Week 9

Agenda

1. Wrap up supervised learning
2. Check-in project 4
3. K-means review
4. If time: Notebook

Unsupervised vs. Supervised Learning

1. What is it?
2. When do we use it?
3. Common use cases?

Breakout: project 2

K-means - Lloyd's algorithm

- Goal: assign each of *N* points/observations to one of *K* clusters, where *K* is determined a priori
- Each cluster has a centroid μ_k
- Loss function (Euclidean distance):

$$J = \sum_{i=1}^{N} \sum_{k=1}^{K} r_{ik} \|x_i - \mu_k\|_2^2$$

- How to minimize loss function:
  - Choose centroids μ_k
  - Assign each data point to centroid
  - Realign centroid to center of mass
  - Repeat last two steps until complete
- This process will converge to a local minimum
- Pseudo-code:

```
Initially choose k points that are likely to be in different
clusters;
Make these points the centroids of their clusters;
FOR each remaining point p DO
   find the centroid to which p is closest;
   Add p to the cluster of that centroid;
   Adjust the centroid of that cluster to account for p;
END;
```

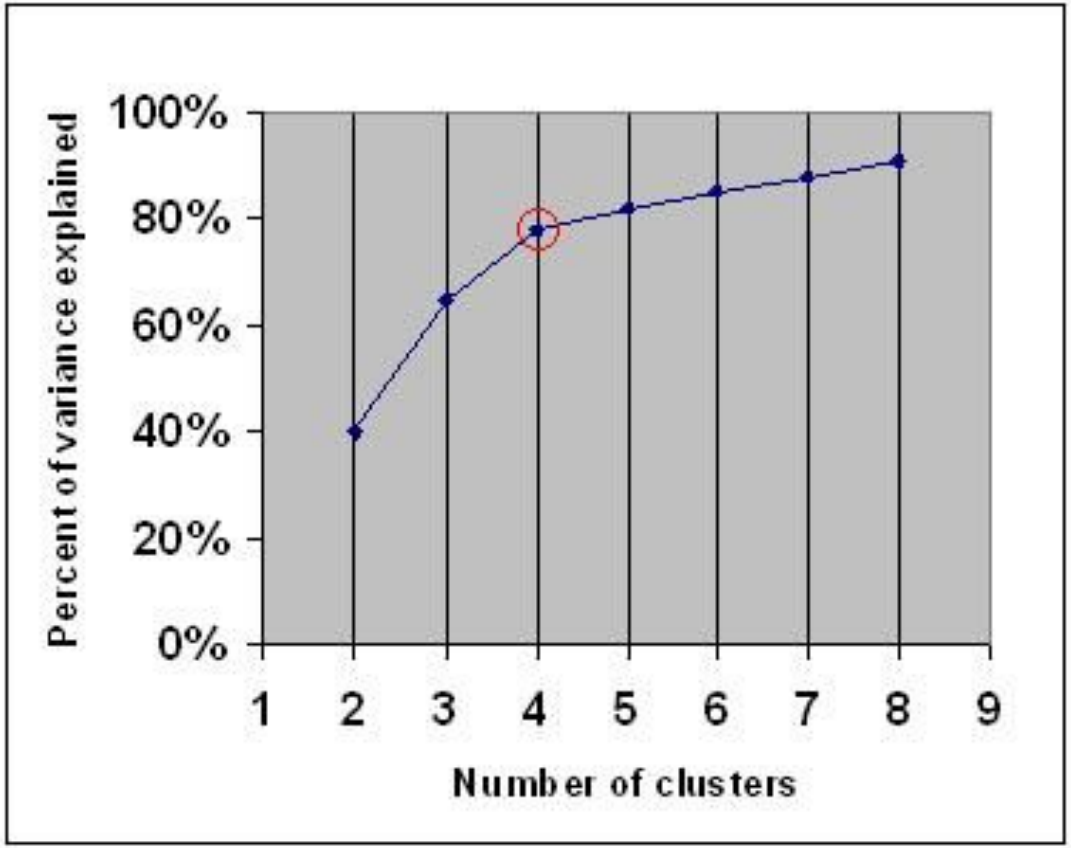
1. Describe the training algorithm.
2. What is the complexity?
3. The underlying problem is NP hard. Algo provides local best solution.

K-means - too simple to be true?

- Choose centroids μ_k

1. What is a good k?
2. What are good initial points?

K-means - find "k" with elbow method



K-means - other ways to "k"

1. Silhouette coefficient: mean intra-cluster distance and mean nearest-cluster distance -> How far is a sample from its own cluster center and another cluster center?
2. Density plot or Kernel density estimation: see notebook

K-means - Find "best" starting points with K-Means++

1. Randomly select first centroid.
2. Compute distance from chosen centroid for each data point.
3. Select next centroid: probability of choosing a point as centroid directly proportional to its distance from nearest centroid
4. Repeat 2 and 3 until k centroids

Breakout

K-means

k-Means Clustering: Perspective

- Pros:
  - Fast, reasonable approximation for spherical data
  - Intuitive
  - Guaranteed to converge
  - Each point assigned to exactly one cluster
- Cons:
  - Points assigned to exactly one cluster
    - Assignment can be sensitive
    - Clusters can be sensitive to data, especially outliers