Assignment 2

CSC2512—Winter 2020

Out: March 9th, 2020
Due: April 3rd (email me your write-up)

In this assignment you will be modeling the correlation clustering problem in MaxSat and using a MaxSat solver to find solutions.

1 Correlation Clustering

Correlation clustering considers grouping a set of data points into some number of clusters, so that related points are placed in the same cluster. However, if x is related to y and y is related to z but x and z are not related then there is a tradeoff for which cluster to place y in: with x or with z, or all three together.

Formally, the clustering problem is set up as an optimization problem, where the aim is to find a clustering that minimizes some cost function. One formulation is as follows:

Let there be n datapoints v_1, ..., v_n. Associated with each pair of datapoints $\{v_i, v_j\}$ is a similarity weight w_{ij}. When $w_{ij} < 0$, v_i and v_j are dissimilar and if possible should be placed in different clusters. When $w_{ij} > 0$, v_i and v_j are similar and if possible should be placed in the same cluster. Finally, when $w_{ij} = 0$, v_i and v_j are neither similar nor dissimilar and we are indifferent about placing them in the same or different clusters.

Let P^+ be the set of datapoint pairs that have negative weight, and P^- be the set of datapoint pairs that have positive weight. The aim is to assign each data point to a cluster so as to minimize the following cost function.

$$\sum_{(v_i,v_j)\in P^+ \text{ and } v_i,\, v_j \text{ are in different clusters}} w_{ij} \quad + \sum_{(v_i,v_j)\in P^- \text{ and } v_i,\, v_j \text{ are in the same cluster}} -w_{ij}$$

In other words we incur a cost of w_{ij} for every similar pair that are placed in different clusters, and $-w_{ij}$ for every dissimilar pair ($w_{ij} < 0$) that are placed in the same cluster.

The solution of the correlation clustering problem is an assignment of data-points to clusters such that this cost function is minimized.

2 Your Assignment

1. Write a script that given a set of datapoints and collection of non-zero integer[1] weights for pairs of datapoints (negative or positive), produces a MaxSat instance that encodes the problem so that the MaxSat solution solves the clustering problem.

[1] Most MaxSat solvers are limited to integer weights.

Note that various encodings have been experimented with in the literature, but you will learn more if you try develop your own (odds are it will be one of those already developed in the literature).

2. Create a small number of problem instances of different sizes, use your script to convert them to MaxSat instances, choose a MaxSat solver, and test how well your encoding scales.

3. In a short write (max 4 pages excluding citations) up describe your encoding, your experimental setup, and your experimental results. Draw some conclusions on the success of your encoding.

4. Email me your write up.

3 Other information

The input format of most MaxSat solvers is similar to the DIMACS sat encoding with some modifications to encode the clause weights. See `https://maxsat-evaluations.github.io/2019/rules.html#input` for a description of the format.

There are a number of different MaxSat solvers that you can use. For example, any of the **complete** solvers entered into the 2019 MaxSat evaluation. Source code for these solvers is located at `https://maxsat-evaluations.github.io/2019/descriptions.html`.

If you want to use MaxHS you will require access to IBM's CPLEX, see the build instructions at `https://github.com/fbacchus/MaxHS`. If you don't have easy access CPLEX, I can provide a static executable (Linux or Mac) to anyone in the class (just e-mail me).

A number of the other solvers are also quite sufficient for the assignment, and most of them can be built more easily than MaxHS.