Clustering with Minimum Spanning Tree

Slides by Carl Kingsford

Jan. 24, 2014

KT 4.6
Clustering: an application of MST
Clustering

You’re given $n$ items and the distance $d(u, v)$ between each of pair.

$d(u, v)$ may be an actual distance, or some abstract representation of how dissimilar two things are. (E.g. the “distance” between two species.)

**Our Goal:** Divide the $n$ items up into $k$ groups so that the minimum distance between items in different groups is maximized.
Our Goal: Divide the $n$ items up into $k$ groups so that the minimum distance between items in different groups is maximized.
Maximum Minimum Distance

Idea:

- Maintain clusters as a set of connected components of a graph.
- Iteratively combine the clusters containing the two closest items by adding an edge between them.
- Stop when there are $k$ clusters.
Maximum Minimum Distance

Idea:

- Maintain clusters as a set of connected components of a graph.
- Iteratively combine the clusters containing the two closest items by adding an edge between them.
- Stop when there are $k$ clusters.

This is exactly Kruskal’s algorithm.

The “clusters” are the connected components that Kruskal’s algorithm has created after a certain point.

Example of “single-linkage, agglomerative clustering.”
Proof of Correctness

Another way too look at the algorithm: delete the $k - 1$ most expensive edges from the MST.

The spacing $d$ of the clustering $C$ that this produces is the length of the $(k - 1)^{st}$ most expensive edge.

Let $C'$ be a different clustering. We’ll show that $C'$ must have the same or smaller separation than $C$. 
Proof of correctness, 2

Since $C \neq C'$, there must be some pair $p_i, p_j$ that are in the same cluster in $C$ but different clusters in $C'$.

Together in $C \implies \text{path } P \text{ between } p_i, p_j \text{ with all edges } \leq d$.

Some edge of $P$ passes between two different clusters of $C'$.

Therefore, separation of $C' \leq d$. 
Class So Far

6 lectures:

- Graphs, Trees
- Prim’s Minimum Spanning Tree algorithm
- Heaps
- Heapsort
- 2-approximation for Euclidean traveling salesman problem
- Kruskal’s MST algorithm
- Array-based union-find data structure
- Tree-based union-find data structure
- Minimum-Maximum-Distance clustering
- Python implementation of MST algorithms