CS292A Notes

* **Def of Probability Spaces.**

A probability distribution is...

... a set Ω that represents an output space

... a function $Pr: 2^{\Omega} \rightarrow [0,1]$

such that ...

... $\forall$ events $E_1, \ldots, E_n$ where $E_i \cap E_j = \emptyset$ ...

... $Pr[E_1 \cup \ldots \cup E_n] = Pr[E_1] + \ldots + Pr[E_n]$

... $Pr[\Omega] = 1$

* **Union Bound.**

$\forall E_1, \ldots, E_n \quad Pr[E_1 \cup \ldots \cup E_n] \leq \sum_i Pr[E_i]$

* **Mutual Independence.**

A set of Events $E_1, \ldots, E_n$ are mutually independent if

$\forall I \subseteq \{1, \ldots, n\} \quad Pr\left[\bigcap_{i \in I} E_i\right] = \prod_{i \in I} Pr[E_i]$

* **Conditional Probability.**

Think of this as reducing and redefining the probability spaces

$Pr[E_1 | E_2] = Pr[E_1 \cap E_2] / Pr[E_2]$

Observe that when E_1 and E_2 are independent...

... $Pr[E_1 | E_2] = Pr[E_1]$

* **Law of Total Probability.**

If $E_1, \ldots, E_n$ partitions Ω, then $\forall X \subseteq \Omega$ ...

... $Pr[X] = \sum_i Pr[X | E_i] \, Pr[E_i]$

* **Random Variable.**

A function $X: \Omega \rightarrow \mathbb{R}$.

Note that the function doesn't need to map to $\mathbb{R}$ but needs addition and multiplication to define expectation.

CS292A Notes

* Expectation.
  $\mathbb{E}[X] = \Sigma_i \Pr[X=i] \cdot i = \Sigma_{p \in \Omega} \Pr[p] \, X(p)$

* Linearity of Expectation.
  $\mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]$

* Existence of values relating to Expectation.
  $\Pr[X \geq \mathbb{E}[X]] > 0.$
  $\Pr[X \leq \mathbb{E}[X]] > 0.$

* Conditional Expectation.
  $\mathbb{E}[Y \mid Z=z] = \Sigma_y \Pr[Y=y \mid Z=z] \cdot y$

* Principle of Deferred Decisions.
  Think about how some random variables are set for analysis and others are left random for later analysis (conditioned upon the set variables).

* Expectation and Constants.
  $\mathbb{E}[c] = c.$
  $\mathbb{E}[cX] = c\,\mathbb{E}[X].$

* Jensen's Inequality.
  $0 \leq \mathbb{E}[X^2] - (\mathbb{E}[X])^2$

* Markov's Inequality.
  For a rv $X \geq 0$, $\forall a \geq 1$...
  ... $\Pr[X \geq a\,\mathbb{E}[X]] \leq 1/a.$

* Moment.
  $\mathbb{E}[X^k] = \Sigma_i \Pr[X=i] \, i^k$

CS292A Notes

* Variance.

$Var[X] = \mathbb{E}[X - \mathbb{E}[X]]^2 = \mathbb{E}[X^2] - (\mathbb{E}[X])^2.$

* Bernoulli RV.

$$Pr[X] = \begin{cases} p & \text{if success} \\ 1-p & \text{if fail} \end{cases}$$

$Var[X] = p(1-p) \leq \min(p, 1-p)$

$\mathbb{E}[X] = p.$

* Binomial RV.

$Pr[X=j] = \binom{n}{j} p^j (1-p)^{n-j}$ where $j \in [0, n]$

$\mathbb{E}[X] = n \cdot p$

$Var[X] = np(1-p)$

* Geometric RV.

infinite independent trials with success probability p;
want to see number of trials until first success...

... $Pr[X=i] = (1-p)^{i-1} p$

... $\mathbb{E}[X] = 1/p$

... $Var[X] = (1-p)/p^2$

* Memoryless Property.

$Pr[X = n+k \mid X > k] = Pr[X = n]$

* Sum of Geometric Series.

$\Sigma_{i=1}^{n} a \cdot r^{i-1} = a(1-r^n)/(1-r)$

* Covariance.

$Cov(X,Y) = \mathbb{E}[X - \mathbb{E}[X]]\, \mathbb{E}[Y - \mathbb{E}[Y]] = \mathbb{E}[XY] - \mathbb{E}[X]\mathbb{E}[Y]$

If X, Y are independent...

... $Cov(X,Y) = 0.$

CS292A Notes

* Joint Variance.
$Var[X+Y] = Var[X] + Var[Y] + 2\,Cov[X,Y]$
If a set of RV are pairwise independent, then...
... $Var[\Sigma_i X_i] = \Sigma_i Var[X_i]$

* Chebyshev's Inequality.
$\Pr[|X - \mathbb{E}[X]| \geq k\,\sigma(X)] \leq \frac{1}{k^2}$ where $\sigma(X) = \sqrt{Var(X)}$

* Moment Generating Function.
$M_X(t) = \mathbb{E}[e^{tX}] \overset{?}{=} \Sigma_i \Pr[X=i] \cdot e^{ti} \; : \mathbb{R} \to \mathbb{R}$
$\{-e,e\}$ defined s.t. geometric series converges & well defined around 0.
For the k^{th} moment...
... $\mathbb{E}[X^k] = M_X^k(t=0)$.

* Joint Moment Generating Functions.
If X, Y are independent...
... $M_{X+Y}(t) = M_X(t)\,M_Y(t)$

* Chernoff Bounds.
For $t>0$,...
... $\Pr[X \geq a] = \Pr[e^{tX} \geq e^{ta}] \leq \mathbb{E}[e^{tX}]/e^{ta}$
... where we want to pick t s.t. $\min(\mathbb{E}[e^{tX}]/e^{ta})$
For $t<0$,...
... $\Pr[X \leq a] = \Pr[e^{tX} \geq e^{ta}] \leq \mathbb{E}[e^{tX}]/e^{ta}$
... where we want to pick t s.t. $\min(\mathbb{E}[e^{tX}]/e^{ta})$

* Chernoff Bounds for Sum of Independent Bernoulli RV
For $\Pr[X_i] = p_i$, $X = \Sigma_i X_i$, $\mu = \mathbb{E}[X]$,...
... $\Pr[X \geq (1+\delta)\mu] < (e^{\delta}/(1+\delta)^{(1+\delta)})^{\mu}$ for any $\delta > 0$
... $\Pr[X \geq (1+\delta)\mu] \leq (e^{-\mu\delta^2/3})$ for $\delta \in (0,1]$
... $\Pr[X \geq R] \leq 2^{-R}$ for $R \geq 6\mu$

CS292A Notes

* More Chernoff Bounds for Sum of Independent Bernoulli RV
For $\Pr[X_i] = p_i$, $X = \sum_i X_i$, $\mu = \mathbb{E}[X]$, ...
... ~~Pr[X~~ $\Pr[X \leq (1-\delta)\mu] \leq e^{-\mu\delta^2/2}$ for $\delta \in (0,1)$
... $\Pr[X \leq (1-\delta)\mu] \leq (e^{-\delta}/(1-\delta)^{(1-\delta)})^{\mu}$ for $\delta \in (0,1)$

* Hoeffding Bounds.
For $a_i \leq X_i \leq b_i$, $X = \sum_i X_i$, $t > 0$, ...
... $\Pr[|X - \mu| \geq \varepsilon n] \leq 2\exp(-2\varepsilon^2 n^2 / \sum (b_i - a_i)^2)$

* Random Graphs - $G(n,p)$ model.
$G(n,p)$ defines a graph with n vertices, and m edges, each included with probability p...
... $\Pr[G = \hat{G}] = p^m (1-p)^{\binom{n}{2} - m}$
... $\mathbb{E}[m] = \binom{n}{2} p$
... $\mathbb{E}[\text{degree per vertex}] = (n-1)p$

* Random Graphs - $G(n, N)$ model.
$G(n, N)$ defines a graph with n vertices and exactly N edges.

* Probabilistic Methods
If $\Pr[E] > 0$, then $E \neq \emptyset$.
$\forall p \in \Omega$, $\Pr[P = p] \leq x \rightarrow |\Omega| \geq 1/x$
$\forall p \in \Omega$, $\Pr[P = p] \geq x \rightarrow |\Omega| \leq 1/x$

* Stochastic Processes.
A sequence of random variables $X_1, \ldots, X_t$.

* Markov Chain.
A stochastic process is a markov chain if it is memoryless...
... $\forall t, a_1, \ldots, a_t$ $\Pr[X_t = a_t \mid X_{t-1} = a_{t-1}, \ldots, X_1 = a_1] = \Pr[X_t = a_t \mid X_{t-1} = a_{t-1}]$
... this does not state the probability is constant w.r.t time, but instead all information is embedded in previous state.

CS292A Notes

*** Verifying Polynomial Identities.**

Input: Polynomials $F(x)$, $G(x)$ of max degree d

Output: 1 iff $F(x) = G(x)$, 0 otherwise

Algorithm:

- pick x' uar $\in [1, 100d]$
- output 1 when $F(x') = G(x')$, 0 otherwise
- repeat to boost success; ~~only~~ output product of trials

*** Verifying Matrix Multiplication**

Input: $A, B \in \mathbb{R}^{m \times m}$, $C \in \mathbb{R}^{m \times m}$

Output: 1 iff $C = AB$, 0 otherwise

Algorithm:

- pick random $x \in \{0,1\}^n$
- compute Cx
- compute $A(Bx)$
- output 1 when $Cx = A(Bx)$, 0 otherwise
- repeat to boost success; output product of trials

*** Min Cut**

Input: Graph G

Output: set X s.t. $d(X) := |\{uv \in E(G) : u \in X, v \notin X\}|$ is minimized to create 2 disjoint sets

Algorithm:

for ~~i=2 to n~~ | $n-2$ iterations,

- pick $e \in E(G)$ uar
- contract e (combine u,v into single vertex)
- eliminate self loops (connecting u,v)

output edges connecting final 2 vertices.

*** K-Approximation.**

A K-approximation is no worse than the optimal solution by a factor of K.

CS292A Notes

* Generating Random Permutations.

Input: list of n elements.

Output: random permutation s.t. all choices have $Pr[\sigma] = \frac{1}{n!}$

Algorithm:

- for n iterations, with each iteration i,
  - choose $r = \text{random}(0, n-i)$
  - swap $a[n-i]$ with $a[r]$
- output permuted array a

* Max Bisection.

Input: Graph G

Output: Two sets A, B that partition $V(G)$ and $||A|-|B|| \leq 1$.

Algorithm:

- Select an arbitrary $n/2$ points to be in A and the other points to be in B. (n = # of vertices).
- output A, B.

* Max Cut.

Input: Graph G.

Output: Set X s.t. $\partial(X) := |\{uv \in E(G) : u \in X, v \notin X\}|$ is maximized to create two disjoint sets in G.

Algorithm:

- include every edge in X with independent probability .5.
- output X.

* Coupon Collector.

- Given n items, infinitely many boxes, each with equal pull rates $p = 1-(i-1)/n$ of getting new item

$\mathbb{E}[X_i] = n/(n-i+1)$ boxes until new pull

$\mathbb{E}[\Sigma_i X_i] = n\Sigma_i \, 1/i$ total boxes to get one of each

CS292A Notes

* 3-SAT.

Input: set of 3-clauses, literals from choice of n booleans.

Output: an assignment that satisfies or maximizes number of clauses satisfied.. where a clause is satisfied if an assignment assigns $T = x_i$, $F = \bar{x}_i$ $\forall x_i \in$ clause C.

Algorithm:

- iterate through n variables
  - set x_i = True if more clauses use x_i than $\bar{x}_i$, false otherwise
- output assignment of $x_1, \dots, x_n$

* Quicksort.

- Input: $A[1], \dots, A[n]$ list of values.
- Output: Sorted list of values.
- Algorithm:
  - if $|A| \le 1$, return A
  - pick pivot $p \in A$ uar.
  - $L = \{A[i] : A[i] < p\}$
  - $R = \{A[i] : A[i] > p\}$
  - return $[QS(L), p, QS(R)]$

* Quickselect.

- Input: $A[1], \dots, A[n]$, K
- Output: Kth value on the sorted list
- Algorithm:
  - If $|A| = 1$, return $A[1]$
  - pick pivot $p \in A$ u.a.r.
  - $L = \{A[i] : A[i] < p\}$
  - $R = \{A[i] : A[i] > p\}$
  - if $|L| \ge K$ return $QS(L, K)$
  - if $|L| = K-1$ return p
  - else return $QS(R, K-|L|-1)$