Correlation Clustering

Derek Mease

University of Colorado - Boulder

CSCI 5654 Spring 2020

Correlation Clustering [1]

Correlation clustering overview:

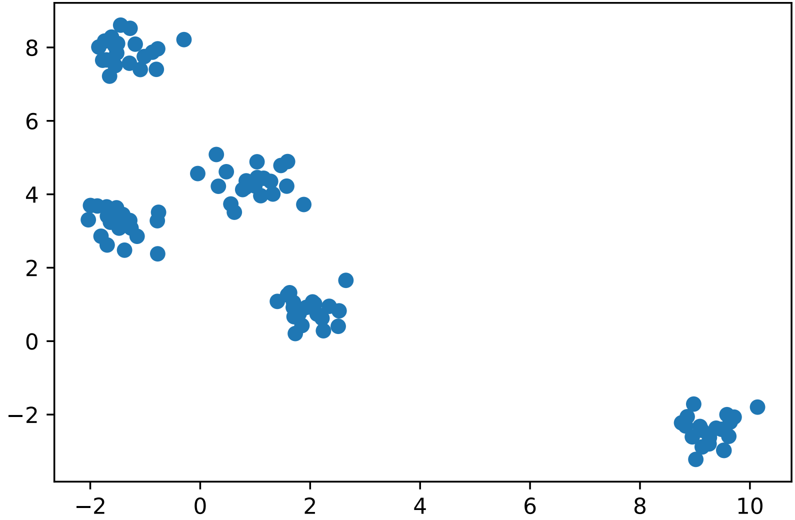
- Sorts data into groups based on their similarity.
- We do *not* need to know the number of groups ahead of time.
- Produces optimal clustering based on a provided similarity matrix.
- NP-Hard

Correlation Clustering [1]

This project will compare various implementations of correlation clustering. Specifically, Integer Programming (IP) implementations will be compared against weighted Max-SAT formulations [2]. State-of-the-art solvers will be utilized (CPLEX, Gurobi, UWrMaxSat). Runtime and memory usage will be analyzed for each algorithm.

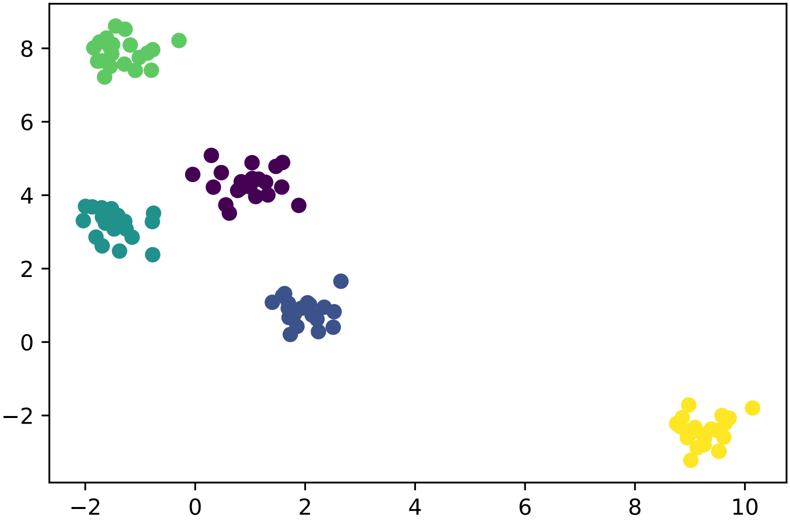
Generate Data

Generate random clusters in 2 dimensions.



K-Means Clustering

For K-Means clustering, we have to specify the number of clusters.



Similarity Matrix

Correlation clustering requires a similarity matrix:

- N x N similarity matrix W, where N is the number of points in the data set.
- w_{ij} is a weight representing the similarity between points v_i and v_j.
- Similar points will have a positive weight.
- Dissimilar points will have a negative weight.
- There are many ways to calculate W.

For this project, we will use a very simple measure of similarity based on Euclidean distance, along with a threshold value. If the distance between points is larger than the threshold, they will receive negative similarity values. Distances smaller than the threshold get positive values.

Integer Linear Programming (ILP) [2]

- Binary variables: $x_{ij} \in \{0, 1\}$ for $1 \leq i < j \leq N$.
- Objective: minimize the weight of dissimilar pairs of co-clustered points.
- Transitivity constraints: $x_{ij} + x_{jk} - x_{ik} \leq 1$.

$$\text{minimize} \sum_{\substack{-\infty < w_{ij} < 0 \\ i<j}} x_{ij}|w_{ij}| - \sum_{\substack{\infty > w_{ij} > 0 \\ i<j}} x_{ij} w_{ij}$$

$$\text{subject to } x_{ij} + x_{jk} - x_{ik} \leq 1 \text{ for all distinct } i, j, k$$

$$x_{ij} \in \{0, 1\} \text{ for all } i, j$$

Complexity

$O(N^2)$ variables and $O(N^3)$ constraints.

Maximum Satisfiability

Maximum Satisfiability (MaxSAT)

Given a formula written in conjunctive normal form, find some assignment of variables that results in the maximum number of clauses evaluating to *true*.

Weighted MaxSat

Each clause is assigned a non-negative weight. We seek to maximize the sum of the weights of satisfied clauses.

Partial Weighted MaxSAT

Same as weighted MaxSAT, except clauses are partitioned into *hard* and *soft* clauses. A solution must satisfy *all* of the hard clauses. We seek to maximize the sum of the weights of all satisfied soft clauses [2].

Transitive Encoding

We must encode our clustering problem into conjuctive normal form. The *transitive* encoding [2] has hard clauses analogous to the ILP transitive constraints. The soft cluases are analogous to the ILP objective function.

Hard Clauses:

- $(\neg x_{ij} \vee \neg x_{jk} \vee x_{ik})$ for all $(v_i, v_j, v_k) \in V^3$ where i, j, k are distinct

Soft Clauses:

- (x_{ij}) for all similar v_i, v_j s.t. $i < j$
- $(\neg x_{ij})$ for all dissimilar v_i, v_j s.t. $i < j$

Soft Clause Weight:

- w_{ij} for all similar v_i, v_j s.t. $i < j$
- $|w_{ij}|$ for all dissimilar v_i, v_j s.t. $i < j$

Complexity

$O(N^2)$ variables and $O(N^3)$ clauses.

Unary Encoding

We can improve the efficiency of the algorithm by setting an upper limit on the number of clusters. We define this maximum as $K \leq N$. With this, we can formulate the *unary* encoding [2]. We use $N \cdot K$ variables y_i^k. We encode logic to ensure that $\sum_{k=1}^{K} y_i^k = 1$ for all $i = 1, \ldots, N$. If $y_i^k = 1$ then v_i is assigned to cluster k. The encoding makes use of a few additional auxiliary variables. For details, refer to the referenced paper.

Complexity

$O(E \cdot K + N \cdot K)$ variables and $O(E \cdot K)$ clauses where E is the number of nonzero values in W.

Binary Encoding

The *binary* encoding [2] is even more compact than the unary. We choose a maximum number of clusters $K = 2^a \leq N$. We define variables b_i^k for $i = 1, \ldots, N$ and $k = 1, \ldots, a$. These variables can be interpreted as the bits in a binary integer. Combining the bits $b_i^a, \ldots, b_i^1$ gives a binary value for the cluster number to which we will assign v_i. Like the unary encoding, the binary encoding employs several additional auxiliary variables. See the referenced paper for a detailed description.

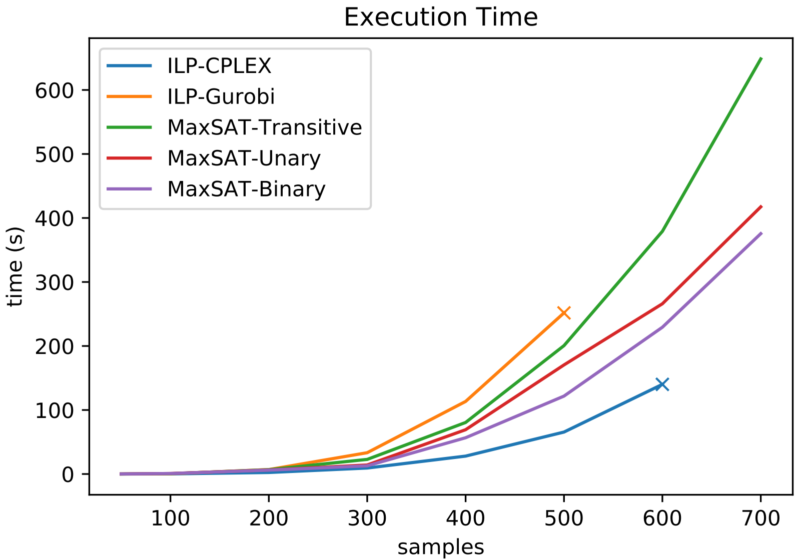
Complexity

$O(E + N \cdot log_2 K)$ variables and $O(E \cdot log_2 K)$ clauses where E is the number of nonzero values in W.

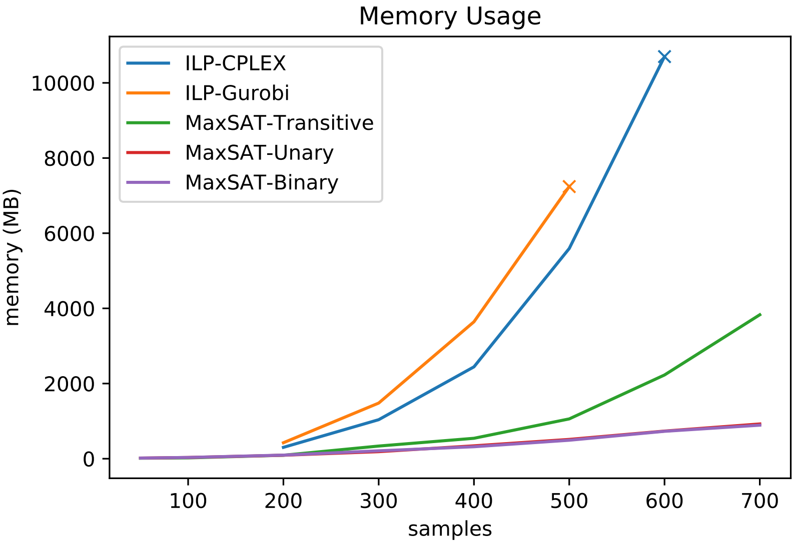
Experiments

- We generate data sets of sizes 50 to 700.
- The ILP formulation was tested using CPLEX and Gurobi.
- Each of the three MaxSAT encodings were tested using UWrMaxSat.
- Execution time and memory usage were measured for each test run.

Results - Execution Time



Results - Memory Usage



Analysis

- MaxSat algorithms utilize far less memory than ILP.
- Limiting the maximum number of clusters using the unary or binary MaxSat encodings significantly reduces memory usage.
- While ILP-CPLEX was faster for smaller data sets, memory consumption becomes prohibitive for larger data sets.
- The reduced memory consumption of MaxSAT algorithms allows us to solve problems with larger data sets without exhausting resources.

References

[1] Nikhil Bansal, Avrim Blum, and Shuchi Chawla. "Correlation Clustering". en. In: *Mach. Learn.* 56.1-3 (June 2004), pp. 89–113. ISSN: 0885-6125. DOI: 10.1023/B:MACH.0000033116.57574.95.

[2] Jeremias Berg and Matti Järvisalo. "Cost-optimal constrained correlation clustering via weighted partial Maximum Satisfiability". en. In: *Artificial Intelligence* 244 (Mar. 2017), pp. 110–142. ISSN: 00043702. DOI: 10.1016/j.artint.2015.07.001.

Code

Code and other resources:
https://github.com/mease/correlation_clustering