UCSD CSE 21, Spring 2014 [Section B00]
Mathematics for Algorithm and System Analysis
Lecture 11
Class URL: http://vlsicad.ucsd.edu/courses/cse21-s14/
Lecture 11 Notes

- **Goals for this week**
  - (Tuesday) Linearity of expectation; Joint distributions
  - (Today) Decision trees, hashing example, ...
  - For quiz: MT solutions, mean/variance, Tuesday material

- Next week: conditional probability, Bayes’ rule; induction and recursion
  - Please do your reading in BW

- Emails sent out to prompt 1-1, small-group tutor sessions.
  - See also IDEA Center (TBP, HKN) tutoring

- **Possibly of interest (“life advice”):**
  - Four parts: Context, Research, Grad School, Life/Career
  
  [Link](http://vlsicad.ucsd.edu/courses/cse101-w14/slides/cse101-w14-gradschoolplus.pdf)
Main Points From Last Time

- **Expectation** $E(X) = \sum_k p_k x_k$
- **Variance** $\sigma_X^2 = E(X^2) - (E(X))^2$

- **Linearity of Expectation**: $E(aX) = aE(X)$
  - $E(aX + bY) = aE(X) + bE(Y)$ \quad \text{r.v.’s} X, Y on same sample space
  - “indicator variable” technique $X_j = 1$ if letter $j$ in correct envelope

- **Joint distribution**: $h_{X,Y}: \text{Im}(X) \times \text{Im}(Y) \to [0,1]$
  - $h_{X,Y}(i,j) = \text{probability of } ((X = i) \text{ AND } (Y = j))$

- **Independence**: $X$ and $Y$ are independent r.v.’s iff
  - $h_{X,Y}(i,j) = f_X(i) \cdot f_Y(j)$ for all $(i,j) \in \text{Im}(X) \times \text{Im}(Y)$
  - Info about one r.v. doesn’t give info about the other
  - $E(X \cdot Y) = E(X) \cdot E(Y)$ only if $X$, $Y$ independent
iClicker Quiz, Week 6

please don’t do this...

Talk about being a good friend
Linearity of Expectation: Baseball Cards

• A bubble gum company puts one random major league baseball player’s card into every pack of its gum. There are N major leaguers in the league this year. If X is the number of packs of gum that a collector must buy up until she has at least one of each player’s card, what is E(X) ?

Think about the card collecting in “phases”

• Phase 1: collect the first unique card
• Phase 2: collect the second unique card
• …
• Phase k-1: collect the (k-1)st unique card
• Phase k: collect the kth unique card
  – P(next pack ends this phase) =
  – E(# packs to buy in this phase) =
Linearity of Expectation: Baseball Cards

• A bubble gum company puts one random major league baseball player’s card into every pack of its gum. There are N major leaguers in the league this year. If X is the number of packs of gum that a collector must buy up until she has at least one of each player’s card, what is E(X) ?

• Define r.v. $X_k = \text{#packs bought in Phase } k$

• $E(X) = E(\sum_{k=1..n} X_k) = \sum_{k=1..n} E(X_k)$  linearity of expectation!
  
  $= \sum_{k=1..n} \frac{N}{N - k + 1}$
  
  $= N \cdot \left(\frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \ldots + \frac{1}{N}\right)$
  
  $= N \cdot H_N$
  
  $H_N = N^{th} \text{ harmonic number}$

$\Rightarrow \sim N \log N \text{ packs}$
Decision Trees

- Decision trees are a systematic way of listing templates / possibilities
- Decision trees are a type of graph
  - Tree: graph with \( n \) vertices: \( n - 1 \) edges, connected, no cycles

- Can be used for enumeration (counting things systematically)

- Can be used to capture recursive solution approaches (algorithms)

- Vehicle for correspondences between induction and recursion
From Lecture 4: Sorting (w/Comparisons)

- Input: sequence of numbers
- Output: a sorted sequence
- **Sorting == Identifying a Permutation**

1. This DECISION TREE is a sorting algorithm!
2. To correctly sort any input, this tree needs at least as many leaves as there are possible outcomes – i.e., permutations of n elements
3. #comparisons needed to get to the leaf at greatest depth is the worst-case complexity of the sorting algorithm
Bounds on Complexity of Sorting (w/ comparisons)

- In any binary tree with \( n! \) leaves, some leaf has depth at least \( \log_2(n!) \)

  From P5.2 and P5.3: \( \log_2(n!) \sim n \log n \)

  = Lower bound on worst-case complexity of sorting

- P11.3: In any binary tree with \( M \) leaves, the average leaf depth is at least \( \log_2 M \).
  - If all \( n! \) permutations are equally likely, then average time to sort is at least \( \log_2(n!) \sim n \log n \)

  = Lower bound on average-case complexity of sorting
A Traveling Salesperson’s Tours

• Recall from Lecture 2 slides: **Bob the salesman starts at his home in San Diego, and tours 6 cities (Abilene, Bakersfield, Corona, Denver, Eastville and Frankfort), visiting each city exactly once before returning home. In how many possible ways could Bob make his tour?**
Number of Non-Increasing Functions

• How many functions from $A = \{1,2,3,4\}$ to itself are non-increasing? (similar to HW6 #7)
• Each function can be defined by a sequence of decisions:
  
  What is $f(1)$
  What is $f(2)$
  What is $f(3)$
  What is $f(4)$ …
  … but these are not independent decisions!

  – If $f(1) = 1$, then we must have $f(2) = f(3) = f(4) = 1$
  – If $f(1) = 2$, then we must have $f(2) \in \{1,2\}$
  – If $f(2) = 2$, then $f(3) \in \{1,2\}$
  – Etc.

  (There are 35 such functions)
Decision Tree: Non-Attacking Queens

- Put N queens on an N x N chessboard such that no queen attacks another queen or, “in how many ways can …”

A queen in chess can *attack* any square on the same row, column, or diagonal. Given an n x n chessboard, we seek to place n queens onto squares of the chessboard, such that no queen attacks another queen. The example shows a placement (red squares) of four mutually non-attacking queens.
Non-Attacking Queens: Leaf in Tree = Solution
Non-Attacking Queens Decision Tree

Comments

• Our tree corresponded to a solution space of size $4^4 = 256$

• What if we numbered the squares as 1, 2, …, 16 and tried to enumerate all $P(16,4)$ choices?
  – Solution space would have size $16 \times 15 \times 14 \times 13 = 43680$

• Our decision tree was “smarter”
  – We realized that there must be exactly one queen per row, so the decision for the $k^{th}$ row (i.e., $k^{th}$ queen) boils down to picking a square in the row

• Our illustration of the tree did not show any infeasible configurations
A graph is legally colored with k colors if each vertex has a color (label) between 1 and k (inclusive), and no adjacent vertices have the same color.

Example: scheduling classes into classrooms
Each class = vertex in graph
Edge between two vertices if class times overlap
#colors = #distinct classrooms needed
A graph is legally colored with k colors if each vertex has a color (label) between 1 and k (inclusive), and no adjacent vertices have the same color.
A graph is legally colored with $k$ colors if each vertex has a color (label) between 1 and $k$ (inclusive), and no adjacent vertices have the same color.
Hashing

- **Definition**: A hash function maps data of variable length to data of a fixed length. The values returned by a hash function are called *hash values* or simply *hashes*.

- **Open Hashing**: Each item hashes to a particular location in an array, and locations can hold more than one item. *Multiple input data values can have the same hash value*. These are called *collisions*!
Number of Items Per Location

• n items, hash to table of size k

• What is expected number of items per location?

• n independent trials with outcome = location in table (i.e., hash value)

• Let r.v. $X = \#$ inputs that hash to location 1
  – Let’s write $X$ as the sum of $X_i$’s:
    • $X_i = 1$ if the $i^{th}$ input hashes to location 1
    • $X_i = 0$ otherwise
  \[
  X = X_1 + X_2 + \ldots + X_n
  \]
  \[
  \Rightarrow E(X) = E(X_1) + E(X_2) + \ldots + E(X_n)
  \]
  \[
  = n \cdot 1/k = n/k
  \]
Number of Empty Locations

• What is expected number of empty locations?

Define r.v. $Y = \# \text{ empty locations}$. We want $E(Y)$

Example: 20 items into table of size 5: $E(Y) = 5 \cdot (4/5)^{20} \approx 0.06$
Example: 20 items into table of size 10: $E(Y) = 10 \cdot (9/10)^{20} \approx 1.22$
Toward Conditional Probability, Bayes’ Rule

- Probability of A and B, i.e., \( P(A \cap B) \)

\[
P(A \cap B) = P(B | A) \cdot P(A) \quad \text{probability of B given A, times probability of A}
\]

\[
P(A \cap B) = P(A | B) \cdot P(B) \quad \text{(symmetrical observation)}
\]

\[
P(A | B) \cdot P(B) = P(B | A) \cdot P(A)
\]

\[
\Rightarrow P(A | B) = \frac{P(B | A) \cdot P(A)}{P(B)} \quad \text{Bayes’ Rule / Theorem}
\]
Toward Conditional Probability, Bayes’ Rule

• If a student knows 80% of the material on an exam, what is the probability that he answers a given true-false question correctly?

Knows material: correct with \( P = 1 \)

Doesn’t know material: correct with \( P = 0.5 \)

Total: 0.9
Toward Conditional Probability, Bayes’ Rule

• A drug test detects PED use 95% of the time. However, 15% of PED-free individuals also test positive (= false-positive rate). 10% of NFL players use PEDs. Your NFL player friend tests positive. What is the probability that he actually did use steroids?

Event A: your friend used steroids

Event B: your friend tests positive

We want: \( P(A \mid B) \)
Problems 11

• P11.1 Draw a convex n-gon P, along with all possible diagonals in P. Suppose that no three diagonals pass through any single point. Give an expression (and an explanation) for the number of distinct triangles that you can find in your drawing.

• P11.2 2n points are chosen on a circle. In how many ways can you join pairs of points by nonintersecting chords? [Hint: this is actually similar to one of the MT review questions!]

• P11.3 Prove that in any binary tree with M leaves, the average leaf depth is at least \( \log_2 M \). (Note: The root is at depth = 0, its children are at depth = 1, etc.)