Midterm

Name:

SUID Number:

[This is a closed-notes/closed-computer exam, though you may refer to 1 page (or 2 sides) of 8.5 x 11 notes that you have prepared. You must work alone on the exam.]

The following theorems might be helpful:

**Theorem 1** (Lovasz Local Lemma). Consider a set of 0/1 random variables, $A_1, \ldots, A_n$ defined over some probability space such that the maximum degree of the corresponding dependency graph is $d$. If, for all $i$, $\Pr[A_i = 1] \leq \frac{1}{e(d+1)}$, then $\Pr[0 = A_1 = A_2 = \ldots = A_n] \geq (1 - \frac{1}{e(d+1)})^n > 0$. (Namely, there is a positive probability that all the events $A_i$ are, simultaneously, avoided.)

**Theorem 2** (Chernoff Bounds). Let $X$ be the sum of independent random variables that take values in $[0, 1]$:

- If $E[X] \leq \mu$, then for $\delta \in (0, 1)$: $\Pr[X \geq (1 + \delta)\mu] \leq e^{-\delta^2 \mu/3}$.
- If $E[X] \geq \mu$, then for $\delta \in (0, 1)$: $\Pr[X \leq (1 - \delta)\mu] \leq e^{-\delta^2 \mu/2}$.

<table>
<thead>
<tr>
<th>Problem</th>
<th>#1</th>
<th>#2</th>
<th>#3</th>
<th>#4</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Score</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum</td>
<td>36</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>62</td>
</tr>
</tbody>
</table>
1. Short-answer questions: (3 points each)

(a) [3 points] Recall that a Monte Carlo randomized algorithm has a bounded runtime but has some probability of outputting an incorrect answer, whereas a Las Vegas randomized algorithm is always correct, but has a runtime that might be unbounded but has bounded expectation. Given a Las Vegas algorithm Alg that solves problem $X$ in expected time $T$, describe a Monte Carlo algorithm for problem $X$ that runs in time at most $10T$ and outputs the correct answer with probability at least 0.9. Prove the correctness of your answer in 1 sentence. [Hint: feel free to call Alg as a subroutine.]

Consider the algorithm that runs Alg until either Alg outputs an answer, or has run for $10T$ steps, whichever happens first; if Alg did output an answer, then we output that value, otherwise we output 'FAIL'. The probably we are correct is the probability that we do not output 'FAIL' (since Alg is always correct), which is the probability that Alg takes more than $10T$ steps to output an answer. By Markov’s inequality, this is bounded by $1/10$, since the expected runtime of Alg is $T$.

(b) [3 points] (The Midterm Paradox) Suppose that every class chooses the date of its midterm uniformly at random from the 14 days in the last week of October/first week of November. Suppose you are taking $C$ classes. What is the expected number of pairs of your classes that will have identical midterm dates? Roughly what does $C$ need to be for you to expect at least one conflict? [Give the expression for the expected number of conflicts, a rough value of $C$ for the expectation to be $\geq 1$, and a one sentence explanation/proof.]

There are $C(C - 1)/2$ pairs of classes that could conflict, and the probability that a given pair conflicts is $1/14$. Hence by linearity of expectation, the expected number of conflicting pairs is $C(C - 1)/28$, which is greater than 1 if $C \geq 6$.

(c) [3 points] Describe an algorithm which takes as input an integer $n$, and with probability at least 0.99999 will return a uniformly random prime number between 1 and $n$ (and with the remaining probability is allowed to output anything). The algorithm you describe must have expected runtime bounded by some polynomial in $\log n$, for example, at most $O((\log n)^5)$. Clearly describe the algorithm, and feel free to refer to any of the algorithms that we’ve covered in class. Provide one or two sentences explaining why the algorithm works, and why it has a polylog($n$) runtime. [You do not need to worry about the exact runtime, as long as it is polynomial in $\log n$.]

Consider the algorithm that repeats the following two steps until a prime number is found: choose a uniformly random number between 1 and $n$, checks whether it is prime (using, for example, the randomized Rabin-Miller primality test with failure parameter $10^{-10}/\log(n)$) and outputs the number if it is prime. Since a $\theta(1/\log n)$ fraction of the number between 1 and $n$ are prime, in expectation we will need to repeat the two
steps only $O(\log n)$ times, and the primality check takes time polylogarithmic in $n$, so the whole runtime is polylog $n$.

(d) [3 points] Prove that for any real-valued random variable $X$ and any number $a$, $\Pr[|X - E[X]| > a] < \frac{E[(X - E[X])^4]}{a^4}$. [Hint: recall the proof of Chebyshev’s inequality that bounds $\Pr[|X - E[X]| > a]$ by the ratio of the variance of $X$ to $a^2$.]

$$\Pr[|X - E[X]| > a] = \Pr[(X - E[X])^4 > a^4], \text{ and hence applying Markov’s inequality to the random variable } (X - E[X])^4 \text{ yields the desired claim.}$$

(e) [3 points] Suppose you want to estimate the probability of a rare event $A$ happening (say you want to test the failure probability of your algorithm). How many trials would it take for you to certify that $\Pr(A) \leq p$, with large constant probability, say, confidence at least 0.99? Please give your answer as a function of $p$, and feel free to ignore constant factors. Justify your answer in a few sentences.

If the event has a true probability that is significantly less than $p$, then $O(1/p)$ samples will suffice to certify that the probability is less than $p$—for example, if we take $10/p$ trials, and don’t observe a single bad outcome, we can conclude that the probability is less than $p$, because if the probability was more than $p$, the probability of having seen no bad outcomes in $10/p$ would be less than $(1 - p)^{10/p} \approx e^{-10}$. Of course, to estimate the probability of the event to error $\epsilon$, if the true probability is constant, and much larger than $\epsilon$, we would need $O(1/\epsilon^2)$ samples.

(f) [3 points] Suppose we toss $n$ balls into $n$ bins uniformly at random, and we want to argue that the number of bins with exactly one ball is very tightly concentrated about its expectation (i.e. tails that decay inverse exponentially). Explain why we cannot immediately use a standard Chernoff bound, and then give a 2-4 sentence high-level description of a good approach to proving this.

We cannot immediately use a Chernoff bound, since the indicator variables of whether or not each bin has exactly one ball are not independent. One way to prove this is to first prove the claim in the “Poissonized” setting, where the bin loads are now independent (and hence we can apply a Chernoff bound), and then note that a Poisson random variable of expectation $n$ is close to $n$ with high probability, and this $o(n)$ discrepancy in the number of balls cannot effect the number of bins with exactly 1 ball by more than $o(n)$.

(g) [3 points] Word embeddings are a mapping from words to vectors (typically $\approx 500$-dimensional vectors), such that the distances between a pair of vectors roughly coincides with some notion of their “semantic distance”. In 1-2 sentences, explain why we might
not be so surprised that this is possible. [You might want to make a specific mention of Bourgain’s metric embedding result—its alright if you don’t remember the specific...]

The point of Bourgain’s metric embedding result is that any metric can be represented in Euclidean space, up to some relatively modest distortion (at most $O(\log n)$ for any metric on $n$ points/words). Hence we should not be too surprised that the semantic distances can be represented/approximated via the Euclidean distances between some carefully chosen vectors.

(h) [3 points] Suppose an applied machine learning researcher comes to you and says the following: “Since there are over 100,000 words, do you think we could find a much better embedding if we map words to 10,000 or 100,000 dimensional vectors instead of just 500 dimensional vectors?” In 2-4 sentences, explain one reason why you might not expect there to be much benefit in considering such high dimensional embeddings. (For full credit, you must specifically reference material from class—don’t just say ”because we don’t have enough data to train it” or something like that...)

The Johnson-Lindenstrauss metric embedding shows that for any embedding of $n$ points in $\mathbb{R}^d$ (under Euclidean distance), they can be embedded into only $O(\log n/\epsilon^2)$ dimensions without changing the pairwise distances by too much. If we are considering a set of 100,000 words, then maybe we need $\log(100000)$ times some constant (maybe 50) dimensions to represent these relationships. In this sense, going to higher dimension will not let us represent any extra distance-structure. Of course, this does not address the question of whether it is easier to directly find a word embedding into a larger space, though one could always apply dimension reduction after such an embedding were found.

(i) [3 points] Prove that the graph metric corresponding to the complete graph on $n$ vertices (i.e. the metric on $n$ points, where the distance between any pair of distinct points is 1) can be embedded into $(\mathbb{R}^{O(\log n)}, \ell_2)$ with distortion at most 2 (i.e. for some constant $c$, there is a mapping from the $n$ points to points in $\mathbb{R}^{c \log n}$ such that for every pair of points, the Euclidean distance between their images differs from their distance in the original metric by at most a factor of 2). Your proof should be 2-3 sentences long.

First note that this graph metric can be trivially embedded into $\mathbb{R}^n$ under Euclidean distance, by mapping each point to one of the $n$ basis vectors, scaled by a factor of $1/\sqrt{2}$. Now recall that the Johnson-Lindenstrauss transformation guarantees that we can map these $n$ vectors to vectors in $\mathbb{R}^{O(\log n)/\epsilon^2}$ such that the Euclidean distances in the reduced space are distorted by at most a factor of $1 + \epsilon$. Taking $\epsilon = 1$ completes the proof.

(j) [3 points] Given a SAT formula over $n$ variables $x_1, \ldots, x_n$, with $a$ clauses that consist
of a single variable [e.g. the clause \((x_i)\), \(b\) clauses with two variables [e.g. the clause \((x_i \lor x_j)\)], and \(c\) clauses with exactly three variables, give an expression for the expected number of clauses that would be satisfied if you assign each variable to be T/F independently with probability 1/2 each. Justify with one sentence of explanation.

The expected number of satisfied clauses would be \(\frac{1}{2}a + \frac{3}{4}b + \frac{7}{8}c\), since the probability that a clause with \(k\) literals is satisfied by a random assignment is \(1 - \frac{1}{2^k}\), and this formula then follows from linearity of expectation.

(k) [3 points] Given the setup of the previous problem, give an informal sketch of a deterministic algorithm that will find an assignment satisfying at least the claimed fraction of clauses, which runs in time polynomial in the number of variables and clauses. [You can still do this part even if you did not do the previous part.]

We proceed by doing a “greedy” algorithm, whereby we iteratively assign \(x_1\) to be T/F, then \(x_2\), then \(x_3\), and so on. For the decision of \(x_i\), we use the formula in the previous part to determine whether \(x_i = T\) or \(x_i = F\) has a higher expected number of satisfied clauses (and assign it to whichever is larger), where the expectation is over the randomness of the assignment to \(x_{i+1}, x_{i+2}, \ldots\).

(l) [3 points] In 2-4 sentences, explain the following statement: For settings in which there are only “local” dependencies, the Lovasz Local Lemma “interpolates” between the pessimistic union bound, and the case where all events are independent.

Given \(d\) events that each occur with probability \(p\), if they have arbitrary dependencies the union bound can be tight, and the probability of simultaneously avoiding all of them can be as small as \(1 - dp\), which is strictly greater than 0 provided \(p > 1/d\). On the other hand, no matter how many events we have, if they are independent (and each occur with probability strictly less than 1), then there is a positive probability of avoiding all of them. The LLL “interpolates” between these two extremes in the sense that, to have a positive probability of avoiding all the events, it suffices to ensure that the union bound holds (with an extra 1/e factor slack) for each of the neighborhoods of the dependency graph.
2. (Thresholds in Random Graphs) On PS4 you proved that in the Erdos-Renyi random graph model $G(n, p)$ over $n$-node graphs where each edge is independently present with probability $p$, if $p > 1/n$ then you expect a large connected component, and if $p < 1/n$ you only expect small connected components. Here, you investigate the threshold phenomena concerning when $G(n, p)$ is (strongly) connected vs disconnected, which occurs at $p = \Theta\left(\frac{\log n}{n}\right)$. [Throughout this problem, feel free to use the fact that $1 - x \approx e^{-x}$ for small values of $x$.]

(a) [2 points] Prove that if $p < \frac{\log n}{2n}$ then the expected number of vertices with degree 0 is at least $\sqrt{n}$.

Let $Z$ denote the random variable representing the number of deg 0 nodes. By Linearity of expectation

$$E[Z] = n \Pr[v \text{ has deg 0}] = n(1-p)^{n-1} \geq n(1-\frac{\log n}{2n})^n \approx ne^{-\log n/2} = n/\sqrt{n} = \sqrt{n}.$$

(b) [2 points] Prove that if $p > 100\frac{\log n}{n}$ then with probability at least $1 - o(1/n)$ the random graph $G(n, p)$ will have no degree 0 nodes. Namely, for every partition of the nodes into sets $S_1, S_2$ with $|S_1| = 1$, there is at least one edge from $S_1$ to $S_2$.

We will do a union bound over the $n$ such partitions. For each such partition, the probability that there is at least one edge crossing the cut is $1 - (1-p)^{n-1} > 1 - e^{-p(n-1)} > 1 - n^{-10}$, where the exponent of $-10$ is a very pessimistic bound. Union bounding over the $n$ partitions yields that the probability that all such partitions have an edge is at least $1 - n^{-9}$.

(c) [2 points] Prove that for any $k \in [1, \ldots, n/2]$, if $p > 100\frac{\log n}{n}$ then with probability at least $1 - o(1/n)$ for every partition of the nodes into sets $S_1, S_2$ with $|S_1| = k$, there is at least one edge from $S_1$ to $S_2$. [Hint: note that $\binom{n}{k} \leq n^k$.]

Same as above, only now we need to union bound over the $\binom{n}{k} \leq n^k$ partitions, and the probability that a given partition has at least one crossing edge is $1 - (1-p)^{k(n-k)} \leq 1 - (1-p)^{nk/2}$, since there are exactly $k(n-k)$ possible edges crossing the cut, and each is present independently with probability $p$. Plugging in the value of $p$, we get that the probability a bad cut exists is at most $1 - n^k e^{-pnk/2} \geq 1 - n^k n^{-10k} = 1 - o(1/n)$.

(d) [2 points] Using the previous part, prove that if $p > 100\frac{\log n}{n}$, then with probability $1 - o(1)$, a random $G(n, p)$ graph will be connected.

We do a union bound over the $n/2$ different possible values of $k = 1, 2, \ldots, n/2$ (where $k$, as above, denotes the size of the offending cut). In the previous part we showed that a bad partition of each size exists with probability $o(1/n)$ hence a union bound yields that the probability any of these bad partitions exist is $o(n \cdot (1/n)) = o(1)$. 


3. Suppose we are investigating the social habits in a group of \( n \) chimpanzees, and after months of observations, for some pairs of chimpanzees \( A \) and \( B \), we know whether \( A \) has spent more time grooming \( B \) or whether \( B \) has spent more time grooming \( A \). We wish to aggregate these results into a single ranking of the “altruism” of each chimpanzee that minimizes the number of “inconsistent pairs”, where a pair \( A, B \) is inconsistent/violated if \( A \) is above \( B \) in the ranking but has spent less time grooming \( B \) than \( B \) spent grooming \( A \).

(a) [5 points] Prove that there exists a ranking that violates at most half the pairwise relationships. [Hint: probabilistic method!]

Consider choosing a uniformly random ranking. By linearity of expectation, in expectation this will violate exactly half the relationships. Since any random variable must attain (or exceed) its expectation with non-zero probability, there must be at least one such ranking that violates at most half the relationships. Alternatively, you could just note that if a ranking satisfies less than half the relationships, by reversing it, you will now satisfy more than half...

(b) [4 points] Prove that for sufficiently large \( n \), there exists a set of grooming habits such that for every ranking, at least 49% of the pairwise relationships would be violated. [Hint: probabilistic method—choose a distribution over the grooming habits, then argue that the probability a random ranking violates significantly less than half the pairwise relationships is so small, that there is a good chance you picked a set of grooming habits with the property that no “good” ranking exists.]

Consider assigning the grooming habits uniformly at random. For any fixed ranking, over the randomness in the grooming habits, we expect the ranking to violate half the relationships—namely \( n(n - 1)/4 \) of them. Since the habits were chosen independently at random, a Chernoff bound applies, and the probability that the actual number of violates habits deviates from this expectation by more than a factor of 0.01 is bounded by \( e^{-cn^2} \), for some constant \( c = 0.012^2/3 \). There are only \( n! < n^n = e^{n \log n} \) rankings, hence, via a union bound over them, the probability that our random grooming habits are more than 51% consistent with any ranking is at most \( e^{n \log n} e^{-cn^2} = o(1) \). (So, not only does such a set of grooming habits exists, but a random set has this property with high probability.)

(c) [bonus 2 points] Prove that there exists a ranking that violates strictly less than half the pairwise relationships.

Consider the greedy algorithm (via “conditional de-randomization” as we referred to it in class) that iteratively builds a ranking by adding the \( i \)th chimpanzee to either the top or bottom of the list containing the first \( i - 1 \) chimpanzees. Note that at each step, the number of \( i \)’s relationships with the \( i - 1 \) earlier chimps that are respected is exactly 50% in expectation. Hence to decide whether to go to the top or the bottom, it suffices to
simply evaluate the number of respected relationships with the $i-1$ earlier chimpanzees and assign greedily. To establish the desired claim, we just need to show that there is one step where the expectation strictly increases (since it starts from equalling exactly half the relationships). When we insert the second chimpanzee, we insert it above the first one or below according to the relationship between them, and the expectation over the remaining ones will increase by exactly $1/2$.

4. An edge coloring of an (undirected) graph $G = (V, E)$ assigns exactly one color to each edge of the graph. We say that a colored path in the graph is symmetric if the path has an even number of edges, and the second half of the path is colored identically to the first half of the path (i.e. the sequence of colors in the second half of the path is the same sequence as in the first half). [Throughout this problem, by “path” we refer only to simple paths—ie paths that do not re-use any edges.]

(a) [5 points] Prove that for any graph whose maximum degree is $d$, there exists a coloring using $10 \cdot d^2$ colors such that there are no “symmetric” paths of length 4 (i.e. no repeating paths consisting of 4 distinct edges). [Hint: use the Lovasz Local Lemma—define a probability distribution over colorings, define a set of “bad events”, argue about the maximum degree of the dependency graph corresponding to your events, etc.]

Consider the uniformly random distribution over coloring. Let bad event $A_i$ denote the event that the $i$th path of length 4 in the graph is symmetric. Consider the dependency graph where any paths that share a vertex have an edge between their corresponding events. This is clearly a valid dependency graph, as the non-neighbors of $A_i$ involve a disjoint set of vertices as path $i$, and hence the probability the path is symmetric is exactly $1/(10d^2)^2$ (since the second two vertices must be colored according to the symmetric pattern of the first two), and is independent of whatever happened in the rest of the graph. The degree of the dependency graph is at most $4d(2d)(2d) = 16d^3$, since any length 4 path that intersects the $i$th path can be formed by first choosing one of the 4 nodes, then choosing one of the two endpoints to “grow”, then choosing one of the $d$ neighbors, etc. By the LLL, since $Pr[A_i] = 1/100d^4 < 1/(e(\cdot 16d^3 + 1))$ (for sufficiently large $d$), the claim holds.

(b) [4 points] Given the setup in the previous part, give an algorithm that will find such a coloring in expected time polynomial in the size of the graph. (No need to justify the correctness of the algorithm, just give pseudo-code.) [Hint: Constructive Lovasz Local Lemma!!]

Start with any coloring of the graph; while there exists a symmetric path (which can be found via brute-force search in time $O(n^4)$, which is polynomial), choose the colors of the nodes in that path independently at random from the $10d^2$ colors.

(c) [1 point] Prove that there is some constant $C$ such that for any graph whose maximum
degree is $d$, there exists a coloring using $C \cdot d^2$ colors such that there are no “symmetric” paths (of any length).

No one got this part, and we actually didn't even count it towards the exam total of 62 points....