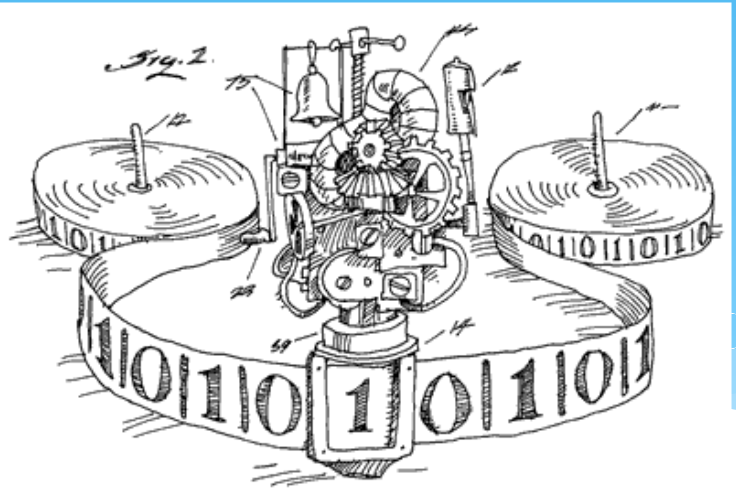
EECS 376: Foundations of Computer Science

Chris Peikert
27 March 2023



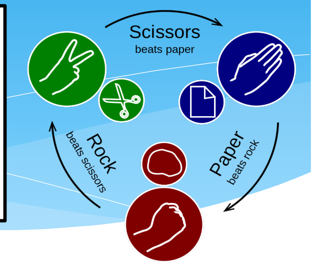
“God does not play dice with the universe.”
- Albert Einstein

“Wanna bet?”
- Quantum Mechanics

Randomized Algorithms

Example: Rock-Paper-Scissors

Moral: Randomization can sometimes help us avoid "worst-case" behavior.
(**Fact:** It can also enable things that are *impossible* deterministically!)



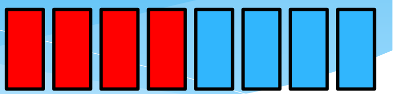
- **Goal:** Maximize our odds of *not losing* in a best-of-one game of rock-paper-scissors against an opponent that *knows our strategy*.
- **Idea:** play (uniformly) at *random*.
- **Analysis:** if opponent plays rock (other cases similar):

$$\begin{aligned}\Pr[\text{we lose} \mid \text{they play rock}] &= \Pr[\text{we play scissors} \mid \text{they play rock}] \\ &= \Pr[\text{we play scissors}] \\ &= 1/3\end{aligned}$$

So overall, $\Pr[\text{we lose}] = 1/3$.

Example: Cards

Moral: Probability lets us quantify how "unlucky" we are.
(worst-case vs. average-case vs. w/ high probability)



- $2n$ cards, n red and n blue, are shuffled, face-down
- **Goal:** Find a blue card by flipping cards over, one at a time, in any order we want.
- **Q:** How many flips do we need, *in the worst case*?
- **Q:** What if we *randomly* chose cards to flip?
  - How many flips do we need, *on average (expectation)*?
  - How many flips do we need, *99% of the time (w.h.p.)*?
- **Analysis:** geometric(-ish) distribution

Maximum 3CNF Satisfiability

* **Problem:** Given a 3CNF formula, find an assignment that satisfies the *maximum* number of clauses.
* **Example:** An *unsatisfiable* 3CNF formula, but any assignment satisfies *7 out of 8* clauses:

$$(x \vee y \vee z) \wedge (\neg x \vee y \vee z) \wedge (x \vee \neg y \vee z) \wedge (x \vee y \vee \neg z) \\ \wedge (\neg x \vee \neg y \vee z) \wedge (\neg x \vee y \vee \neg z) \wedge (x \vee \neg y \vee \neg z) \wedge (\neg x \vee \neg y \vee \neg z)$$

Theorem: There is an efficient algorithm that, given *any* 3CNF formula *with distinct variables in each clause*, outputs an assignment satisfying $\geq$ 7/8ths of clauses.
(***Expectation maximization***, a derandomization technique.)

Random Assignments

* Fix a 3CNF formula $\phi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ with m clauses, each of which contains *distinct* variables.
* **Claim:** If we pick a *random assignment* of ϕ, then we satisfy at least 7/8ths of the clauses, "on average".
* Let N be the number of satisfied clauses.
  * This is a *random variable*.
* **Goal:** Show that the *expected value* of N is $7m/8$.
* Let's first review these terms…

Review: Random Variables

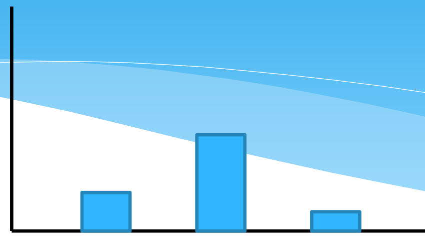
- A ***random variable*** is a *quantity* determined by the *outcome* of *a random experiment*.
- **Example:** Let N be the number of satisfied clauses of 3CNF formula ϕ when we generate an assignment of ϕ by assigning its variables T/F independently and uniformly at random.

$$\phi = (x \vee x \vee x) \wedge \left(y \vee y \vee y\right) \wedge \left(\neg x \vee \neg y \vee \neg z\right)$$

Note: this ϕ *doesn't* satisfy our theorem's hypothesis.

Outcome	FFF	FFT	FTF	FTT	TFF	TFT	TTF	TTT
Sat?	NNY	NNY	NYY	NYY	YNY	YNY	YYY	YYN
N	1	1	2	2	2	2	3	2

Review: Distribution of an RV



- The ***probability*** that a random variable equals some *fixed value* is the *sum of the probabilities* of all *outcomes that result in that value*.
- **Example:** N = number of satisfied clauses of a 3CNF formula ϕ when we generate an assignment by assigning its variables T/F independently and uniformly at random.

$$\phi = (x \vee x \vee x) \wedge \left(y \vee y \vee y\right) \wedge \left(\neg x \vee \neg y \vee \neg z\right)$$

Outcome	FFF	FFT	FTF	FTT	TFF	TFT	TTF	TTT
Sat?	NNY	NNY	NYY	NYY	YNY	YNY	YYY	YYN
N	1	1	2	2	2	2	3	2

$$\Pr[N=1] = \frac{2}{8} \qquad \Pr[N=2] = \frac{5}{8} \qquad \Pr[N=3] = \frac{1}{8}$$

EECS M ECE CSE

Review: Expected Value of an RV

- The ***expected value*** of a random variable is the *weighted average* of its values (value's weight = its probability):
- $$\mathbb{E}[N] = \sum_v v \cdot \Pr[N = v].$$
- **Example:** N = number of satisfied clauses of a 3CNF formula ϕ…

$$\Pr[N=1] = \frac{2}{8} \qquad \Pr[N=2] = \frac{5}{8} \qquad \Pr[N=3] = \frac{1}{8}$$

$$\mathbb{E}[N] = 1 \cdot \frac{2}{8} + 2 \cdot \frac{5}{8} + 3 \cdot \frac{1}{8} = \frac{1}{8}(1+1+2+2+2+2+3+2)$$

"*In expectation*, a random assignment satisfies 15/8 clauses of ϕ."

Analyzing $\mathbb{E}[N]$

- Fix *any* 3CNF formula $\phi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ with m clauses, each of which contains *distinct* variables.
- Suppose we generate a random assignment of ϕ (i.e., set its variables to T/F independently, uniformly at random).
- $N =$ number of clauses satisfied by the assignment
- **Goal:** Show that $\mathbb{E}[N] = 7m/8$.
- **Q:** How can we analyze $\mathbb{E}[N]$?
- **Very useful tricks:** *linearity of expectation* + *indicator random variables*

Review: 0/1 random variables

Useful Property: If Z is an indicator r.v., then
$$\mathbb{E}[Z] = 1 \cdot \Pr\left[Z = 1\right] + 0 \cdot \Pr\left[Z = 0\right] = \Pr[Z = 1]$$

- An ***indicator*** random variable is *always* either 0 or 1.
- **Example:** random assignment in 3CNF formula $\phi = C_1 \wedge C_2 \wedge \ldots \wedge C_m$.
  - For $1 \leq i \leq m$, let N_i be the indicator random variable for whether clause C_i is satisfied by the assignment.

$$\phi = (x \vee x \vee x) \wedge \left(y \vee y \vee y\right) \wedge \left(\neg x \vee \neg y \vee \neg z\right)$$

Assignment	FFF	FFT	FTF	FTT	TFF	TFT	TTF	TTT
N	1	1	2	2	2	2	3	2

Observation: The number of clauses satisfied by the assignment is $N = N_1 + N_2 + N_3 + \cdots + N_m$.

Review: Linearity of Expectation

Observation: The number of clauses satisfied by the assignment is $N = N_1 + N_2 + N_3 + \cdots + N_m$.

- ***Linearity of*** $\mathbb{E}$**:** for *any* (possibly dependent!) r.v.s N_i ,
- $$\mathbb{E}\Big[\sum_i N_i\Big] = \sum_i \mathbb{E}[N_i].$$
- **Example:** random assignment to 3CNF formula $\phi = C_1 \wedge C_2 \wedge \ldots \wedge C_m$.
  - For $1 \le i \le m$, let N_i be the indicator random variable for whether clause C_i is satisfied by the assignment.

$$\phi = (x \vee x \vee x) \wedge (y \vee y \vee y) \wedge (\neg x \vee \neg y \vee \neg z)$$

Outcome	FFF	FFT	FTF	FTT	TFF	TFT	TTF	TTT
N	1	1	2	2	2	2	3	2

$\mathbb{E}[N_1] = 1/2$ $\mathbb{E}[N_2] = 1/2$ $\mathbb{E}[N_3] = 7/8$

$$\mathbb{E}[N] = \mathbb{E}[N_1] + \mathbb{E}[N_2] + \mathbb{E}[N_3] = 15/8$$

Review: Independence

- An ***event*** is a set of outcomes.
- **Example:** Let V be an r.v. for the sum of two fair dice.
  - $V = 5$ is an event: the set of outcomes $\{(1,4),\ (2,3),\ (3,2),\ (4,1)\}$
  - $\Pr[V = 5] = 4/36 = 1/9$
- **Informally:** Two events are ***independent*** if the occurrence of one does not affect the probability of the other occurring.
- **Formal Definition:** Events A and B are ***independent*** if $\Pr[A \cap B] = \Pr[A] \cdot \Pr[B]$.
- A _collection of r.v.s_ $Z_1, \ldots, Z_n$ is ***independent*** if for _every_ $b_1, \ldots, b_n : \Pr[Z_1 = b_1,\ \ldots, Z_n = b_n] = \prod_i \Pr[Z_i = b_i]$.

Analyzing $\mathbb{E}[N]$

- Fix *any* 3CNF formula $\phi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ with m clauses, each of which contains *distinct* variables.
- Let r.v. $N = \sum N_i$ = #clauses satisfied by a random assignment, where N_i is the indicator r.v. for whether clause C_i is satisfied.
- **Claim:** $\mathbb{E}[N] = \mathbb{E}[N_1] + \mathbb{E}[N_2] + \cdots + \mathbb{E}[N_m] = 7m/8$
  - $\mathbb{E}\left[N_i\right] = \Pr\left[N_i = 1\right] = 1 - \Pr\left[N_i = 0\right]$

    $= 1 - \Pr\left[\ell_{i1} = 0, \ell_{i2} = 0, \ell_{i3} = 0\right]$ ($N_i = 0$ iff all of C_i's literals are false)

    $= 1 - \Pr\left[\ell_{i1} = 0\right] \bullet \Pr\left[\ell_{i2} = 0\right] \bullet \Pr\left[\ell_{i3} = 0\right]$ (independence: vars are *distinct*)

    $= 1 - (1/2)^3 = 7/8$
- Therefore, a random assignment satisfies 7/8ths of the clauses of ϕ *in expectation*, as claimed.

Quote of the Day #2

"Every student's score was above average"

– No professor ever

Fact: For any RV X there *exists* an outcome a s.t. $X(a) \geq \mathbb{E}[X]$.

Fact: For any RV X there exists an outcome b s.t. $X(b) \leq \mathbb{E}[X]$.

(Note: these don't imply anything about the *likelihood* of X being "above/below average (expectation).")

Averaging Argument

- Since a random assignment satisfies $\geq$ 7/8ths of the clauses *on average*, there *exists an assignment* that satisfies $\geq$ 7/8ths of the clauses.
  - **Analogy:** If the average money of a group of people is $100, then someone in that group has at least $100!
  - This is called an ***averaging argument***, or "probabilistic method."
- Similarly, *at least one of* $\mathbb{E}[N \mid$ first variable was set to $T]$ and $\mathbb{E}[N \mid$ first variable was set to $F]$ is $\geq 7m/8$.
  - $\mathbb{E}[N]$ is the *average* of the two expressions!
    (**Fact:** for any RV X and event A,
    $\mathbb{E}[X] = \Pr[A] \bullet \mathbb{E}\left[X \middle| A\right] + \Pr[\overline{A}] \bullet \mathbb{E}\left[X \middle| \overline{A}\right]$)

Derandomizing the Algorithm

- Fix any 3CNF formula $\phi = C_1 \wedge C_2 \wedge \ldots \wedge C_m$ on n variables $x_1, x_2, \ldots, x_n$, with distinct variables in each clause.
- We can *deterministically* build an assignment for ϕ by setting $x_i = a_i$ iteratively ($i = 1, \ldots, n$) as follows:
  - If $\mathbb{E}[N \mid x_1 = a_1, \ldots, x_{i-1} = a_{i-1}, x_i = T] \geq \mathbb{E}[N \mid x_1 = a_1, \ldots, x_{i-1} = a_{i-1}, x_i = F]$ then set $a_i = T$; Otherwise, set $a_i = F$.
    (Can compute these efficiently by linearity of expectation!)
- **Key:** Each step, we fix one variable to keep (for remaining vars) the expected number of satisfied clauses $\geq 7m/8$.

Theorem: There is an efficient *deterministic* algorithm that outputs an assignment satisfying 7/8ths of the clauses.

Markov's Inequality

- **Example:** The average score on the midterm was 60. What's the *maximum* fraction of students that could have a score of at least 90? (there are no negative scores)
  - 1/2? 2/3? 3/4? 99/100?
- ***Markov's Inequality:*** If X is a <u>*non-negative*</u> random variable and $a > 0$, then $\Pr[X \geq a] \leq \mathbb{E}[X]/a$.
  - **Proof:** choose a random student
  - X = score of student, a = some arbitrary score
  - $\mathbb{E}[X] \geq a\Pr[X \geq a]$. Divide by a.

How Often Does the Randomized Max-3SAT Algorithm "Do Well"?

* **Theorem:** There is an efficient randomized algorithm that, given any 3CNF formula ϕ with distinct variables in each clause, outputs an assignment that satisfies 7/8ths of the clauses, *in expectation*.
* **Q:** What is (a bound on) the probability that $\geq$ half of the clauses are satisfied?
* Let N be number of clauses satisfied. How to bound $\Pr\left[N \geq \frac{m}{2}\right]$?
* **Markov's Inequality:** If X is a non-negative random variable and $a > 0$, then $\Pr[X \geq a] \leq \mathbb{E}[X]/a$.
  * Therefore: $\Pr\left[N \geq \frac{m}{2}\right] \leq \left(\frac{7m}{8}\right)/\left(\frac{m}{2}\right) = 1.75\ldots$ unhelpful!
* What about # of *unsatisfied* clauses $N' = m - N \geq 0$?
  * $\Pr\left[N' \geq \frac{m}{2}\right] \leq \left(\frac{m}{8}\right)/\left(\frac{m}{2}\right) = \frac{1}{4}$, hence $\Pr\left[N > \frac{m}{2}\right] \geq 3/4$.

Verifying Matrix Multiplication

Goal: Given n-by-n matrices A, B, C, check whether $AB = C$.

Trivial: Compute AB, check if $AB = C$. Naïve matrix-mult time: $O(n^3)$.

Using randomization, can do it in $O(n^2)$ time! **Algorithm:**

* Choose a uniformly random vector r with each entry 0 or 1.
* Check if $A(Br) = Cr$.

Running time: $O(n^2)$. (Compute $v = Br$, then Av.)

Correctness: If $AB = C$, we accept with certainty.
Claim: If $AB \neq C$, then $\Pr[\text{accept}] \leq 1/2$. (Repeat to reduce!)

Proof of Claim

Claim: If $AB \neq C$, then $\Pr[ABr = Cr] \leq 1/2$.

Proof: Let $D = AB - C \neq 0$. (D does not have all-zero entries.)
We want to show that $\Pr[Dr \neq \mathbf{0}] \geq 1/2$.

Suppose (wlog) that column $D_1 \neq \mathbf{0}$.
Fix *any* choice of the entries $r_2, \ldots, r_n$ (so only random r_1 remains).

$$Dr = r_1 D_1 + \underbrace{r_2 D_2 + \cdots + r_n D_n}_{\text{fixed } z}.$$

Conclusion: Dr cannot be **0** for *both* $r_1 = 0$ and $r_1 = 1$. QED.