INFO 6205
Program Structure and Algorithms

Nik Bear Brown

Random Algorithms

Topics

Probability

Random variables

Permutations

Combinations

Tail Bounds

Bayesian Framework

Las Vegas Algorithms

Monte Carlo Algorithms

RP Class

Probability

- *Probability* is a measure of the likelihood of a random phenomenon or chance behavior. Probability describes the long-term proportion with which a certain outcome will occur in situations with short-term uncertainty.
- Probability is expressed in numbers between 0 and 1. Probability = 0 means the event never happens; probability = 1 means it always happens.
- The total probability of all possible event always sums to 1.

Probability

1. The probability of any event E, $P(E)$, must be between 0 and 1 inclusive. That is,

$$0 \leq P(E) \leq 1.$$

2. If an event is **impossible**, the probability of the event is 0.

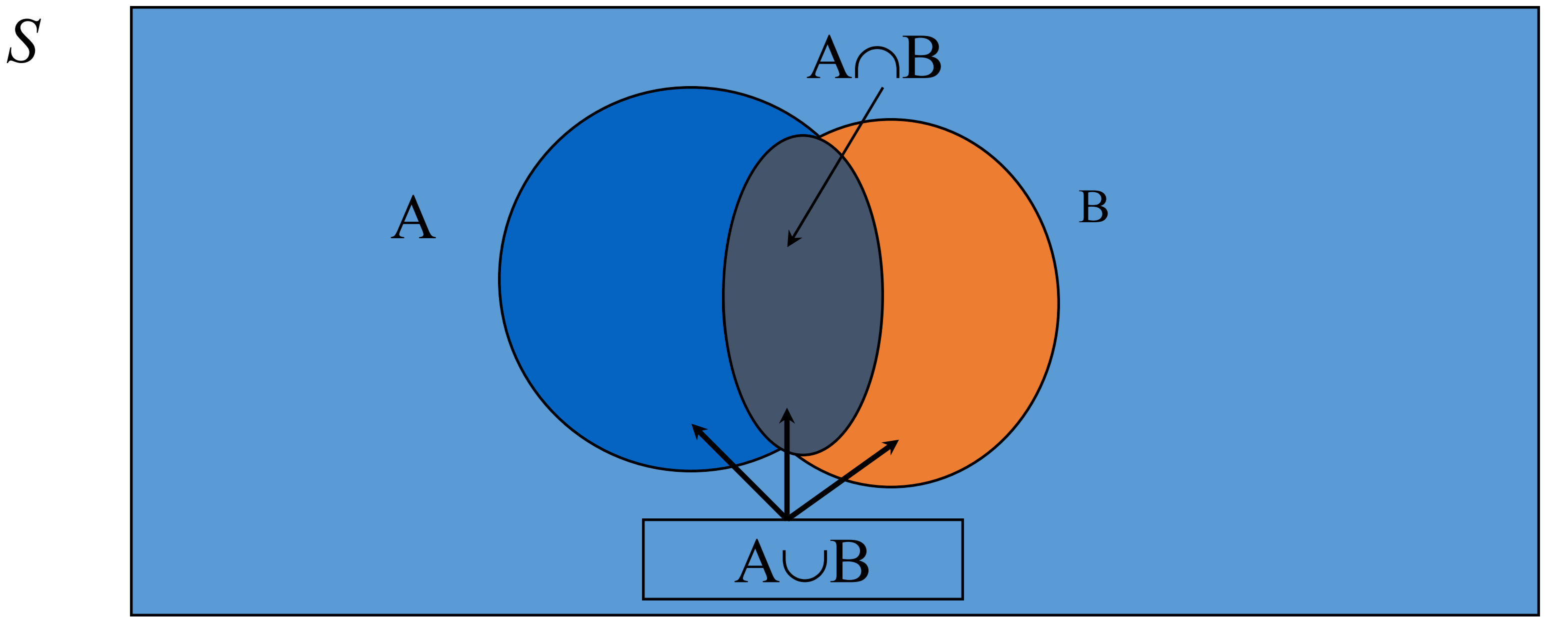
3. If an event is a **certainty,** the probability of the event is 1.

4. If $S = \{e_1, e_2, \ldots, e_n\}$, then

$$P(e_1) + P(e_2) + \ldots + P(e_n) = 1.$$

Unions and Intersections

AND Rule of Probability

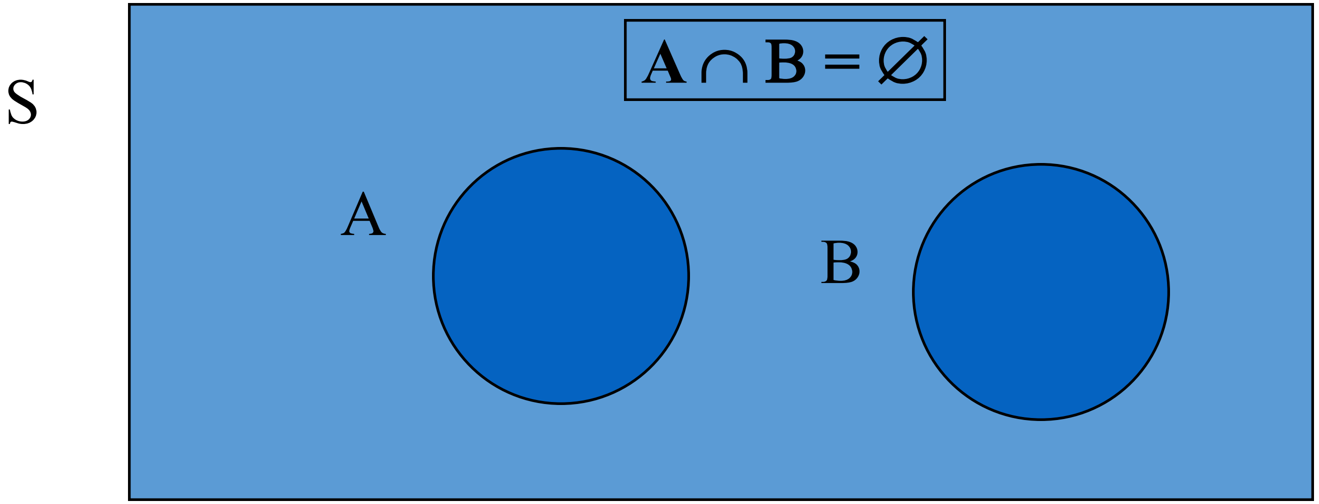


Probability - The OR Rule of Probability

- The probability that either one of 2 different events will occur is the sum of their separate probabilities.
- For example, the chance of rolling either a 2 or a 3 on a die is 1/6 + 1/6 = 1/3.

Mutually Exclusive Events

- The OR Rule of Probability
- Mutually exclusive events-no outcomes from *S* in common

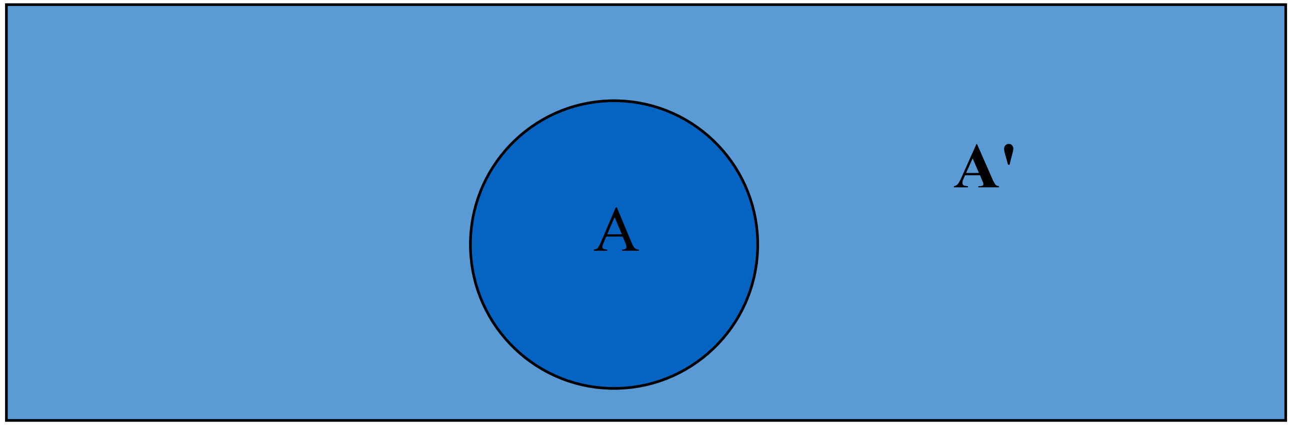


Probability - NOT Rule

- The chance of an event not happening is 1 minus the chance of it happening.
- For example, the chance of not getting a 2 on a die is 1 - 1/6 = 5/6.
- This rule can be very useful. Sometimes complicated problems are greatly simplified by examining them backwards.

P(A’) = 1 - P(A)

For an event A, A’ is the **complement of A**; A’ is everything in *S* that is not in A.



Probability

- if A and B are mutually exclusive events:

  P(A or B) = P(A) + P(B)

  ex., die roll: P(1 or 6) = 1/6 + 1/6 = .33

- possibility set:

  sum of all possible outcomes

  ~A = anything other than A

  P(A or ~A) = P(A) + P(~A) = 1

Probability

- one event has no influence on the outcome of another event
- if events A & B are independent

  then P(A&B) = P(A)*P(B)
- if P(A&B) = P(A)*P(B)

  then events A & B are independent
- coin flipping

  if P(H) = P(T) = .5 then

  P(HTHTH) = P(HHHHH) =

  $.5*.5*.5*.5*.5 = .5^5 = .03$

Random variables

Random variables assign a real number to each outcome:

$$\begin{aligned} X : \Omega &\to \mathbb{R} \\ \omega &\to X(\omega) \end{aligned}$$

Random variables can be:

Discrete: if it takes at most countably many values (integers).

Continuous: if it can take any real number.

Random variables

Distribution of a random variable $F(x) = F_X(x) = P(X \leq x)$

(i) $F(x) \to 0$ when $x \to -\infty$

(ii) $F(x) \to 1$ when $x \to +\infty$

(iii) $F(x)$ is nondecreasing.

$$x_1 < x_2 \Rightarrow F(x_1) \leq F(x_2)$$

(iv) $F(x)$ is right-continuous.

$F(x) \to F(x_0)$ when $x \to x_0$

$x > x_0$

Random variables

- For a random variable, we define
- Probability function
- Density function,

- depending on whether it is discrete or continuous

Random variables

Probability function

$$p(x) = p_X(x) = P(X = x)$$

verifies

$$(i)\ p(x) \geq 0$$

$$(ii) \sum_x p(x) = 1$$

Random variables

Probability density function

$$f(x)$$

verifies

$$(i) \quad f(x) \geq 0$$

$$(ii) \quad \int_{-\infty}^{+\infty} f(x)dx = 1$$

We have

$$F(x) = \int_{-\infty}^{x} f(t)dt \ \text{ and } f(x) = F'(x).$$

Random variables

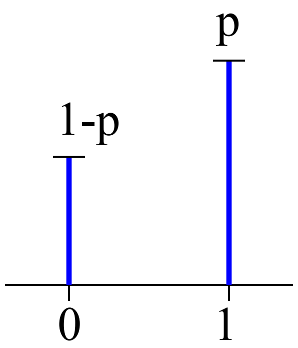
F completely determines the distribution of a random variable.

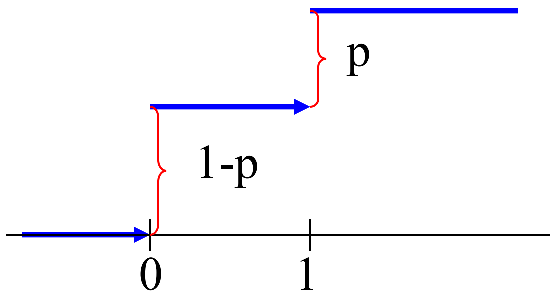
$$P(a < X \le b) = F(b) - F(a) = \begin{cases} \sum_{a<x\le b} p(x) \\ \int_a^b f(t)dt \end{cases}$$

Random variables

Bernoulli

$$X \equiv B(1, p)$$
$$P(X = 1) = p$$
$$P(X = 0) = 1 - p$$





Random variables

Binomial

Successes in *n* independent Bernoulli trials with success probability *p*

$$X \equiv B(n, p)$$

$$P(X = x) = \binom{n}{x} p^x (1 - p)^{n-x} \qquad x = 0,1,2,\ldots,n$$

$$with \quad \binom{n}{x} = \frac{n!}{x!(n-x)!}$$

Random variables

Geometric

Time of first success in a sequence of independent Bernoulli trials with success probability *p*

$$X \equiv G(p)$$
$$P(X = x) = (1 - p)^{x-1} \cdot p \qquad x = 1,2,3,\ldots$$

Random variables

Poisson

X expresses the number of “ rare events”

$$X \equiv P(\lambda), \quad \lambda > 0$$

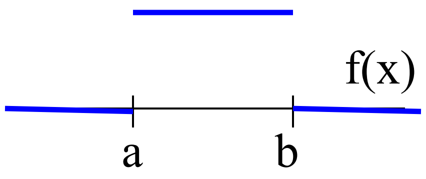
$$P(X = x) = \frac{e^{-\lambda}\lambda^{x}}{x!} \qquad x = 0, 1, 2, \ldots$$

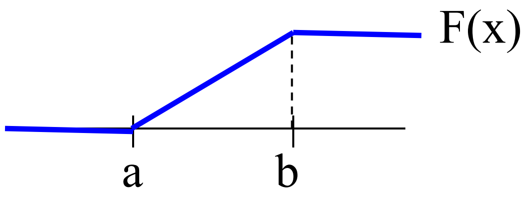
Random variables

Uniform

$$X \equiv U[a,b] \qquad f(x) = \begin{cases} \dfrac{1}{b-a} & for \quad a < x < b \\ 0 \quad otherwise & \end{cases}$$

$$F(x) = \begin{cases} 0 \quad for & x < a \\ \dfrac{x-a}{b-a} & for \quad a \le x < b \\ 1 \quad for & x \ge b \end{cases}$$



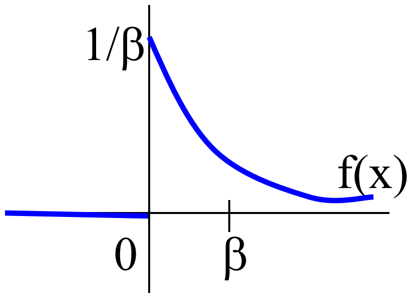


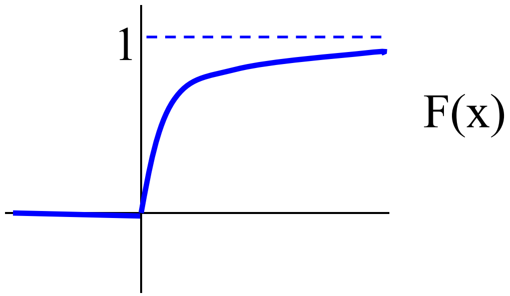
Random variables

Exponential

$$X \equiv \exp(\beta) \qquad f(x) = \begin{cases} \dfrac{1}{\beta} e^{\frac{-x}{\beta}} & for \quad x > 0 \\ 0 \quad for \quad x \leq 0 \end{cases}$$

$$F(x) = \begin{cases} 0 \quad for \quad x < 0 \\ 1 - e^{\frac{-x}{\beta}} \quad for \quad x \geq 0 \end{cases}$$





Random variables

Normal

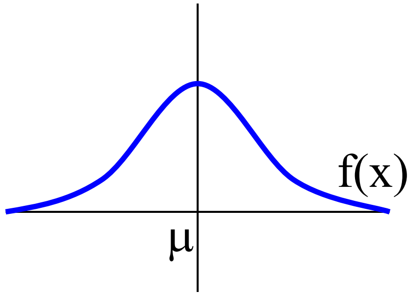
$$X \equiv N(\mu, \sigma^2)$$

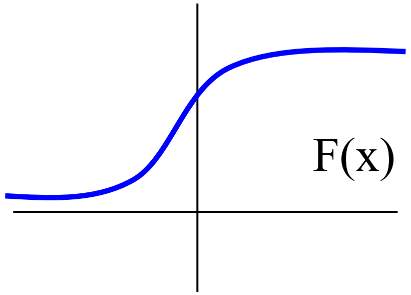
$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2} \right)$$

$$x \in \mathbb{R}$$

$$\mu \in \mathbb{R}$$

$$\sigma^2 \geq 0$$





Random variables

Properties of normal distribution

(i) $$\frac{X-\mu}{\sigma} \equiv N(0,1)$$ standard normal

(ii) $$Z \equiv N(0,1) \Rightarrow \sigma Z + \mu \equiv N(\mu, \sigma^2)$$

(iii) $$X_i \equiv N(\mu_i, \sigma_i^2)$$
$$\Rightarrow \sum_i X_i \equiv N(\sum_i^n \mu_i, \sum_i^n \sigma_i^2)$$ independent i=1,2,...,n

Random variables

Two random variables X and Y are independent if and only if:

$$p(x, y) = p_X(x) p_Y(y)$$

$$f(x, y) = f_X(x) f_Y(y),$$

for all values x and y.

Random variables

Discrete variables

$$p(x \mid y) = P(X = x \mid Y = y) = \frac{p(x, y)}{p(y)}$$

Continuous variables

$$f(x \mid y) = \frac{f(x, y)}{f(y)}$$

If *X* and *Y* are independent:

$$p(x \mid y) = p(x)$$

$$f(x \mid y) = f(x)$$

Random variables

$$EX = \mu_X = \sum_x xp(x)$$

$$EX = \mu_X = \int xf(x)dx$$

Properties:

(i) $E\sum_i \alpha_i X_i = \sum_i \alpha_i E X_i \qquad i = 1, \ldots, n$

(ii) If $X_i, i = 1, \ldots, n$ are independent then:

$$E\prod_i X_i = \prod_i EX_i$$

Random variables

Moment of order k

$$EX^k = \sum_x x^k p(x)$$

$$EX^k = \int x^k f(x)dx$$

Random variables

Variance

Given X with $\mu = EX$

$$VX = \sigma_X^2 = E(X-\mu)^2$$

$$\sigma_X = \sqrt{VX} = (E(X-\mu)^2)^{1/2}$$

standard deviation

Permutations

A B C D E

- How many ways can we choose 2 letters from the above 5, without replacement, when the order in which we choose the letters is important?
- $\underline{5} \times \underline{4} = 20$

Permutations (cont.)

$$\underline{5}\times\underline{4}=20=\frac{5!}{(5-2)!}=\frac{5!}{3!}=5\times 4$$

$$Notation:\ {}_5P_2=\frac{5!}{(5-2)!}=20$$

Combinations

A B C D E

- How many ways can we choose 2 letters from the above 5, without replacement, when the order in which we choose the letters is not important?
- $\underline{5} \times \underline{4} = 20$ when order important
- Divide by 2: $(5 \times 4)/2 = 10$ ways
- N choose K (5 choose 2)

Bounding Numbers of Combinations

$$\binom{n}{k} = \frac{n!}{k!\,(n\text{-}k)!}$$

= number of (unordered)
combinations of n objects
taken k at a time

- N choose K

$$\binom{n}{k} \sim \frac{n^k e^{-\frac{k^2}{2n}-\frac{k^3}{6n^2}}}{k!}(1-o(1)) \quad \text{for } k = o\left(n^{\frac{3}{4}}\right)$$

Combinations (cont.)

$$\binom{5}{2} = {}_5C_2 = \frac{5!}{(5-2)!2!} = \frac{5!}{3!2!} = \frac{5\times 4}{1\times 2} = \frac{20}{2} = 10$$

$$\binom{n}{r} = {}_nC_r = \frac{n!}{(n-r)!r!}$$

Tail Bounds

- In the analysis of randomized algorithms, we need to know how much does an algorithms run-time/cost deviate from its expected run-time/cost.
- That is we need to find an upper bound on Pr[X deviates from E[X] a lot]. This we refer to as the tail bound on X.

Markov and Chebychev Inequalities

- Markov Inequality (uses only mean)

$$\text{Prob}\ (\text{A} \geq \text{x}) \leq \frac{\mu}{\text{x}}$$

- Chebychev Inequality (uses mean and variance)

$$\text{Prob}\ (|\text{A} - \mu| \geq \Delta)\ \leq \frac{\sigma^2}{\Delta^2}$$

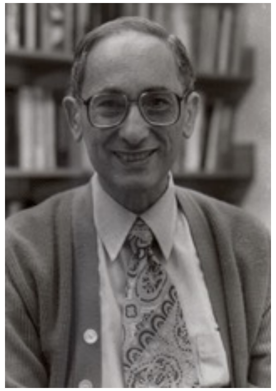
Markov and Chebychev Inequalities

- Example, if B is a Binomial with parameters n,p

$$\text{Then Prob }(\mathrm{B} \geq \mathrm{x}) \leq \frac{\mathrm{np}}{\mathrm{x}}$$

$$\text{Prob }(|\mathrm{B}-\mathrm{np}| \geq \Delta) \ \leq \frac{\mathrm{np}\ (1\text{-}\mathrm{p})}{\Delta^2}$$

Chernoff bounds



The Chernoff bound for a random variable X is obtained as follows: for any t >0,

$$\Pr[X \text{ ⍰ } a] = \Pr[e^{tX} \text{ ⍰ } e^{ta}] \leq E[e^{tX}] / e^{ta}$$

Similarly, for any t <0,

$$\Pr[X \text{ ⍰ } a] = \Pr[e^{tX} \text{ ⍰ } e^{ta}] \leq E[e^{tX}] / e^{ta}$$

The value of t that minimizes $E[e^{tX}] / e^{ta}$ gives the best possible bounds.

Chernoff Bound of Random Variable A

- Uses all moments
- Uses moment generating function

$$\text{Prob } (A \geq x) \leq e^{-sx} \; M_A(s) \text{ for } s \geq 0$$

$$= e^{\gamma(s) - sx} \; \text{where } \gamma(s) = \ln (M_A(s))$$

$$\leq \; e^{\gamma(s) - s\,\gamma'(s)}$$

By setting $x = \gamma'(s)$

1st derivative minimizes bounds

Chernoff Bound of Discrete Random Variable A

$$\text{Prob}\,(A \geq x) \leq z^{-x}\; G_A(z) \quad \text{for } z \geq 1$$

- Choose z = z_0 to minimize above bound
- Need Probability Generating function

$$G_A(z) = \sum_{x \geq 0} z^x\; f_A(x) = E(z^A)$$

Chernoff Bounds for Binomial B with parameters n,p

- Above mean $x \geq \mu$

$$\text{Prob }(B \geq x)$$

$$\leq \left(\frac{n\text{-}\mu}{n\text{-}x}\right)^{n\text{-}x} \left(\frac{\mu}{x}\right)^{x}$$

$$\leq \ e^{x\text{-}\mu} \left(\frac{\mu}{x}\right)^{x} \text{ since } \left(1 - \frac{1}{x}\right)^{x} < e^{-1}$$

$$\leq \ e^{\text{-}x\text{ - }\mu} \ \text{ for } x \geq \mu e^{2}$$

Chernoff Bounds for Binomial B with parameters n,p

- Below mean x ≤ μ

$$\text{Prob } (B \le x)$$

$$\le \left(\frac{\text{n-}\mu}{\text{n-x}}\right)^{\text{n-x}} \left(\frac{\mu}{\text{x}}\right)^{\text{x}}$$

Birthday Problem

- What is the smallest number of people you need in a group so that the probability of 2 or more people having the same birthday is greater than 1/2?
- Answer: 23

No. of people	23	30	40	60
Probability	.507	.706	.891	.994

Birthday Problem

- A={at least 2 people in the group have a common birthday}
- A’ = {no one has common birthday}

$$3 \ \ people \quad : P(A') = \frac{364}{365} \times \frac{363}{365}$$

$$23 \ \ people \quad :$$

$$P(A') = \frac{364}{365} \times \frac{363}{365} \times \ldots \frac{343}{365} = .498$$

$$so \ \ P(A) = 1 - P(A') = 1 - .498 \ \ = .502$$

The Bayesian Framework

- The Bayesian framework assumes that we always have a prior distribution for everything.
  - The prior may be very vague.
  - When we see some data, we combine our prior distribution with a likelihood term to get a posterior distribution.
  - The likelihood term takes into account how probable the observed data is given the parameters of the model.
    - It favors parameter settings that make the data likely.
    - It fights the prior
    - With enough data the likelihood terms always win.

Basic Probability Formulas

- Product rule

$$P(A \wedge B) = P(A \mid B)P(B) = P(B \mid A)P(A)$$

- Sum rule

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$$

- Bayes theorem

$$P(h \mid D) = \frac{P(D \mid h)P(h)}{P(D)}$$

- Theorem of total probability, if event *Ai* is mutually exclusive and probability sum to 1

$$P(B) = \sum_{i=1}^{n} P(B \mid A_i)P(A_i)$$

Bayes Theorem

- Given a hypothesis *h* and data *D* which bears on the hypothesis:

$$P(h \mid D) = \frac{P(D \mid h)P(h)}{P(D)}$$

- *P(h)*: independent probability of *h*: *prior probability*
- *P(D)*: independent probability of *D*
- *P(D|h)*: conditional probability of *D* given h: *likelihood*
- *P(h|D)*: conditional probability of *h* given *D*: *posterior probability*

Maximum A Posterior

- Based on Bayes Theorem, we can compute the *Maximum A Posterior* (MAP) hypothesis for the data
- We are interested in the best hypothesis for some space H given observed training data D.

$$
\begin{aligned}
h_{MAP} &\equiv \underset{h \in H}{\operatorname{argmax}}\, P(h \mid D) \\
&= \underset{h \in H}{\operatorname{argmax}} \frac{P(D \mid h) P(h)}{P(D)} \\
&= \underset{h \in H}{\operatorname{argmax}}\, P(D \mid h) P(h)
\end{aligned}
$$

H: set of all hypothesis.

Note that we can drop *P(D)* as the probability of the data is constant (and independent of the hypothesis).

Naïve Bayes Background

- There are two main methods for training a classifier:

a) Discriminative Classifiers

Examples: k-NN, decision trees, Neural Networks, SVM

b) Generative Classifiers

Example: Bayesian approaches (naive Bayes…)

Discriminative approach seems easier, as the task is easier; you don’t need to model classes (observation distribution of features in those classes), just need to find where the query instance belongs to.

Naïve Bayes

- The *Naïve Bayes Assumption*: Assume that all features are independent **given the class label Y**
- Equationally speaking:

$$P(X_1, \ldots, X_n | Y) = \prod_{i=1}^{n} P(X_i | Y)$$

- (We will discuss the validity of this assumption later)

Las Vegas Algorithms

- Always gives the true answer.
- Running time is random.
- Running time is bounded.
- Random Quick sort is a Las Vegas algorithm.

Monte Carlo Algorithms

- It may produce incorrect answer!
- We are able to bound its probability.
- By running it many times on independent random variables, we can make the failure probability arbitrarily small *at the expense of running time*.

Monte Carlo Example

- Suppose we want to find a number among n given numbers which is larger than or equal to the median.

Suppose $A_1 < \ldots < A_n$.

We want A_i, such that $i \geq n/2$.

It's obvious that the best deterministic algorithm needs O(n) time to produce the answer.

n may be very large!

Suppose n is 100,000,000,000 !

Monte Carlo Example

- Choose 100 of the numbers with *equal probability*.
- find the maximum among these numbers.
- Return the maximum.

- The running time of the given algorithm is O(1).
- The probability of Failure is $1/(2^{100})$.
- Consider that the algorithm may return a wrong answer but the probability is very smaller than the hardware failure or even an earthquake!

RP Class (randomized polynomial)

- Bounded polynomial time in the worst case.
- If the answer is Yes; Pr[return Yes] > ½.
- If the answer is No; Pr[return Yes] = 0.
- ½ is not actually important.

PP Class (probabilistic polynomial)

- Bounded polynomial time in worst case.
- If the answer is Yes; Pr[return Yes] > ½.
- If the answer is No; Pr[return Yes] < ½.
- Unfortunately the definition is weak because the distance to ½ is important but is not considered.

Graph Connectivity

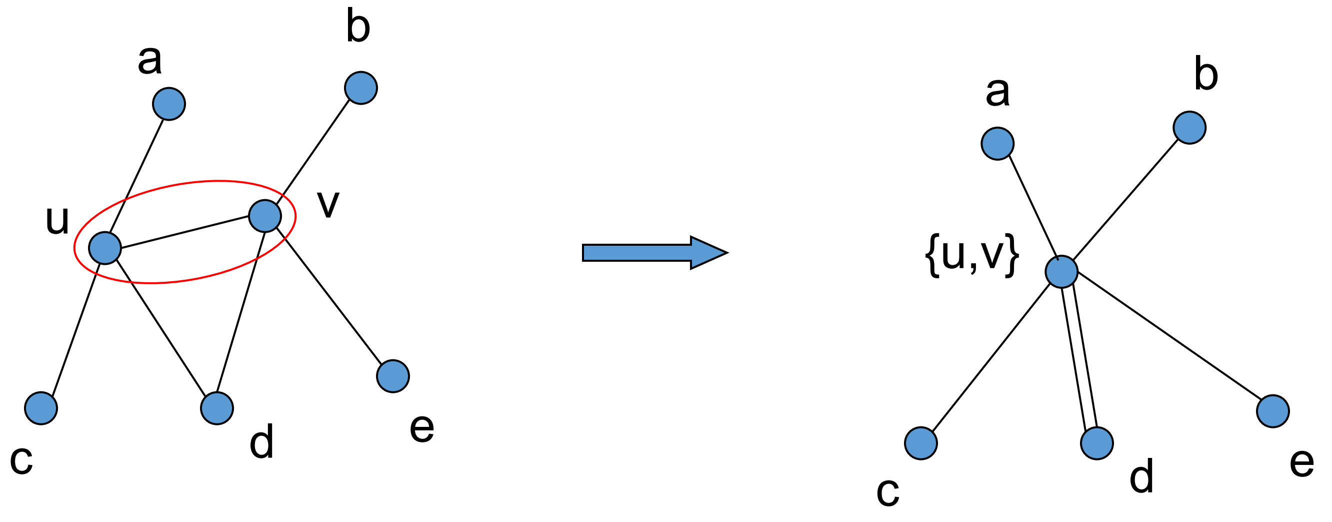
- You want to check if two vertices u and v are in the same connected component.
- Start a random walk from v.
- Have a random walk of length $2n^3$.
- If you haven’t visited u, the probability of u to be in this component is less than ½.
- By repeating this algorithm, you can make the probability of failure arbitrarily small.
- Running time of algorithm is $O(n^3)$.
- Required space is O(logn).

Graph Contraction

For an undirected graph G, we can construct a new graph G’ by *contracting* two vertices u, v in G as follows:

- u and v become one vertex {u,v} and the edge (u,v) is removed;
- the other edges incident to u or v in G are now incident on the new vertex {u,v} in G’;

Note: There may be multi-edges between two vertices. We just keep them.



Graph G

Graph G’

Karger’s Min-cut Algorithm

```
For i = 1 to $100n^2$
    repeat
        randomly pick an edge (u,v)
        contract u and v
    until two vertices are left
    $c_i$ ← the number of edges between them
Output mini $c_i$
```

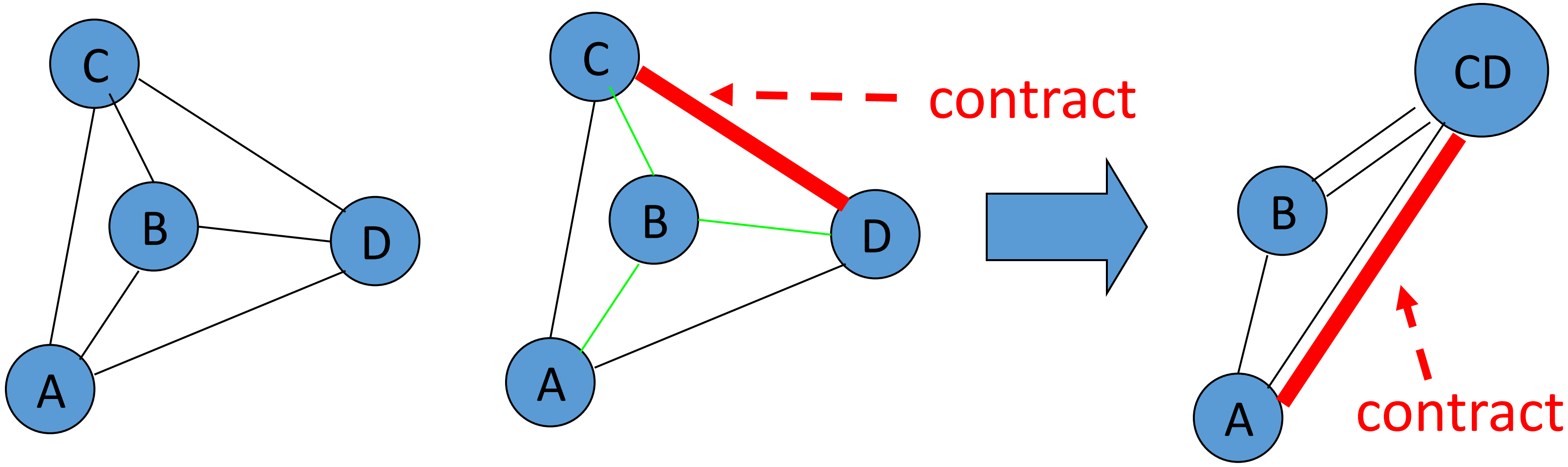
Key Idea

- Let $C^* = \{c_1^*, c_2^*, \ldots, c_k^*\}$ be a min-cut in G and C^i be a cut determined by Karger's algorithm during some iteration i.

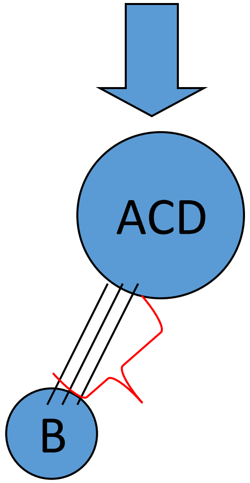
- C^i will be a min-cut for G if during iteration "i" none of the edges in C* are contracted.

- If we can show that with *prob. $\Omega(1/n^2)$, where $n = |V|$, C^i will be a min-cut*, then by *repeatedly obtaining min-cuts $O(n^2)$ times* and taking minimum gives the min-cut with high prob.

Karger’s Min-cut Algorithm



(i) Graph G (ii) Contract nodes C and D (iii) contract nodes A and CD



Note: C is a cut but not necessarily a min-cut.

C is a cut, but not necessarily a min-cut.

(Iv) Cut C=*{(A,B), (B,C), (B,D)}*

Analysis of Karger's Algorithm

- In step 1, Pr [no crossing edge picked] >= 1 – 2/n
- Similarly, in step 2, Pr [no crossing edge picked] ≥ 1-2/(n-1)
- In general, in step j, Pr [no crossing edge picked] ≥ 1-2/(n-j+1)
- Pr {the n-2 contractions never contract a crossing edge}
  - = Pr [first step good]

    * Pr [second step good after surviving first step]

    * Pr [third step good after surviving first two steps]

    * …

    * Pr [(n-2)-th step good after surviving first n-3 steps]

    ≥ (1-2/n) (1-2/(n-1)) … (1-2/3)

    = [(n-2)/n] [(n-3)(n-1)] … [1/3] = 2/[n(n-1)] = $\Omega(1/n^2)$