CS 70 Discrete Mathematics and Probability Theory
Summer 2016 Dinh, Psomas, and Ye HW 7

Due Wednesday August 10 at 1:59 PM

1. Maximum Likelihood (14 points: 4/5/5)

Often, when you observe samples from a random distribution, you don't know the parameters which govern the distribution. You would like to estimate these parameters after having seen some number of samples from the distribution. This is similar to what we saw in polling-related questions where we were trying to estimate the true fraction p of some relevant subset of some population.

In this question, you are given the type of the distribution and n samples drawn from it. You are to determine the most likely parameters of the distribution which would give those samples.

For example, consider the case of n tosses of a biased coin which gives heads with probability p. You want to find the most likely p, given the number of heads we see from the coin tosses.

We define the likelihood function $l(p)$ as the likelihood of getting the results we observed, as a function of p (or more generally, all the parameters of the distribution).

In this case, if we observe k heads and $n-k$ tails from our tosses, $l(p) = p^k \cdot (1-p)^{n-k}$. The most likely value of p would be:

$$p' = \texttt{argmax}_p l(p)$$

We can do this by differentiating $l(p)$ with respect to p and setting it to 0. This would give us $p' = \frac{k}{n}$ as the only maximum, which is what we expect.

Now, do the same maximum likelihood estimate for the following. Note, this information might be useful: $\texttt{argmax}_p l(p) = \texttt{argmax}_p \log(l(p))$

(a) Consider the uniform distribution with parameters a and b ($a < b$) given by the PDF:

$$f(x) = \begin{cases} \frac{1}{b-a} & a \le x \le b \\ 0 & \text{otherwise} \end{cases}$$

Given n samples $X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n$ drawn from a uniform distribution, what are the maximum likelihood estimates of a and b?

(b) Consider the exponential distribution with parameter λ given by the PDF:

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & x \ge 0 \\ 0 & x < 0 \end{cases}$$

Given n samples $X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n$ drawn from an exponential distribution, what is the maximum likelihood estimate of λ?

(c) Consider the Gaussian distribution with parameters μ and σ^2 given by the PDF:

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

Given n samples $X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n$ drawn from a Gaussian distribution, what are the maximum likelihood estimate of μ and σ^2? Find μ first by differentiating the log-likelihood with respect to μ and then find σ^2.

2. **Proof or Counterexample** (6 points: 4/2)

   For each of the following statements determine whether it is true or false and justify your answer (either by a proof or a counter-example).

   (a) Given a non-negative continuous random variable X the following holds:

   $$\mathbf{E}[X] = \int_0^\infty \Pr[X \geq t]\, dt.$$

   Note: this is just the continuous analogue of a similar statement about discrete random variables. One approach is to try the same proof as in the discrete case with summations replaced by integrals.

   (b) There is a finite number that bounds *all* density functions $f(x)$ of continuous random variables, i.e., there exists a constant $0 < C < \infty$ such that for any probability density function f, $f(x) \leq C$ for all x.

3. **Extend it!** (6.5 points: 1.5/2/2/1)

   In this problem we will consider extending the gcd algorithm.

   (a) Note that $x \bmod y$, by definition, is always x minus a multiple of y. So, in the execution of Euclid's algorithm, each newly introduced value can always be expressed as a "combination" of the previous two, like so:
   $gcd(2328, 440)$
   $= gcd(440, 128)$ [$128 \equiv 2328 \bmod 440 \equiv 2328 - 5 \times 440$]
   $= gcd(128, 56)$ [$56 \equiv 440 \bmod 128 \equiv 440 -$ ____ $\times 128$]
   $= gcd(56, 16)$ [$16 \equiv 128 \bmod 56 \equiv 128 -$ ____ $\times 56$]
   $= gcd(16, 8)$ [$8 \equiv 56 \bmod 16 \equiv 56 -$ ____ $\times 16$]
   $= gcd(8, 0)$ [$0 \equiv 16 \bmod 8 \equiv 16 - 2 \times 8$]
   $= 8$.
   (Fill in the blanks)

   (b) Now working back up from the bottom, we will express the final gcd above as a combination of the two arguments on each of the previous lines:
   8
   $= 1 \times 8 + 0 \times 0 = 1 \times 8 + (16 - 2 \times 8)$
   $= 1 \times 16 - 1 \times 8$
   $=$ ____ $\times 56 +$ ____ $\times 16$ [Hint: Remember, $8 = 56 - 3 \times 16$. Substitute this into the above line...]

   $=$ ____ $\times 128 +$ ____ $\times 56$ [Hint: Remember, $16 = 128 - 2 \times 56$]

   $=$ ____ $\times 440 +$ ____ $\times 128$

   $=$ ____ $\times 2328 +$ ____ $\times 440$

(c) In the same way as just illustrated in the previous two parts, calculate the gcd of 17 and 38, and determine how to express this as a "combination" of 17 and 38.

(d) What does this imply, in this case, about the multiplicative inverse of 17, in arithmetic mod 38?

4. **(The last digit)** (8 points: 4/4)
Let a be a positive integer. Consider the following sequence of numbers x defined by:

$$\begin{aligned} x_0 &= a \\ x_n &= x_{n-1}^2 + x_{n-1} + 1 \text{ if } n > 0 \end{aligned}$$

(a) Show that if the last digit of a is 3 or 7, then for every n, the last digit of x_n is respectively 3 or 7.

(b) Show that there exist $k > 0$ such that the last digit of x_n for $n \geq k$ is constant. Give the smallest possible k,
no matter what a is.

5. **Modular arithmetic** (3 points: 0.5/0.5/0.5/0.5/1)

Solve the following equations (i.e., find all solutions) for x and y modulo the indicated modulus, or show that no solution exists. Show your work.

(a) $8x \equiv 1 \pmod{15}$.

(b) $6x + 14 \equiv 11 \pmod{23}$.

(c) $5x + 13 \equiv 4 \pmod{20}$.

(d) $5x + 14 \equiv 4 \pmod{20}$.

(e) The system of simultaneous equations $2x + 3y \equiv 0 \pmod{7}$ and $3x + y \equiv 4 \pmod{7}$.

6. **Enjoy the little theorems in life** (6 points: 2/3/1)

(a) What is $569^{570^{571}} \pmod{571}$?

(b) What is $5^{8^{11}} \pmod{9}$?

(c) What is $7^{2014} \pmod{31}$?

7. **Romulans Don't Know How To Count!** (9.5 points: 4.5/5)

The Romulans have reached a peace treaty with the Federation. The Romulans embarassingly admit that they do not know how to count and can only count up to five. The federation decides to teach them the art of counting and nominates Mr. Spock to help them out. Spock decides that the first course of action would be to teach them to count up to 15 with the help of Chinese Remainder Theorem. Spock decides that his first students would be Nero and Nevala. He devises the following plan. He tells one of them to count up to three and go back to one when the number counted exceeds that. He then tells the other to do the same thing up to five. Then he can put together the answers (say, 2 and 4) to find the unique number between 1 and 15 that has these remainders $\pmod 3$ and $\pmod 5$, respectively. (In this example, the only such number is 14.)

Here we show that this method always works.

(a) Let $N = p_1 \times ... \times p_k$ be the product of k distinct primes. Now consider two integers x, y between 1 and N, and their remainders $x_i \equiv x \pmod{p_i}$,$y_i \equiv y \pmod{p_i}$. Show that, if $y_i = x_i, \forall i = 1, ..., k$, then $x = y$. (Hint: What does $x_i = y_i$ imply for $x - y$?)

(b) Show that there is a one-to-one correspondence between integers between 1 and N and their k-tuple of remainders modulo $p_i, i = 1, ..., k$. (Hint: How many k-tuples are there? Then use the previous part.)

(c) Extra challenge, **no credit**: Find an efficient way of going back from the k-tuple of remainders $x_i = x \pmod{p_i}$ to the unique integer between 1 and N that has these remainders.

8. **Poker Mathematics** (5 points)

A *pseudo-random number generator* is a way of generating a large quantity of random-looking numbers, if all we have is a little bit of randomness (known as the *seed*). One simple scheme is the *linear congruential generator*, where we pick some modulus m, some constants a, b, and a seed x_0, and then generate the sequence of outputs $x_1, x_2, x_3, x_4 \ldots$ according to the following equation:

$$x_{t+1} = (ax_t + b) \bmod m$$

(Notice that $0 \leq x_t < m$ holds for every t.)

You've discovered that a popular web site uses a linear congruential generator to generate poker hands for its players. For instance, it uses x_0 to pseudo-randomly pick the first card to go into your hand, x_1 to pseudo-randomly pick the second card to go into your hand, and so on. For extra security, the poker site has kept the parameters a and b secret, but you do know that the modulus is $m = 2^{31} - 1$ (which is prime).

Suppose that you can observe the values x_0, x_1, x_2, x_3, and x_4 from the information available to you, and that the values $x_5, \ldots, x_9$ will be used to pseudo-randomly pick the cards for the next person's hand. Describe how to efficiently predict the values $x_5, \ldots, x_9$, given the values known to you.

9. **Tweaking RSA** (9 points: 3/2/4)

(a) You are trying to send a message to your friend, and as usual, Eve is trying to decipher what the message is. However, you get lazy, so you use $N = p$, and p is prime. Similar to the original method, for any message $x \in \{0, 1, \ldots, N-1\}$, $E(x) \equiv x^e \pmod{N}$, and $D(y) \equiv y^d \pmod{N}$. Show how you choose e and d in the encryption and decryption function, respectively. Prove that the message x is recovered after it goes through your new encryption and decryption functions, $E(x)$ and $D(y)$.

(b) Can Eve now compute d in the decryption function? If so, by what algorithm? (Eve knows the public key.)

(c) Now you wonder if you can modify the RSA encryption method to work with three primes ($N = pqr$ where p, q, r are all prime). Explain how you can do so and prove that you can successfully decode the encoded message.

10. **Polynomial Interpolations** (6 points: 3/3)

(a) Consider the set of four points $\{(0, 1), (1, -2), (3, 4), (4, 0)\}$, construct the unique degree-3 polynomial (over the reals) that passes through these four points by writing down and solving a system of linear equations.

(b) Use Lagrange interpolation to find a polynomial $p(x)$ of degree at most 2 that passes through the points $(1,2)$, $(2,3)$, and $(3,5)$, working in $GF(7)$. In other words, we want $p(x)$ to satisfy $p(1) \equiv 2 \pmod 7$, $p(2) \equiv 3 \pmod 7$, and $p(3) \equiv 5 \pmod 7$. Show your work clearly and use the same notations as in Lecture Note 22.

11. **Alice wants to talk** (11 points: 4/3/4)

Alice has a message of length $n = 3$ for Bob. She also has another message of length $n = 3$ for Charles. Her message for Bob is $a_0 = 4$, $a_1 = 3$, and $a_2 = 2$. And her message for Charles is $a_0 = 1$, $a_1 = 2$, and $a_2 = 2$.

(a) If Alice accounts for $k = 1$ general errors, then what are Alice's augmented messages to Bob and Charles (each modulo 5)?

(b) Alice now transmits the augmented message intended for Bob over an erasure channel. Bob receives only $P(0)$, $P(2)$, and $P(4)$. The rest are erased. How does Bob recover Alice's message? Show your work in detail.

(c) Alice then transmits the augmented message intended for Charles over a noisy channel. Charles receives the entire message but now $P(2)$ is corrupted to $P(2)+2 \pmod 5$. Charles doesn't know where the error is but he does know that at most one error has occured. How will Charles recover Alice's message?

12. **Why work with primes?** (20 points: 3/5/4/5/3)

In class, you learned about erasure codes and error correcting codes, and prime numbers played a central role in both kinds of codes – since all calculations were supposed to be done modulo a *prime number*. In this problem, we will see why this is a crucial requirement, and explore what happens if this requirement is relaxed in a naive manner.

For this problem, assume that Alice wants to send n packets to Bob, across an "erasure channel"modeled as discussed in class. Let us say all calculations are done modulo $N = 12$ (note that this is *not* a prime number).

As discussed in class, let us say Alice sends $n+1$ packets to Bob, and Bob receives at least n of these packets intact. That is, the channel can erase at most 1 packet, and if it does so, Bob gets to know which packet was erased (although he does not know the contents of the erased packet).

(a) Suppose $n = 1$. That is, Alice wants to send only 1 packet to Bob (plus one redundant packet to compensate for erasure). Would the scheme discussed in class work with $N = 12$? What are all the possible 2-packet lists that Alice could transmit? In each case, would Bob be able to recover Alice's message in spite of a possible erasure? Would Alice or Bob face any problems because they are doing their calculations modulo 12?

(b) Now suppose $n = 2$. That is, Alice now wants to send 2 packets to Bob (plus one redundant packet to compensate for erasure). Now, would there be any problems because $N = 12$?

(c) Now let $n = 3$ (3 packets plus one additional packet to compensate for erasure). Assume that Alice wants to encode messages into "systematic" codewords (with the first few evaluations of the polynomial being the message itself). Prove that Alice can no longer send arbitrary messages of her liking to Bob, by showing that it would be impossible for Alice to send the message $(11,6,2)$. Find 2 other examples of messages that Alice cannot send to Bob.

(d) Why does Lagrange interpolation fail when Alice tries to send the messages in part (c) above? Does Lagrange interpolation fail for any message in part (a) or part (b)? Why or why not?

(e) Suppose Alice chooses a value for n, and restricts herself to only the messages that are possible to send in modulo 12 arithmetic. For any such n, and any such message, would Bob be able to recover Alice's message in spite of a possible packet erasure? If so, how? If not, provide a counter-example.