Exercise Set XII, Advanced Algorithms 2022

These exercises are for your own benefit. Feel free to collaborate and share your answers with other students. Solve as many problems as you can and ask for help if you get stuck for too long. Problems marked * are more difficult but also more fun :).

These problems are taken from various sources at EPFL and on the Internet, too numerous to cite individually. **The problems are not ordered with respect to difficulty.**

1 Let $G = (V, E)$ be a d-regular undirected graph, and let $M = \frac{1}{d}A$ denote its normalized adjacency matrix. Let $1 = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ denote the eigenvalues of M. Prove that if $\lambda_{n-1} = -1$, then G is disconnected.

Solution: See solutions to Problem 4a on Final 2020 on moodle.

2 Let $G = (V, E)$ be a d-regular undirected graph, and let $M = \frac{1}{d}A$ denote its normalized adjacency matrix. Let $1 = \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$ denote the eigenvalues of M. Prove that if $\lambda_n > -1/2$, then G is not tripartite (recall that a graph is tripartite if its vertex set can be partitioned into three disjoint sets such that all of its edges connect vertices in different components).

Solution: See solutions to Problem 4b on Final 2020 on moodle.

3 Consider a family $\mathcal{H}$ of hash functions $h : [n] \to [n]$. We want to analyze the quantity

$$\text{maxload(h)} := \max_{b \in [n]} |\{h(i) = b\}|$$

when $n \geq 10$.

a Assuming $\mathcal{H}$ is a pairwise independent family of hash function, prove that

$$\Pr_{h \in \mathcal{H}}[\text{maxload(h)} > 11\sqrt{n}] \leq 1/100\,.$$

b Assuming that $\mathcal{H}$ is the family of completely random hash functions, prove that

$$\Pr_{h \in \mathcal{H}}[\text{maxload(h)} > 9\ln(n)] \leq 1/2\,.$$

Solution: Proof of bound when $\mathcal{H}$ is pairwise independent

- We analyze a fixed "bucket b" we then we use the union bound.
- Let X be the random variable $|\{h(i) = b\}|$ and X_i the indicator random variable that $h(i) = b$. We have

$$\mu := \mathbb{E}_{h\in\mathcal{H}} X = \mathbb{E}_{h\in\mathcal{H}} \sum_{i\in[n]} X_i = \sum_{i\in[n]} \mathbb{E}_{h\in\mathcal{H}} X_i = \sum_{i\in[n]} \frac{1}{n} = 1$$

- We now wish to bound $\Pr[X > 11\sqrt{n}]$. By Chebyechev's inequality, we have

$$\Pr[X > 11\sqrt{n}] \leq \Pr[|X - \mu| > 10\sqrt{n}] \leq \frac{\mathrm{Var}[X]}{100n}$$

- So let's analyze $\mathrm{Var}[X]$:

$$\begin{aligned}
\mathrm{Var}[X] &= \mathbb{E}[X^2] - (\mathbb{E}[X])^2 \\
&= \sum_{i,j} (\mathbb{E}[X_i X_j] - \mathbb{E}[X_i]\mathbb{E}[X_j]) \\
&= \sum_{i} \left(\mathbb{E}[X_i^2] - (\mathbb{E}[X_i])^2\right) \qquad \text{(by pairwise independence)} \\
&= \sum_{i} \left(\frac{1}{n} - \frac{1}{n^2}\right) \leq 1
\end{aligned}$$

- Hence, $\Pr[X > 11\sqrt{n}] \leq \frac{1}{100n}$. Now by union bound over the n buckets, we get

$$\Pr_{h\in\mathcal{H}}[\mathrm{maxload(h)} > 11\sqrt{n}] \leq 1/100\,.$$

Proof of bound when $\mathcal{H}$ is completely random

- Very similar proof as in previous case and we use the same notation. The only difference is that we know wish to prove that

$$\Pr[X > 9\ln(n)] \leq \frac{1}{100n}$$

where $X = (X_1 + X_2 + \ldots + X_n)$ is now the sum of n *independepent* random variables taken values in $\{0, 1\}$.

- Hence we apply a Chernoff bound to obtain

$$\Pr[X > 9\ln(n)] < \Pr[X > (1 + 8\ln(n)) \cdot \mu] \leq e^{-\frac{8\ln(n)}{3}} \leq \frac{1}{2},$$

for $n \geq 10$.

4 Consider an undirected graph $G = (V, E)$. Show that the following linear program, that has a variable x_v for each $v \in V$, can be solved in polynomial time

$$\begin{aligned} \textbf{maximize} \quad & \sum_{v \in V} x_v \\ \textbf{subject to} \quad & \sum_{v \in S} x_v \leq |\delta(S)| \qquad \text{for every } S \subsetneq V \\ & x_v \geq 0 \end{aligned}$$

Recall that $\delta(S)$ is the set of edges crossing the cut S.

By Ellipsoid method this reduces to the following separation problem: Given x, design a polytime algorithm that verifies whether x is feasible or, if not, outputs a violated constraint.

Solution:

- To verify that x is non-negative is trivial to verify in polynomial time and if not simply output a violated inequality $x_v \geq 0$.
- It remains to deal with the exponential set of inequalities

$$\sum_{v \in S} x_v \leq |\delta(S)| \qquad \text{for every } S \subsetneq V.$$

- We can rewrite the above

$$0 \leq f(S) \qquad \text{for every } S \subsetneq V,$$

  where $f(S) = |\delta(S)| - \sum_{v \in S} x_v$.
- The key insight is that f is a submodular set function:
  - $|\delta(S)|$ is the cut function (as seen in class)
  - $-\sum_{v \in S} x_v$ is a modular function and thus trivially submodular (the values are independent)
  - The sum of two submodular functions are submodular as seen in class
- We can thus minimize f in polynomial time (since we can evaluate $f(S)$ in polynomial time). If the minimizer is ≥ 0 then x is feasible. Otherwise the minimizer S corresponds to a violated constraint if $S \neq V$.
- To deal with that we don't allow $S = V$ we can do n submodular function minimizations over the submodular functions $g_1, g_2, \ldots, g_n$ where the i-th function g_i is defined over the ground set $V \setminus \{v_i\}$ and $g(S) = f(S)$ for every $S \subseteq V \setminus \{v_i\}$. Here, we name the vertices so that $V = \{v_1, \ldots, v_n\}$.