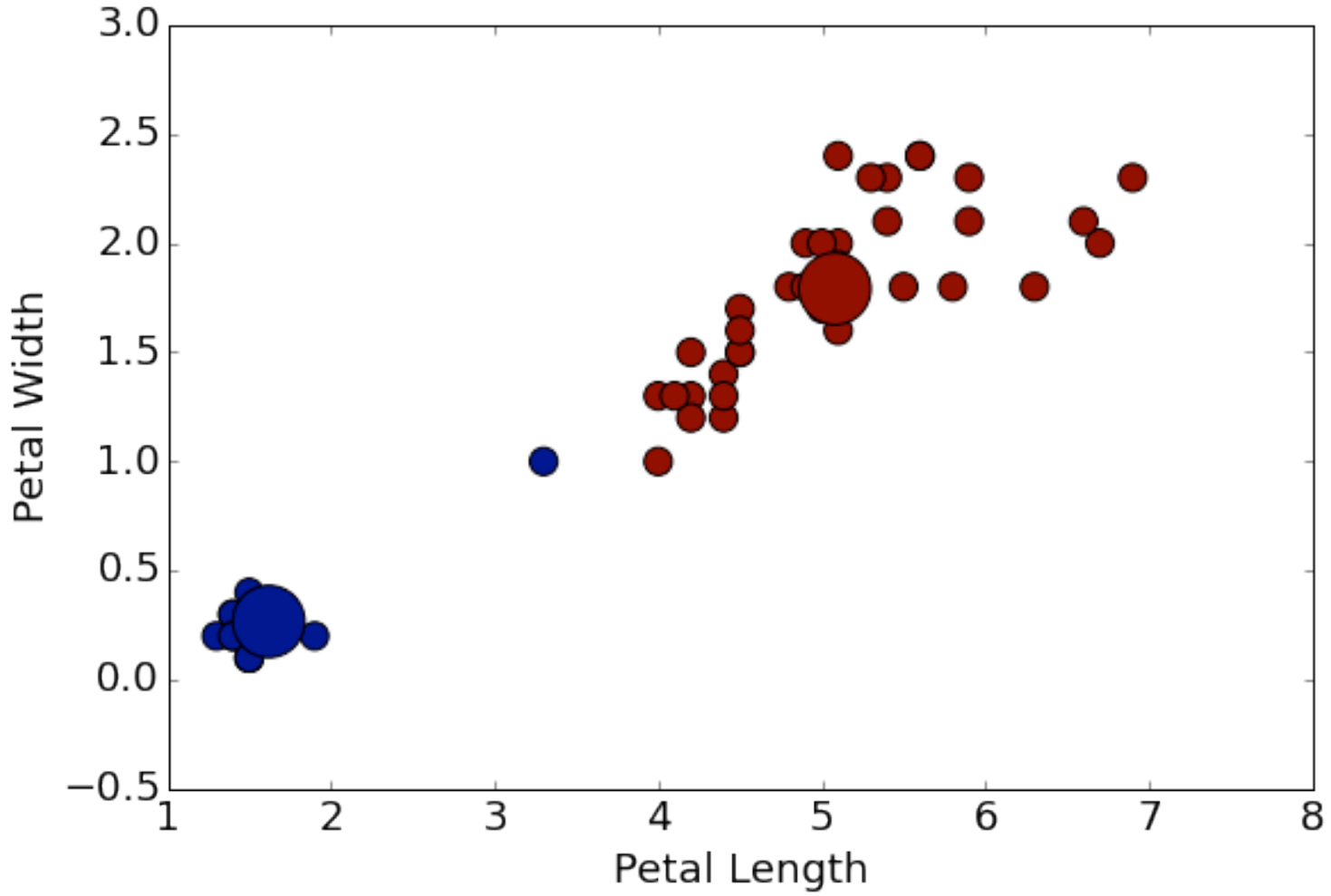
K-means Clustering Algorithm

- And again ...



K-means Clustering Algorithm

- ... and again.



Objective Function

Objective function: members of a cluster must be as close to its centroid as possible

$$J = \sum_{k=1}^{K} \sum_{x \in k} ||x - \mu_k||^2$$

- J: distortion measure
- K: number of clusters
- $x \in k$: members of cluster k
- μ_k : centroid of cluster k

Practical Questions

- When to stop?
  - There are no more changes in the cluster structure/membership
  - The objective function reaches a certain level
- How many clusters?
  - An informed guess?
  - Trying different numbers to see which one yields the best value for the objective function
- Can we speed up the algorithm?
  - Update the centroids incrementally
  - Select the initial centroids wisely (how?)

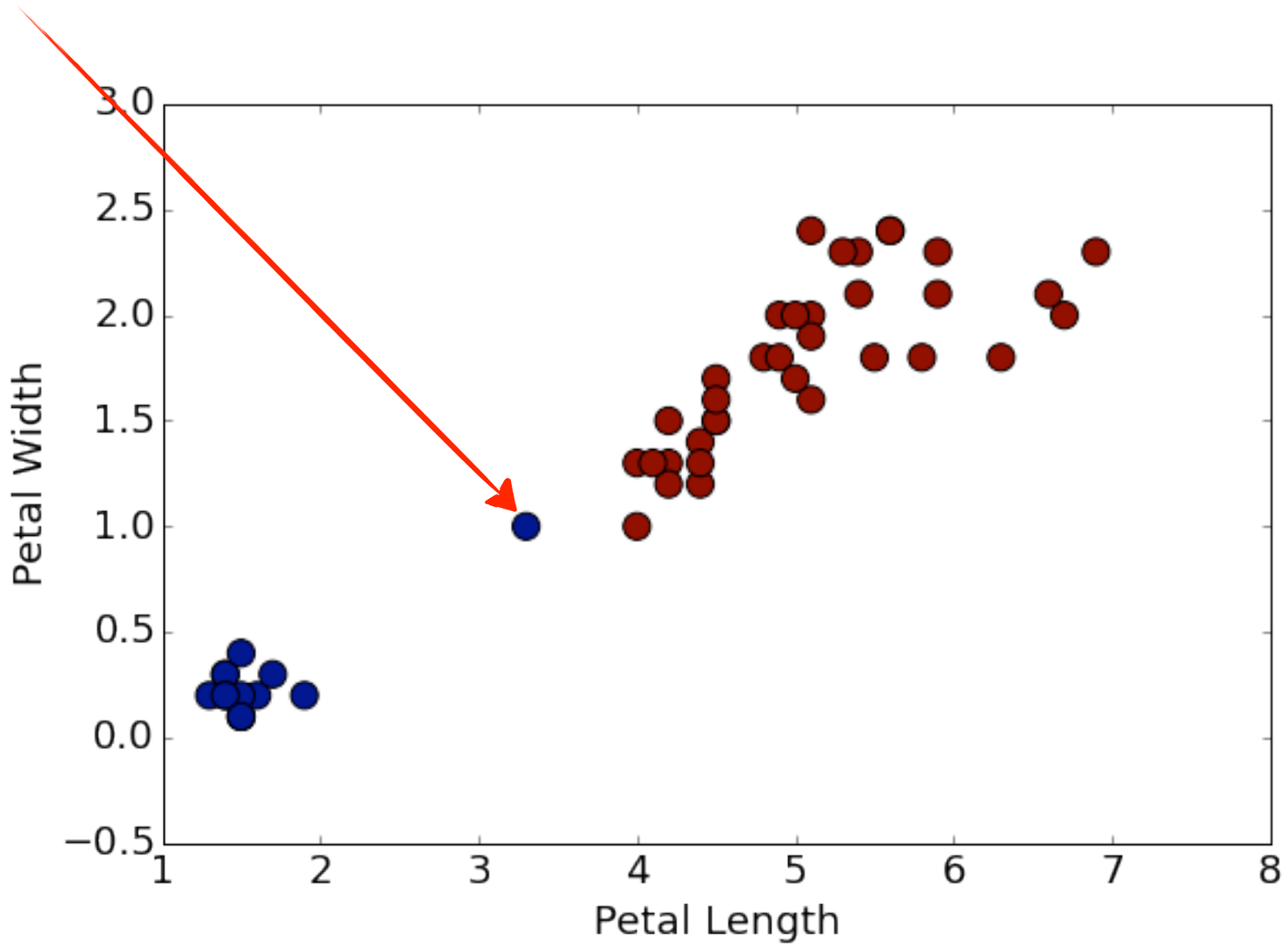
KMeans in *sklearn*

```
from sklearn.cluster import KMeans

model = KMeans(n_clusters=2)
model.fit(data)

print “Cluster assignment: “, model.predict(data)
print
print “Cluster centroids: “, model.cluster_centers_
```

```
Cluster assignment:  [1 1 0 1 1 1 1 1 1 0 0 1 1 1 0 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 0 1
0 0 0 1 1 0 0 0 1 1 1 1 1 0 1 1]

Cluster centroids:  [[ 1.62        0.26666667] [ 5.08        1.79142857]]
```

Hard vs. Soft Clustering

- Could this point also belong to the other cluster?



Soft Clustering

- We can estimate a "membership degree", or a probability that a point belongs to a cluster.

  h_{ik}: the probability that the data point x_i belongs to cluster k

- This probability can show the reliability of a prediction:
  - The flower looks kind of like an iris, but I'm only 73% sure.

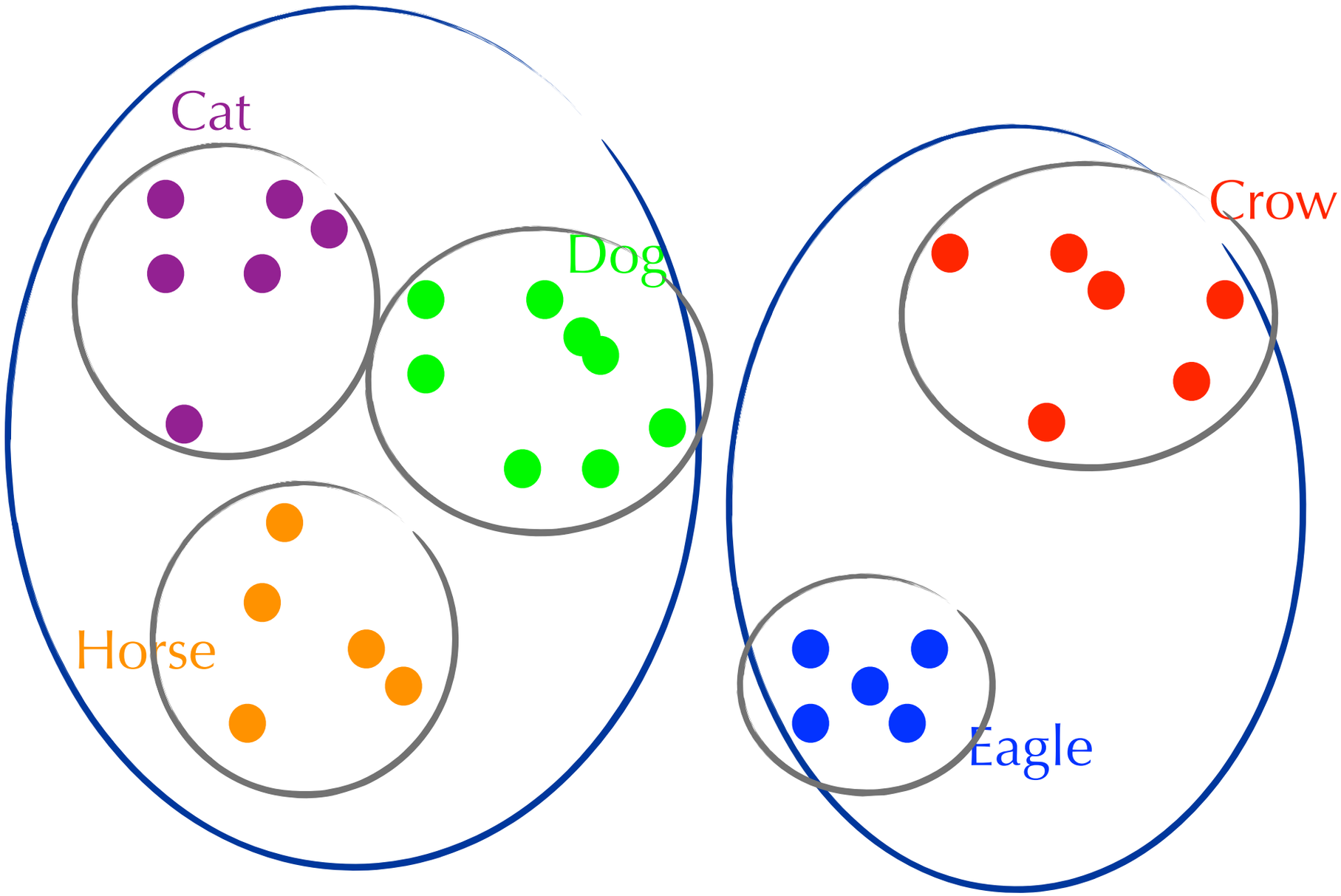
- ... or it can show an actual multi-membership:
  - An email can be 65% related to work and 35% related to friends
  - A document can be 82% about politics and 18% about science

EM Algorithm in *sklearn*

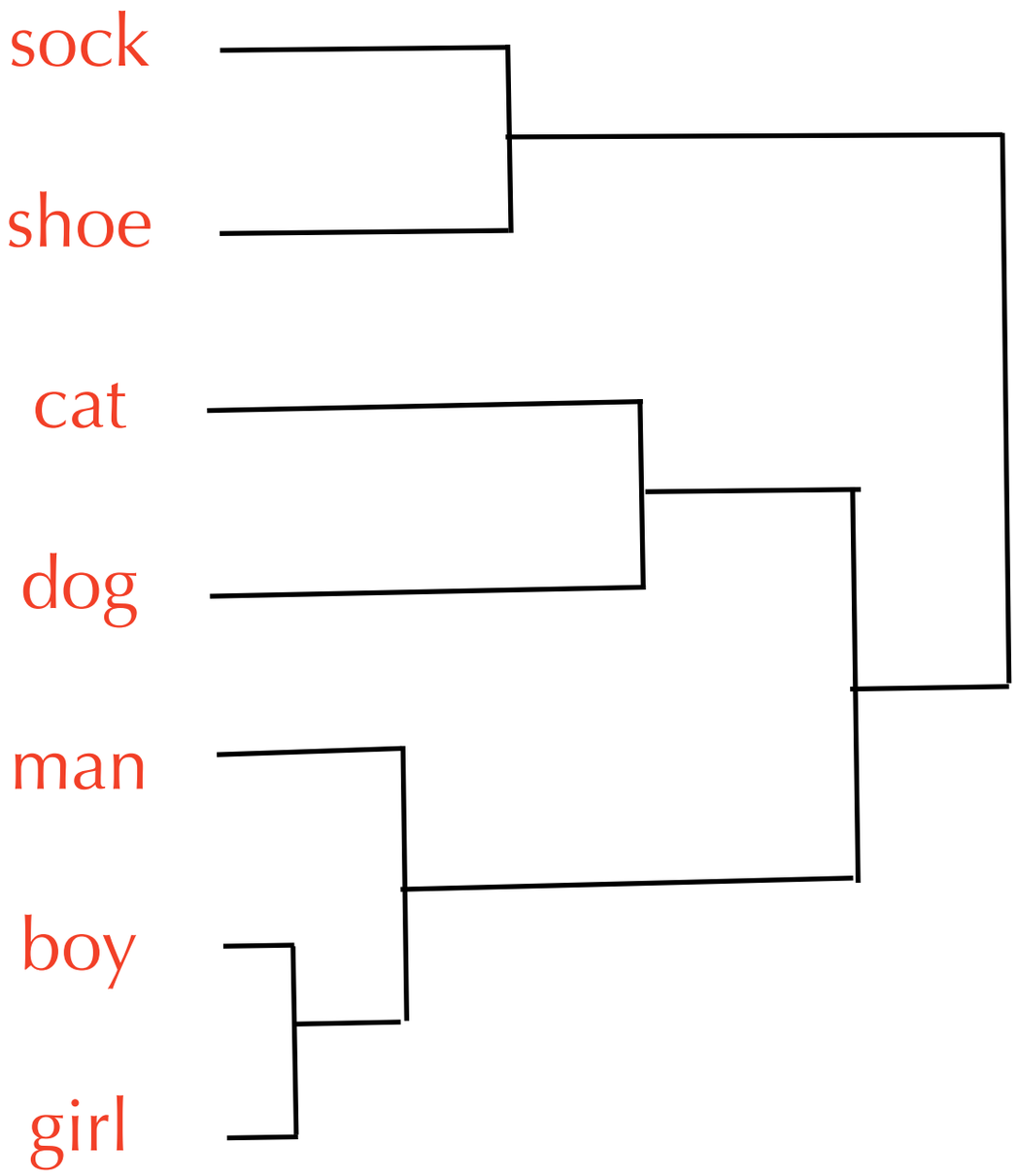
```
from sklearn.mixture import GMM

model = GMM(n_components=2, n_iter=20)
model.fit(data)

print "Membership probabilities: ",
  model.predict_proba(data)

print "Cluster prediction: ", model.predict(data)
```

```
Membership probabilities:  [[  1.00000000e+000   3.45284645e-107]
 [  1.00000000e+000   4.74279740e-102]
 [  5.47886151e-009   9.99999995e-001]
    ...

Cluster prediction:  [0 0 1 0 0 0 0 0 0 1 1 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 1
1 0 0 1 1 1 0 0 0 0 0 1 0 0]
```

Flat vs. Hierarchical Clustering



Hierarchical Clustering



Evaluating Cluster Quality

- How do we evaluate the quality of the induced clusters?
  - There is no gold standard to compare our clusters against, therefore no precision/recall estimates
  - We can use an objective function as a measure of coherence

- If the induced clusters are used by another task, the performance in that task can be an indicator of the quality of clusters

Many Cognitive Tasks are Unsupervised

- Image processing:
  - Recognizing edges, texture, shadows, ...
  - Estimating distance, overlap, spatial relations, ...
  - Identifying objects

- Formation of concepts:
  - categorizing visual entities (e.g., furniture, humans, food) based on their features (shape, color, size, movement, etc.)
  - categorizing relations (e.g., causal movement, manner of motion, change of state) based on their participants
  - categorizing abstract concepts (e.g., emotions, artistic style, etc)

Example: Learning Language

Words that behave similarly belong to the same category

You can eat the apples.

He bought some apples from the market.

Apples are yummy.

She can eat the oranges.

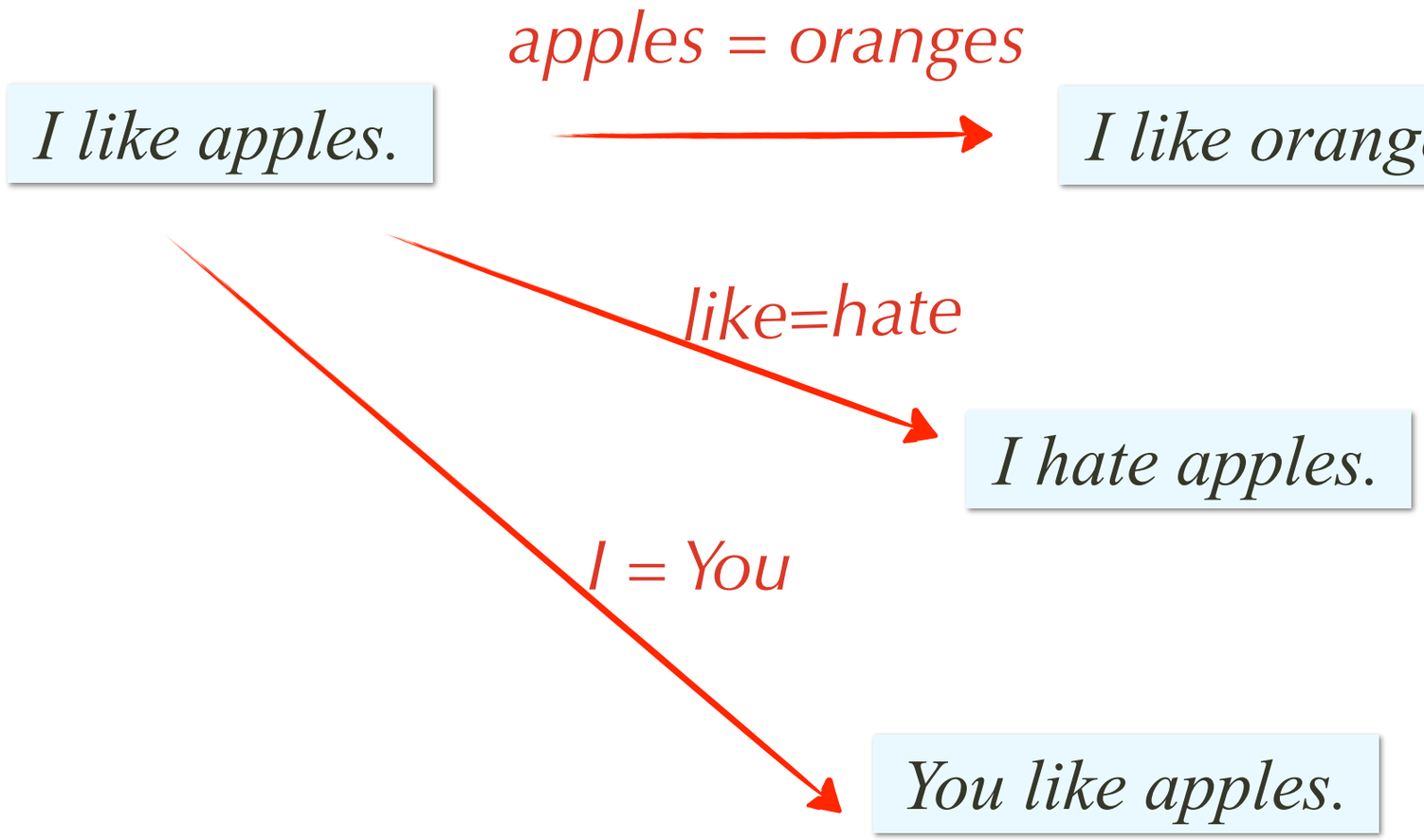
I bought some oranges from the shop.

Oranges are yummy.

apples = oranges

Example: Learning Language

Word categories can facilitate language learning and use:

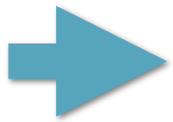


Learning Lexical Categories

- Word usage: a vector of content and context features:

-2 -1 0 1 2

"want to put them on"



-2=want	-1=to	0=put	1=them	2=on
1	1	1	1	1

- A lexical category is a cluster of word usages
- Centroid: the mean of the distribution vectors of its members

-2=want	-2=have	-1=to	0=go	0=sit	0=show	0=send	1=it	...
0.25	0.75	1	0.25	0.25	0.25	0.25	0.5	...

Top 10 Words for 10 Clusters

do are have can not go put did get play
is that it what not there he was where put
you not I the we what it they your a
to you we and I will not can it on
it a that the not he this right got she
are do is have on in can want did going
one I not shall there then you are we it
is in are on oh with and of have do
the a your of that it this some not very
going want bit go have look got will at little

Questions?