Announcement

• Those of you who submitted HW1 and are still on waitlist email me.
Clustering

- Grouping sets of data points s.t.
  - points in same group are similar
  - points in different groups are dissimilar

- A form of unsupervised classification where there are no predefined labels
Kary clustering is a partition of $x_1, \ldots, x_n$ into $K$ groups.

For now assume the magical $K$ is given to use.

Clustering given by $C_1, \ldots, C_K$, the partition of data points.

Given a clustering, we shall use $c(x_t)$ to denote the cluster identity of point $x_t$ according to the clustering.

Let $n_j$ denote $|C_j|$, clearly $\sum_{j=1}^{K} n_j = n$. 
How do we formalize a good clustering objective?
How do we formalize?

Say dissimilarity($x_t, x_s$) measures dissimilarity between $x_t$ & $x_s$.

Given two clustering $\{C_1, \ldots, C_K\}$ (or $c$) and $\{C'_1, \ldots, C'_K\}$ (or $c'$)

How do we decide which is better?

- points in same cluster are not dissimilar
- points in different clusters are dissimilar
Clustering Criterion

- Minimize total within-cluster dissimilarity
  \[ M_1 = \sum_{j=1}^{K} \sum_{s,t \in C_j} \text{dissimilarity}(x_t, x_s) \]

- Maximize between-cluster dissimilarity
  \[ M_2 = \sum_{x_s, x_t: c(x_s) \neq c(x_t)} \text{dissimilarity}(x_t, x_s) \]

- Maximize smallest between-cluster dissimilarity
  \[ M_3 = \min_{x_s, x_t: c(x_s) \neq c(x_t)} \text{dissimilarity}(x_t, x_s) \]

- Minimize largest within-cluster dissimilarity
  \[ M_4 = \max_{j \in [K]} \max_{s,t \in C_j} \text{dissimilarity}(x_t, x_s) \]
Minimize average dissimilarity within cluster

\[ M_6 = \sum_{j=1}^{K} \frac{1}{|C_j|} \sum_{s \in C_j} \text{dissimilarity} \left( x_s, C_j \right) \]

\[ = \sum_{j=1}^{K} \frac{1}{|C_j|} \sum_{s \in C_j} \left( \sum_{t \in C_j, t \neq s} \text{dissimilarity} \left( x_s, x_t \right) \right) \]

\[ = \sum_{j=1}^{K} \frac{1}{|C_j|} \sum_{s \in C_j} \left( \sum_{t \in C_j, t \neq s} \| x_s - x_t \|_2^2 \right) \]

Minimize within-cluster variance: \( \mathbf{r}_j = \frac{1}{n_j} \sum_{x \in C_j} x \)

\[ M_5 = \sum_{j=1}^{K} \sum_{t \in C_j} \| x_t - \mathbf{r}_j \|_2^2 \]
How different are these criteria?
minimizing $M_1 \equiv$ maximizing $M_2$

minimizing $M_5 \equiv$ minimizing $M_6$
Multiple clustering criteria all equally valid
Different criteria lead to different algorithms/solutions
Which notion of distances or costs we use matter
Let's build algorithm for two criteria

1. \[ M_5 = \sum_{j=1}^{K} \sum_{t \in C_j} \| x_t - r_j \|_2^2 \]

2. \[ M_3 = \min_{x_s, x_t : c(x_s) \neq c(x_t)} \text{dissimilarity}(x_t, x_s) \]
Let's build an Algorithm

\[ M_5 = \sum_{j=1}^{K} \sum_{t \in C_j} \left\| x_t - r_j \right\|^2_2 \]

where \( r_j = \frac{1}{|C_j|} \sum_{t \in C_j} x_t \)
Demo
Demo
Demo
Demo
Demo
Demo
K-means Clustering

- For all $j \in [K]$, initialize cluster centroids $\hat{r}_j^1$ randomly and set $m = 1$
- Repeat until convergence (or until patience runs out)
  1. For each $t \in \{1, \ldots, n\}$, set cluster identity of the point
     
     $$\hat{c}^m(x_t) = \arg\min_{j\in[K]} \|x_t - \hat{r}_j^m\|$$

  2. For each $j \in [K]$, set new representative as
     
     $$\hat{r}_j^{m+1} = \frac{1}{|\hat{C}_j^m|} \sum_{t \in \hat{C}_j^m} x_t$$

  3. $m \leftarrow m + 1$
K-means Convergence

- K-means algorithm converges to local minima of objective

\[
O(c; r_1, \ldots, r_K) = \sum_{j=1}^{K} \sum_{c(x_t) = j} \|x_t - r_j\|^2
\]

- Proof:
  Clustering assignment improves objective:

\[
O(\hat{c}^{m-1}; r_1^m, \ldots, r_K^m) \geq O(\hat{c}^m; r_1^m, \ldots, r_K^m)
\]

(By definition of \(\hat{c}^m(x_t)\))

Computing centroids improves objective:

\[
O(\hat{c}^m; r_1^m, \ldots, r_K^m) \geq O(\hat{c}^m; r_1^{m+1}, \ldots, r_K^{m+1})
\]

(By the fact about centroid)
Let's build an Algorithm

\[ M_3 = \min_{x_s, x_t : c(x_s) \neq c(x_t)} \text{dissimilarity}(x_t, x_s) \]
Demo
\[ \text{dissimilarity}(C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s) \]
demo

\[
\text{dissimilarity}(C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s)
\]
\[
\text{dissimilarity}(C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s)
\]
\[
\text{dissimilarity}(C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s)
\]
Demo

dissimilarity\( (C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s) \)
dissimilarity($C_i, C_j$) = \( \min_{t \in C_i, s \in C_j} \) dissimilarity($x_t, x_s$)
\[
\text{dissimilarity}(C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s)
\]
dissimilarity($C_i, C_j$) = \[ \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s) \]
Demo

dissimilarity\( (C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s) \)
\[ \text{dissimilarity}(C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s) \]
Demo
**Single Link Clustering**

- Initialize $n$ clusters with each point $x_t$ to its own cluster

- Until there are only $K$ clusters, do
  1. Find closest two clusters and merge them into one cluster
  2. Update between cluster distances (called proximity matrix)
**Single Link Clustering**

- Initialize $n$ clusters with each point $x_t$ to its own cluster

- Until there are only $K$ clusters, do
  1. Find closest two clusters and merge them into one cluster
  2. Update between cluster distances (called proximity matrix)

$$\text{dissimilarity}(C_i, C_j) = \min_{t \in C_i, s \in C_j} \text{dissimilarity}(x_t, x_s)$$
Objective for single-link:

\[ M_3 = \min_{x_s, x_t : c(x_s) \neq c(x_t)} \text{dissimilarity}(x_t, x_s) \]

Single link clustering is optimal for above objective!