Probability and Computing

Lecturers

Samuel Baguley, Andreas Goebel, Panagiotis Aivasiliotis

Notes

Simon Cyrani

Version
git: ab41269-*
compiled: Wednesday 24th April, 2024 14:49

Abstract

The following lecture notes are my personal (and therefore unofficial) write-up for 'Probability and Computing' aka 'ProbComp', which took place in summer semester 2024 at Hasso-Plattner-Institut. I do not guarantee correctness, completeness, or anything else. Importantly, note that I willfully changed some specific notations, reordered some material, and left out parts that I didn't found worth typing down.

If you miss something, feel free to contribute in the repository!

Contents

Summary of lectures

Part I

Lecture notes

Lecture 1
Mo 15 Apr 2024

1 Probabilistic method

This section will introduce how probability can be used to solve problems that at first glance do not have anything to do with probability. Consider following combinatorial problem.

Definition 1.1 (Ramsay numbers)**.** We define R_k as the smallest integer n such that any graph with n vertices must contain either a clique or an independent set of size K. We call R_k the (symmetric) **Ramsey numbers**.

Let us first look at some examples to get a feel for what this section is about.

Example 1.2. The first few Ramsay numbers are given as

- $R_2 = 2$: Obviously, $R_k \geq k$. Furthermore, there are only two different graphs with two nodes, i.e. with or without an edge. In the former case, both nodes form a clique, in the latter they form an independet set of size 2.
- $R_3 = 6$: First, let us show $R_3 \geq 6$. Consider a cycle of 5 nodes. Then, there is no clique of size 3 (since there is no 3-cycle). Also, among any three nodes two nodes are connected by an edge, so there is no independent set. Therefore, this is a counterexample. Now, consider a graph with 6 nodes. Suppose there is no clique of size 3. Then it suffices to show that there is an independent set of 3 nodes. Indeed, with an ugly case distinction this is possible: If there are no cycles, the graph is bipartite, so there is an independent set of at least 3. Otherwise, there exists at least a 4-cycle, but no 3-cycle (i.e. chord-free). We can then select at least two independent nodes from the cycle, and if needed the missing third node from the non-cycle nodes such that they form an independent set.
- $R_4 = 18$. Trust me, we do not want the proof here.
- $R_5 \in [43, 48]$. The exact value is indeed still unknown!

As we can see, even for small k, it is not trivial to determine their Ramsay number. Instead, let us try to at least find some bound for their value.

Theorem 1.3. For every $k \geq 1$ holds $R_k > 2^{k/2}$.

Proof. Consider a uniform distribution over all graphs with n vertices (i.e. the Erdős–Rényi random graph $\mathcal{G}(n, \frac{1}{2})$). Each edge in particular exists with probability $\frac{1}{2}$. Now, have a look at $p \coloneqq P(G \sim \mathcal{G}(n, \frac{1}{2})$ has a k-clique or k independent set). If we can show that this probability p is less than 1, then this means there is a graph with n vertices such that the property is *not* satisfied, and therefore $R_k > n$.

Let S be a k-tuple of vertices. S is per definition a k-clique if *all* or *none* of its edges is existent. Therefore, its probability of being either one is given as

$$P(S \text{ is } k\text{-clique or } k\text{-independent set}) = 2 \cdot \frac{1}{2^{\binom{k}{2}}}. \tag{1}$$

The total number of k-subsets given n vertices is given as $\binom{n}{k}$, so by basic properties of probability and binomial coefficients we see

$$p \leq \binom{n}{k} \cdot \frac{1}{2^{\binom{k-1}{2}}} \leq \frac{n^k}{k!} 2^{1-\frac{k^2-k}{2}} \tag{2}$$

For $n = 2^{k/2}$ the right-hand side reduces to $\frac{2^{k+2}}{k!}$, which can be shown easily to be smaller than 1 for $k \geq 3$. ☐

Notice how we suddenly imposed a probabilistic view on this presumably deterministic problem! This technique of using a suitable random model to demonstrate the existence and/or non-existence of certain properties is known as **Probabilistic Method**.

Before we have a look at another example, we need following lemma.

Lemma 1.4. Let X be a discrete random variable over a set Ω with $\mathbb{E}[X] = \mu$. Then, $P(X \geq \mu) > 0$ and $P(X \leq \mu) > 0$.

Proof. Assume $P(X \geq \mu) = 0$. Then $P(X = x) = 0$ for $x \geq \mu$. By definition and assumption therefore

$$\mathbb{E}[X] = \sum_{x \in X(\Omega)} xP(X = x) = \sum_{x < \mu} xP(X = x) \tag{3}$$

$$< \sum_{x < X(\Omega)} \mu P(X = x) = \mu \sum_{x \in X(\Omega)} P(X = x) = \mu. \tag{4}$$

This is a contradiction! ☐

Consider following problem.

Definition 1.5 (Max-Cut)**.** Given a graph $G = (V, E)$, find a set A maximizing the number of edges between A and $V \setminus A$. We call this the **Maximum Cut Problem**.

As it is usual in lectures of this kind, its canonical decision variant is indeed a NP-complete problem. Again, let us try instead to find a "good" cut.

Theorem 1.6 (Minimal Max-Cut)**.** Given a graph $G = (V, E)$ with $|E| = m$. There exists $A \subseteq V$ with at least $\frac{m}{2}$ cut size.

Proof. Choose A uniformly over $\mathcal{P}(V)$, i.e. every node is chosen with probability $\frac{1}{2}$. Then, every edge is with probability $\frac{1}{2}$ included in the cut, which happens iff exactly one of the nodes of the edge is in A. Let X be the number of cut edges, and X_e be the indicator variable for e being in the cut. By linearity of expectancy,

$$\mathbb{E}[X] = \mathbb{E}\left[\sum_{e\in E} X_e\right] = \sum_{e\in E} \mathbb{E}[X_e] = m \cdot \frac{1}{2}. \tag{5}$$

Using Lemma 1.4 there is positive probability for a choice of A having cut size at least $m/2$. □

You might have noticed that these are non-constructive results, so naturally following question emerges: Can we use these results to construct a solution? Indeed, there is a simple answer!

Definition 1.7 (Las-Vegas Algorithm)**.** Let T be the run-time of an algorithm and n its input size. We call a **Las-Vegas Algorithm** an algorithm which

1. always returns the correct answer with $\mathbb{E}[T] \in \mathcal{O}(n^k)$ for some k, or
2. has runtime always in $\mathcal{O}(n^k)$ for some k, and returns a correct answer with probability at least $\delta > 0$ (and otherwise returns no answer).

Remark 1.8. Both definitions are equivalent, which stems from the fact the second variant can be seen as a geometric process: Consider the number T of attempts until the algorithm returns a correct result. Then, $\mathbb{E}[T] = \frac{1}{1-\delta}$, so the total runtime is still $\mathcal{O}(n^k)$ in expectancy.

Turning back to the Max-Cut problem, let us transform our result of Theorem 1.6 into a Las-Vegas algorithm. Simply generate A randomly as constructed in the proof, and check if it has enough cut edges in polynomial time. The probability that this works is $P(X_A \geq \frac{m}{2}) = p$. However, this is just an abstract value - let us try to find a more meaningful way of expressing p:

$$\frac{m}{2} = \mathbb{E}\left[X_A\right] = \sum_{i<\frac{m}{2}} i \cdot P(X_A = i) + \sum_{i\geq\frac{m}{2}} i \cdot P(X_A = i) \tag{6}$$

$$\leq \left(\frac{m}{2} - 1\right) \sum_{i<\frac{m}{2}} P(X_A = i) + m \sum_{i\geq\frac{m}{2}} P(X_A = i) = \left(\frac{m}{2} - 1\right)(1-p) + mp \tag{7}$$

Here we just upper-bound the corresponding first factors in each sum, and then use $P(X_A \geq \frac{m}{2}) = p$. Using some easy algebra, we deduce $p \geq \frac{1}{\frac{m}{2}+1}$, so our Las-Vegas approach seems to get gradually worse the more edges our graph has.

Interestingly, we do not even need a randomized algorithm to find a big cut. Instead, using probabilistic arguments we can construct a deterministic algorithm still running in polynomial time. This technique is known as **Derandomization**.

Algorithm 1: Find Big-Cut of $G = (V, E)$

$A \leftarrow \emptyset, B \leftarrow \emptyset$
for $k = 1, \dots, n$ **do**
  if $|\{(v_k, u) \in E \mid u \in A\}| \leq |\{(v_k, u) \in E \mid u \in B\}|$ **then**
    $A \leftarrow A + v_k$
  end
  else
    $B \leftarrow B + v_k$
  end
end
return A

The idea is to greedily decide for each vertex if we want it in our cut set or not based on a case distinction using conditional expectation.

Theorem 1.9. Given a graph $G = (V, E)$. Let $A \sim \mathcal{U}_{\mathcal{P}(V)}$, C_A the amount of cuts, and $x_1, \dots, x_n$ indicate if the vertices $v_1, \dots, v_n \in A$. Then Algorithm 1 satisfies following statements:

1. $\mathbb{E}[C_A \mid x_1, \dots, x_k] \leq \mathbb{E}[C_A \mid x_1, \dots, x_k, x_{k+1}]$ after iteration $k+1$ of the for-loop (such that x_i is fixed by the algorithm).
2. The algorithm runs in $\mathcal{O}(n + m)$ and returns a big cut of size at least $m/2$.

Proof. Notice for $1 \leq k < n$ by definition of conditional expectancy

$$\mathbb{E}[C_A \mid x_1, \dots, x_k] = \frac{1}{2}\mathbb{E}[C_A \mid x_1, \dots, x_k, x_{k+1} = 1] + \frac{1}{2}\mathbb{E}[C_A \mid x_1, \dots, x_k, x_{k+1} = 0].$$

So, at least one of the choices for x_{k+1} satisfy the required lower bound for the conditional expectancy. It remains to prove that our algorithm also chooses this value, i.e. the choice with larger condtional expectancy. Let us observe how the expected number of cut edges can change if we fix the position of vertex v_{k+1}. Consider following cases for an edge $e \in E$:

- e does not contain v_{k+1}. Then, by indepence, the expected value of its Is-Cut-Edge indicator X_e conditioned on all fixed vertices does not change by fixing v_{k+1} (i.e 1 or 0 if both vertices are determined, else $\frac{1}{2}$).
- e contains v_{k+1}, but the other vertex is not fixed. Again, aforementioned expected value does not change since there is still a $\frac{1}{2}$ probability for the other vertex being in A.
- e contains v_{k+1}, but the other vertex *is* fixed. Then we suddenly fix the value of X_e depending on the choice of x_{k+1}. In particular, this changes the conditional expectancy of X_e, increasing it to 1 or decreasing it to 0, and therefore the conditional expectancy on the number of cut edges changes in the same way.

In summary, the change in conditional expectancy by fixing x_{k+1} only depends on the neighborhood of fixed vertices of v_{k+1}. If there are more neighbors that are fixed to be A than not in A, then

$$\mathbb{E}\left[C_A \mid x_1, \ldots, x_k, x_{k+1} = 0\right] \geq \mathbb{E}\left[C_A \mid x_1, \ldots, x_k, x_{k+1} = 1\right],$$

otherwise the other way around, adhering to our algorithm design.

In fact, by induction it immediately follows that $\frac{m}{2} \leq \mathbb{E}\left[X_A\right] \leq \mathbb{E}\left[X_A \mid x_1, \ldots, x_n\right]$ for $x_1, \ldots, x_n$ being chosen according to the algorithm. So, our algorithm works, and has mentioned runtime. □ Lecture 2
Mi 17 Apr 2024

2 Probabilistic method 2

Definition 2.1 (SAT problem)**.** Given a boolean formula in k-CNF (conjunctive normal form). The decision problem if there is an assignment that the formula is true is called k-SAT.

Assume we have a k-SAT instance φ, and assume each variable appears in exactly one clause. Then it is possible to show $\varphi \in \mathsf{SAT}$.

Example 2.2. For 3-SAT consider $(x_1 \vee x_2 \vee \neg x_3) \wedge (x_4 \vee \neg x_5 \vee x_6)$. Then, a valid assignment is $x_1 = x_2 = x_3 = 1, x_4 = x_5 = x_6 = 0$.

It is rather intuitive this statement must hold, since we enforce some form of indepence. This makes it interesting to look at from the lense of the probability, and its notion of indepence.

So, let us assume a uniform random assignment α for such a φ (with n clauses). Let E_i be the event that clause i is not satisfied by α. Then, $\bigcup E_i$ is the event that φ is not satisfied by α, and thus $\bigcap \overline{E_i}$ denotes φ being satisfied by α. Similarly to previous applications of the probabilistic method we see that

$$P\left(\bigcap_{i=1}^{n} \overline{E_i}\right) = \prod_{i=1}^{n} P(\overline{E_i}) = \prod_{i=1}^{n} 1 - 2^{-k} > 0 \tag{8}$$

shows that there *are* assignments that satisfy α. However, we were now able to use indepence of E_i in this case.

While this example does not seem too spectacular at first, we can now try to leviate our conditions and only assume limited independence. Let us introduce a new construct.

Definition 2.3 (Dependency graph)**.** Consider a set of events $E_1, \ldots, E_n$. We call $G = (V, E)$ with $V = \{1, \ldots, n\}$ the **Dependency Graph** if E satisfies that for every E_i it holds that $\{E_j \mid (i,j) \neq E_i\}$ are mutually independent.

Lemma 2.4. If events $E_1, \ldots, E_n$ are mutually independent, then its counterparts $\overline{E_1}, \ldots, \overline{E_n}$ are mutually independent.

Using these notions, we can now introduce a rather strong tool!

Lemma 2.5 (Symmetric **Loràsz Local Lemma**)**.** Let $E_1, \ldots, E_n$ be a set of events and assume

1. $P(E_i) \leq p$ for some fixed p,
2. the maximum degree of the dependency graph of these events is d, and
3. $4dp \leq 1$.

Then $P(\bigcap_{i=1}^{n} \overline{E_i}) > 0$.

The proof is rather technical, but the main idea is to use two nested inductions, and rewrite our result as a fancy product using conditional probabilities.

Proof. Let $S \subseteq \{1, \ldots, n\}$. We will show by induction on the size s of S that for all $k \notin S$ it holds

$$P(E_k \mid \bigcup_{j \in S} \overline{E_j}) \leq 2p. \tag{9}$$

For the base case $s = 0$ by assumption $P(E_k) \leq p \leq 2p$.

For the induction step we first need to show $P(\bigcap_{j \in S} \overline{E_j}) > 0$. Again, we use another induction. For $s = 1$ this is clear by assumption. (Notice $p < 1$). Now, w.l.o.g. let $S = \{1, \ldots, s\}$. By definition of conditional expectancy,

$$\begin{aligned} P(\bigcap_{j \in S} \overline{E_j}) &= P(\overline{E_s} \mid \bigcap_{j=1}^{s-1} \overline{E_j}) \cdot P(\bigcap_{i=1}^{s-1} \overline{E_j}) \\ &= \underbrace{\left(1 - P(E_s \mid \bigcap_{j=1}^{s-1} \overline{E_j})\right)}_{>0 \text{ by outer induction}} \cdot \underbrace{P(\bigcap_{i=1}^{s-1} \overline{E_j})}_{>0 \text{ by inner induction}} > 0 \end{aligned}$$

This concludes the inner induction, so let us continue with the outer induction step. Let us split S into $S_1 = \{j \in S \mid (k,j) \in E\}, S_2 = S \setminus S_1$. Denote with $F_k = \bigcup_{i=1}^{k} \overline{E_k}$. Then

$$P(E_k \mid F_s) = \frac{P(E_k \cap F_s)}{P(F_s)} = \frac{P(E_k \cap F_{S_1} \cap F_{S_2})}{P(F_{S_1} \cap F_{S_2})} = \frac{P(E_k \cap F_{S_1} \mid F_{S_2})}{P(F_{S_1} \mid F_{S_2})}$$

Consider both parts of the final fraction, and let us bound them:

$$
\begin{aligned}
&P(E_k \cap F_{S_1} \mid F_{S_2}) \leq P(E_k \mid F_{S_2}) \leq p \\
&\quad P(F_{S_1} \mid F_{S_2}) = 1 - P(\overline{F_{S_1}} \mid F_{S_2}) = 1 - P(\bigcup_{i \in S_1} E_i \mid F_{S_2}) \\
&\qquad\qquad \geq 1 - \sum_{i \in S_1} P(E_i \mid \bigcap_{j \in S_2} \overline{E_j}) \geq 1 - |S_1| \cdot 2p \geq 1 - 2pd \geq \frac{1}{2}
\end{aligned}
$$

Applying these results yields $2p$ as an upper bound for $P(E_k \mid F_s)$, which also concludes the outer induction.

Continuing with the actual statement, we get

$$
P(\bigcap_{i=1}^{n} \overline{E_i}) = P(\overline{E_n} \mid \bigcap_{i=1}^{n-1} \overline{E_i}) \cdot P(\bigcap_{i=1}^{n-1} \overline{E_i}) = \prod_{i=1}^{n} P(\overline{E_i} \mid \bigcap_{j=1}^{i-1} \overline{E_i}) \geq (1-2p)^n > 0.
$$

□

Turning back to our k-SAT problem, we are now able to show a stronger version.

Theorem 2.6. Given a k-SAT instance φ (with n clauses), and assume that no variable appears in more than $\frac{2^k}{4k}$ clauses. Then, $\varphi \in \mathsf{SAT}$.

Proof. We motivate Lemma 2.5. Firstly, we see $P(E_i) \leq 2^{-k}$. For each vertex of the dependency graph its degree is at most $\frac{2^k}{4k} \cdot k$. Therefore, $4 \cdot 2^{-k} \cdot \frac{2^k}{4k} k \leq 1$, and $P(\bigcap_{i=1}^{n} \overline{E_i}) > 0$ implies the existence of a valid assignment. □

Some more applications.

Theorem 2.7. Assume n pairs of vertices need to be connected using n disjoint paths on a given network E. Each pair i can choose from a collection F_i of m paths. If any path in F_i shares edges by at most k paths in F_j and $\frac{8nk}{m} \leq 1$, then we can always choose an edge-disjoint collection of paths.

Proof. Consider a probability space where every pair i chooses a path in F_i uniformly distributed, i.e. with probability $1/m$. We define $E_{i,j}$ as the "bad" event that paths i, j share any edges, which occurs with probability k/m. The degree of the corresponding dependency graph then is $2n$. By Lemma 2.5, we are done.

□ Lecture 3
Mo 22 Apr 2024

3 Random graphs

In this section we want to try answer following question.

Question 3.1. What does the "average" graph look like?

We can introduce two notions for distributions over graphs.

Definition 3.2.
1. The $\mathcal{G}_{n,m}$ model is a probability distribution over $\mathcal{G}$ given by a uniform distribution over all graphs with n nodes and m edges.
2. The $\mathcal{G}_{n,p}$ model is a probability distribution over $\mathcal{G}$ for graphs with n nodes such that the existence of every edge is drawn with probability p.

However, for our goals it is easier to work with the second notion. If $G \sim \mathcal{G}_{n,p}$, then $\mathbb{E}\left[|E(G)|\right] = \binom{n}{2}p$.

Lemma 3.3. For all $G \sim \mathcal{G}_{n,p}, G' \sim \mathcal{G}_{n,m}$ it holds that for any graph H

$$P(H = G \mid |E(H)| = m) = P(H = G') \tag{10}$$

Proof. Using some simple transformations and thinking about the probabilities of our graphs we see

$$\begin{aligned} P(H = G \mid E(H) = m) &= \frac{P(H = G \cap E(H) = m)}{P(E(H) = m)} = \frac{P(H = G)}{P(E(H)) = m} \\ &= \frac{p^m(1-p)^{\binom{n}{2}-m}}{\binom{\binom{n}{2}}{m}} p^m (1-p)^{\binom{n}{2}-m} \\ &= \frac{1}{\binom{\binom{n}{2}}{m}} = P(H = G') \end{aligned}$$

□

This gives us the ability to try prove well-known graph problems on random graphs. For example, we can ask outselves if $G \sim \mathcal{G}_{n,p}$ contains K_4. Consider $C \subseteq V(G)$ such that $|C| = 4$, and let X_c be the indicator variable if $G[C] = K_4$. Furthermore, let X be the numbe of 4-cliques in G.

We easily see that $P(X_C = 1) = p^6 = \mathbb{E}\left[X_C\right]$, and $\mathbb{E}\left[X\right] = \binom{n}{4}p^6 \in \theta(n^4p^6)$. Therefore

- $p << n^{-2/3}$ implies $\mathbb{E}\left[X\right] \longrightarrow_{n\to\infty} 0$, and
- $p >> n^{-2/3}$ implies $\mathbb{E}\left[X\right] \longrightarrow_{n\to\infty} \infty$.

What happens though in the case of equality, i.e $p(n) = n^{-2/3}$?

Definition 3.4. We call $f(n)$ a **Threshold** for a property Q in $\mathcal{G}_{n,p}$ if

- $p >> f(n)$ implies $P(G \sim \mathcal{G}_{n,p} \text{ has } Q) \longrightarrow_{n\to\infty} 1$,
- $p << f(n)$ implies $P(G \sim \mathcal{G}_{n,p} \text{ has } Q) \longrightarrow_{n\to\infty} 0$.

Let us show $p(n) = n^{-2/3}$ is a threshold for the existence of a 4-clique.

1. Case $p << n^{-2/3}$: Then using Markov's inequality we immediately see

$$P(X > 0) = P(X \geq 1) \leq \frac{\mathbb{E}[X]}{1} \longrightarrow_{n\to\infty} 0. \tag{11}$$

2. Case $p >> n^{-2/3}$: Then using Tschebychev's inequality we see

$$P(X = 0) \leq P(|X - \mathbb{E}[X]| \geq \mathbb{E}[X]) \leq \frac{\mathbb{V}[X]}{\mathbb{E}[X]^2}. \tag{12}$$

Having a closer at the variance by smart reordering, we conclude

$$\begin{aligned}
\mathbb{V}[X] = \mathbb{E}\left[X^2\right] - \mathbb{E}[X]^2 &= \mathbb{E}\left[\left(\sum_C X_C\right)^2\right] - \mathbb{E}\left[\sum_C X_C\right]^2 \\
&= \sum_C \left(\mathbb{E}\left[X_C^2\right] - \mathbb{E}[X_C]^2\right) + \sum_{C\neq D} \left(\mathbb{E}[X_C X_D] - \mathbb{E}[X_C]\,\mathbb{E}[X_D]\right) \\
&= \sum_C \mathbb{V}[X_C] + \sum_{C\neq D} \mathbf{Cov}[X_C, X_D]
\end{aligned}$$

It suffices to show that both sums independently tend to 0 for $n \to \infty$ if divided by $\mathbb{E}[X]^2$ as seen in (12).

Notice $\mathbb{V}[X_C] = \mathbb{E}\left[X_C^2\right] - \mathbb{E}[X_C]^2 \leq \mathbb{E}\left[X_C^2\right] = p^6$, so taken over all $\binom{n}{4}$ instances of C the first sum has its upper bound in $\Theta(n^4 p^6) \subseteq \Theta(n^8)$. Since $\mathbb{E}[X]^2 = p^{12} >> n^{-8}$, the first sum converges indeed to 0 for $n \to \infty$.

For the second sum, we need a case distinction over the overlap of C and D. Notice that we only need to consider $\mathbb{E}[X_C X_D] \geq \mathbf{Cov}[X_C, X_D]$.

- $|C \cap D| \leq 1$: Then $\mathbf{Cov}[X_C, X_D] = 0$.
- $|C \cap D| = 2$: Then $\mathbb{E}[X_C X_D] = P(X_C X_D = 1) = p^{11}$. This happens $\binom{n}{6}\binom{6}{4}\binom{4}{2} \in \Theta(n^6)$-times.
- $|C \cap D| = 3$: Then $\mathbb{E}[X_C X_D] = p^9$. This happens $\binom{n}{5}\binom{5}{4}\binom{4}{3} \in \Theta(n^5)$-times.

Analoguously, this concludes the convergence.

Another interesting property is the largest connected component of a random graph.

Theorem 3.5. For $G \sim \mathcal{G}_{n,p}$ it holds that $f(n) := \frac{1}{n}$ is a threshold for G having a connected component of size $\Theta(n)$. Furthermore, for $p = \frac{c}{n}$, it holds that

$$\text{Largest connected component is} \begin{cases} \Theta(\log n), & c < 1 \\ \Theta(n^{\frac{2}{3}}), & c = 1 \\ \Theta(n), & c > 1 \end{cases}$$

Lecture 4
Mi 24 Apr 2024

Theorem 3.6. In almost every $G \sim \mathcal{G}_{n,\frac{1}{2}}$, the largest clique has size approximately $2\log_2(n)$.

Proof sketch. Let X_k be the number of k-cliques in $G \sim \mathcal{G}_{n,\frac{1}{2}}$. As previously shown for 4-cliques, we easily generalize

$$g(k) := \mathbb{E}\left[X_k\right] = \binom{n}{k} 2^{-\binom{k}{2}}.$$

Let $K_0(n)$ be the largest k such that $g(k) \geq 1$. If we show $K_0(n) \approx 2\log n$, then $g(k) \approx 2^{k\log n - k^2/2}$. Furthermore, let c be a constant (to be determined later), and

$$K_1(n) := K_0(n) - c, \qquad K_2(n) := K_0(n) + c. \tag{13}$$

We will show that

$$P(X_{K_1(n)} > 0) \xrightarrow{n\to\infty} 1, \tag{14}$$
$$P(X_{K_2(n)} > 0) \xrightarrow{n\to\infty} 0. \tag{15}$$

One can show (we just believe it) that

$$\mathbb{E}\left[X_{K_1}\right] \xrightarrow{n\to\infty} \infty, \tag{16}$$
$$\mathbb{E}\left[X_{K_2}\right] \xrightarrow{n\to\infty} 0. \tag{17}$$

Apparently following part gives us some intuition for that?

$$\frac{g(k+1)}{g(k)} = \frac{\binom{n}{k+1} 2^{-\binom{k+1}{2}}}{\binom{n}{k} 2^{-\binom{k}{2}}} = \frac{n-k}{k+1} \cdot 2^{-k}, \tag{18}$$

so for $k \approx 2\log n$ this is approximately

$$\frac{g(k+1)}{g(k)} \approx \frac{n}{2\log n} n^{-2} \xrightarrow{n\to\infty} 0. \tag{19}$$

Using Markov's inequality (**First Moment Method**) we conclude

$$P(X_{K_2(n)} \geq 1) \leq \frac{\mathbb{E}\left[X_{K_2}\right]}{1} \xrightarrow{n\to\infty} 0. \tag{20}$$

which shows (15). Using Tschebychev's inequality (**Second Moment Method**) we can show

$$P(X_{K_1(n)} = 0) \le P(|X_{K_1} - \mathbb{E}[X_{K_1}]| \ge \mathbb{E}[X_{K_1}]) \le \frac{\mathbb{V}[X_{K_1}]}{\mathbb{E}[X_{K_1}]^2} \xrightarrow{n\to\infty} 1. \tag{21}$$

To show the limit actually holds, we use the same decomposition trick as in : ref

$$\mathbb{V}[X_{K_1}] = \mathbb{V}\left[\sum_S X_S\right] = \sum_S \mathbb{V}[X_S] + \sum_{S \neq D} \mathbf{Cov}[X_S, X_D] \tag{22}$$

Let us define $S \vDash D$ if $|S \cap D| \ge 2$. We simplify further

$$\begin{aligned}
\sum_S \mathbb{V}[X_S] + \sum_{S\neq D} \mathbf{Cov}[X_S, X_D] &= \sum_S \mathbb{V}[X_S] + \sum_{S \vDash D} \mathbf{Cov}[X_S, X_D] \\
&\le \sum_S \mathbb{E}\left[X_S^2\right] + \sum_{S\vDash D} \mathbb{E}[X_S X_D] \\
&= \sum_S \mathbb{E}[X_S] + \sum_{S\vDash D} \mathbb{E}[X_S X_D] \\
&= \mathbb{E}[X_{K_1}] + \sum_{S\vDash D} \mathbb{E}[X_S X_D].
\end{aligned}$$

Notice that we divide this term by $\mathbb{E}[X_{K_1}]^2$ in (21), so the first summand of previous result reduces to $1/\mathbb{E}[X_{K_1}]$ which tends to 0 by (16).

For the second sum, using even more reordering and conditional probabilities

$$\begin{aligned}
\sum_{S\vDash D} \mathbb{E}[X_S X_D] &= \sum_{S\vDash D} P(X_S = 1 \cap X_D = 1) = \sum_{S\vDash D} P(X_S = 1 \mid X_D = 1) P(X_D = 1) \\
&= \sum_D P(X_D = 1) \left(\sum_{S:S\vDash D} P(X_S = 1 \mid X_D = 1) \right) \\
&= \sum_D P(X_D = 1) \left(\sum_{S:S\vDash D_0} P(X_S = 1 \mid X_{D_0} = 1) \right) \text{ for fixed } D_0 \\
&= \sum_{S:S\vDash D_0} P(X_S = 1 \mid X_{D_0} = 1) \cdot \sum_D P(X_D = 1) \\
&= \sum_{S:S\vDash D_0} P(X_S = 1 \mid X_{D_0} = 1) \cdot \mathbb{E}[X_{K_1}].
\end{aligned}$$

The factor vanishes after dividing by $\mathbb{E}[X_{K_1}]$, so we only remain with one last sum.

$$\sum_{S:S\vDash D_0} P(X_S = 1 \mid X_{D_0} = 1) = \sum_{i=2}^{K_1 - 1} 2^{-\binom{K_1}{2} + \binom{i}{2}} \binom{n - K_1}{K_1 - i} \binom{K_1}{i}$$

Dividing by the second $\mathbb{E}\left[X_{K_1}\right]$ using some binomial magic we conclude

$$\frac{\sum_{S:S\vDash D_0} P(X_S = 1 \mid X_{D_0} = 1)}{\mathbb{E}\left[X_{K_1}\right]} = \sum_{i=2}^{K_1-1} \underbrace{2^{\binom{i}{2}} \frac{\binom{K_1}{i}\binom{n-K_1}{K_1-i}}{\binom{n}{K_1}}}_{:=f(i)} \leq K_1 \max_{2\leq i\leq K_1-1} f(i) \tag{23}$$

For our previously introduced c, if chosen large enough, then the maximum is reached for $i = 2$, thus

$$f(2) = \frac{K_1!}{(K_1-2)!} \cdot \frac{K_1!}{(K_1-2)!} \cdot \frac{(n-K_1)!(n-K_1)!}{n!(n-2K_1-2)!} \approx \frac{K_1^2 K_1^2}{n^2}$$

and we upper bound (23) by

$$\frac{K_1^5}{n^2} \approx \frac{\log n^5}{n^2} \xrightarrow{n\to\infty} 0$$

showing (14) and concluding this wonderful proof. □

Let us find a notion for randomizing $k - \mathsf{SAT}$.

Conjecture 3.7. For all k there exists a threshold value $r_k^\star \in \mathbb{R}$ such that

$$P(\varphi_k(n,m) \in \mathsf{SAT}) \xrightarrow{n\to\infty} \begin{cases} 0, & r > r_k^\star \\ 1, & r < r_k^\star \end{cases}. \tag{24}$$

Note. We know that $3.52 \leq r_3^\star \leq 4.51$.

Part II

Appendix

A Exercise sheets

1. exercise sheet

Exercise 1.1. Placeholder.

Index