Randomized Algorithms

Lecture 10: the probabilistic method, ramsey numbers, and random graphs

Kousha Etessami

Theorem

[Ramsey,1928] *(a special case, for graphs) For any positive integer, k, there is a positive integer, n, such that in any undirected graph with at least n vertices:*

- *either there are k vertices that form a k-clique.*
- *or, there are k vertices that form a k-independent-set.*

For each integer $k \geq 1$, let $R(k)$ be the smallest such integer $n \geq 1$ such that every undirected graph with n or more vertices has either a k-clique or a k-independent-set as an induced subgraph.

The numbers $R(k)$ are called diagonal Ramsey numbers.

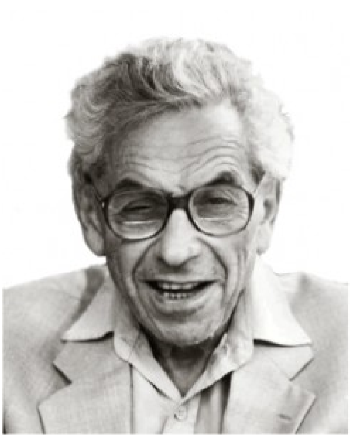
Proof of Ramsey's Theorem: Consider any integer $k \geq 1$, and any graph, $G_1 = (V_1, E_1)$ with at least $n = 2^{2k}$ vertices.

Initialize: $S_{Clique} := \{\}$; $S_{IndSet} := \{\}$;
for $i := 1$ to $2k - 1$ **do**
Pick any vertex $v_i \in V_i$;
if (v_i has at least 2^{2k-i} neighbors in G_i) **then**
$S_{Clique} := S_{Clique} \cup \{v_i\}$; $V_{i+1} := \{\text{neighbors of } v_i\}$;
else (* in case v_i has at least 2^{2k-i} non-neighbors in G_i *)
$S_{IndSet} := S_{IndSet} \cup \{v_i\}$; $V_{i+1} := \{\text{non-neighbors of } v_i\}$;
end if
Let $G_{i+1} = (V_{i+1}, E_{i+1})$ be the subgraph of G_i induced by V_{i+1};
end for

At the end, all vertices in S_{Clique} form a clique, and all vertices in S_{IndSet} form an independent set. Since $|S_{Clique} \cup S_{IndSet}| = 2k - 1$, either $|S_{Clique}| \geq k$ or $|S_{IndSet}| \geq k$. Q.E.D. □

Remarks on the proof, and on Ramsey numbers

- The proof establishes that $R(k) \leq 2^{2k} = 4^k$.
- **Question:** Can we give a better upper bound on $R(k)$?
- **Question:** Can we give a good lower bound on $R(k)$?



Paul Erdös (1913-1996)

Immensely prolific mathematician,
eccentric nomad,
father of the probabilistic method in combinatorics.

Lower bounds on Ramsey numbers: the birth of the Probabilistic Method

Theorem (Erdös,1947)

For all $k \geq 3$,

$$R(k) > 2^{k/2}$$

The proof uses the probabilistic method.

Lower bounds on Ramsey numbers: the birth of the Probabilistic Method

Theorem (Erdös,1947)

For all $k \geq 3$,

$$R(k) > 2^{k/2}$$

The proof uses the probabilistic method.

Recall the **general idea of the probabilistic method:** to show the existence of a hard-to-find object with a desired property, Q, try to construct a probability distribution over a sample space Ω of objects, and show that with positive probability a randomly chosen object in Ω has the property Q.

Random Graphs

Definition

The $G_{n,p}$ random graph model

A random graph $G = (V, E)$ sampled from $G_{n,p}$ is obtained as follows:

- G has $n = |V|$ nodes.
- For each of the $\binom{n}{2}$ possible pairs, $\{u, v\}$, with $u, v \in V$ and $u \neq v$, to determine whether or not $\{u, v\} \in E$, we flip an (independent) coin, which lands heads with probability p (and tails with probability $(1 - p)$). If it lands heads then $\{u, v\} \in E$; otherwise $\{u, v\} \notin E$.

Proof that $R(k) > 2^{k/2}$ using the probabilistic method:

Consider a random graph $G = (V, E)$ sampled from $G_{n,\frac{1}{2}}$.

(We will later determine that letting $n \leq 2^{k/2}$ suffices.)

Let $V = \{v_1, \ldots, v_n\}$. Note that for $v_i \neq v_j$, $\Pr(\{v_i, v_j\} \in E) = \frac{1}{2}$.

There are $\binom{n}{k}$ subsets of V of size k.

Let $S_1, S_2, \ldots, S_{\binom{n}{k}}$ be an enumeration of these subsets of V.

For $i = 1, 2, \ldots, \binom{n}{k}$, let E_i be the event that S_i forms either a k-clique or a k-independent-set in the graph. Note that:

$$\Pr(E_i) = 2 \cdot 2^{-\binom{k}{2}} = 2^{-\binom{k}{2}+1}$$

Proof of $R(k) > 2^{k/2}$ (continued):

Note that $E = \bigcup_{i=1}^{\binom{n}{k}} E_i$ is the event that there exists either a k-clique or a k-independent-set in the graph. But:

$$\Pr(E) = \Pr(\bigcup_{i=1}^{\binom{n}{k}} E_i) \leq \sum_{i=1}^{\binom{n}{k}} \Pr(E_i) = \binom{n}{k} \cdot 2^{-\binom{k}{2}+1}$$

Question: How small must n be so that $\binom{n}{k} \cdot 2^{-\binom{k}{2}+1} < 1$?

For $k \geq 2$: $$\binom{n}{k} = \frac{n(n-1)\ldots(n-k+1)}{k(k-1)\ldots 1} < \frac{n^k}{2^{k-1}}$$

Thus, if $n \leq 2^{k/2}$, then

$$\binom{n}{k} \cdot 2^{-\binom{k}{2}+1} < \frac{(2^{k/2})^k}{2^{k-1}} \cdot 2^{-\binom{k}{2}+1} = \frac{2^{k^2/2}}{2^{k-1}} \cdot 2^{-k(k-1)/2+1}$$

$$= 2^{\frac{k^2}{2}-k+1} \cdot 2^{-\frac{k^2}{2}+\frac{k}{2}+1} = 2^{-\frac{k}{2}+2}$$

Completion of the proof that $R(k) > 2^{k/2}$:

For all $k \geq 4$, $2^{-\frac{k}{2}+2} \leq 1$.

So, for $k \geq 4$, $\Pr(E) < 1$, and thus $P(\overline{E}) = 1 - P(E) > 0$.

But note that $P(\overline{E})$ is the probability that in a random graph of size $n \leq 2^{k/2}$, there is no k-clique and no k-independent-set.

Thus, since $\Pr(\overline{E}) > 0$, such a graph must exist for any $n \leq 2^{k/2}$.

Hence, $R(k) > 2^{k/2}$, for $k \geq 4$.

It is easy to argue "by hand" that $R(3) = 6$, and clearly $6 > 2^{3/2} = 2.828\ldots$.

Hence, for all $k \geq 3$, $\quad R(k) > 2^{k/2}$. $\square$

A randomized algorithm?

- The proof directly yields a randomize Monte Carlo algorithm for generating a random graph $G \sim G_{n,1/2}$ of size $n << 2^{k/2}$ which, with high probability, will have no k-clique and no k-independent set.
- However, checking whether a graph, G has a k-clique (or k-independent set), given both G and k as input, is **NP-complete**. So, we can't check it efficiently for large k.
- Hence, we have no way to convert this Monte Carlo algorithm to an efficient randomized Las Vegas algorithm that always produces a graph with no k-clique and no k-independent set.

Remarks on Ramsey numbers

- We have shown $2^{k/2} = (\sqrt{2})^k < R(k) \leq 4^k = 2^{2k}$.

Remarks on Ramsey numbers

- We have shown $$2^{k/2} = (\sqrt{2})^k < R(k) \leq 4^k = 2^{2k} .$$
- Despite decades of research by many combinatorists, nothing significantly better was known until very recently! In particular:

  no constant $c > \sqrt{2}$ is known such that $c^k \leq R(k)$, and

  no constant $c' < 4$ was known such that $R(k) \leq (c')^k$.

  Major breakthrough (‼) announced this year:
  [Campos,Griffiths,Morris, Sahasrabudhe,2023]: *There is a fixed constant $\epsilon > 0$ (specifically, $\epsilon = 2^{-7}$), such that for all sufficiently large k:* $$R(k) \leq (4 - \epsilon)^k .$$
- For specific small k, more is known:

$$R(1) = 1 \quad ; \quad R(2) = 2 \quad ; \quad R(3) = 6 \quad ; \quad R(4) = 18$$

$$43 \leq R(5) \leq 48$$

$$102 \leq R(6) \leq 165$$

$$\cdots$$

Why can't we just compute $R(k)$ exactly, for small k?

For each k, we know that $2^{k/2} < R(k) < 2^{2k}$,

So, for small fixed k, we could try to check, exhaustively, for each r such that $2^{k/2} < r < 2^{2k}$, whether there exists a graph G with r vertices such that G has no k-clique and no k-independent set.

Question: How many graphs on r vertices are there?

There are $2^{\binom{r}{2}} = 2^{r(r-1)/2}$ (labeled) graphs on r vertices.

So, for $r = 2^k$, we would have to check $2^{2^k(2^k-1)/2}$ graphs!!

So for $k = 5$, just for $r = 2^5$, we have to check 2^{496} graphs !!

Quote attributed to Paul Erdös:

> *Suppose an alien force, vastly more powerful than us, landed on Earth demanding to know the value of $R(5)$, or else they would destroy our planet.*

Quote attributed to Paul Erdös:

> *Suppose an alien force, vastly more powerful than us, landed on Earth demanding to know the value of $R(5)$, or else they would destroy our planet.*
>
> *In that case, I believe we should marshal all our computers, and all our mathematicians, in an attempt to find the value.*

Quote attributed to Paul Erdös:

> *Suppose an alien force, vastly more powerful than us, landed on Earth demanding to know the value of $R(5)$, or else they would destroy our planet.*
>
> *In that case, I believe we should marshal all our computers, and all our mathematicians, in an attempt to find the value.*
>
> *But suppose instead they asked us for $R(6)$.*

Quote attributed to Paul Erdös:

> *Suppose an alien force, vastly more powerful than us, landed on Earth demanding to know the value of $R(5)$, or else they would destroy our planet.*
>
> *In that case, I believe we should marshal all our computers, and all our mathematicians, in an attempt to find the value.*
>
> *But suppose instead they asked us for $R(6)$.*
>
> *In that case, I believe we should attempt to destroy the aliens.*

Maximum Satisfiability (MAXSAT)

A propositional boolean formula in Conjunctive Normal Form (CNF), is a conjunction of disjunctive clauses, where each disjunctive clause is a "Or" of literals: $\{x_1, \ldots, x_n\} \cup \{\neg x_1, \ldots, x_n\}$.
An example of a CNF formula looks something like this:

$$(x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_1 \vee \neg x_2) \wedge (x_1 \vee x_2 \vee x_3 \vee x_4) \wedge \ldots$$

The MAX-*k*-SAT problem: Given a CNF formula, φ, with n variables and m clauses, where each clause has at most k literals, what is the maximum number clauses that can be simultaneously satisfied by a true/false assigment to all the variables?

Theorem: MAX-*k*-SAT is NP-hard, for all $k \geq 2$. In fact, it is NP-hard even to approximate the maximum number of clauses within some constant factor (the constant depending on k when there are exactly k literals in each clause).

Theorem

Given a CNF boolean formula with m clauses, where each clause contains at least k literals, there exists a truth assignment to the variables that satisfies at least $m \cdot (1 - \frac{1}{2^k})$ clauses.

(In particular, note that this means that for a 3-CNF formula where every clause contains exactly 3 literals, there exists an assignment that satisfies a 7/8 fraction of the clauses.)

Proof: Randomly assign true or false, with probability 1/2 each, independently, to each of the n variables.

The probability that the i'th clause, with k_i literals, is satisfied is $(1 - \frac{1}{2^{k_i}})$. Hence, the expected total number of clauses that are satisfied (using linearity of expectation) is:

$$\sum_{i=1}^{m} (1 - 2^{-k_i}) \geq m(1 - 2^k). \qquad \square$$

- This proof can be converted to a randomized Las Vegas algorithm (with expected polynomial running time) for computing such a truth assignment that satisfies $7/8$ fraction of the clauses, when every clause has exactly 3 literals (MAX-E-3SAT).
- Furthermore, the algorithm can be derandomized, using the method of conditional expectations.

Astonishingly:

Theorem

[Hastad,2001] If for any $\epsilon > 0$ there exists a polynomial-time $(\frac{7}{8} + \epsilon)$-approximation algorithm for MAX-E-3SAT, then $\mathbf{P} = \mathbf{NP}$.

The proof (beyond the scope of this course) involves much of the deep theoretical developments behind the **PCP** ("Probabilitically Checkable Proof") characterization of **NP**.

References

- Chapter 6, sections 6.1-6.3 of [MU].
- We will continue with Chapter 6 and the probabilistic method next time.