STAT 325 Introduction to Probability Models Spring 2012

Investigation 2: Hog the Dice (assigned on Tues Apr 17, due on Thur Apr 26)

You may work with one partner, submitting one report with both names, provided that you both contribute substantially to the work. Word-processed reports are strongly preferred to hand-written ones. Please integrate computer output into your report.

The dice game called "hog" is played as follows: You decide how many (fair, six-sided) dice to roll. If any of your dice land on 1, then your score is 0. But if none of your dice land on 1, then your score is the sum of the dots landing up on the dice. That's all there is to it. The strategy, of course, comes in deciding how many dice to roll. You will now investigate the question of how many dice to roll in order to maximize the expected value of your score.

1. Introduction:

a) Provide intuitive arguments for why it's probably not optimal to roll either 1 die or 100 dice.

Let n denote the number of dice to be rolled, and let the random variable Y represent your score.

b) For the case of $n = 1$ die, determine the probability mass function for Y and calculate E(Y).

c) Repeat b) for the case of $n = 2$ dice. Include an explanation for the probabilities that you calculate in the pmf.

2. Simulation:

The file dicehog.txt contains the following R code for simulating this dice game:

```
# must first enter n = number of dice, N = number of repetitions
#
score = rep(NA, times = N)
die = (1:6)
for (i in 1:N) {
   dice = sample(die, n, replace=TRUE)
   isitone = (dice == 1)
   anyones = (sum(isitone) > 0)
   if(anyones == TRUE) {score[i]=0} else {score[i] = sum(dice)}
}
table(score)
hist(score)
mean(score)
```

d) Write an explanation for what each line in this program does.

e) Execute this program for 100,000 repetitions with $n = 1$ and with $n = 2$. Submit the table and histogram of results, and also report the average score in the 100,000 repetitions (for each value of n). Also comment on whether the simulation results are consistent with your answers to b) and c).

f) Now run this program for 100,000 repetitions with $n = 3, 6, 10$, and 20. Submit the table and histogram of results, and also report the average score in the 100,000 repetitions (for each value of *n*).

g) Now run this program for 100,000 repetitions for all remaining integer values of *n* between 1 and 20. Do not bother to submit all of these tables and histograms. Do create a list of all integer values of *n* from 1 to 20, along with the average score achieved for that value of *n*. Also produce and submit a graph of these average scores vs. number of dice.

h) According to your simulation results, what value of *n* maximizes the long-run average score for this game?

3. Exact analysis:

Now you will solve this problem analytically, using the following result:

Let Y be a random variable, and let A be any event. Then $E(Y) = E(Y\|A)\times\Pr(A) + E(Y\|A^c)\times\Pr(A^c)$

Continue to let Y be the score, and let A be the event that at least one die (among *n* dice rolled) lands on 1.

i) Determine $\Pr(A)$ and $\Pr(A^c)$.

j) Determine $E(Y|A)$. [*Hint*: This is trivial, because given A, there's only one possible value for Y.]

k) Determine $E(Y|A^c)$. [*Hint*: First think of Y as $X_1 + X_2 + \ldots + X_n$, where X_i is the result of die *i*. And then determine $E(X_i|\, A^c)$, which is the expected result of a die roll given the knowledge that the die did not land on 1.]

l) Now use your answers to i), j), and k), along with the result given above, to determine E(Y) as a function of the number of dice *n*.

m) Produce and submit a graph of this function for values of *n* from 1 to 20. Is this graph similar to the graph of your simulation results in g)?

n) What value(s) of *n* maximizes the expected score for this game, and what is that expected score?

4. Generalization:

Now suppose that you use fair *k*-sided dice, where *k* is an integer > 1.

Again start with a simulation analysis.

o) Modify the R code in order to conduct a simulation analysis for determining the optimal number of dice (*n*) based on any value of *k* that the user might enter. Submit your code, and highlight the new or revised parts of the code.

p) Run the code with 100,000 repetitions for the following values of *k*: 10, 20, 50. In each case produce a graph of the average points as a function of the number of dice (*n*), and report the optimal value(s) of *n* along with the average number of points for that value.

Now proceed to an analytical analysis.

q) Determine an expression for E(Y) as a function of *k* and *n*. Be sure to show the steps in your derivation. [*Hint*: Use the same result that you used for six-sided dice above.]

r) Use this expression to determine E(Y) as a function of *n* for the following values of *k*: 10, 20, 50. In each case produce a graph of E(Y) as a function of the number of dice (*n*), for reasonable values of *n*. Also report the optimal value(s) of *n* and their associated expected numbers of points.

s) Based on your analysis thus far, make a conjecture for the optimal number of *k*-sided dice to roll.

t) Use calculus to prove your conjecture.