Chris Piech
CS109

Handout #15
April 18, 2016

Problem Set #3

Due: 2:30pm on Wednesday, April 27th

<u>For each problem, explain/justify how you obtained your answer in order to obtain full credit.</u> In fact, most of the credit for each problem will be given for the derivation/model used as opposed to the final answer. Make sure to describe the distribution and parameter values you used (e.g., Bin(10, 0.3)), where appropriate. It is fine for your answers to include summations, products, factorials, exponentials, or combinations, unless you are specifically asked for a computed numeric answer.

1. Recall the game set-up in the "St. Petersburg's paradox" discussed in class: there is a fair coin which comes up "heads" with probability $p = 0.5$. The coin is flipped repeatedly until the first "tails" appears. Let N = number of coin flips before the first "tails" appears (i.e., N = the number of consecutive "heads" that appear). Given that no one really has infinite money to offer as payoff for the game, consider a variant of the game where you win MIN($\$2^N$, X), where X is the maximum amount that the game provider will pay you after playing. Compute the expected payoff of the game for the following values of X. Show how you derived your answer.

   a. X = $5.

   b. X = $500.

   c. X = $4096.

2. When a bit string is sent over a network, each bit in the string will independently be corrupted (flipped) with probability p. Say we come up with a protocol for sending strings over the network where if we have an original string s of length n bits, we create the message ss (just two copies of the original message in a row, so ss has length $2n$ bits) and send that message over the network instead. Thus, the recipient can detect an error if there are any discrepancies between the first and second halves of the string they receive. Note that it is possible for the recipient to not be able to detect an error if a bit and its corresponding duplicate in the second half of the message are both corrupted (flipped).

   a. What is the expression (in terms of n and p) for the probability that the message ss is received without any corruption? Also, compute the numerical value for your expression for $n = 4$ and $p = 0.05$.

   b. What is the expression (in terms of n and p) for the probability that the recipient receives a corrupted message and is <u>not</u> able to detect that it is corrupted? Also, compute the numerical value for your expression for $n = 4$ and $p = 0.05$.

   c. What is the expression (in terms of n and p) for the probability that the recipient receives a corrupted message where the recipient <u>can</u> detect that some sort of corruption took place? Also, compute the numerical value for your expression for $n = 4$ and $p = 0.05$.

3. Say we have an integer array **arr[10]** (indexed from 0 to 9), which contains the numbers 1 through 10 in sorted order. Now, say **key** is a randomly generated integer value between 1 and 10 (inclusive), where each value is equally likely.

   a. What is the expected number of times that the "equality test" (as noted by the comment in the code) is executed in the function **linear** below (assuming **linear** is passed the array **arr** and the randomly chosen value **key**). Give an exact value (not a big-Oh running time or an approximation) for the expectation, and explain how you derived your answer.

```
int linear(int arr[], int key) {
   for(int i = 0; i < 10; i++) {
      if (arr[i] == key) return i;  // Equality test: (arr[i] == key)
   }
   return -1;  // Will never get here when key is in [1,10]
}
```

   b. Under the same conditions for array **arr** and the randomly chosen value **key**, what is the expected number of times that the "equality test" is executed in the function **binary** below. Give an exact value (not a big-Oh running time or an approximation) for the expectation, and explain how you derived your answer.

```
int binary(int arr[], int key) {
   int low = 0;
   int high = 9;
   while (low <= high) {
      int mid = (low + high) / 2;
      if (arr[mid] == key) return mid;  // Equality test: (arr[mid] ==
key)
      else if (arr[mid] < key) low = mid + 1;
      else high = mid - 1;
   }
   return -1;  // Will never get here when key is in [1,10]
}
```

4. Consider a hash table with n buckets. Now, m strings are hashed into the table (with equal probability of being hashed into any bucket).

   a. Let n = 8,000 and m = 24,000. What is the (Poisson approximated) probability that the first bucket has 0 strings hashed to it?

   b. Let n = 8,000 and m = 24,000. What is the (Poisson approximated) probability that the first bucket has 10 or fewer strings hashed to it?

   c. Let m = 24,000. What is largest integer value n such the Poisson approximated probability that an arbitrary bucket in the hash table will have no strings hashed to it is less than 0.5 (= 50%)?

   d. Let X be a Poisson random variable with parameter λ, that is: $X \sim \mathrm{Poi}(\lambda)$. What value of λ maximizes $P(X = 5)$? Show formally (mathematically) how you derived this result. (Hint: at some point in your derivation you should be differentiating with respect to λ).

(Questions such as this allow us to compute appropriate sizes for hash tables in order to get good performance with high probability in applications where we have a ballpark idea of the number of elements that will be hashed into the table).

5. Given our recent analysis of Justice Breyer's probabilistic arguments regarding jury selection, let's consider a situation involving juries. Suppose it takes at least 9 votes from a 12-member jury to convict a defendant. Suppose also that the probability that a juror votes that an actually guilty person is innocent is 0.2, whereas the probability that the juror votes that an actually innocent person is guilty is 0.1. If each juror acts independently and if 75% of defendants are actually guilty, find the probability that the jury renders a correct decision. Also determine the percentage of defendants found guilty by the jury.

6. Consider a computer cluster (data center) of 500 web servers, where incoming requests are randomly assigned to servers with equal probability. Based on historical averages, each server in the data center receives requests at a rate of 4 per second. Some buggy server code was just deployed to all the servers in the cluster and as a result any server will crash if it receives more than 10 requests in a second. What is the approximate probability that no servers have crashed 1 second after the buggy code is deployed?

   (Hint: If you're using Wolfram to compute this probability, you're not approximating).

7. An urn contains 5 white balls and 5 black balls. Two balls are drawn randomly (without replacement) from the urn. If they are the same color, you win \$2.00. If they are different colors, you lose \$1.00 (i.e., you win –\$1.00). Let X = the amount you win.

   a. What is E[X]?

   b. What is Var(X)?

8. The number of times a person's computer crashes in a month is a Poisson random variable with $\lambda = 5$. Suppose that a new operating system patch is released that reduces the Poisson parameter to $\lambda = 3$ for 75% of computers, and for the other 25% of computers the patch has no effect on the rate of crashes. If a person installs the patch, and has his/her computer crash 2 times in the month thereafter, how likely is it that the patch has had an effect on the user's computer (i.e., it is one of the 75% of computers that the patch reduces crashes on)?

9. Say there are k buckets in a hash table. Each new string added to the table is hashed to bucket i with probability p_i, where $\sum_{i=1}^{k} p_i = 1$. If n strings are hashed into the table, find the expected number of buckets that have at least one string hashed to them. (Hint: Let X_i be a binary variable that has the value 1 when there is at least one string hashed to bucket i after the n strings are added to the table (and 0 otherwise). Compute $E\left[\sum_{i=1}^{k} X_i\right]$.)

10. Say we have a cable of length n. We select a point (chosen uniformly randomly) along the cable, at which we cut the cable into two pieces. What is the probability that the shorter of the two pieces of the cable is less than 1/4th the size of the longer of the two pieces? Explain formally how you derived your answer.

11. Let X be a continuous random variable with probability density function:

$$f(x) = \begin{cases} c\,(3 - 2x^2) & -1 < x < 1 \\ 0 & \text{otherwise} \end{cases}$$

   a. What is the value of c?
   b. What is the cumulative distribution function (CDF) of X?
   c. What is E[X]?

12. Let X be a Normal random variable with $\mu = 5$. If $P(X > 9) = 0.3$, what is the approximate value of σ?

13. The **median** of a continuous random variable having cumulative distribution function F is the value m such that $F(m) = 0.5$. That is, a random variable is just as likely to be larger than its median as it is to be smaller. Find the median of X (in terms of the respective distribution parameters) in each case below.
   a. $X \sim \text{Uni}(a, b)$
   b. $X \sim N(\mu, \sigma^2)$
   c. $X \sim \text{Exp}(\lambda)$

14. Say the lifetimes of computer chips produced by a certain manufacturer are normally distributed with parameters $\mu = 1.5 \times 10^6$ hours and $\sigma = 9 \times 10^5$ hours. The lifetime of each chip is independent of the other chips produced. What is the approximate probability that a batch of 100 chips will contain at least 65 whose lifetimes are less than 2.1×10^6 hours?

15. A disk has just 0 and 1 bits written over its entire surface. A voltage reading taken by the disk drive head at a random bit that is actually a 0 will give a reading that is normally distributed with $\mu = 4$ and $\sigma^2 = 4$. A voltage reading taken at a bit that is actually a 1 will give a reading that is normally distributed with $\mu = 6$ and $\sigma^2 = 9$. Say that α is the fraction of bits on the disk that are actually 1s. Now say that the voltage is measured at a randomly chosen bit on the disk surface and gives a voltage reading of 5. For what value of α would the probability of making an error regarding the bit value be the same, regardless of whether one concluded that the bit was supposed to be a 0 or a 1?

    More formally, let event A = bit measured is actually a 1, and event B = bit measured is actually a 0. Let random variable R = voltage value read at the bit. The probability of making an "error" is the same whether the chosen bit was a 1 or a 0 when:

    $P(A \mid R = 5) = P(B \mid R = 5) = 0.5.$

So, basically, you want to determine the value of α when P(A | R = 5) = P(B | R = 5) = 0.5.