Lecture 12

Approximate counting, sampling, and volume estimation

12.1 Approximate counting and sampling

A *counting problem* is specified by a function $f : \Sigma^* \to \mathbb{N}$, where Σ is a finite alphabet. For instance, f could be the function that takes as input a (properly encoded) 3-SAT formula φ and returns the number of satisfying assignments to φ. Given an input $x \in \Sigma^*$, the goal is to return $f(x)$.

A *sampling problem* is a function $f : \Sigma^* \to \{S, \pi\}$ which returns a pair formed by a finite set S and a distribution π on S. Given an input $x \in \Sigma^*$, the goal is to produce an element $s \in S$ distributed according to π.

Counting and sampling problems tend to be very hard. For instance, even though 2-SAT can be solved in polynomial time (as we saw in the previous lecture), the problem of counting the number of solutions to a 2-SAT formula is $\#P$-hard, where $\#P$ is the analogue of NP for counting problems. The same holds for counting the number of cycles in an undirected graph. On the other hand, counting the number of matchings is known to be in P, via a simple computation.

So we settle for approximation schemes. For counting problems this is called a *fully polynomial randomized approximation scheme* (FPRAS). For sampling it is called a *fully polynomial almost uniform sampler* (FPAUS). More formally,

Definition 12.1. Given a counting problem f, a FPRAS for f is a randomized algorithm which given as input x, ε, and δ returns a value y such that $(1-\varepsilon)f(x) \leq y \leq (1+\varepsilon)f(x)$ in time that is polynomial in $|x|$, $1/\varepsilon$, and $\log(1/\delta)$.

Given a sampling problem f, a FPAUS for f is a randomized algorithm which given as input x and δ runs in time polynomial in x and $\log 1/\delta$ and returns an element $s \in S$ that is distributed according to a distribution $p = p(x, \delta)$ such that $\|p - \pi\|_1 \leq \delta$, where $f(x) = (S, \pi)$.

A famous theorem by Jerrum, Valiant and Vazirani shows that approximate counting and approximate sampling are essentially equivalent:

Theorem 12.2. *For "nicely behaved" counting problems (the technical term is "downward self-reducible"), the existence of an FPRAS is equivalent to the existence of an FPAUS.*

Proof sketch. For concreteness we prove the theorem for the problem of counting the number of satisfying assignments to any formula φ.

FPAUS $\Longrightarrow$ *FPRAS.* Take a polynomial number of satisfying assignments for φ sampled by the FPAUS. This lets us estimate p_0 and p_1, the probability that φ is satisfiable conditioned on $x_1 = 0$ and $x_1 = 1$ respectively. Assume $p_0 \geq 1/2$, the other case being symmetric. Make a recursive call to approximate the number of satisfying assignments to $\varphi_{|x_1=0}$. Let $\hat{N}_0$ be the estimate returned, and output $\hat{N}_0/p_0$.

It is clear that this is correct on expectation. Moreover, using that p_0 is not too small a Chernoff bound shows that the estimate obtained from the FPAUS samples will be very accurate with good probability. It is then not hard to prove by induction that provided a polynomial number of samples are taken at each of the n recursive calls (where n is the number of variables, the overall estimate can be made sufficiently accurate, with good probability.

FPRAS $\Longrightarrow$ *FPAUS.* First we run the FPRAS to obtain good estimates for the number of satisfying assignments N to φ and N_0 to $\varphi_{|x_1=0}$. We can assume the estimates returned are such that $\hat{N}_0/\hat{N} \geq 1/2$, as otherwise we exchange the roles of 0 and 1. Next we flip a contain with bias $\hat{N}_0/\hat{N}$. If it comes up heads we set $x_1 = 0$ and recurse; if it comes up tails we set $x_1 = 1$ and recurse. Provided the estimates $\hat{N}, \hat{N}_0, \ldots$ are accurate enough the distribution on assignments produced by this procedure will be close in statistical distance to the uniform distribution on satisfying assignments. □

12.2 Volume estimation

Let $K \subseteq \mathbb{R}^n$ be convex. Our goal is to estimate the volume $\mathrm{Vol}_n(K)$: we would like to devise an efficient procedure that, given K, returns two real numbers α, β that are such that $\alpha \leq \mathrm{Vol}_n(K) \leq \beta$ and β/α is as small as possible. Before diving into this, however, a basic question — how is K specified?

In order to abstract out the details and provide an algorithm that works in a general context we will assume that the only access we are given to K is through one of the following type of "oracle":

- *Membership oracle* (resp. *weak membership oracle*): given a query $x \in \mathbb{R}^n$, the oracle answers whether $x \in K$ (resp. $x \in K$ or $d(x, K) \geq \varepsilon$; the oracle is allowed to fail whenever neither condition is satisfied).

- *Separation oracle* (resp. *weak separation oracle*): given a query x, the oracle returns the same answer as the membership oracle, but in case $x \notin K$ (resp. $d(x, K) \geq \varepsilon$) it also returns an $y \in \mathbb{R}^n$ such that $y^T z > y^T x \ \forall z \in K$ (resp. $y^T z > y^T x - \varepsilon/2 \ \forall z \in K$).

It is always possible to derive a weak separation oracle from a weak membership oracle, and in this lecture we won't worry about the difference — in fact all we'll need is a weak membership oracle.

But is this enough? What if we query x, and we learn $x \notin K$, with $y = (1, 0, \ldots, 0)$? Then we know we should increase x_1. Say we double every coordinate, to $x' = 2x$. But suppose we get the same answer, again and again. We never know how far K is! It seems like a boundedness assumption is necessary, so we'll assume the following: there exists (known) values $r, R > 0$ such that $B_\infty(0, r) \subseteq K \subseteq B_\infty(0, R)$ with $R/r < 2^{\mathrm{poly}(n)}$, where $B_\infty(0, r)$ denotes the ball of radius r for the ℓ_∞ norm (a square box with sides of length $2r$). In fact, by scaling we may as well assume $r = 1$, and for simplicity we'll also assume $R = n^2$. It is not so obvious at first this is without loss of generality, but it is — with some further re-scaling and shifting of things around it is not too hard to reduce to this case.

(The idea is to slowly grow a simplex inside K. At each step we can perform a change of basis so that $\mathrm{Conv}(e_1, \ldots, e_n) \subseteq K$. Then for $i = 1, \ldots, n$ we check if there is a point $x \in K$ such that $|x_i| \geq 1 + 1/n^2$. If so we include it, rescale, and end up with a simplex of volume at least $1 + 1/n^2$ times the previous one, and this guarantees we won't have to go through too many steps. If there is no such point, we stop, as we've achieved the desired ratio. Finally we need to check that we can find such point, if it exists, in polynomial time; this is the case as if $x \in K$ then $(0, \ldots, 0, x_i, 0, \ldots, 0) \in K$ as well (using convexity and the assumption that K contains the simplex), so it suffices to call the membership oracle n times.)

Now that we have a proper setup in place, can we estimate $\mathrm{Vol}(K)$? Say we only want a multiplicative approximation. An idea would be to use the separation oracle to get some kind of a rough approximation of the boundary of K using hyperplanes, then do some kind of triangulation, and estimate the volume by counting triangles. In fact there is a very strong no-go theorem for deterministic algorithms:

Theorem 12.3. *For any polynomial-time algorithm such that on input a convex body $K \subseteq \mathbb{R}^n$ (specified via a separation oracle) the algorithm returns $\alpha(K)$, $\beta(K)$ such that $\alpha(K) \leq \mathrm{Vol}(K) \leq \beta(K)$, it must be that there exists a constant $c > 0$ and a sequence of convex bodies $\{K_n \subseteq \mathbb{R}^n\}_{n \geq 1}$ such that for all $n \geq 1$,*

$$\frac{\beta(K_n)}{\alpha(K_n)} \geq \left(\frac{cn}{\log n}\right)^n.$$

This is very bad: even an approximation within an *exponential* factor is ruled out! Note however that a key to the above result is that the only access to K is given by a separation oracle — if we have more knowledge about K then a polynomial-time algorithm might be feasible (though we don't know any).

Proof idea. The idea for the proof is to design an oracle that answers the queries made by any deterministic algorithm in a way that is consistent with the final convex body being one of two possible bodies, K or K°, whose volume ratio is exponentially large; if we manage to do this then the algorithm cannot provide an estimate that will be accurate for both K and K°.

The oracle is very simple: upon any query $x \in \mathbb{R}^n$ it is very generous and says that $x/\|x\| \in K$, $-x/\|x\| \in K$, and moreover K is included in the "slab" $\{y : -\|x\| \leq \langle y, x\rangle \leq \|x\|\}$. Note that these answers are all consistent with K being the Euclidean unit ball.

Now, if points $x_1, \ldots, x_m$ have been queried, define K to be the convex hull of $\pm x_i/\|x_i\|$, $\pm e_j$ where e_j are the unit basis vectors. Define $K^\circ = \{y : \langle y, z\rangle \leq 1 \, \forall z \in K$. Then you can check that the oracle's answers are all consistent with K and K°. But their volumes are very different, and one can show that $\mathrm{Vol}(K^\circ)/\mathrm{Vol}(K)$ is roughly of order $(n/\log(m/n))^n$; as long as m is not exponential in n this is exponentially large. □

If we allow randomized algorithms the situation is much better:

Theorem 12.4 (Dyer, Frieze, Kannan 1991)**.** *There exists a fully polynomial randomized approximation scheme for approximating Vol(K).*

A fully polynomial randomized approximation scheme (FPRAS) means that $\forall \varepsilon, \delta > 0$ the algorithm returns a $(1 \pm \varepsilon)$-multiplicative approximation to the volume with probability at least $1 - \delta$, and runs in time $\mathrm{poly}(n, 1/\varepsilon, \log 1/\delta)$. Volume estimation is one of these relatively rare problems for which we have strong indication that randomized algorithms can be exponentially more efficient than deterministic ones (primality testing used to be another such problem before the AKS algorithm was discovered!).

The algorithm of Dyer, Frieze and Kannan had a running time that scaled like $\sim n^{23}$. Since then a lot of work has been done on the problem, and the current record is $\sim n^5$. In principle it's possible this could be lowered even more, to say $\sim n^2$; there are no good lower bounds for this problem.

Proof sketch. The main idea is to use random sampling. For instance, suppose we'd place K inside a large, fine grid, as in Figure 12.1.

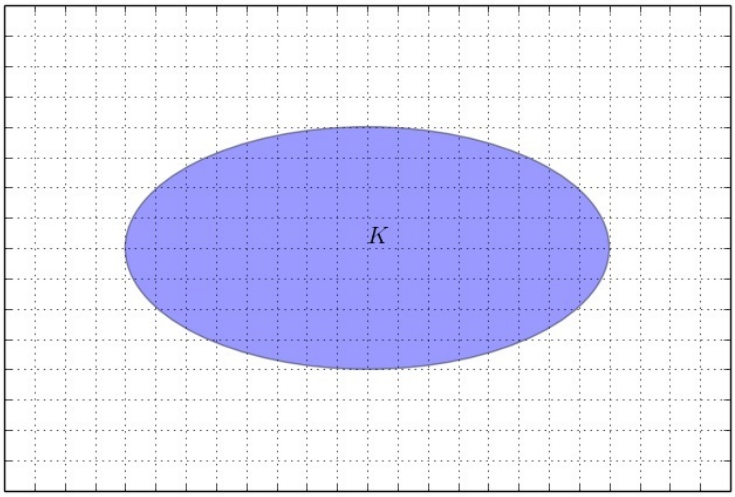

Figure 12.1: The region K is the feasible region.

We could then run a random walk on the grid until it mixes to uniform. This will take time roughly $n(R/\delta)^2$, where δ is the grid spacing; given we assumed $r = 1$ something like $\delta = 1/n^2$ would be reasonable.

Exercise 1. Show that the mixing time of the lazy random walk on $\{1, \ldots, N\}^n$, the n-dimensional grid with sides of length N, is $O(n^2N^2)$.

At the end of the walk we can call the membership oracle to check if we are in K. Since $\text{Prob}(x \in K) \sim \frac{\text{Vol}_n(K)}{\text{Vol}_n(\text{grid})}$, by repeating the walk sufficiently many times we'd get a good estimate. While this works fine in two dimensions (you can estimate $\pi = \text{Area(unit disk)}$ in this way!), in higher dimensions it fails dramatically, as all we know about $\text{Vol}(K)$ is that it is at least 2^n (since it contains the unit ball for ℓ_∞), but the grid could have volume as large as $(2R)^n = (2n^2)^n$, so even assuming perfectly uniform mixing the probability that we actually obtain a point in K is tiny, it's exponentially small.

This is still roughly how we'll proceed, but we're going to have to be more careful. There are three important steps. Here is a sketch:

- Step 1: Subdivision.
  Set $K_0 = B_\infty(0,1) \cap K = B_\infty(0,1), K_1 = B_\infty(0, 2^{1/n}) \cap K, \ldots, K_{2n\log n} = B_\infty(0, n^2) \cap K = K$. Then

$$\text{Vol}(K) = \text{Vol}(K_{2n\log n}) = \frac{\text{Vol}(K_{2n\log n})}{\text{Vol}(K_{2n\log n-1})} \cdot \frac{\text{Vol}(K_{2n\log n-1})}{\text{Vol}(K_{2n\log n-2})} \cdots \frac{\text{Vol}(K_2)}{\text{Vol}(K_1)} \cdot 2^n,$$

since $\text{Vol}(K_1) = 2^n$. So, we have reduced our problem to the following: given $K \subseteq L$ both convex such that $\text{Vol}(K) \geq \frac{1}{2}\text{Vol}(L)$, estimate $\text{Vol}(K)/\text{Vol}(L)$. This eliminates the "tiny ratio" issue we had initially, but now we have another problem: the enclosing set L is no longer a nice grid, but it is an arbitrary convex set itself. Are we making any progress?

- Step 2: A random walk.
  Our strategy will be as follows. Run a random walk on a grid that contains L, such that the stationary distribution of the random walk satisfies the two conditions that $\mathbf{Pr}(x \in L)$ is not too small, and the stationary distribution is close to uniform, conditioned on lying in L. If we can do this we're done: we repeatedly sample from the stationary distribution sufficiently many times that we obtain many samples in L, and we check the fraction of these samples that are also in K: $\frac{\text{Vol}(K)}{\text{Vol}(L)} \sim \frac{\mathbf{Pr}(x \in K)}{\mathbf{Pr}(x \in L)} = \mathbf{Pr}(x \in K | x \in L)$.

  So the challenge is to figure out how to define this random walk around L. Here is a natural attempt. Start at an arbitrary point $x^{(0)} \in L$, say the origin. Set $x^{(1)}$ to be a random neighbor of $x^{(0)}$ on the grid, subject to $x^{(1)} \in L$ (we have $2n$ neighbors to consider, and for each we can call the membership oracle for L). Repeat sufficiently many times. This is the right idea — note that we really want to stay as close to L as possible, because if we allow ourselves to go outside too much we'll get this "tiny ratio" issue once more — but the boundary causes a lot of problems:

  (a) Some points are never reached. L could be very pointy, in which case there could be a grid point that lies in L, but none of its neighbors does. And this cannot be solved just by making the grid finer; it is really an issue with the kinds of angles that are permitted in L.

  (b) Some grid cubes have much bigger intersection with L than others.

  (c) The degree of the graph underlying our walk is not constant (it tends to be smaller close to the boundary), so the stationary distribution will not be uniform.

  It turns out we can fix all of these issues by doing a bit of "smoothing out" on L. Let δ be the width of the grid, and consider $L' = (1 + \delta\sqrt{n})L$, where $\delta\sqrt{n}$ is the diameter of a cube. Assuming $\delta \leq n^{-2}$, this doesn't blow up the volume by much, so in terms of volume ratio we're fine. Moreover, you can check easily that:

  – Any grid point inside L has all of its neighbors in L',
  – All $p \in L$ belong to a grid cube $\subseteq L'$.

  These two points get rid of issue (a) above: all points in L are now reached by the walk. Moreover, we can easily get rid of the degree issue by adding self loops. This guarantees that the stationary distribution will be uniform on grid points in L, clearing (c). Remains (b), the issue of uneven intersection between grid cubes and L. For this we do the following:

  – Do a random walk on $\delta\mathbb{Z}^n \cap L'$ as described before.

– Arrive at a random grid point p. Choose random vector $q \in B_\infty(0,1)$ and output the point $p + \delta q$ if it is in L. Otherwise, restart the walk.

As a result the stationary distribution is uniform on L: if we have a cube C that partially intersects L its points are sampled with probability precisely $\sim \text{Vol}(L \cap C)$. We also need to make sure that there are not too many restarts — what if $\text{Vol}(L \cap C)/\text{Vol}(C)$ is again tiny? Now here we can show this is ok if it happens, because the point is we're trying to estimate the whole volume of L, not just the volume for that intersection. If it's tiny, we never see it, but that's fine.

- Step 3: Mixing Time.
So far the random walk would work even if K is not convex, in the sense that as long as the walk mixes it will let us estimate the ratio $\text{Vol}(K \cap L)/\text{Vol}(L)$. But now we need to understand the conductance of our graph: by Cheeger's inequality and the analysis of mixing time we saw in the previous lecture, as long as the conductance is at least 1/poly, we are good to go.

Since the graph is regular, given a set of vertices S such that $|S| \leq n/2$, we have $\phi(S) = \frac{|\partial S|}{d|S|}$. If V is the volume of a cube, then $d|S| \sim \frac{\text{Vol}(S)}{V}$, where $\text{Vol}(S)$ is the sum of volumes of cubes at vertices in S, and $|\partial S| \sim \frac{\text{Area}(\partial S)}{A}$, where A is the surface area ($(n-1)$-volume) of a single cube. Since $V/A = 2\delta \approx 2/n^2$, we just need to lower bound $\frac{\text{Area}(S)}{\text{Vol(S)}}$. Fix a value of $\text{Vol}(S)$. How small can $\text{Area}(S)$ be? This is called an *isoperimetric inequality.*

Theorem 12.5. *Let $L \subseteq \mathbb{R}^n$ be convex with diameter d . Let $S \subseteq L$ be such that $Vol(S) \leq Vol(L)/2$. Then, $Area(S) \geq \frac{1}{d} Vol(S)$.*

If we look at a body L that is a very thin needle, we see that for S that cuts across half the needle the volume will be roughly $(d/2)$ times the area, so the estimate provided in the theorem is essentially optimal. The proof of the theorem is not hard but it does involve quite a bit of re-arranging to argue that the needle is indeed the "worst-case scenario", and we'll skip it.

In our setting we have $B_\infty(0,1) \subseteq K \subseteq L \subseteq L' = (1+\delta\sqrt{n})L \subseteq (1+\delta\sqrt{n})B_\infty(0,n^2)$. So, our grid has sides of length at most $\lesssim \frac{2(1+\delta\sqrt{n})n^2}{\delta} = O(n^4)$, and the diameter is $O(n^4)$. Thus the isoperimetry theorem implies that our random walk will mix in polynomial time.

□