SCUM - Probabilistic Method

Rikhav Shah

April 2018

The probabilistic method is a proof technique pioneered by Paul Erdos. There is a textbook titled the *Probabilistic Method* by Noga Alon and Joel Spencer.

1 Introduction

The probabilistic method is used for proving existence theorems, i.e. theories of the form "if condition C, then there exists an X such that property P holds".

For events of the form $A(c) \equiv f(c) \gtrless k$, we can use the expected value of f. If $E[f(c)] = k$, then there must be a positive probability that $f(c) \geq k$, and a positive probability that $f(c) \leq k$

$$E[f(c)] = k \implies \Pr[f(c) \leq k], \Pr[f(c) \geq k] > 0 \tag{1}$$

The proof of this follows immediately from considering the expectation as the average of f and noting that f must at some point be equal to its average, more than its average, or less than its average.

2 Warm up: Independent Sets

An independent set I of a graph $G = (V, E)$ is a set of vertices such that there are no edges between them. The 'independence number' $\alpha(G)$ is the size of the largest independent set, the MIS. Computing $\alpha(G)$ is NP-hard, so we'll focus on bounding it. If we can find an independent set of size k, then $\alpha(G) \geq k$. We can ask a natural question: under what conditions on G are we guaranteed to be able to find an independent set of size k?

2.1 Bound from $|V|$ and $|E|$

Let $n = |V|, m = |E|$. Finding the MIS is NP-hard, so instead we will randomly pick a subset of vertices $S \subset V$. Of course, this set may not be independent. However, we can easily construct one from it: for every edge with both endpoints in S, simply remove one of the endpoints from S. At the end of this procedure, there will be no edges with both endpoints in S, and so S will be independent. Let's call S the original random set and S' the set after all the deletions. If there were X vertices in S and Y edges with both endpoints in S, then S' will be of size $X - Y$ since we delete one vertex for every edge.

By using expectation, we are thus guaranteed that there exists an independent set of size at least $E[X - Y]$.

Now see that X and Y are random variables that are based on the method of randomly selecting S. Let

For a graph $G = (V, E)$ on n nodes and m edges, place each vertex in $S \subset V$ with equal independent probability p, which we'll specify later. The expected number of vertices in S is np and the expected number of edges is mp^2. This gives $E[X - Y] = np - mp^2$. Recall that we want this quantity to be small and that we get to pick p. We simply set the derivative wrt p to 0 and see that $p = \frac{n}{2m}$ is the best choice. This gives $E[X - Y] = \frac{n^2}{4m}$, so $\Pr[X - Y \geq \frac{n^2}{4m}] > 0$.

2.2 Turan's Theorem

We could construct our random set S a little bit more intelligently. Lets label the nodes 1 through n and include each vertex in S if and only if its label is larger than all of its neighbors labels. It should be clear that this is an independent set: for every edge, the endpoint with the smallest label is not eligible for membership in S. The event we're interested in is $|S| \geq k$. We'll use expectation again. If X_v is the indicator for $v \in S$ then $|S| = \sum_v X_v$ and we can use linearity of expectation.

$$E[S] = \sum_v \Pr[v \in S]$$

If d_v is the degree of vertex v then $\Pr[v \in S]$ is the probability that the label of v is the minimum of $d_v + 1$ values selected uniformly randomly, so is $1/(d_v + 1)$. Plugging this into the sum gives the final result: $\alpha(G) \geq \sum_{v \in V} \frac{1}{d_v + 1}$

3 Lovász Local Lemma

For events in the form $A = \overline{B_1} \wedge \overline{B_2} \wedge \cdots \wedge \overline{B_k}$ where each B_i is a 'bad' event, we can use the Lovász Local Lemma. Pick $p < 1$ and $d > 0$ so that $\Pr[B_i] \leq p$ for every i, and each B_i is dependent on at most d of the other B_j events. Let e be the base of the natural logarithm.

$$ep(d + 1) \leq 1 \implies \Pr[A] = \Pr[\overline{B_1} \wedge \overline{B_2} \wedge \cdots \wedge \overline{B_k}] > 0 \tag{2}$$

In words, this says that when bad events are not too likely, and when bad events are not too correlated, there's a positive probability that we simultaneously avoid all bad events. To prove this, we begin by using Bayes rule to expand the probability of A,

$$\Pr[A] = \Pr[\overline{B_1}] \cdot \Pr[\overline{B_2} \mid \overline{B_1}] \cdot \Pr[\overline{B_3} \mid \overline{B_1} \wedge \overline{B_2}] \cdot \Pr[\overline{B_k} \mid \overline{B_1} \wedge \cdots \wedge \overline{B_{k-1}}] \cdot \tag{3}$$

We want to bound each factor below by some positive value. We will instead bound the complementary probabilities above by a value less than 1. In particular, we will show that for every i,

$$\Pr[B_i \mid \overline{B_1} \wedge \cdots \wedge \overline{B_{i-1}}] < \frac{1}{d + 1}$$

In fact, it will be easier to show the more general statement that for every subset of events $\mathcal{B}$ excluding B_i that

$$\Pr[B_i \mid \bigwedge_{B_j \in \mathcal{B}} \overline{B_j}] < \frac{1}{d+1} \tag{4}$$

We will prove this using induction on the size of $\mathcal{B}$. The base case of $|\mathcal{B}| = 0$ is simple since $\Pr[B_i] \leq p < \frac{1}{e(d+1)} < \frac{1}{d+1}$. Now fix any $\mathcal{B}$ with $|\mathcal{B}| \geq 1$. We can renumber the events so that $\mathcal{B} = \{B_1, \cdots, B_{|\mathcal{B}|}\}$ and B_i depends on $\overline{B_1}, \cdots, \overline{B_{d'}}$ and not on $B_j, \cdots, B_{|\mathcal{B}|}$ (by assumption $d' \leq d$). We can again apply Bayes rule.

$$\Pr[B_i \mid \overline{B_1} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] = \frac{\Pr[B_i \wedge \overline{B_1} \wedge \cdots \wedge \overline{B_{d'}} \mid \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}]}{\Pr[\overline{B_1} \wedge \cdots \wedge \overline{B_{d'}} \mid \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}]}$$

We will now upper bound the numerator and lower bound the denominator.

Numerator:

$$\begin{aligned}
\Pr[B_i \wedge \overline{B_1} \wedge \cdots \wedge \overline{B_{d'}} \mid \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] &\leq \Pr[B_i \mid \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] && (5)\\
&= \Pr[B_i] && (6)\\
&\leq p && (7)
\end{aligned}$$

where (6) follows since event $(B_i \wedge \overline{B_1} \wedge \cdots \wedge \overline{B_{d'}})$ implies the event B_i; (7) follows since B_i is independent of the variables being conditioned on; and (8) holds by assumption. The denominator again uses Bayes rule (last time I promise!)

Denominator:

$$\begin{aligned}
\Pr[\overline{B_1} \wedge \cdots \wedge \overline{B_{d'}} \mid \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] &= \Pr[\overline{B_1} \mid \overline{B_2} \wedge \cdots \wedge \overline{B_{d'}} \wedge \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] && (8)\\
&\quad \cdot \Pr[\overline{B_2} \mid \overline{B_3} \wedge \cdots \wedge \overline{B_{d'}} \wedge \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] && (9)\\
&\quad \vdots && (10)\\
&\quad \cdot \Pr[\overline{B_{d'}} \mid \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] && (11)
\end{aligned}$$

Now see that in every factor on the RHS, that $\left(\overline{B_j} \wedge \cdots \wedge \overline{B_{d'}} \wedge \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}\right)$ contains fewer than $|\mathcal{B}|$ events, so by the inductive hypothesis their complements are are bounded above by $\frac{1}{d+1}$, and thus are bounded below by $1 - \frac{1}{d+1}$. Recalling that $d' \leq d$ we arrive at a nice bound on the denominator.

$$\Pr[\overline{B_1} \wedge \cdots \wedge \overline{B_{d'}} \mid \overline{B_{d'+1}} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] \geq \left(1 - \frac{1}{d+1}\right)^{d'} \geq \left(1 - \frac{1}{d+1}\right)^{d} \geq \frac{1}{e} \tag{12}$$

Combining (8) and (13) shows that

$$\Pr[B_i \mid \overline{B_1} \wedge \cdots \wedge \overline{B_{|\mathcal{B}|}}] \leq \frac{p}{1/e} = ep \leq \frac{1}{d+1}$$

where the final step follows from assumption. Lets plug this into (4)

$$\Pr[A] \geq \left(1 - \frac{1}{d+1}\right)^k$$

Ironically, when the events are all independent we can set $d = 0$ and this lemma gives a trivial bound, but for all other d this gives $\Pr[A] > 0$ as desired. Now let's turn to some specific problems where we can use these tools.

4 Ramsey Numbers

$R(k, k)$ is the smallest integer such that any red/blue edge coloring of the complete graph on $R(k, k)$ vertices must contain a monochromatic clique of size k. Bounding $R(k, k)$ is an open problem with the difference between the upper and lower bounds being quite large. We will prove a lower bound.

Call a coloring 'good' if it avoids monochromatic k-cliques. If we can find a good coloring of K_n, then $R(k, k) > n$. We could try to come up with a really clever algorithm for coloring, but instead we're just going to color the graph randomly. Intuitively, if a good coloring exists, then there's some (possibly exponentially small) positive probability that we'll happen to guess it with our random coloring. Conversely, if there are no good colorings, then there's no way our random coloring will be good.

Let C be the set of all colorings and let $f(c)$ be the number of monochromatic k-cliques when colored by c. One way to write the event that a coloring is good is $f(c) = 0$. Since we know $f(c)$ is nonnegative, this event is the same as $f(c) \leq 0$. This form of the event suggests us to use the expected value.

4.1 Lower bound due to Erdos: $R(k, k) \geq \frac{k}{e\sqrt{2}}\, 2^{k/2}$

First we write $f(c)$ as the sum of indicators $X_H(c)$ for the event that H is a monochromatic clique.

$$\begin{aligned} E[f(c)] &= \sum_{H \text{ size } k} \Pr[H \text{ is monochromatic}] \\ &= \binom{n}{k} \frac{1}{2^{\binom{k}{2}-1}} \end{aligned}$$

What happens when the expected value is less than 1? Well $f(c)$ is an integer, so if $E[f(c)] < 1$, there must be a nonzero probability that $f(c)$ is 0. This allows us to conclude that some coloring is good when $\binom{n}{k} \frac{1}{2^{\binom{k}{2}-1}} < 1$.

We can use Stirling's approximation and the inequality $\binom{n}{k} \leq \frac{n^k}{k!}$ to turn this into an asymptotic: $n < \frac{k}{e\sqrt{2}} 2^{k/2}$ (details are left to the reader). There is another way of thinking about the event $f(c) = 0$ which will allow us to use Lovasz.

4.2 Lower bound due to Spencer: $R(k, k) \geq \frac{k\sqrt{2}}{e}\, 2^{k/2}$

Recall that $f(c)$ is the sum of indicators X_H. The only way for this sum to be zero is if every X_H is 0. In other words, we need each size k subset to not be monochromatic. Thus the X_H-s

are our 'bad' events. L[3] tells us that the probability of avoiding every bad event is nonzero when $ep(d+1) \leq 1$. In our case, we've already stated that $p = 2^{1-\binom{k}{2}}$. For d, see that X_{H_1} and X_{H_2} are dependent only when H_1 and H_2 share an edge, i.e. share at least two vertices. For a fixed H_1, the number of ways for H_2 to share at least two vertices is less than $\binom{k}{2}\binom{n-2}{k-2}$, which makes that a suitable selection for d. Therefore, a sufficient condition on the existence of a good coloring is $e\,2^{1-\binom{k}{2}}\binom{k}{2}\binom{n-2}{k-2} \leq 1$. Using Stirling's approximation again gives the result.

5 k-Satisfiability

Say we have given a boolean formula in k-cnf form. These formulas are made of three parts: variables x_i which can be assigned true or false; literals y_i which are either a variable or it's negation (i.e. $y_i = x_i$ or $y_i = \bar{x}_i$; clauses C_j which are made up of k variables strung together with the 'or' ($\vee$) operation (we impose that each variable can only appear in a given clause once). The clauses are strung together with the 'and' ($\wedge$) operation to create the formula. The $k - SAT$ problem asks if a given formula has an assignment of its variables that makes the entire formula evaluate to true. Such an assignment is called a satisfying assignment.

Instead of trying to find a satisfying assignment, we will simply independently uniformly randomly assign true values to each variable. We can let C be the set of all boolean assignments and $f(c)$ the number of unsatisfied clauses. As before, the event we are after is $f(c) = 0$, which can be thought of as $f(c) \leq 0$. If we look at expectation, note that each clause is only false when each literal is false. Since each literal is assigned independently of others, this occurs with probability $1/2^k$. Thus we are guaranteed that there is an assignment which leaves at most $1/2^k$ of the clauses unsatisfied – but this is not strong enough!

Alternatively, we can view the failure to satisfy a clause as a 'bad' event and use LLL. Say m is the max number of times a variable appears in the formula. Then, $p = 2^{-k}$ and $d = k(m-1)$ so LLL tells us that it is possible to avoid all bad events when $epk(m-1) + ep \leq 1$. This turns into a nice condition $m \leq \frac{2^k}{ke} - \frac{1}{k} + 1$. Unfortunately, for the important 3-SAT problem this only guarantees us a satisfying assignment when each variable appears at most once. On the other hand, this bound is exponential in k and independent of the number of clauses. For 100-SAT each variable can appear 4.66×10^{27} times and no matter what, there *will* exist a satisfying assignment.

6 Games and Constructions

So far, the probabilistic method has only given us guarantees that solutions exist, but no indication about how to go about constructing them. We now turn our attention to an instance in which the the probabilistic method can be used to create an algorithm to construct a solution.

The Liar game works like this: Alice picks secret $x \in [n]$. Bob is allowed to as q questions to Alice about x. His questions must be of the form "is x in S?" where $S \subset [n]$. If Alice answers truthfully, then Bob can determine x if $q \geq \lg n$ by using binary search. To make the problem more interesting, Alice is allowed to lie a total of k times. We say that Bob wins if he is able to uniquely determine x, and that Alice wins if after q questions and answers, there are more than one possible value for x that are consistent with all the answers. We can modify the problem slightly by allowing Alice to cheat: instead of picking x at the beginning, she simply gives whatever answers to Bob's questions she pleases. If Alice isn't careful, it may end up being the case that *no* value of x satisfies

the answers shes given, and so Bob can prove that she cheated and thus wins by default. This perspective will make our analysis clearer: Bob wins if he's able to reduce the number of possible values of x to 1 or 0.

A natural question to ask is for which n, q, k do Alice or Bob have winning strategies? When $q \geq k + (k+1)\lg n$, Bob can win using binary search and just repeating every question many times. On the other hand, when $q < \lg n$ Alice will win even if $k = 0$. Instead of trying construct strategies for Alice and Bob, we instead allow them to play randomly. We will focus on Alice's strategy. First, let's reformulate the game:

There are a row of $k+1$ buckets numbered 0 through k placed on a table. n balls numbered 1 through n are placed in bucket k. Each ball represents a possible value of x. Bob's queries consist of selecting a subset S of the balls. If Alice says "yes, $x \in S$", then Bob moves all the balls in $[n] \backslash S$ down one bucket. If Alice says "no, $x \notin S$", then Bob moves all the balls in S down one bucket. If a ball is in the 0th bucket and is moved down, it's simply removed from the table. Note that ball i is moved off the table when (and only when) Alice has indicated that $x \neq i$ more than k times, and so it must be true that $x \neq i$. Thus, Alice wins if she can keep more than 1 ball on the table by the end of the game. If she answers each question randomly, we can ask the expected number of balls on the table. If X_i is the indicator for ball i staying on the table.

$$E[\text{number of balls left on table}] = \sum_i \Pr[X_i] = n \Pr[X_1]$$

For a given question, no matter what Bob selects S to be, the probability that any particular ball is moved down one bucket is $1/2$. In order for $X_1 = 1$, of q questions, at most k of Alice's responses must result in the first ball being moved down a bucket. The number of sequences of responses by Alice that result in the ball being moved m times is $\binom{n}{m}$. Summing gives the desired probability

$$\Pr[X_1] = \sum_{m=0}^{k} 2^{-q} \binom{q}{m} = \frac{\binom{q}{0} + \binom{q}{1} + \cdots + \binom{q}{k}}{2^q}$$

Since $\sum_{i=1}^{n} X_i$ must be an integer, if its expectation is more than 1, there is a positive probability it's at least 2. Therefore, in that case, there must exist a sequence of responses which results in at least 2 balls remaining on the table. A sufficient condition for Alice having a winning strategy is thus

$$n \frac{\binom{q}{0} + \binom{q}{1} + \cdots + \binom{q}{k}}{2^q} > 1$$

Again, however, this alone doesn't tell us Alice's winning strategy. To find it, let's generalize the game. Instead of placing all n chips in bucket k, let's put n_i balls in bucket i. Of the balls in bucket i, the expected number of balls on the table at the end of the game follows analogously as before:

$$E[\text{number of balls left on table from bucket } i] = n_i \frac{\binom{q}{0} + \binom{q}{1} + \cdots + \binom{q}{i}}{2^q}$$

Summing over i gives

$$E[\text{number of balls left on table}] = \sum_{i=0}^{k} \left(n_i \, \frac{\binom{q}{0} + \binom{q}{1} + \cdots + \binom{q}{i}}{2^q} \right)$$

Note that as Alice and Bob play the original game, every intermediate state is an instance of the generalized game: Bob has a certain number of questions remaining and the balls are distributed among the many buckets. We can define the easily computable function $f(\vec{n}, q) = \sum_{i=0}^{k} \left(n_i \, \frac{\binom{q}{0}+\binom{q}{1}+\cdots+\binom{q}{i}}{2^q} \right)$ with $\vec{n} = n_1 \cdots n_k$. When asked a question, Alice has two possible answers: "yes" and "no". Each answer would change the state of balls $\vec{n}$, call the possible new states $\vec{n}^{YES}, \vec{n}^{NO}$. If Alice responds randomly, then $E[f(\vec{n}, q)] = \frac{1}{2}E[f(\vec{n}^{YES}, q-1)] + \frac{1}{2}E[f(\vec{n}^{NO}, q-1)]$. Therefore, when $E[f(\vec{n}, q)] > 1$, one of $E[f(\vec{n}^{YES}, q-1)]$ or $E[f(\vec{n}^{NO}, q-1)]$ must also be more than 1. Recall that Alice wins when $E[f(\vec{n}, q)] > 1$. At every step, she computes both of $E[f(\vec{n}^{YES}, q-1)]$ and $E[f(\vec{n}^{NO}, q-1)]$ and selects "yes" or "no" based on which expectation is more than 1. She can continue this process until the end of the game. When Bob has no more questions remaining, $f(\vec{n}, 0) > 1$ shows that at least 2 balls remain on the table, and so Alice wins.