Issued: March 7, 2003   # Problem Set 5   Due: March 14, 2003

Problem 1: The Registrar's Worst Nightmare

To reinforce Bayes' Theorem, which was covered last week, here's another probability problem.

There are two ways for MIT students to register for 2.110/6.050. Either put it on your Registration Form on Registration Day (call this event R) or use an Add Form after Registration Day (event A). (Assume you cannot do both.) Those who for some reason don't want to take 2.110/6.050 do neither (call this event N). For MIT students, events R, A, and N form a partition. After processing by the Registrar, students may actually be enrolled in 2.110/6.050 (call it event E) or unenrolled (event U). Events U and E also form a partition. (Students can discover whether they are enrolled by checking their status of registration.)

You would probably want things to work so that all Forms are handled properly, i.e., $p(E \mid R) = 1$, $p(E \mid A) = 1$, and $p(U \mid N) = 1$. Unfortunately, one year, 20% of the Add Forms got lost in the mail, 15% of the 2.110/6.050 entries on Registration Forms were illegible because of ink smears, and errors in typing enrolled 2% of the people who didn't want to take the course, by mistake. Knowledge you gain in one part of the problem may be applied to all subsequent parts of the problem.

a. If your friend Alice submitted an Add Form for 2.110/6.050, what is the probability $p(E \mid A)$ that she is enrolled?

b. A study showed that of those that were enrolled in 2.110/6.050, 25% did so by using an Add Form. If your friend Bob is enrolled, what is the probability $p(A \mid E)$ that he submitted an Add Form?

c. Suppose that 5% of the students are enrolled in 2.110/6.050, so $p(E) = 0.05$. Write symbolically and find numerically the probability that a random freshman, say your friend Carol, BOTH submitted an Add Form AND is enrolled.

d. What is the probability that Carol submitted an Add Form for 2.110/6.050?

e. You have just learned that Carol is actually enrolled in 2.110/6.050, a fact which you did not know in parts c. and d. Now what is the probability that she submitted an Add Form?

f. What is the probability that a student, David, selected at random from the whole of the student body, did not put 2.110/6.050 on his Registration Form and did not use an Add Form?

g. What is the probability that David is enrolled in 2.110/6.050?

h. What is the probability that Ed, who did not put 2.110/6.050 on the Registration form and who also did not use an Add Form, is enrolled in 2.110/6.050?

Problem 2: Special Orders Don't Upset Us

Buzz, the hot new dining spot on campus, emphasizes simplicity. It only has two items on the menu, burgers and zucchini. Customers make a choice as they enter (they are not allowed to order both), and inform the cooks in the back room by shouting out either "B" or "Z". Unfortunately the two letters sound similar so 8% of the time the cooks misinterpret what was said. The marketing experts who designed the restaurant guess that 90% of the orders will be for burgers and 10% for zucchini.

The cooks can hear one order per second. The customers arrive at the rate of one per second. One of the chefs says that this system will never work because customers can only send one bit per second, the rate at which orders can be accepted, so you could barely keep up even if there were no noise in the channel. You are hired as an outside consultant to deal with the problem.

    a. What is the channel capacity C of this communication channel in bits per second?

    b. What is the information content per order?

    c. Is it possible, through some coding arrangement, to transmit orders reliably at your target rate of one per second?

    d. If the restaurant becomes more popular, what is the maximum rate your system can handle, measured in orders per second (assuming you could find an effective source code and channel code)?

    e. You decide to process orders two at a time. Devise a Huffman code in which you transmit the information for two orders in less than 2 bits on average. Will this scheme work, given the channel capacity?

Problem 3: New, "Improved" Web Protocol

Ben Bitdiddle is designing a new protocol (a replacement for TCP) to transmit Web pages from a server to a client. A page consists of a file of any length.

All packets sent by the client and by the server include cyclic redundancy checksums (CRC) so that errors in a packet may be detected. Assume that this always works correctly, and that any packets with an error are immediately discarded. Ben realizes that the Internet is error-prone, and any packet sent out may get lost and not arrive at its destination. He also realizes that some packets may travel faster than others. He is designing his protocol to deal with these problems.

Here's how the server works:

- Wait for a request from a client, either for an entire Web page or a list of specific numbered packets from a Web page.

- Send an acknowledgement packet to the requesting client.

- Put the IP addresses of the requesting server at the start of the file and save the altered file (this serves as a log of activity)

- Break up the file (as just altered)into packets of 1000 bytes each (the last packet may be smaller). Add a number to each packet so that the client can put them back together in case they arrive out of order. The packets are numbered in ascending order, from 1 through the number of packets in the file.

- If the request was for the entire Web page, send the packets, one at a time, in any order, to the client without waiting for any response from the client.

- If the request was for specific packets, send those, one at a time, in any order, to the client.

Here's how the client works:

- Send the server a request for an entire Web page.

- Discard the acknowledgement packet.

- When packet 1 is recieved, strip off all the accumulated IP addresses the server had put on the file.

- Store each received packet, along with its number.

- If ten seconds go by without receiving any packets, this is called a timeout. This happens, for example, after all packets have been received. After a timeout, see if you have packets with all numbers from 1 through the greatest-numbered packet you have received. If so, display the page for the user. If not, send a request to the server for the missing packets, and go back to accepting packets as before.

Ben's protocol is incomplete, since it does not cover the case where no packets are received before timeout.

a. Complete the protocol by saying what the client should do in this case. There are many possible answers; which one you give may influence your answers to the rest of this problem.

b. Despite Ben's attentiveness to detail, he finds that every so often, a Web page requested via his protocol is not displayed quite right. Apparently his protocol has bugs or, as they are sometimes called, issues. Does Ben's protocol properly handle the possibility that the client's requests might get lost? If so, explain how recovery from this error happens. Or if not, suggest a way to fix this bug.

c. Please describe all the other issues you can find with the protocol above. We are aware of several; you will get full credit for any one issue correctly identified and described. (In this problem we are only concerned about whether the file is eventually transferred without error, and not about how fast the transfer happens or how many resources are needed.)

d. Provide a patch for at least one of the issues you identify. That is, describe a (hopefully simple) way to alter Ben's protocol in order to prevent at least one of the issues from part c. Be sure to say which of those issues your patch addresses.

Turning in Your Solutions

If you used MATLAB for this problem set, you may have some M-files and a diary. Name the M-files with names like `ps5p1.m`, `ps5p2.m`,..., and name the diary `ps5diary`. You may turn in this problem set by e-mailing your M-files and diary. Do this either by attaching them to the e-mail as *text* files, or by pasting their content directly into the body of the e-mail (if you do the latter, please indicate clearly where each file begins and ends). The deadline for submission is the same no matter which option you choose.

Your solutions are due 5:00 PM on Friday, March 14, 2003. Later that day, solutions will be posted on the course website.

6.050J / 2.110J Information and Entropy
Spring 2008