DS-GA 3001: Homework Problem Set 2

Special Topics in Data Science: (Spring 2018)

MATHEMATICS OF DATA SCIENCE: Graphs and Networks

Afonso S. Bandeira
bandeira@cims.nyu.edu
http://www.cims.nyu.edu/~bandeira

Shuyang Ling
sling@cims.nyu.edu
http://www.cims.nyu.edu/~sling

Due on February 27, 2018

This homework problem set is due on February 27 at 5pm sent by email to the graders (lv800@nyu.edu and cl3898@nyu.edu)

Please take a look at the homework as soon as possible as there may be things that are not clear, please ask us if you have questions (Piazza preferred).

This homework is a **coding challenge**. All the questions consist of giving solutions (or upper bounds) for optimization problems on graphs. There are 6 graphs of various sizes $(5, 20, 50, 50, 500, 500)$, they are available at https://piazza.com/nyu/spring2018/dsga3001/resources. They are encoded by their A adjacency matrices, meaning $A_{ij} = 1$ if $(i, j) \in E$ is an edge of G and 0 otherwise. Solve each of the problems below for all 6 graphs. You are welcome to **use any programming language**, but the output should be in a csv file, similarly to the input, more on that below. **Hope you have fun solving this homework!**

You are allowed to use optimization solvers such as cvx to solve intermediate linear programs or semidefinite programs (etc), but please do not use available packages to solve exactly the problem that is being asked. The whole goal is to learn from the process.

You can discuss with others about ideas algorithms but not which values they give (or what value you are submitting). Each student must write his/her own code based on his/her own algorithm.

Late submissions will be graded with a penalty of 10% per day late.

Description of the general problem and submission procedure.

For each of the problems below $G = (V, E)$ will correspond to the graph (one of the 6 graphs given), you should solve every problem for every graph. We formulate every problem as a maximization problem $\max_{x \in \text{set}} f(x)$ where x corresponds to a subset of vertices for consistency. For each of the problems and each of the graphs you are asked to do two things:

1. Give a solution for it that has as large of an objective value as possible. Send a `csv` file with the solution (x), not the value it has. **For all problems x is enconded as a subset of the vertices, please encode this is a vector with the length equal to the total number of vertices and taking the value 1 on vertices in the set, and the value 0 on vertices not in the set.**. Then, in the handwritten or pdf submission, write the objective value your solution has $f(x)$ and a brief description of your algorithm. The highest objective function solution will have the best grade.
2. Give an upper bound U on the maximum of the optimization problem (such that $\max_{x \in \text{set}} f(x) \leq U$. This should be a value of the objective function that you are 100% that no one will be able to come up with a solution that does best that this upper bound. Notice that you do not need to give a solution that achievs this, in fact it may be that no solution exists whose value is U. The value U should be submitted on the handwritten or pdf submission. There should also be a description of the algorithm that computed U and a reasoning for why it is an upper bound. The lowest upper bound will have the best grade. **Important**: Given the nature of the upper bound, if there is a submission of an upper bound U for which there is another submission of a solution x such that $f(x) > U$ then the submission of U receives negative points equal to three times the value of the question. **So be sure of your upper bound!** **Name each file as `problemXgraphY.csv`, for example** `problem2graph5.csv`

You should send only one handwritten or pdf submission. For the submission of the csv files there should be one file per problem per graph (so a total of 30 files)

Problems

Let $n = |V|$, we think of $x \in \{0, 1\} \subset \mathbb{R}^n$ and $S = \text{supp}(x)$ so that $S \subset V$. We formulate some of the problems in terms of S. Also A_{ij} is the adjacency of G, meaning $A_{ij} = 1$ if (i, j) is an edge and 0 otherwise. Recall that S^c

corresponds to the set of vertices in V not in S, i.e.: $S^c = V \setminus S$.

$$cut(S) = \sum_{i \in S, j \in S^c} A_{ij},$$

$$edg(S) = \frac{1}{2} \sum_{i \in S, j \in S} A_{ij},$$

and

$$vol(S) = \sum_{i \in S, j \in V} A_{ij}.$$

Note that $vol(S) = 2edg(S) + cut(S)$.

Problem 1.1 (Maximum Cut)

$$\max_{\{S \subset V\}} cut(S)$$

Notice the different encoding of $x \in \{0,1\}$ instead of $x \in \{-1,1\}$ as in class.

Problem 1.2 (Conduunctance)

$$\max_{\{S \subset V:\ vol(S) \leq vol(S^c)\}} \frac{vol(S)}{cut(S)}$$

Problem 1.3 (Maximum Clique)

$$\max_{\{S \subset V: 2edg(S) = |S|^2 - |S|\}} |S|$$

Notice the requirement simply asks that every pair of nodes in S has an edge.

Problem 1.4 (Small cluster)

$$\max_{\{S \subset V: |S| = |V|/5\}} \frac{1}{cut(S)}$$

Problem 1.5 (Dense graph)

$$\max_{\{S \subset V: 2edg(S) \geq (|S|^2 - |S|)/2\}} |S|$$