ECE 3077, Summer 2014

Homework #3

Due Thursday June 5, in class

Reading: B&T Chapter 1.6–2.2

1. Using you class notes, prepare a 1-2 paragraph summary of what we talked about in class in the last week. I do not want just a bulleted list of topics, I want you to use complete sentences and establish context (Why is what we have learned relevant? How does it connect with other things you have learned here or in other classes?). The more insight you give, the better.

2. Suppose that I select n people at random to attend a surprise party. Assume that every person I select has an equal probability of being born on any day during the year, independent of everyone else, and ignore for the moment the existence of February 29 (so that there are only 365 possible birthdays, and they occur with equal probability). Find an expression for the probability that each person at the party has a different birthday (as a function of n). Using this expression, tell me how many people I need to select to ensure that the probability that two people share a birthday is greater than 50%? Are you surprised?

3. A well-shuffled 52-card deck is dealt to 4 players. Find the probability that each of the players gets an ace.

4. Suppose that the Georgia Tech baseball team has a 58% chance of winning any given game. Modelling a series of games as a sequence of Bernoulli trials, what is the probability that Georgia Tech will win at least 3 of the next 5 games? What is the probability they win at least 4 of the next 7? What about winning at least 16 of the next 31?[1]

5. Suppose that a family has 5 natural children and has adopted 2 girls. Each natural child has equal probability of being a girl or a boy, independent of the other children. Find the pmf of the number of girls out of the 7 children.

6. Let C be a random variable associated with the result of a fair coin flip; if the coin lands on "tails", $C = -1$, if the coin lands on "heads", $C = 1$ and so

$$p_C(k) = \begin{cases} 1/2, & k = -1 \\ 1/2, & k = 1 \\ 0, & \text{otherwise.} \end{cases}$$

[1] Feel free to use MATLAB to do the calculation — just document what you did. You might find the command `nchoosek` useful.

Let D be a random variable associated with a fair die roll:

$$p_D(k) = \begin{cases} 1/6, & k = 1, 2, 3, 4, 5, 6 \\ 0, & \text{otherwise.} \end{cases}$$

(a) Find the pmf for $X = C \cdot D$. Sketch it.

(b) Find the pmf for $X = C + D$. Sketch it.

7. Roy makes free throws with probability p. Let X be the number of free throws it takes for Roy to make 10. Find the pmf for X, and then use MATLAB to make a stem plot of $p_X(k)$ for $p = 0.75$ and a reasonable range of k.
(Hint: Start slowly. Clearly, $X \geq 10$. What is $p_X(10)$? What is $p_X(11)$?)

8. We are rolling a fair die, and keeping a running sum (that is, the sum of all of the results so far). Let X be the number of rolls it takes for this sum to be ≥ 100. The pmf for X is complicated; in this question, we will estimate it through simulation using MATLAB. Note that $X <= 100$.

   Here is a summary of what we will do:

   - Allocate an array of length 100 called `pX` that will hold our estimates of $p_X(k)$ for $k = 1, \ldots, 100$. (Note that the first 16 entries will always remain 0.)
   - Set the number of experiments `Q`. Each experiment consists of rolling the die until the sum is 100.
   - Loop over `Q` iterations. At each iteration, we set the roll count `x=0` and the running sum `sum=0`, then generate random die rolls, incrementing `sum` by the result and `x` by one after each, until `sum >= 100`. This inner loop is easily accomplished with a `while` loop. A single die roll can be simulated using[2] `ceil(6*rand(1))`.
   - Increment the corresponding entry of `pX`.
   - At the end, make sure to normalize `pX` so it sums to 1. (Take your tallies and divide them by `Q`).

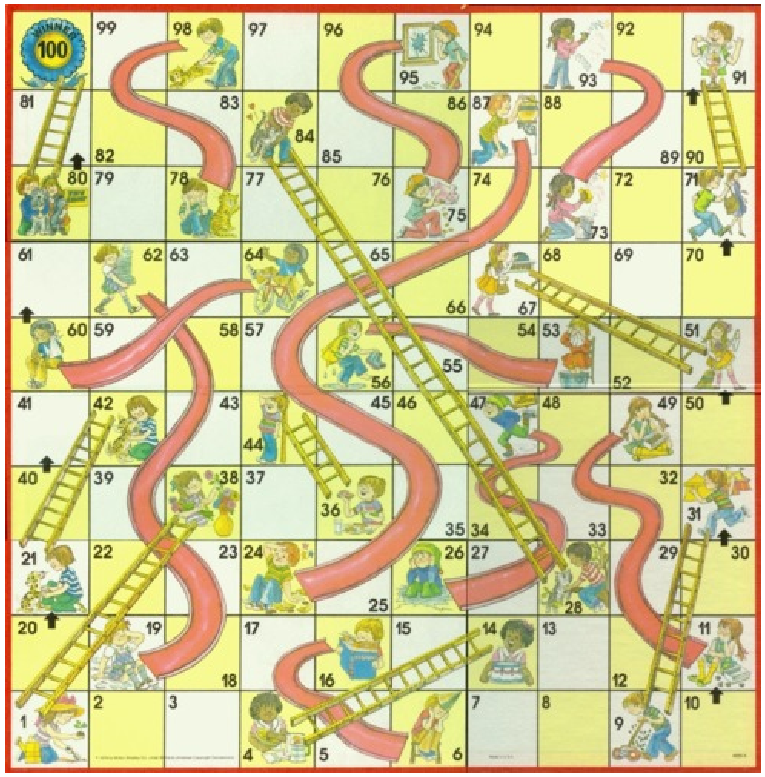
   Run your script for `Q=100000`, and make a stem plot of the resulting `pX`. Turn in your code and your plot.

9. Chutes & Ladders is a board game for young children; I am sure that many of you have played it (my four year old son loves it). It is simple: there are 100 squares on the board labeled $1, 2, \ldots, 100$, you start just outside the square labeled 1, flick a spinner that returns a number between 1 and 6 (presumably uniformly at random), and then move your piece forward that number of spots. Whoever gets to square 100 first wins.

   What makes the game fun is that there are a network of chutes and ladders. If you land on a square with a ladder, you get to move up to a square with a higher value

[2] This command generates a continuous-valued real number between 0 and 1, multiplies it by 6 so it is between 0 and 6 and then rounds up to make it an integer.

(wherever that ladder leads). If you land on a square with a chute, you move down to a square of a lower value (again, wherever that chute leads). Here is a picture of the board[3].



Let X be a random variable for the number of turns it takes for you to reach (or pass) square 100. Write a MATLAB script the estimates the pmf of `X` through simulation[4] by simulating one million games. Turn in a stem plot of your estimated pmf $p_X(k)$ for a reasonable range of k. How long was the longest game in your one million sims? (There is no upper limit on the value of X in this case.) Which value of X is most probable?

You will find the file `chutesladders_map.mat` useful; the vector `map` codifies where the chutes and ladders take you — if you land on square `i`, you move to square `map(i)`, and if there is no chute or ladder there, then `map(i) = i`.

[3]You can also download the file `chutesladders_board.jpg` from T-square

[4]Note that without the chutes and ladders, this is exactly the same as the previous question.