Miscellaneous Results

Contents

1 Catalan Objects and Dyck Paths

Dyck paths are one interpretation of the Catalan numbers. Here, we will instead consider a more general form of Dyck Paths, which correspond to numbers in the *Catalan Trapezoid*.

A Dyck path can be constructed as a $2n$ step one-dimensional random walk (Figure 1). Each step in the walk moves one unit along the positive x-axis and one unit up or down the positive y-axis. Given these restrictions, we would obtain a 1D random walk pinned to zero on both sides. A Dyck path also has the additional restriction that the y-coordinate of any point on the random walk is ≥ 0. i.e. the walk is always north of the origin.

The number of possible Dyck paths is the n^{th} Catalan number –

$$C_n = \frac{1}{n+1} \cdot \binom{2n}{n}$$

We will attempt to support queries to a uniformly random instance of a Dyck path. Specifically, we will want to answer queries of the form HEIGHT(i), which returns the position of the path after i steps.

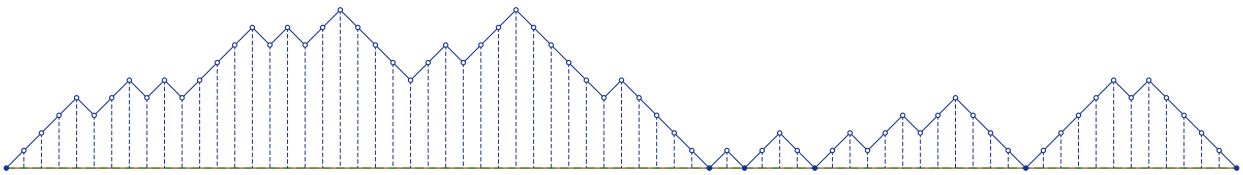

Figure 1: Simple Dyck path with $n = 35$.

1.1 Catalan Trapezoids and Generalized Dyck Paths

First, we define Catalan trapezoids as presented in [Reu14]. Let $C_k(n, m)$ be the $(n, m)^{th}$ entry of the Catalan trapezoid of order k, where $C_1(n, m)$ corresponds to the Catalan triangle.

The interpretation is as follows. Consider a sequence of n up-steps and m down-steps, such that the sum of any initial sub-string is not less than $1-k$. This means that we start our Dyck path at a height of $k-1$, and we are never allowed to cross below zero (Figure 2). The total number of such paths is exactly $C_k(n, m)$. For $k = 1$, we obtain the definition of the simple Dyck path (Figure 1).

Now, we state a result from [Reu14] without proof

$$C_k(n,m) = \begin{cases} \binom{n+m}{m} & 0 \leq m < k \\ \binom{n+m}{m} - \binom{n+m}{m-k} & k \leq m \leq n+k-1 \\ 0 & m > n+k-1 \end{cases}$$

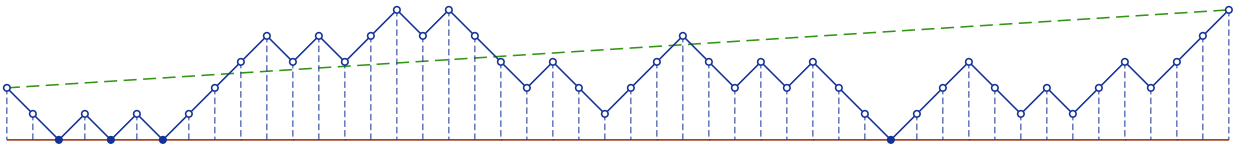

Figure 2: Complex Dyck path with $n = 25$, $m = 22$ and $k = 3$. Notice that the boundary is shifted.

1.2 Generating Dyck Paths

Our general recursive step is as follows. We consider a sequence of length $2S$ comprising of $2U$ up moves $(+1)$ and $2D$ down moves (-1). Additionally, the sum of any initial sequence prefix? canon be less than $k-1$. Without loss of generality, let's assume that $2D \leq S$. If this were not the case, we could simply flip the sequence and negate the elements. This essentially means that the overall Dyck path is non-decreasing.

Lemma 1. $S - 2D = \mathcal{O}(\log n\sqrt{S}) \implies U - D = \mathcal{O}(\log n\sqrt{S})$

We want to sample the height of this path after S steps. This is the same as sampling the number of $(+1)$s that get assigned to the first half of the elements in the sequence. We define p_d as the probability that exactly $D - d$ (-1)s get assigned to the first half. This means that exactly $U + d$ $(+1)$s get assigned to the first half. Consequently, the second half will contain exactly $D + d$ (-1)s and $U - d$ $(+1)$s.

What is d is negative?

Let us first compute this probability.

$$p_d = \frac{D_{left} \cdot D_{right}}{D_{tot}}$$

Here, D_{left} denotes the number of valid starting sequences (first half) and D_{right} denotes the number of valid ending sequences. Here, *valid* means that each half sequence gets the appropriate number of ups and downs and the initial sums never drop below $1 - k$. For, D_{right}, we will start the Dyck path from the end of the $2S$ sequence. In this case the invalidation threshold will be a different k'. This k' is the final height of the $2S$ sequence. So, $k' = k + 2U - 2D = k + 4S - 2D$. We will use this fact extensively moving forward.

Also, D_{tot} is the total number of possible sequences of length $2S$, given the initial conditions. Note that in this case the threshold remains at k.

We will use the following rejection sampling lemma from [GGN10].

Frequently?

Lemma 2. *Let $\{p_i\}$ and $\{q_i\}$ be distributions satisfying the following conditions*

1. *There is a poly-time algorithm to approximate p_i and q_i up to $\pm n^{-2}$*
2. *Generating an index i according to q_i is closely implementable.*
3. *There exists a poly(logn)-time recognizable set S such that*

- $1 - \sum_{i \in S} p_i$ *is negligible*
- *There exists a constant* c *such that for every* i*, it holds that* $p_i \leq \log^{\mathcal{O}(1)} n \cdot q_i$

Then, generating an index i *according to the distribution* $\{p_i\}$ *is closely-implementable.*

Algorithm 1 Naïve Generator

procedure SPLIT(U, D, k)
  $S \leftarrow U + D$
  $d \sim \left\{ \frac{\binom{S}{S-d}\binom{S}{D+d}}{\binom{2S}{2D}} \right\}_d$
  $k' \leftarrow k + U - D$
  $p_d \leftarrow \frac{\binom{S}{D-d} - \binom{S}{D-d-k}\binom{S}{U-d} - \binom{S}{U-d-k'}}{\binom{2S}{2D} - \binom{2S}{2D-k}}$
  $q_d \leftarrow \frac{\binom{S}{S-d}\binom{S}{D+d}}{\binom{2S}{2D}}$
  if $p_d < q_d$ **then**
    return d
  draw $X \sim \mathsf{Bern}(p_d/q_d)$
  if $X = 0$ **then**
    return d
  return SPLIT(U, D, k, k')
procedure HEIGHT(x)
  if $x \in heights$ **then**
    return $heights[x]$
  $l \leftarrow$ LOWER-BOUND(x)
  $r \leftarrow$ UPPER-BOUND(x)
  $h_l \leftarrow$ HEIGHT(l)
  $h_r \leftarrow$ HEIGHT(r)
  $extra \leftarrow (r - l) - (h_r - h_l)$
  $U \leftarrow (h_r - h_l) + extra/2$
  $D \leftarrow extra/2$
  $k \leftarrow 1 + h_l$
  $d \leftarrow$ SPLIT(U, D, k)
  $heights[(r + l)/2] \leftarrow h_l + U + d$
  return HEIGHT(x)

1.2.1 The Simple Case

The problem of sampling reduces to the binomial sampling case when $k > \mathcal{O}(\log n)\sqrt{S}$ for some constant c. This is because with high probability, will never dip below the threshold. In this case, the we can simply approximate the probability as

$$\frac{\binom{S}{D-d} \cdot \binom{S}{D+d}}{\binom{2S}{2D}}$$

This is because unconstrained random walks will not dip below the $1-k$ threshold with high probability. This problem was solved in [GGN10] using $\mathcal{O}(poly(\log n))$ resources.

1.2.2 Path Segments Close to Zero

The problem arises when we $k < \mathcal{O}(\log n)\sqrt{S}$. In this case we need to compute the actual probability, Using the formula from [Reu14], we find that.

$$D_{left} = \binom{S}{D-d} - \binom{S}{D-d-k} \qquad D_{right} = \binom{S}{U-d} - \binom{S}{U-d-k'} \qquad D_{tot} = \binom{2S}{2D} - \binom{2S}{2D-k} \tag{1}$$

Here, $k' = k + 2U - 2D$, and so $k' = \mathcal{O}(\log n)\sqrt{S}$ (using Lemma 1).

The final distribution we wish to sample from is given by $\{p_d\}_d$ where $p_d = \frac{D_{left} \cdot D_{right}}{D_{tot}}$. To achieve this, we will use Lemma 2 from [GGN10]. An important point to note is that in order to apply this lemma, we must be able to compute the p_d values. For now, we will assume that we have access to an oracle that will compute the value for us. Later, in Section **??**, we will see how to construct such an oracle.

Fix reference

In this process, we will first disregard all values of d where $|d| > \Theta(\log n\sqrt{S})$. The probability mass associated with these values can be shown to be negligible .

bound variance of path

Next, we will construct an appropriate $\{q_i\}$ and show that $p_d < \log^{\mathcal{O}(1)} n \cdot q_d$ for all $|d| < \Theta(\sqrt{S})$ and some constant c. We will use the following distribution

$$q_d = \frac{\binom{S}{D-d} \cdot \binom{S}{D+d}}{\binom{2S}{2D}} = \frac{\binom{S}{D-d} \cdot \binom{S}{U-d}}{\binom{2S}{2D}}$$

It is shown in [GGN10] that this distribution is closely implementable.

First, we consider the case where $k \cdot k' \leq 2U+1$. In this case, we use loose bounds for $D_{left} < \binom{S}{D-d}$ and $D_{right} < \binom{S}{U-d}$. We also use the following lemma (proven in Section A).

Lemma 3. *When $kk' > 2U+1$, $D_{tot} > \frac{1}{2} \cdot \binom{2S}{2D}$.*

Combining the three bounds we obtain $p_d < \frac{1}{2} q_d$. Intuitively, in this case the dyck boundary is far away, and therefore the number of possible paths is only a constant factor away from the number of unconstrained paths (no boundary).

The case where the boundaries are closer (i.e. $k \cdot k' \leq 2U + 1$) is trickier, since the individual counts need not be close to the corresponding binomial counts. However, in this case we can still ensure that the sampling probability is within poly-logarithmic factors of the binomial sampling probability. We use the following lemmas (proven in Section A).

Lemma 4. *$D_{left} \leq c_1 \frac{k \cdot \log n}{\sqrt{S}} \cdot \binom{S}{D-d}$ for some constant c_1.*

Lemma 5. *$D_{right} < c_2 \frac{k' \cdot logn}{\sqrt{S}} \cdot \binom{S}{U-d}$ for some constant c_2.*

Lemma 6. *When $kk' \leq 2U + 1$, $D_{tot} < c_3 \frac{k \cdot k'}{S} \cdot \binom{2S}{2D}$ for some constant c_3.*

We can now put these lemmas together to show that $p_d/q_d \leq \Theta(\log^2 n)$. Now, we can apply Lemma 2 to sample the value of d, which gives us the height of the Dyck path at the midpoint of the two given points.

Theorem 1. *There is an algorithm that given two points at distance a and b (with $a < b$) along a Dyck path of length $2n$, with the guarantee that no position between a and b has been sampled yet, returns the height of the path halfway between a and b. Moreover, this algorithm only uses $\mathcal{O}(poly(\log n))$ resources.*

Proof. If $b - a$ is even, we can set $S = (b - a)/2$. Otherwise, we first sample a single step from a to $a + 1$, and then set $S = (b - a - 1)/2$. Since there are only two possibilities for a single step, we can explicitly compute an approximation of the probabilities, and then sample accordingly. Now, if $S > \Theta(\log^2 n)$ we can simply use the rejection sampling procedure described above to obtain a $\mathcal{O}(poly(\log n))$ algorithm. Otherwise, we sample each step induividually. Since there are only $2S = \Theta(\log^2 n)$ steps, the sampling is still efficient. □

Theorem 2. *There is an algorithm that provides sample access to a Dyck path of length $2n$, by answering queries of the form* HEIGHT(x) *with the correctly sampled height of the Dyck path at position x using only $\mathcal{O}(poly(\log n))$ resources per query.*

Proof. The algorithm maintains a successor-predecessor data structure (e.g. Van Emde Boas tree) to store all positions x that have already been queried. Each newly queried position is added to this structure. Given a query HEIGHT(x), the algorithm first finds the successor and predecessor (say a and b) of x among the alredy queried positions. This provides us the quarantee required to apply Theorem 1, which allows us to query the height at the midpoint of a and b. We then binary search by updating either the successor or predecessor of x. Once the interval length becomes less than $\Theta(\log^2 n)$, we perform the full sampling (as in Theorem 1) which provides us the height at position x. □

1.3 Supporting "First Return" Queries

We might want to support more complex queries to a Dyck path. Specifically, in addition to querying the height of a position, we might want to know the next time the path return to that

height (if at all). We introduce a new query First-Return(x) which returns the first time the walk returns to Height(x) if the step from x to $x+1$ is an up-step.

The utility of this kind of query can be seen in other interpretations of Catalan objects. For instance, if we interpret it as a well bracketed expression, First-Return(x) returns the position of the bracket matching the one started at x. If we consider a uniformly random rooted tree, the fucntion effectively returns the next child of a vertex.

Explain why

We will use the following asymptotic formula for *close-to-central* binomial coefficients.

Lemma 7. *If $k = \frac{n \pm c\sqrt{n}}{2}$ where $c = o(n^{1/6})$, we can approximate $\binom{n}{k}$ up to constant factors by the expression:*

$$\frac{2^n}{\sqrt{n}} \cdot e^{-c^2/2}$$

We maintain a threshold $\mathcal{T} = \Theta(\log^7 n)$. If an un-sampled interval in the Dyck path has length less than $\mathcal{T}$, then we sample the entire interval. So, for intervals with length $S > \mathcal{T}$, the maximum deviations are bounded by $\mathcal{O}(\log n\sqrt{S}) = \mathcal{O}(\log^{4.5} n)$ with high probability. Specifically, this means that if we write the deviation as $c\sqrt{n}$, we see that $c = \log n$ which is $o(S^{1/6})$.

Formalize this notion of deviations

1.3.1 Maintaining a Boundary Invariant

Why?

Consider all positions that have been queried already $\langle x_1, x_2, \cdots, x_m \rangle$ (in increasing order) along with their corresponding heights $\langle h_1, h_2, \cdots, h_m \rangle$. We maintain an invariant that for each $i < m$, the Dyck path between positions x_i and x_{i+1} is constrained to lie above $min(h_i, h_{i+1})$.

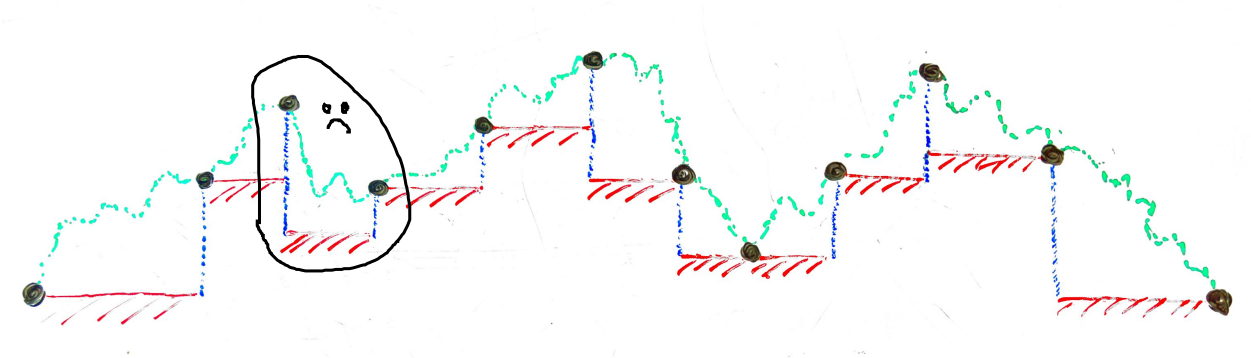

Figure 3: Error in third segment.

It is not even clear that this is always possible. After sampling the height of a particular position x_i as h_i (with $x_{i-1} < x_i < x_{i+1}$), the invariant is potentially broken on either side of x_i. We will re-establish the invariant by sampling an additional point on either side. This proceeds as follows for the interval between x_i and x_{i+1} (see error in Figure 3):

1. Sample the lowest height h achieved by the walk between x_i and x_{i+1}.

2. Sample a position x such that $x_i < x < x_{i+1}$ and $\textsc{Height}(x) = h$.

Since h is the minimum height along this interval, sampling the point x suffices to preserve the invariant.

1.3.2 Sampling the Lowest Achievable Height

For the first step, we need to sample the lowest height of the walk between x_i and x_{i+1}. Notice that we can assume $x_i < x_{i+1}$ without loss of generality (if $x_i > x_{i+1}$, swap them and proceed). Let's say that the boundary is currently $k' - 1$ units below h_i.

We know how to count the numnber of possible Dyck paths for any given boundary. Dividing by the total number of possible paths gives us precisely the CDF we need. This allows us to binary search to find the boundary.

We will use D_k to denote the number of paths that respect a boundary which is $k - 1$ units below h_i. So, in the first step, we compute $p = D_{k'/2}/D_{k'}$. This means that with probability p, the path never reaches height $h_i - k'/2$. Otherwise, the path must reach $h_i - k'/2$ but not $h_i - k'$. Note that we can also calculate the total number of such paths as $D_{k'} - D_{k'/2}$. We repeat this procedure, essentially performing binary search, until we find a k such that the path reaches height $h_i - k + 1$ (potentially multiple times), but never goes below it.

1.3.3 Sampling the Position of First Return

Now that we have a "*mandatory boundary*" k, we just need to sample a position x with height $h = x_i - k + 1$. In fact, we will do something stronger by sampling the *first* time the walk touches the boundary after x_i.

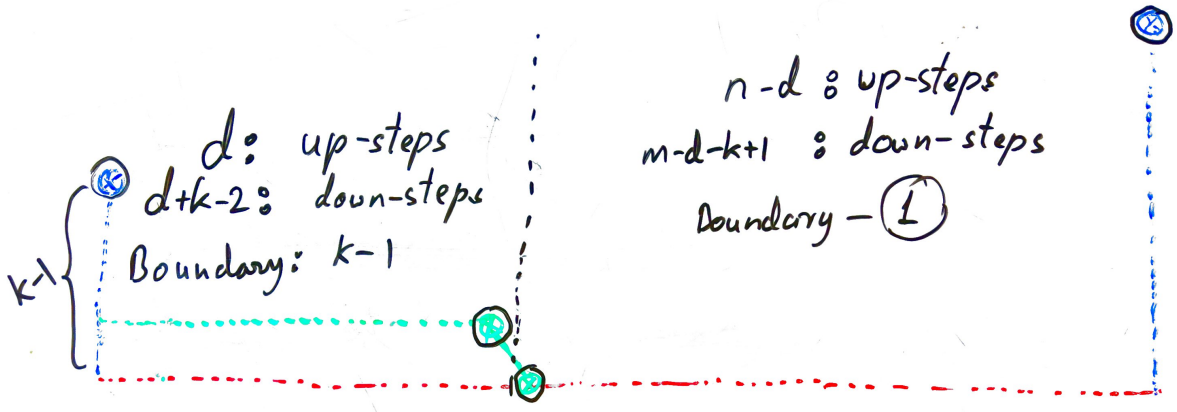

Figure 4: Zooming into (and flipping) the error in Figure 4

We will parameterize the position x the number of up-steps between x_i and x (See Figure 4). This

quantity will be referred to as d such that $x - x_{i+1} = 2d + k - 1$. Given a specific d, we want to compute the number of valid paths that result in d up-steps before the first approach to the boundary. We will calculate this quantity by counting the total number of paths to the left and right of the first approach and multiplying them together.

Since we only care about getting a asymptotic (up to poly$(\log n)$ factors) estimate of the probabilities, it suffices to estimate the number of paths asymptotically as well.

Lemma 8. $D_{left}(d) = \Theta\left(\frac{2^{2d+k}}{\sqrt{d}} e^{-r_{left}(d)} \cdot \frac{k-1}{d+k-1}\right)$ *where* $r_{left}(d) = \frac{(k-2)^2}{8(2d+k-2)}$.

Lemma 9. $D_{right}(d) = \Theta\left(\frac{2^{n+m-2d-k}}{\sqrt{n}} e^{-r_{right}(d)} \cdot \frac{n-m+k}{n-d+1}\right)$ *where* $r_{right}(d) = \frac{(n-m-k-1)^2}{8(n+m-2d-k+1)}$.

1.3.4 Estimating the CDF

Lemma 10. $D_{left}(d) \cdot D_{right}(d) = \Theta\left(\frac{2^{n+m}}{dn} e^{-r(d)} \cdot \frac{k-1}{d+k-1} \cdot \frac{n-m+k}{n-d+1}\right)$ *where* $r(d) = \mathcal{O}(\log^2 n)$.

Proof. This follows from the fact that both $r_{left}(d)$ and $r_{right}(d)$ are $\mathcal{O}(\log^2 n)$. □

Corollary 1. *The probability p_d of sampling d as the number of up-steps before the first approach to the boundary can be approximated as:*

$$p_d = \Theta\left(\frac{2^{n+m} \cdot \frac{(k-1)(n-m+k)}{dn(d+k-1)(n-d+1)} \cdot e^{-\lfloor r(d) \rfloor}}{\binom{n+m}{n} - \binom{n+m}{m-k}}\right)$$

This is because the floor function only affects the value of the exponential by a factor of at most e.

D_{total} is not correct

Corollary 2. *We define a piecewise continuous function*

$$\hat{q}(d) = \frac{2^{n+m} \cdot \frac{(k-1)(n-m+k)}{dn(d+k-1)(n-d+1)} \cdot e^{-\lfloor r(d) \rfloor}}{\binom{n+m}{n} - \binom{n+m}{m-k}}$$

We claim that $p_d = \Theta\left(\int_d^{d+1} \hat{q}(d)\right)$. *Note that this integral has a closed form for a fixed value of* $\lfloor r(d) \rfloor$.

Let the maximum value of $r(d)$ be $r_{max} = \mathcal{O}(\log^2 n)$.

Corollary 3. *We can compute the approximate normalized probabilities*

$$q_d = \frac{\int_d^{d+1} \hat{q}(d)}{\int_1^n \hat{q}(d)}$$

such that $p_d = \Theta(q_d)$. *Furthermore, we can also compute the CDF of* q_d *as:*

$$Q_d = \frac{\int_1^{d+1} \hat{q}(d)}{\int_1^n \hat{q}(d)}$$

This allows us to sample from the distribution q_d *and use Lemma 2 to indirectly sample from* p_d.

References

[GGN10] Oded Goldreich, Shafi Goldwasser, and Asaf Nussboim. On the implementation of huge random objects. *SIAM Journal on Computing*, 39(7):2761–2822, 2010.

[Reu14] Shlomi Reuveni. Catalan's trapezoids. *Probability in the Engineering and Informational Sciences*, 28(03):353–361, 2014.

A Dyck Path Generator

A.1 Dyck Path Boundaries and Deviations

Consider a contiguous *sub-path* of a simple Dyck path of length $2n$ that comprises of U up-steps and D down-steps with $U + D = 2S$, such that the sum of any initial sub-string is not less than $1 - k$ for some valid $U, D, k < n$. This means that we start our Dyck path at a height of $k - 1$, and we are never allowed to cross below zero (Figure 2).

Lemma 11. *Both $|S - U|$ and $|S - D|$ are $\mathcal{O}(\log n\sqrt{S})$ with high probability.*

Proof. □

Lemma 12. *There exists a constant c such that if $k > c \log n\sqrt{S}$, then the distribution of paths sampled without a boundary (normal random walks) is statistically $1/n^2$-close in L_1 distance to the distribution of Dyck paths.*

Proof. □

A.2 Computing Probabilities

We start with Stirling's approximation which states that

$$m! = \sqrt{2\pi m}\left(\frac{m}{e}\right)^m \left(1 + \mathcal{O}\left(\frac{1}{m}\right)\right)$$

We will also use the logarithm approximation when a better approximation is required:

$$\log(m!) = m \log m - m + \frac{1}{2}\log(2\pi m) + \frac{1}{12m} - \frac{1}{360m^3} + \frac{1}{1260m^5} - \cdots \tag{2}$$

Oracle for estimating probabilities:

Lemma 13. *Given a probability expression of the form $p_d = \frac{D_{left} \cdot D_{right}}{D_{total}}$ and a parameter n, there exists a* poly$(\log n)$ *time oracle that returns a $\left(1 \pm 1/n^2\right)$ multiplicative approximation to p_d.*

Point to section referencing the left right/ total.

Proof. We first compute a $1/n^2$ *additive* approximation to $\log p_d$. Note that this is possible because $\log p_d$ can be written as a sum of logarithms of factorials . So, we can use the series expansion from Equation 2 up to $\mathcal{O}(\log n)$ terms.

Not obvious how

Now, we can exponentiate the approximation to obtain $p_d \cdot e^{\mathcal{O}(1/n^2)} \approx p_d\left(1 + \mathcal{O}(1/n^2)\right)$ □

A.3 Approximating Close-to-Central Binomial Coefficients

This immediately gives us an asymptotic formula for the central binomial coefficient as:

$$\binom{n}{n/2} = \sqrt{\frac{2}{\pi n}} 2^n \left(1 + \mathcal{O}\left(\frac{1}{n}\right)\right)$$

Cite Asymptopia

Now, we consider a "off-center" Binomial coefficient $\binom{n}{k}$ where $k = \frac{n+c\sqrt{n}}{2}$. We consider the ratio: $R = \binom{n}{k}/\binom{n}{n/2}$:

$$R = \frac{\binom{n}{k}}{\binom{n}{n/2}} = \frac{(n/2)!(n/2)!}{k!(n-k)!} = \prod_{i=1}^{c\sqrt{n}/2} \frac{n/2 - i + 1}{n/2 + i} \tag{3}$$

$$\implies \log R = \sum_{i=1}^{c\sqrt{n}/2} \log\left(\frac{n/2 - i + 1}{n/2 + i}\right) \tag{4}$$

$$= \sum_{i=1}^{c\sqrt{n}/2} -\frac{4i}{n} + \mathcal{O}\left(\frac{i^2}{n^2}\right) = -\frac{c^2 n}{2n} + \mathcal{O}\left(\frac{(c\sqrt{n})^3}{n^2}\right) = -\frac{c^2}{2} + \mathcal{O}\left(\frac{c^3}{\sqrt{n}}\right) \tag{5}$$

$$\implies \binom{n}{k} = \binom{n}{n/2} e^{-c^2/2} exp\left(\mathcal{O}(c^3/\sqrt{n})\right) \tag{6}$$

A.4 Sampling the Height

fix

- $d < c \cdot \sqrt{S} \log n$
- $k < c \cdot \sqrt{S} \log n \implies U - D < c \cdot \sqrt{S} \log n$
- $k' < c \cdot \sqrt{S} \log n$
- $S > \log^2 n \implies \sqrt{S} \log n < S$

Lemma 14. *For $x < 1$ and $k \geq 1$,*

$$1 - kx < (1-x)^k < 1 - kx + \frac{k(k-1)}{2}x^2.$$

Lemma 4. *$D_{left} \leq c_1 \frac{k \cdot \log n}{\sqrt{S}} \cdot \binom{S}{D-d}$ for some constant c_1.*

Proof. This involves some simple manipulations.

$$D_{left} = \binom{S}{D-d} - \binom{S}{D-d-k} \tag{7}$$

$$= \binom{S}{D-d} \cdot \left[1 - \frac{(D-d)(D-d-1)\cdots(D-d-k+1)}{(S-D-d+k)(S-D-d+k-1)\cdots(S-D-d+1)}\right] \tag{8}$$

$$\leq \binom{S}{D-d} \cdot \left[1 - \left(\frac{D-d-k+1}{S-D+d+k}\right)^k\right] \tag{9}$$

$$\leq \binom{S}{D-d} \cdot \left[1 - \left(\frac{U+d+k-(U-D+d+k-1)}{U+d+k}\right)^k\right] \tag{10}$$

$$\leq \binom{S}{D-d} \cdot \left[1 - \left(\frac{U+d+k-\mathcal{O}(\log n\sqrt{S})}{U+d+k}\right)^k\right] \tag{11}$$

$$\leq \Theta\left(\frac{k\log n}{\sqrt{S}}\right) \cdot \binom{S}{D-d} \tag{12}$$

□

Lemma 5. *$D_{right} < c_2 \frac{k'\cdot logn}{\sqrt{S}} \cdot \binom{S}{U-d}$ for some constant c_2.*

Proof.

$$D_{right} = \binom{S}{U-d} - \binom{S}{U-d-k'} \tag{13}$$

$$= \binom{S}{U-d} \cdot \left[1 - \frac{(U-d)(U-d-1)\cdots(U-d-k'+1)}{(S-U-d+k')(S-U-d+k'-1)\cdots(S-U-d+1)}\right] \tag{14}$$

$$\leq \binom{S}{U-d} \cdot \left[1 - \left(\frac{U-d-k'+1}{S-U+d+k'}\right)^{k'}\right] \tag{15}$$

$$\leq \binom{S}{U-d} \cdot \left[1 - \left(\frac{2D-U-d-k+1}{2U-D+k+d}\right)^{k'}\right] \tag{16}$$

$$\leq \binom{S}{U-d} \cdot \left[1 - \left(\frac{U+k+d-(2U-2D+2d+2k-1)}{U+k+d}\right)^{k'}\right] \tag{17}$$

$$\leq \binom{S}{U-d} \cdot \left[1 - \left(\frac{U+k+d-\mathcal{O}(\log n\sqrt{S})}{U+k+d}\right)^{k'}\right] \tag{18}$$

$$\leq \Theta\left(\frac{k'\log n}{\sqrt{S}}\right) \cdot \binom{S}{U-d} \tag{19}$$

□

Lemma 15. $D_{tot} \geq \binom{2S}{2D} \cdot \left[1 - \left(1 - \frac{k'}{2U+1}\right)^k\right]$.

change statement

Proof.

$$D_{tot} = \binom{2S}{2D} - \binom{2S}{2D-k} \tag{20}$$

$$= \binom{2S}{2D} \cdot \left[1 - \frac{(2D)(2D-1)\cdots(2D-k+1)}{(2S-2D+k)(2S-2D+k-1)\cdots(2S-2D+1)}\right] \tag{21}$$

$$\geq \binom{2S}{2D} \cdot \left[1 - \left(\frac{2D-k+1}{2S-2D+1}\right)^k\right] \tag{22}$$

$$\geq \binom{2S}{2D} \cdot \left[1 - \left(\frac{2U-(2U-2D+k-1)}{2U+1}\right)^k\right] \tag{23}$$

$$\geq \binom{2S}{2D} \cdot \left[1 - \left(\frac{(2U+1)-k'}{2U+1}\right)^k\right] \tag{24}$$

$$\geq \binom{2S}{2D} \cdot \left[1 - \left(1 - \frac{k'}{2U+1}\right)^k\right] \tag{25}$$

$$\tag{26}$$

□

Reference previous lemma

Lemma 3. *When $kk' > 2U+1$, $D_{tot} > \frac{1}{2} \cdot \binom{2S}{2D}$.*

Proof. When $kk' > 2U+1 \implies k > \frac{2U+1}{k'}$, we will show that the above expression is greater than $\frac{1}{2}\binom{2S}{2D}$. Defining $\nu = \frac{2U+1}{k'} > 1$, we see that $(1-\frac{1}{\nu})^k \leq (1-\frac{1}{\nu})^\nu$. Since this is an increasing function of ν and since the limit of this function is $\frac{1}{e}$, we conclude that

$$1 - \left(1 - \frac{k'}{2U+1}\right)^k > \frac{1}{2}$$

□

Lemma 6. *When $kk' \leq 2U+1$, $D_{tot} < c_3 \frac{k \cdot k'}{S} \cdot \binom{2S}{2D}$ for some constant c_3.*

Proof. Now we bound the term $1 - \left(1 - \frac{k'}{2U+1}\right)^k$, given that $kk' \leq 2U+1 \implies \frac{kk'}{2U+1} \leq 1$. Using

Taylor expansion, we see that

$$1 - \left(1 - \frac{k'}{2U+1}\right)^k \tag{27}$$

$$\leq \frac{kk'}{2U+1} - \frac{k(k-1)}{2} \cdot \frac{k'^2}{(2U+1)^2} \tag{28}$$

$$\leq \frac{kk'}{2U+1} - \frac{k^2k'^2}{2(2U+1)^2} \tag{29}$$

$$\leq \frac{kk'}{2U+1}\left(1 - \frac{kk'}{2(2U+1)}\right) \tag{30}$$

$$\leq \frac{kk'}{2(2U+1)} \leq \frac{kk'}{\Theta(S)} \tag{31}$$

$$\tag{32}$$

□

A.5 First Return Sampling

Lemma 8. $D_{left}(d) = \Theta\left(\frac{2^{2d+k}}{\sqrt{d}} e^{-r_{left}(d)} \cdot \frac{k-1}{d+k-1}\right)$ *where* $r_{left}(d) = \frac{(k-2)^2}{8(2d+k-2)}$.

Proof. In what follows, we will drop constant factors: □

Lemma 9. $D_{right}(d) = \Theta\left(\frac{2^{n+m-2d-k}}{\sqrt{n}} e^{-r_{right}(d)} \cdot \frac{n-m+k}{n-d+1}\right)$ *where* $r_{right}(d) = \frac{(n-m-k-1)^2}{8(n+m-2d-k+1)}$.

Proof. □