



Machine Learning - Clustering

CS102
Spring 2020

Data Tools and Techniques

- **Basic Data Manipulation and Analysis**
Performing well-defined computations or asking well-defined questions (“queries”)
- **Data Mining**
Looking for patterns in data
- **Machine Learning**
Using data to build models and make predictions
- **Data Visualization**
Graphical depiction of data
- **Data Collection and Preparation**

Machine Learning

Using data to build models and make predictions

Supervised machine learning

- Set of labeled examples to learn from: training data
- Develop model from training data
- Use model to make predictions about new data

Unsupervised machine learning

- Unlabeled data, look for patterns or structure (similar to data mining)

Clustering

Like classification, data items consist of **values** for a set of **features** (numeric or categorical)

- Medical patients

Feature values: age, gender, symptom1-severity, symptom2-severity, test-result1, test-result2

- Web pages

Feature values: URL domain, length, #images, heading₁, heading₂, ..., heading_n

- Products

Feature values: category, name, size, weight, price

Clustering

Like classification, data items consist of **values** for a set of **features** (numeric or categorical)

- Medical patients

Feature values: age, gender, symptom1-severity, symptom2-severity, test-result1, test-result2

Unlike classification,
there is no label

- Web pages

Feature values: URL domain, length, #images, heading₁, heading₂, ..., heading_n

- Products

Feature values: category, name, size, weight, price

Clustering

Like K-nearest neighbors, for any pair of data items i_1 and i_2 , from their feature values can compute distance function: $distance(i_1, i_2)$

Example:

Features - gender, profession, age, income, postal-code

person₁ = (male, teacher, 47, \$25K, 94305)

person₂ = (female, teacher, 43, \$28K, 94309)

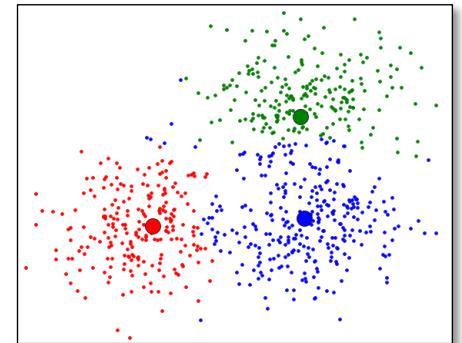
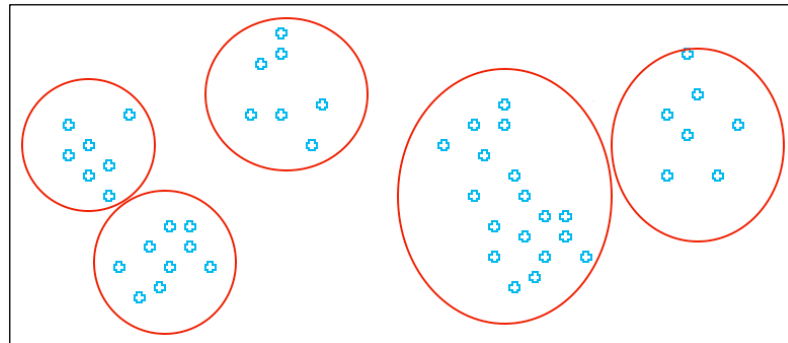
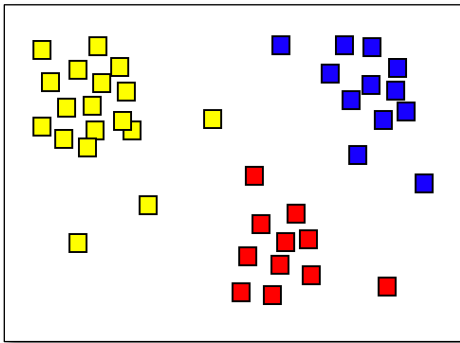
$distance(\text{person}_1, \text{person}_2)$

$distance()$ can be defined as inverse of $similarity()$

Clustering

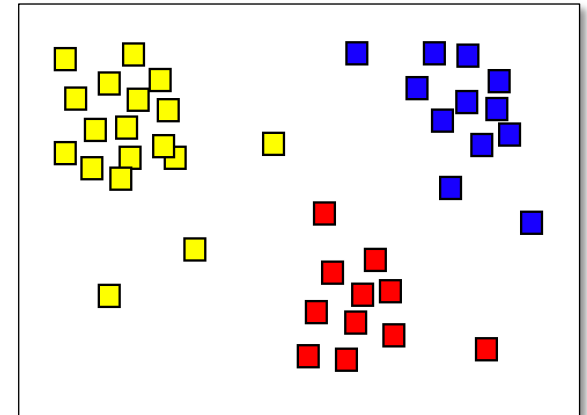
GOAL: Given a set of data items, partition them into groups (= clusters) so that items within groups are close to each other based on distance function

- Sometimes number of clusters is pre-specified
- Typically clusters need not be same size



Some Uses for Clustering

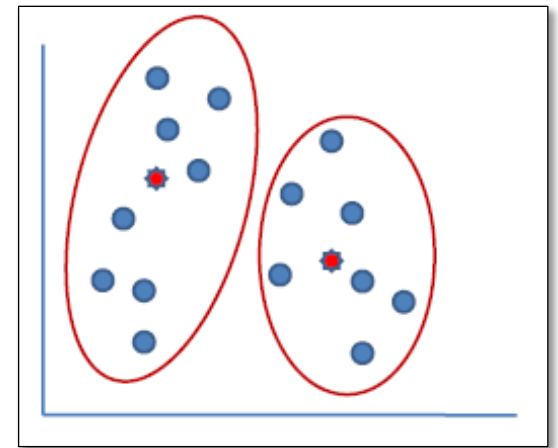
- **Classification!**
 - Assign labels to clusters
 - Now have labeled training data for future classification
- **Identify similar items**
 - For substitutes or recommendations
 - For de-duplication
- **Anomaly (outlier) detection**
 - Items that are far from any cluster



K-Means Clustering

Reminder: for any pair of data items i_1 and i_2 have $distance(i_1, i_2)$

For a group of items, the **mean value (centroid)** of the group is the item i (in the group or not) that minimizes the sum of $distance(i, i')$ for all i' in the group

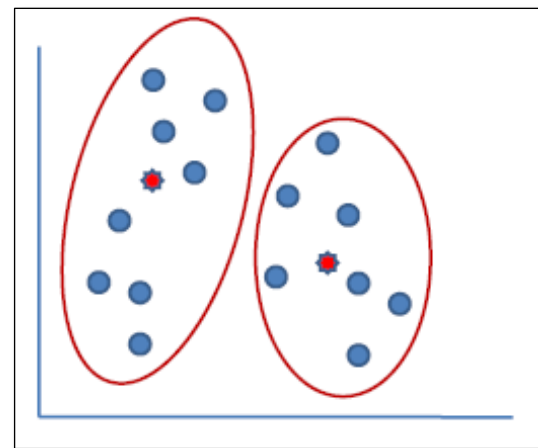


K-Means Clustering

For a group of items, the **mean value (centroid)** of the group is the item i (in the group or not) that minimizes the sum of $distance(i, i')$ for all i' in the group

- **Error** for each item: distance d from the mean for its group; **squared error** is d^2
- **Error** for the entire clustering: **sum of squared errors (SSE)**

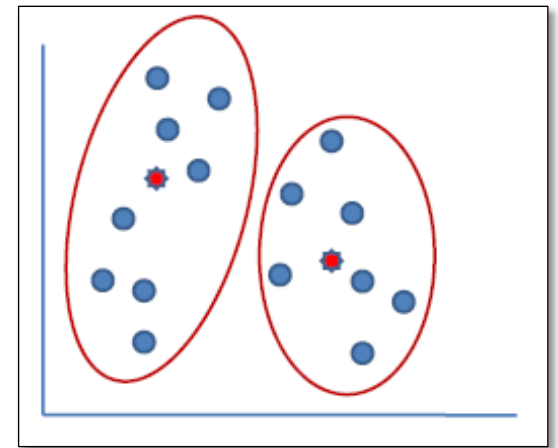
Remind you of anything?



K-Means Clustering

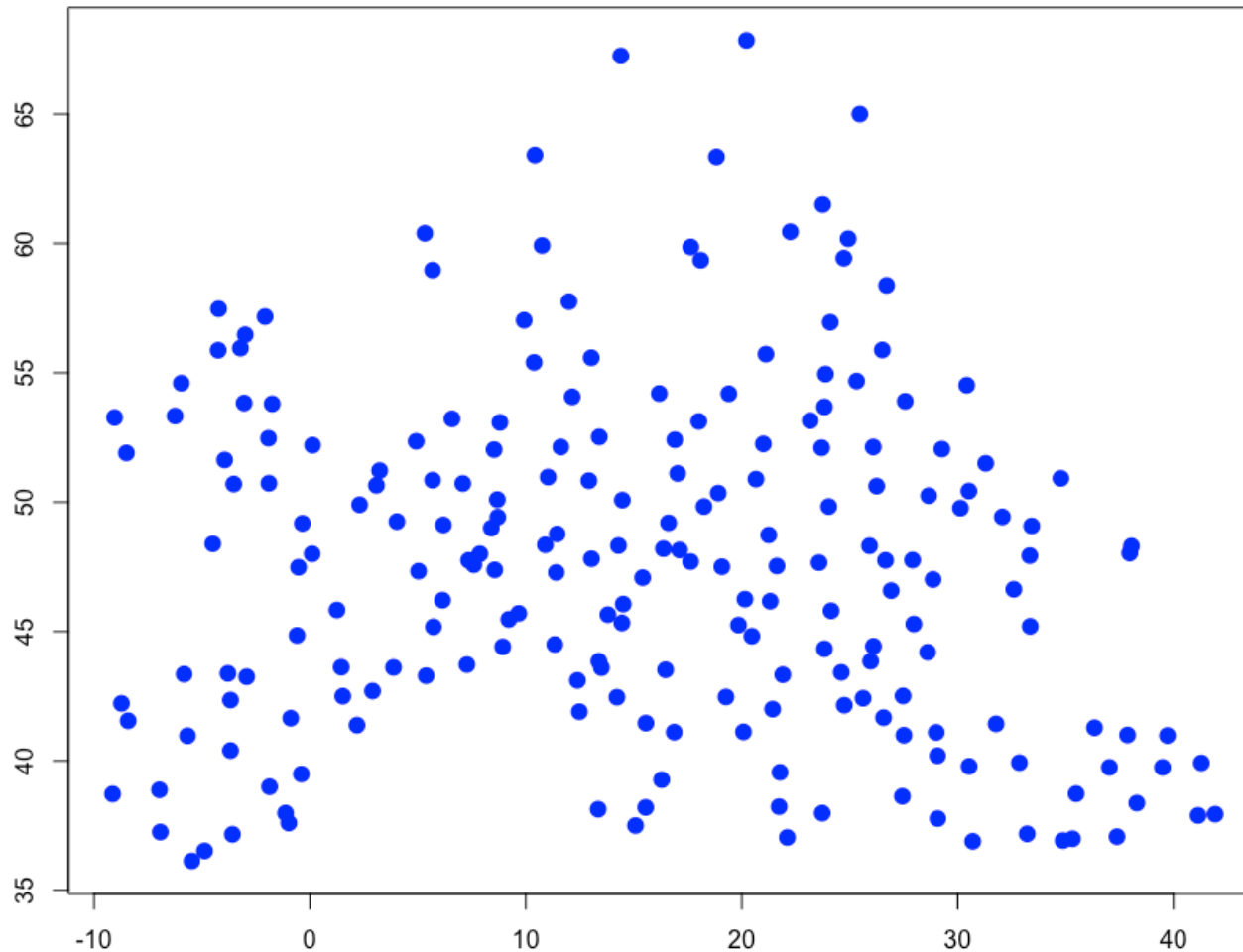
Given set of data items and desired number of clusters k , K-means groups the items into k clusters minimizing the SSE

- Extremely difficult to compute efficiently
 - In fact, impossible
- Most algorithms compute an **approximate** solution (might not be absolute lowest SSE)



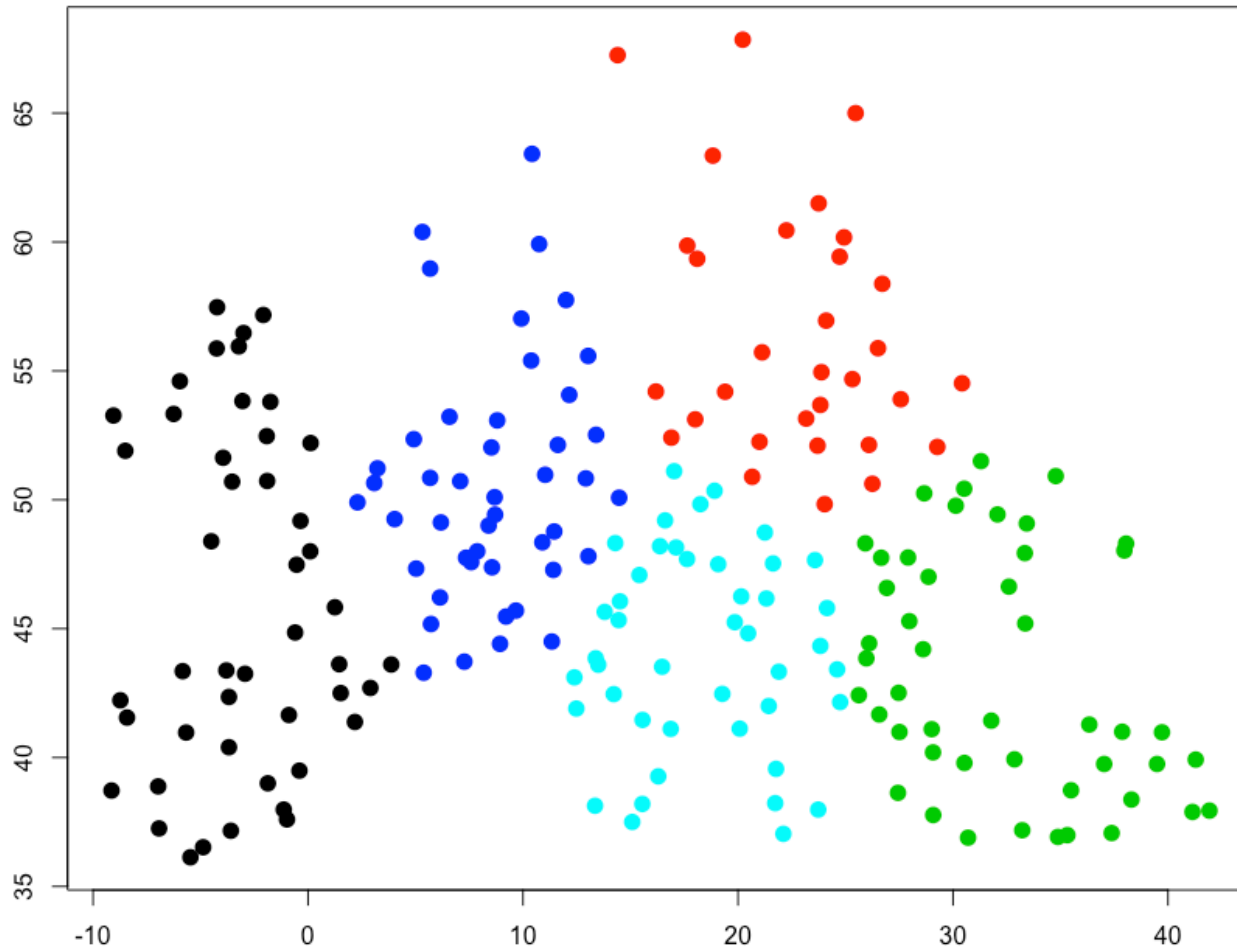
Clustering European Cities

By geographic distance, then by temperature



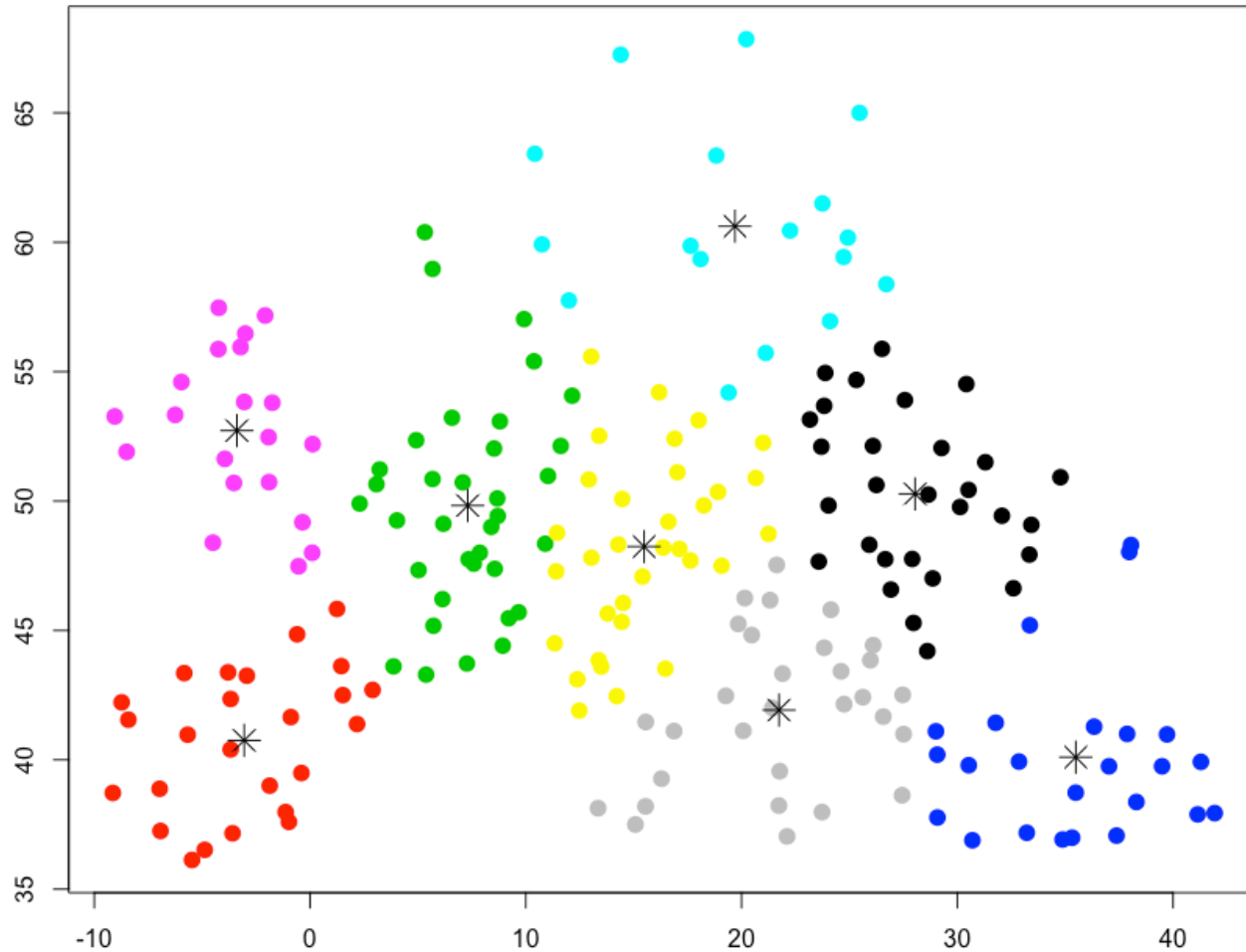
Clustering European Cities

Distance = actual distance, $k = 5$



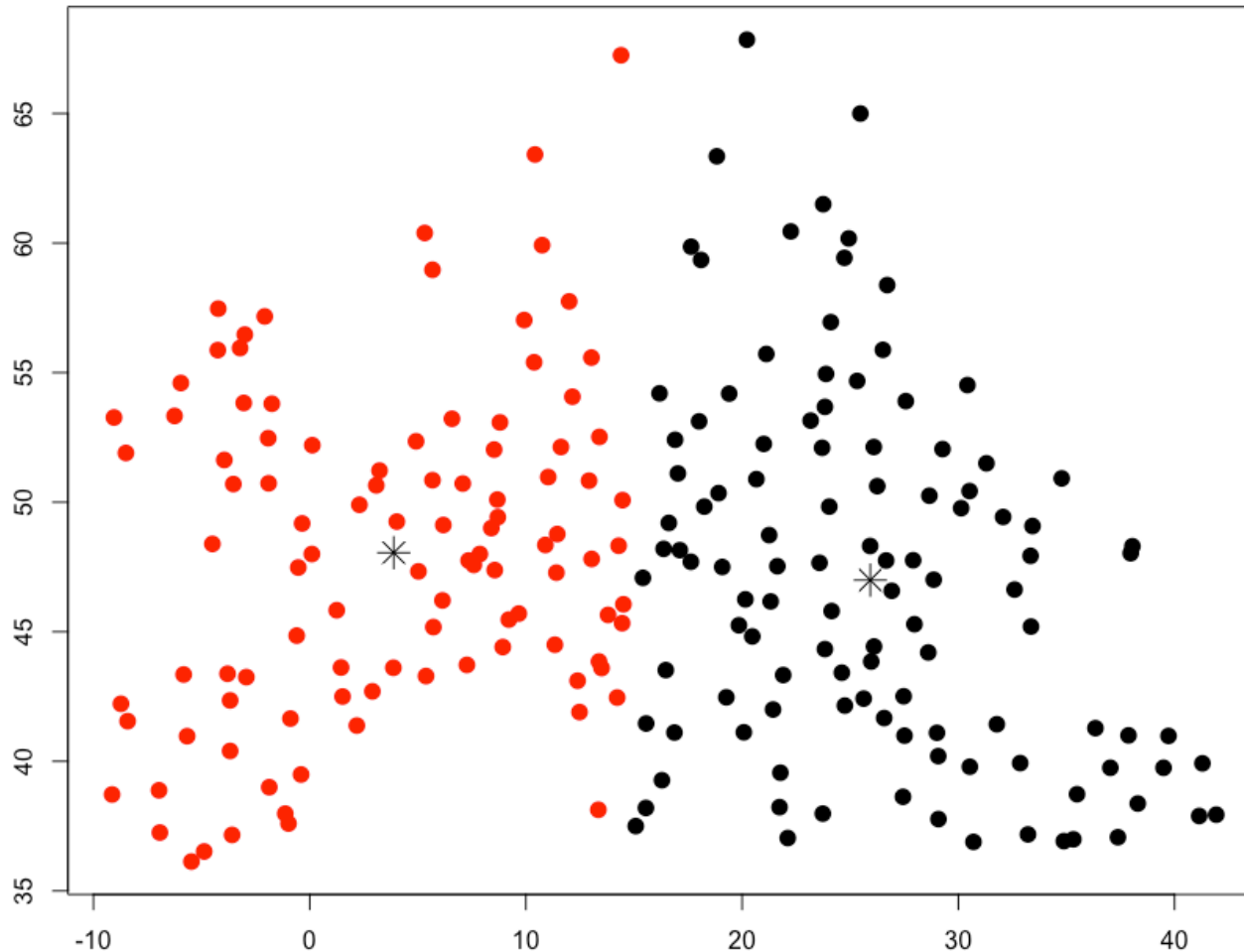
Clustering European Cities

Distance = actual distance, $k = 8$, with cluster means



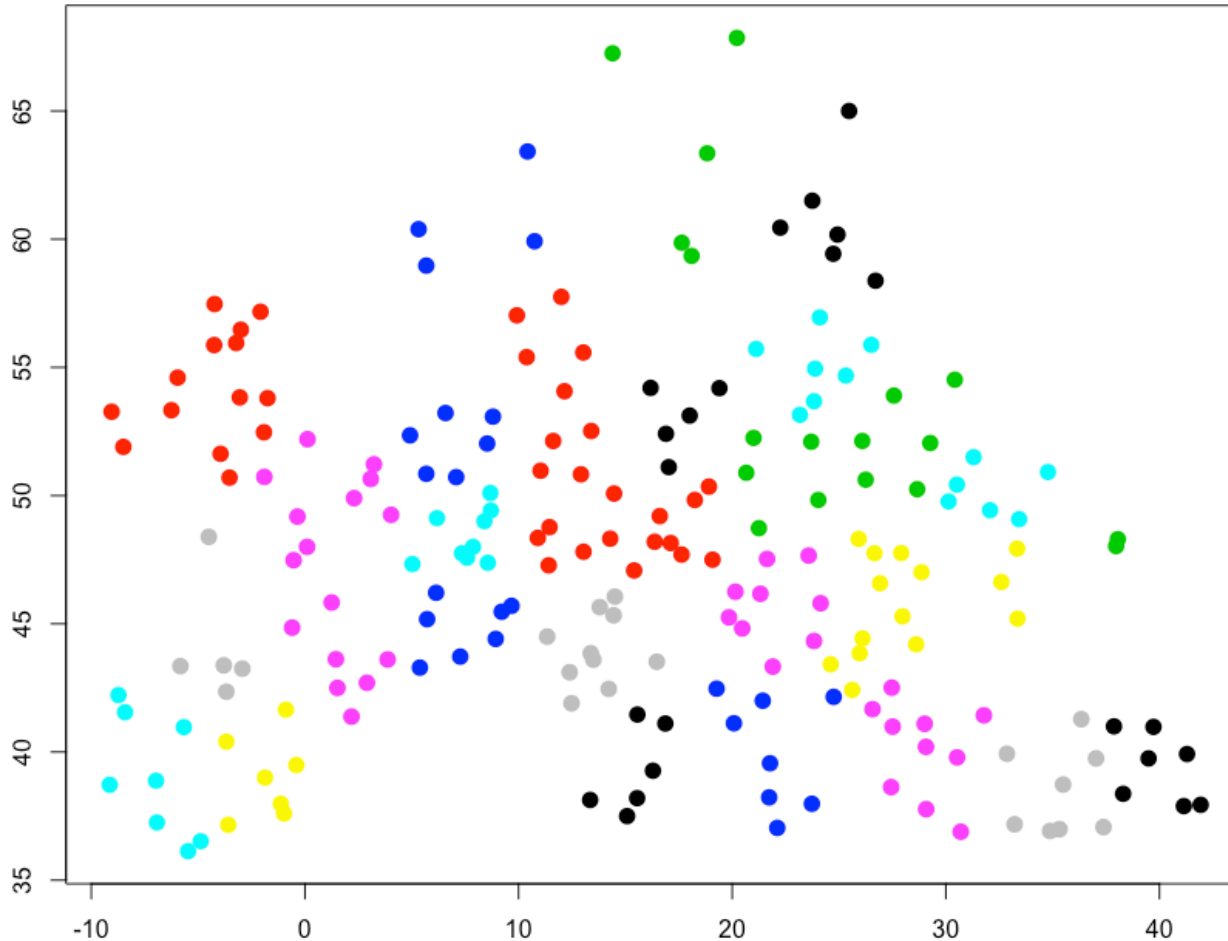
Clustering European Cities

Distance = actual distance, $k = 2$, with cluster means



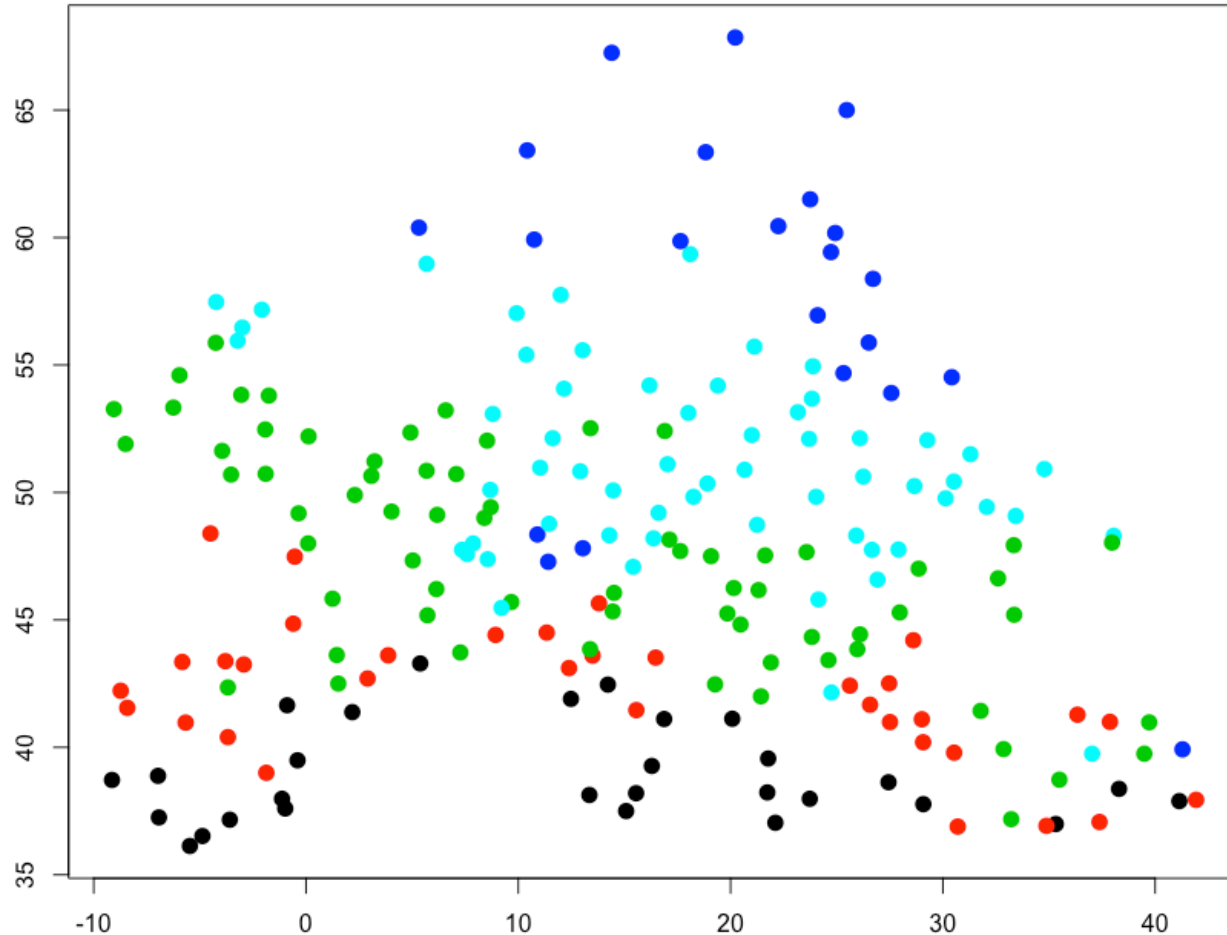
Clustering European Cities

Distance = actual distance, $k = 30$



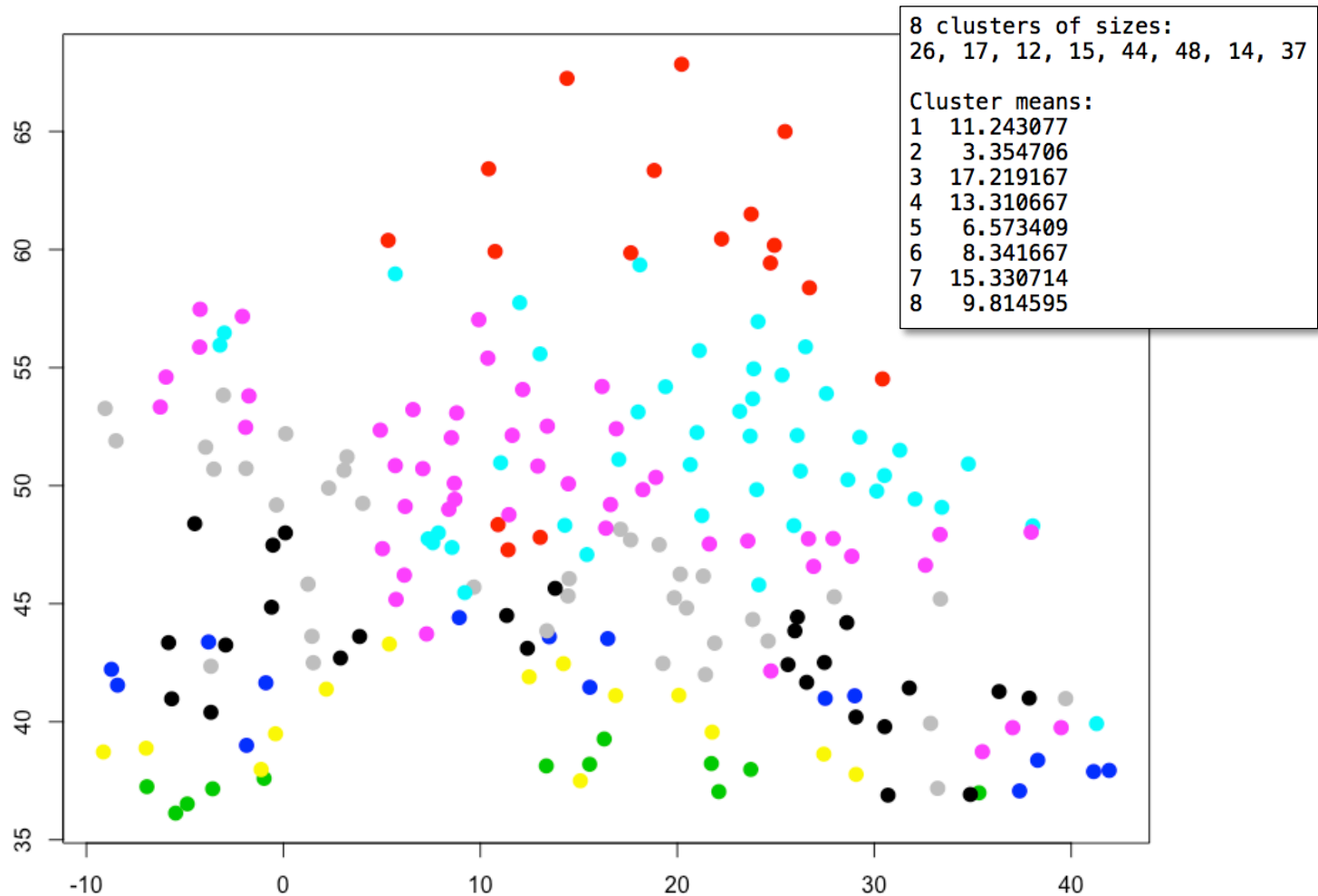
Clustering European Cities

Distance = temperature, $k = 5$



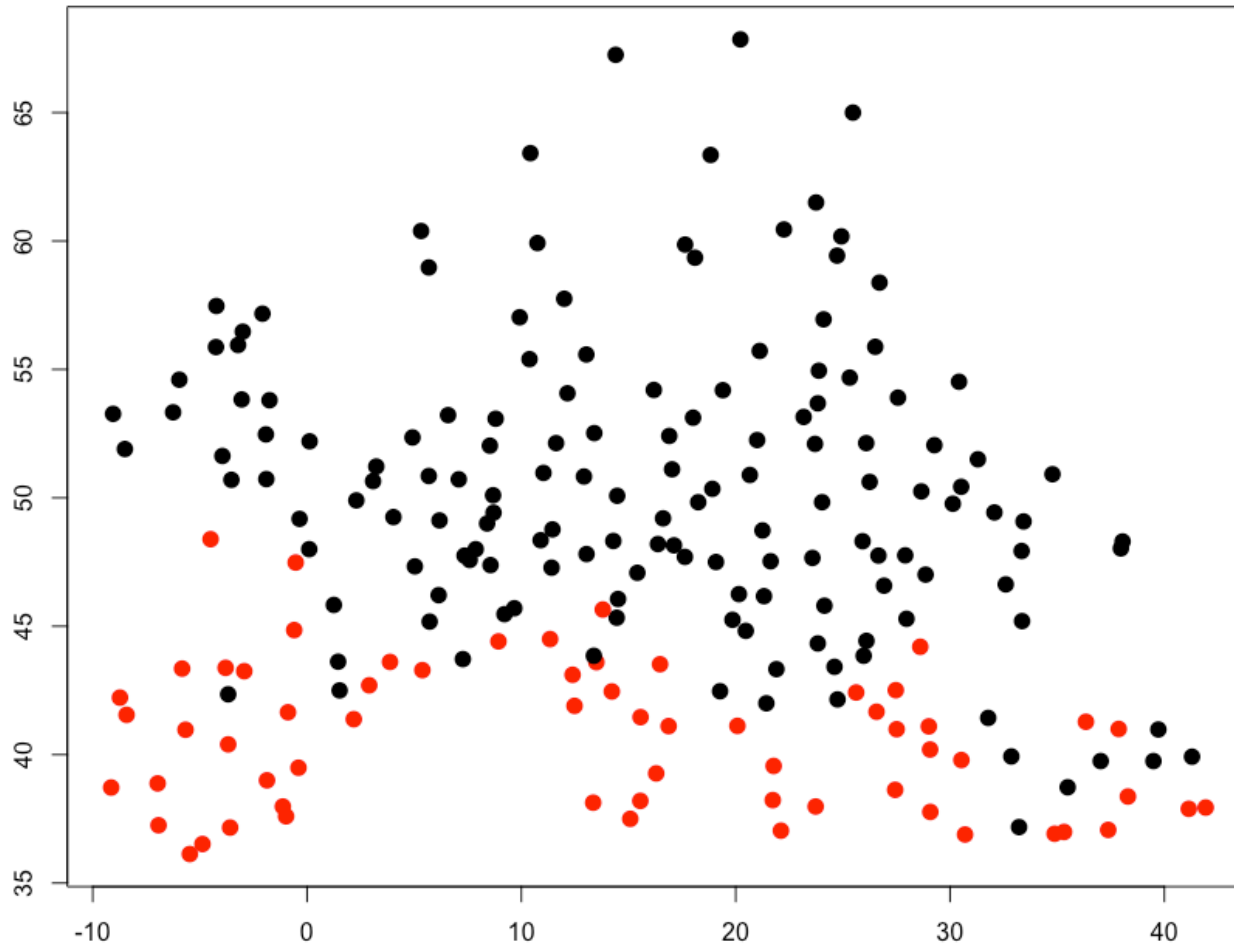
Clustering European Cities

Distance = temperature, $k = 8$, with means



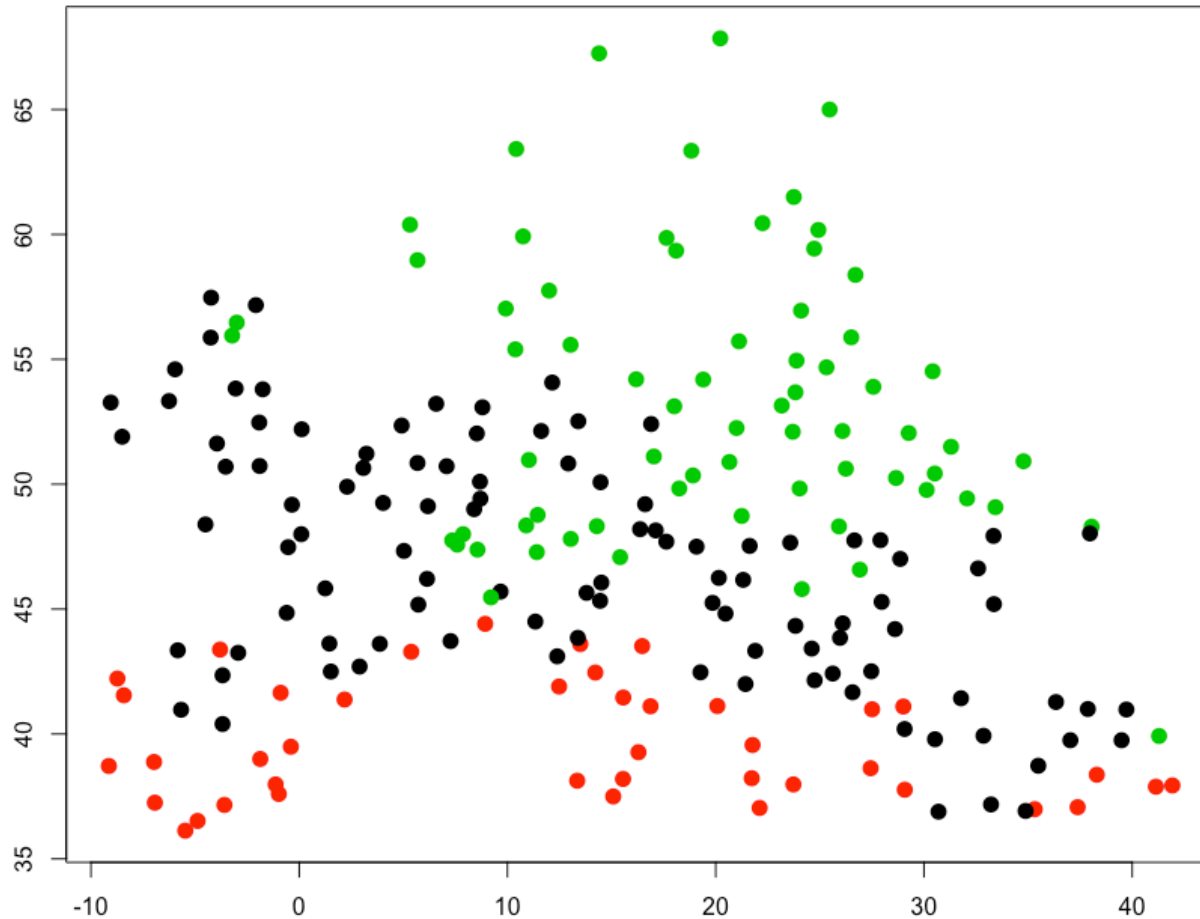
Clustering European Cities

Distance = temperature, $k = 2$



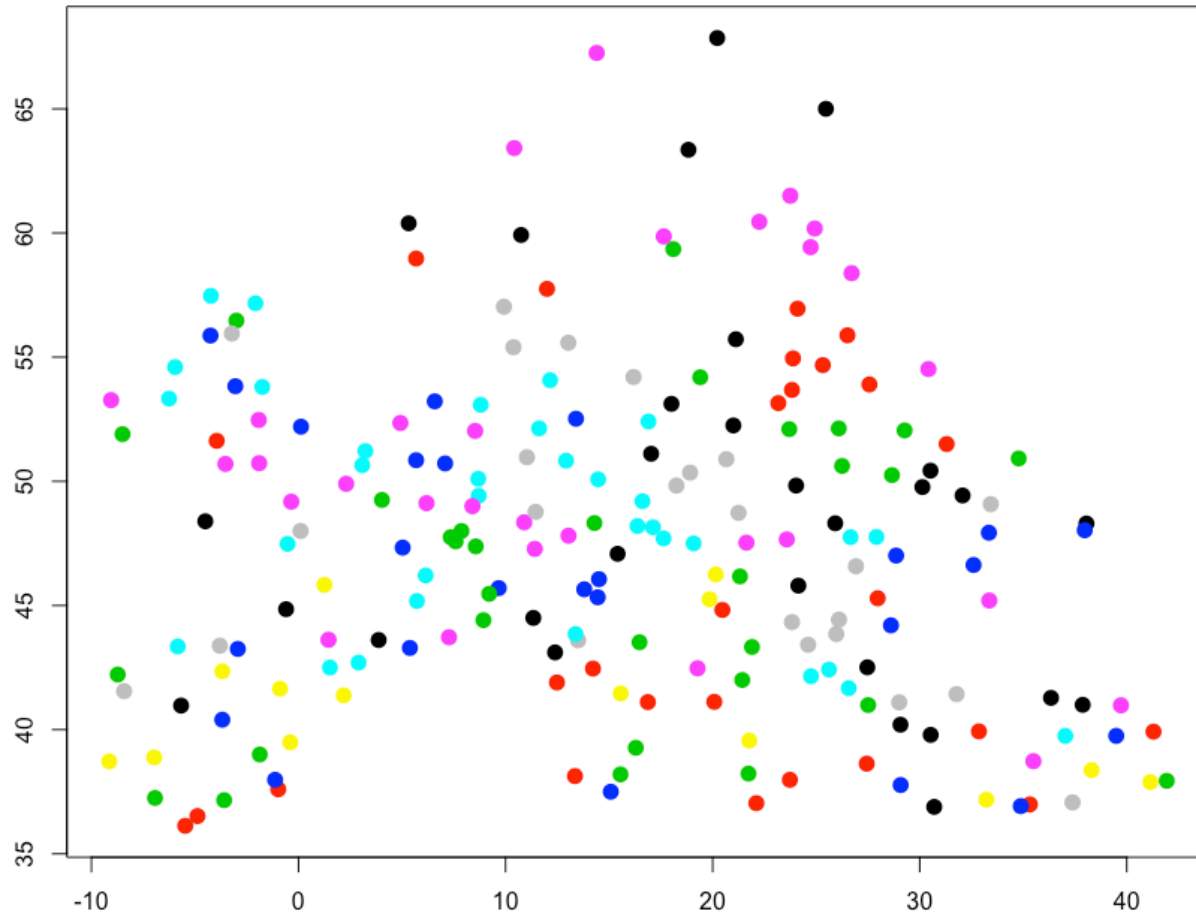
Clustering European Cities

Distance = temperature, $k = 3$



Clustering European Cities

Distance = temperature, $k = 30$



Some Uses for Clustering

- Classification
 - Assign labels to clusters
 - Now have labeled training data for future classification
- Identify similar items
 - For substitutes or recommendations
 - For de-duplication
- Anomaly (outlier) detection
 - Items that are far from any cluster