Randomness in Computing

Lecture 1

Randomness in Computing

- Course information
- Verifying polynomial identities
- Probability Amplification
- Probability Review

Sofya Raskhodnikova
<table>
<thead>
<tr>
<th></th>
<th>Course information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Course staff</td>
</tr>
<tr>
<td>2.</td>
<td>Course website(s)</td>
</tr>
<tr>
<td>3.</td>
<td>Piazza bonus</td>
</tr>
<tr>
<td>4.</td>
<td>Prerequisites</td>
</tr>
<tr>
<td>5.</td>
<td>Textbook(s)</td>
</tr>
<tr>
<td>6.</td>
<td>Syllabus</td>
</tr>
<tr>
<td>7.</td>
<td>Homework logistics</td>
</tr>
<tr>
<td>8.</td>
<td>Collaboration policy</td>
</tr>
<tr>
<td>9.</td>
<td>Exams and grading</td>
</tr>
</tbody>
</table>

Sofya Raskhodnikova; Randomness in Computing
Tips for the course

- Concepts in this course take some time to sink in: be careful not to fall behind.
- Do the assigned reading on each topic before the corresponding lecture.
- Attend the lectures: some material will be presented on the blackboard (and some of it is not in the book).
- Attend the discussions: practice problem solving.
- Take advantage of office hours.
- Be active in lectures and on piazza.
- Allocate lots of time for the course: comparable to a project course, but spread more evenly.
Tips for the course: HW

• Start working on HW early.
• Spread your HW time over multiple days.
• You can work in groups (up to 4 people), but spend 1-2 hours thinking about it on your own before your group meeting.
Tips: learning problem solving

To learn problem solving, you have to do it:

• Try to think how you would solve any presented problem before you read/hear the answer.
• Do exercises in addition to HW.
Tips: how to read a math text

• Not like reading a mystery novel.
• The goal is not to get the answers, but to learn the techniques.
• Always try to foresee what is coming next.
• Always think how you would approach a problem before reading the solution.
• This applies to things that are not explicitly labeled as problems.
Skills we will work on

- Mathematical reasoning
- Expressing your ideas
  - abstractly (suppress inessential details)
  - precisely (rigorously)
- Probabilistic thinking
- Algorithmic thinking
- Problem solving
- Having **FUN** with all of the above!!!
Uses of Randomness in Computing

- To speed up algorithms.
- To enable new applications:
  - Symmetry breaking in distributed algorithms, cryptography, privacy, online games and gambling.
- To simulate real world events in physical systems: model them as happening randomly.
- To analyze algorithms when data is generated from some distribution:
  - learning theory, data compression.
- To analyze algorithms when errors happen randomly
  - error-correcting codes.
- Analyzing statistics from sampling.
Verifying Polynomial Identities

- \((x + 1)(x - 2)(x + 3)(x - 4)(x + 5)(x - 6) \equiv \_\_ x^6 - 7x + 37\)

**Task:** Given two polynomials \(F(x)\) and \(G(x)\), verify if \(F(x) \equiv G(x)\).

**Idea 1 (deterministic):** Convert both polynomials to canonical form \(\sum_{i=0}^{d} c_i x^i\).
- If \(F(x)\) is given as \(\Pi_{i=1}^{d} (x - a_i)\), conversion by consecutively multiplying monomials requires \(\Theta(d^2)\) multiplications of coefficients.

*Faster with Fourier Transform*
§1.1 (MU) Verifying Polynomial Identities

**Task:** Given two polynomials $F(x)$ and $G(x)$, verify if $F(x) \equiv G(x)$.

**Idea 2 (randomized):** Evaluate the polynomials on random integers.

Let $d = \text{max degree of } F(x) \text{ and } G(x)$

1. Pick $r$ uniformly from $\{1, ..., 100d\}$.
2. Compute $F(r)$ and $G(r)$.
3. reject if $F(r) \neq G(r)$; o.w. accept.

**Error Analysis: Probability of accepting incorrectly**

**Fundamental Theorem of Algebra**

A polynomial of degree $d$ has at most $d$ roots.

$$\text{Pr[error]} =$$
Probability space has three components

- Sample space $\Omega$
- Family of allowable events $E \subseteq \Omega$
- A probability function $\Pr$ that maps events $E$ to $\mathbb{R}$ such that
  - $\Pr(E) \in [0,1]$ for any event $E$;
  - $\Pr(\Omega) = 1$;
  - For any finite or countable sequence of pairwise disjoint events $E_1, E_2, \ldots$,\[ \Pr\left(\bigcup_{i \geq 1} E_i\right) = \sum_{i \geq 1} \Pr(E_i). \]
### Review: Inclusion-Exclusion Principle

#### For any two events $E_1$ and $E_2$,

$$\Pr(E_1 \cup E_2) = \Pr(E_1) + \Pr(E_2) - \Pr(E_1 \cap E_2).$$

#### For any three events $E_1, E_2$ and $E_3$,

$$\Pr(E_1 \cup E_2 \cup E_3) = \Pr(E_1) + \Pr(E_2) + \Pr(E_3) - \Pr(E_1 \cap E_2) - \Pr(E_1 \cap E_3) - \Pr(E_2 \cap E_3) + \Pr(E_1 \cap E_2 \cap E_3).$$

#### For any $n$ events $E_1, E_2, \ldots, E_n$,

$$\Pr\left(\bigcup_{i \in [n]} E_i\right) = \sum_{i \in [n]} \Pr(E_i) - \sum_{i < j} \Pr(E_i \cap E_j) + \sum_{i < j < k} \Pr(E_i \cap E_j \cap E_k) - \cdots + (-1)^{n+1} \Pr\left(\bigcap_{i \in [n]} E_i\right).$$
Union Bound

For finite or countable sequence of events $E_1, E_2, \ldots$,

$$\Pr\left(\bigcup_{i \geq 1} E_i\right) \leq \sum_{i \geq 1} \Pr(E_i).$$
Probability Amplification

- Our algorithm for verifying polynomial identities accepts incorrectly with probability $\leq \frac{1}{100}$

**Idea:** Repeat the algorithm and accept if all iterations accept.

$$\Pr[\text{error in all } k \text{ iterations}] \leq \left( \frac{1}{100} \right)^k$$
Independent events

- Two events $E_1$ and $E_2$ are independent if
  $$\Pr(E_1 \cap E_2) = \Pr(E_1) \cdot \Pr(E_2).$$
- Events $E_1, \ldots, E_n$ are mutually independent if, for every subset $I \in [n],$
  $$\Pr(\bigcap_{i \in I} E_i) = \prod_{i \in I} \Pr(E_i).$$
- If all pairs of events among $E_1, \ldots, E_n$ are independent then $E_1, \ldots, E_n$ are pairwise independent.
- Pairwise independence does not necessarily imply mutual independence!

$[n]$ denotes $\{1, 2, \ldots, n\}$.
Independence: Example

- We toss a fair coin twice.
- Let A be the event that the 1st flip is HEADS.
- Let B be the event that the 2nd flip is HEADS.
- Let C be the event that both flips are the same.

1. Are events A, B, C pairwise independent?
2. Are they mutually independent?
The conditional probability of event $E$ given event $F$ is

\[ \Pr(E \mid F) = \frac{\Pr(E \cap F)}{\Pr(F)}. \]

When $E$ and $F$ are independent,

\[ \Pr(E \mid F) = \frac{\Pr(E \cap F)}{\Pr(F)} = \frac{\Pr(E) \cdot \Pr(F)}{\Pr(F)} = \Pr(E). \]

Well defined only if $\Pr(F) \neq 0$.
Review question

- Event E: the numbers on two dice sum to 8.
- Event F: the numbers on two dice are both even.

What is the probability of E given that F occurred, \( \Pr(E|F) \)?

A. Less than 1/3
B. 1/3
C. Greater than 1/3, but less than 2/3
D. 2/3
E. Greater than 2/3
We deal two cards. What is the probability that the second card is an ace, given that the first is an ace?

A. $\frac{3}{52}$
B. $\frac{3}{51}$
C. $\frac{4}{52}$
D. $\frac{5}{52}$
E. None of the answers above are correct.
Product rule

For any two events $E_1$ and $E_2$,

$$\Pr(E_1 \cap E_2) = \Pr(E_1) \cdot \Pr(E_2 | E_1).$$

For all events $E_1, \ldots, E_n$,

$$\Pr(\cap_{i=1}^n E_i) = \Pr(E_1) \cdot \Pr(E_2 | E_1) \cdot \ldots \cdot (E_n | \cap_{i=1}^{n-1} E_i)$$
Sampling without replacement

- Let $E_i$ be the event that we choose a root in iteration $i$

$$\Pr[\text{error in all } k \text{ iterations}] = \Pr[E_1 \cap \cdots \cap E_k]$$

$$= \Pr[E_1] \cdot \Pr[E_2|E_1] \cdot \cdots \cdot \Pr[E_k|E_1 \cap \cdots \cap E_{k-1}]$$

- It is $0$ if $k > d$.
- If $k \leq d$, then

$$\Pr[E_j|E_1 \cap \cdots \cap E_{j-1}] = \frac{d - (j - 1)}{100d - (j - 1)}$$

$$\Pr[\text{error in all } k \text{ iterations}] \leq \left(\frac{1}{100}\right)^k$$