CS 231 Quantum Computation and Quantum Complexity *Instructor: Anurag Anshu*

Homework 0

Due: Thu, January 27, 2022 *TFs: Chi-Ning Chou, Prayaag Venkat*

Introduction: The questions below aim to help examine your prerequisite knowledge for CS 231. These are basic questions in the following topics: linear algebra, constraint satisfaction problems, probability theory & stochastic processes and randomized algorithms. Note that expertise in these topic is *not needed*; just a level of comfort when we start translating some fundamental ideas into the quantum world.

This homework carries no score. Please provide your answer in the space below, and add your comments (if any).

If more than 3 questions are found to be very unfamiliar, we recommend that you talk to us soon. While this course will encourage collaborations, we advise against it for Homework 0 (as the goal is to test *your* prior knowledge).

Notational background: A recurring set in the exercises is '$\mathcal{X}$'. Its any of your favourite sets and we have not specified what its elements actually are. The notation × is used to denote the Cartesian product of two sets. A graph is denoted as $G = (V, E)$, where V is the set of vertices and E is the set of edges. Probability distributions over a set (or a universe) are denoted by capital letters P and Q. With due apology, matrices are also represented by capital letters, such as M. This is for consistency with the upcoming quantum notation and the identity will be clear from context.

1. **Circuit to SAT:** The classic Cook-Levin theorem shows that Circuit-SAT can be reduced to 3SAT. If we apply this reduction to one of the simplest circuits out there: the 'OR' gate, we obtain the following 3SAT instance.

   (a) $(x_1 \vee x_2 \vee \neg x_3) \wedge (x_3 \vee \neg x_1) \wedge (x_3 \vee \neg x_2)$.

   (b) $(x_1 \vee x_2)$.

   (c) $(x_1 \wedge x_2 \wedge x_3)$.

   (d) $(x_1 \wedge x_2 \wedge \neg x_3)$.

   Answer: (a)

   Comments: The classic reduction formula for circuits to 3SAT would give (a), which is satisfied by values to x_1, x_2 which also give $x_1 \vee x_2 = 1$.

2. **Magic square constraint satisfaction:** Consider the constraint satisfaction problem in Figure 1, depicted as a magic square.

   The largest fraction of constraints satisfiable by assignments to (classical) variables x_{ij} is

   (a) 1.

   (b) $\frac{5}{6}$.

   (c) $\frac{2}{3}$.

   (d) $\frac{1}{2}$.

x_{31}	x_{32}	x_{33}
x_{21}	x_{22}	x_{23}
x_{11}	x_{12}	x_{13}

Figure 1: Each variable x_{ij} is in $\{1, -1\}$. There are 6 constraint: across any row, the total parity should be 1 (that is $x_{i1}x_{i2}x_{i3} = 1$ for all $i \in \{1, 2, 3\}$) and across any column the total parity should be -1 (that is $x_{1i}x_{2i}x_{3i} = -1$ for all $i \in \{1, 2, 3\}$).

Answer: (b)

Comments: We can easily find a case that satisfies 5 rows/columns, but satisfying all of them would require all the elements to multiply to both 1 and -1

3. **Searching:** Consider a database of N items, such that one unknown item is marked and the remaining items are unmarked. There is a randomized algorithm that can find the marked item with probability at least $\frac{1}{2}$ by looking at no more than ___ items. However, any randomized algorithm that can find a marked item with probability at least $\frac{1}{2}$ must look at least at ___ items.

   (a) $N \ln 2$ and $N/2$.

   (b) N and $1 + N \ln 2$.

   (c) $\sqrt{N}$ and $\sqrt{\pi N/4}$.

   (d) $N/10$ and $N^{0.99}$.

   Answer: (a)

   Comments: Since there are N elements and only one is marked, we must sample at least $N/2$ elements on average no matter how good our algorithm is.

4. **Boosting:** A (randomized) algorithm $\mathcal{A}$ takes as input a sample r from a distribution R and an input $x \in \mathcal{X}$, to output $\mathcal{A}(x, r) \in \{0, 1\}$. It computes a given function $f : \mathcal{X} \to \{0, 1\}$ in the following sense: $\text{Prob}_{r \sim R}[\mathcal{A}(x, r) = f(x)] \geq \frac{2}{3}$ for all $x \in \mathcal{X}$. Let us say that $\frac{2}{3}$ is the *success probability* of $\mathcal{A}$. It is possible to boost this success probability to $1 - \varepsilon$ (for any $\varepsilon \in (0, 1)$), by running $\mathcal{A}$ many times and taking the majority of the outputs. The number of times $\mathcal{A}$ is run is best written as:

   (a) $\Omega\left(\log \frac{1}{\varepsilon}\right)$.

   (b) $\Omega\left(\sqrt{\log \frac{1}{\varepsilon}}\right)$.

   (c) $\Omega\left(\log \log \frac{1}{\varepsilon}\right)$.

   (d) This scheme will not work in general.

   Answer: (a)

   Comments: We can find this using the CLT on the sum of $A(x, r_i)$ for all samples r_i.

5. **Boosting again:** Now, suppose the randomized algorithm $\mathcal{A}$ computes a given function $f : \mathcal{X} \to \{0,1\}$ in the following sense: $\text{Prob}_{x \sim U, r \sim R}[\mathcal{A}(x,r) = f(x)] \geq \frac{2}{3}$, where U is the uniform distribution over $\mathcal{X}$. It is still possible to boost this success probability (now averaged according to U) to $1 - \varepsilon$, by running $\mathcal{A}$ as many times as before and taking the majority of the outputs.

   (a) I totally agree.

   (b) Yes, but it suffices to run $\mathcal{A}$ way less times.

   (c) Yes, but in some cases $\mathcal{A}$ needs to be run many more times.

   (d) This scheme will not work on some algorithms.

   Answer: (c)

   Comments: This is the same as 4 but we have to end up running it for each $x \in \mathcal{X}$, which takes a lot more time.

6. **Correlation:** Given two n-bit strings x, y, our goal is to design a randomized algorithm $\mathcal{A}$ that outputs a good estimate to $\frac{1}{n}\sum_i x_i y_i$ (the correlation between the strings). More precisely, given an error parameter ε, the algorithm must output a real number $r(x,y)$ such that with probability at least $\frac{2}{3}$, $|r(x,y) - \frac{1}{n}\sum_i x_i y_i| \leq \varepsilon$ (for all x, y). However, an unreasonable constraint is imposed upon the algorithm: it can only see one bit of the inputs x, y at one time. Which of the following best captures the number of bits of input $\mathcal{A}$ needs to see, as well as the tightest lower bound.

   (a) $\Omega\left(\log \frac{1}{\varepsilon}\right)$.

   (b) $\Omega\left(\frac{1}{\varepsilon^2}\right)$.

   (c) $\Omega\left(2^{\frac{1}{\varepsilon}}\right)$.

   (d) None of the above.

   Answer: (b)

   Comments: (a) grows too slow and (c) grows too fast.

7. **Random walks:** A popular algorithmic framework is the random walk, which helps us sample from complicated distributions by taking some elementary steps. Given a connected and undirected graph $G = (V, E)$, a random walk inputs a distribution over V and outputs another distribution over V. More precisely, it is a stochastic matrix $\{M_{x,y}\}_{x,y \in V}$ such that $M_{x,y}$ is non-zero iff $(x,y) \in E$ or $y = x$. The entry $M_{x,y}$ can be interpreted as the probability to reach y starting from x. Note that there is some non-zero probability for self-transition on each x. Choose all the correct options:

   (a) Starting from any distribution, the random walk will converge to the uniform distribution over V.

   (b) Starting from any distribution, the random walk will converge to a non-uniform distribution in which the vertices of higher degree have higher probability.

   (c) The rate of convergence to the stationary distribution is inversely related to the spectral gap[1] of the stochastic matrix M.

[1] The difference between the absolute values of the largest eigenvalue and the second largest eigenvalue.

(d) The rate of convergence to the stationary distribution is proportionally related to the spectral gap of the stochastic matrix M.

Answer: (b), (d)

Comments:

The random walk will converge on a stationary state with higher probability in vertices with many edges with high weights, since more probability will move towards those points every step. Also, the spectral gap determines how fast eigenvector components of the state with eigenvalues less than 1 will decay, so a larger gap will lead to higher decay of those components and faster convergence.

8. **Classical no-cloning:** Continuing the topic of stochastic processes, let's try to design certain stochastic matrix $M_{x,(y,z)}$, where $x \in \mathcal{X}$ and $(y,z) \in \mathcal{X} \times \mathcal{X}$. The goal is to specify the entries of M such that for any probability distribution Q (viewed as a row matrix) over $\mathcal{X}$, $Q \cdot M$ is equal to the distribution $Q \otimes Q$ over $\mathcal{X} \times \mathcal{X}$. Here, the notation $\otimes$ refers to the Kronecker product. Choose a correct option:

   (a) It is possible to find such a stochastic matrix M.

   (b) Such a stochastic matrix can't exist since things should be linear.

   (c) Such a stochastic matrix can't exist since the total variational distance between two distributions Q and Q' is smaller than the total variational distance[2] between $Q \times Q$ and $Q' \times Q'$.

   (d) Such a stochastic matrix can't exist, but for very different reasons not listed above.

   Answer: (b)

   Comments: This operation isn't linear because $(Q + Q') \cdot M \neq Q \otimes Q + Q' \otimes Q'$.

9. **Matrices:** Selection all correct statements about matrices (with entries as complex numbers):

   (a) A square matrix M is diagonalizable if there is another square matrix N and a diagonal matrix D such that $M = NDN^{-1}$.

   (b) Every square matrix has singular-value decomposition, however not every square matrix has an eigen-value decomposition.

   (c) A unitary matrix may not be hermitian, hence it may not have eigenvalues.

   (d) Singular values of a square matrix are always non-negative reals, but eigenvalues of a hermitian matrix may be complex numbers.

   Answer: (a), (b)

   Comments:

[2] Total variational distance measures how different two probability distributions are. Given two distributions P_1, P_2 over a set $\mathcal{X}$, total variational distance is defined as $\frac{1}{2} \sum_{x \in \mathcal{X}} |P_1(x) - P_2(x)|$.