CFCS1: Practical 5
Matlab Probability

Peter Bell

February 21, 2008

1   Introduction

In this practical you will use Matlab to help solve a variety of problems in probability theory. You should be familiar with the material covered in lectures 17 and 18. No other resources are required.

You will use two of Matlab's random-number functions, `rand` and `randperm`, to simulate random experiments. These can be used to check your solutions to simple problems – for more complex problems where it is difficult or impossible to find an analytic solution, these simulation-based methods are often a good alternative.

2   Simple simulation

Carrying out probability experiments is known as *sampling*. In probability theory it is useful to distinguish between *sampling with replacement* and *sampling without replacement*. In the former case, the conditions of a probability experiment remain the same from one experiment to the next, so that the probabilities do not change. In the latter case, the conditions change, based on the outcome of previous experiments. For example, consider selecting balls from a bag containing 3 red balls and 3 green balls. If the balls are replaced after each experiment, the probability of selecting a green or red ball will always be 0.5. However, if the balls are not replaced, selecting a green

ball on the first experiment will make selecting a green ball *less likely* for the second experiment. Examples of sampling with replacement include tossing a coin or throwing a die. Examples of sampling without replacement include lottery draws or selections for a football team.

In Matlab, you can use the `rand` function to simulate sampling with replacement, and `randperm` to simulate sampling without replacement.

2.1 Sampling with replacement

The `rand(n,m)` function generates an $n \times m$ matrix of random numbers, uniformly distributed between 0 and 1. A vector of these numbers can be used to simulate repeated experiments, where each possible outcome has a fixed probability. For example, consider an experiment where event A occurs with probability 0.5, B occurs with probability 0.3 and C occurs with probability 0.2. This experiment could be simulated as follows:

- Set `w = rand(1,1)`.

- Then if $w \leq 0.5$, say event A has occurred, if $0.5 \leq w \leq 0.8$ then say B has occurred, and otherwise say C has occurred.

If the experiment is repeated many times, the results can be used to estimate event probabilities. The more experiments there are, the more accurate the estimation becomes. You will investigate this for the simple example of rolling a four-sided die, where each number from 1 to 4 occurs with probability 0.25. You should:

1. Use `w = rand(100,1)` to generate a 100-dimensional random vector, simulating 100 throws of the die.

2. Say that a 1 has been thrown on the ith experiment if the `w(i)` is less than or equal to 0.25. Count how many times a 4 has been thrown from the 100 experiments – call it n.

3. Estimate the probability of throwing a 4 as:

$$p(4) = \frac{n}{100}$$

4. How close is this estimate to the true value, 0.25? Repeat the steps above for 500, 1000, 5000 and 10000 experiments. What do you find?

If each experiment involved rolling 2 dice, you could simulate N experiments using `rand(N,2)`, with each row corresponding to one experiment.

2.2   Sampling without replacement

The `randperm(m)` function produces a random ordering of the numbers from 1 to m. Suppose you wish to choose two balls from a bag containing three red balls and three green balls. A single experiment can be simulated as follows:

```
bag = [ 1 1 1 2 2 2 ]   % 1=red ball, 2=green ball
perm = randperm(6)       % random ordering of the numbers 1 to 6
                         % simulating the order in which balls are drawn

draw = perm(1:2)         % only consider the first two balls
balls = bag(draw)        % find the actual two balls drawn, in order
```

You will repeat this simulation to estimate the probability of drawing a red ball first, then a green ball second.

1. Write a Matlab program to repeat the above simulation 100 times. Produce a count, n, of how many times that a red ball was drawn first, and a green ball second.

2. After repeating the simulation, estimate the probability using

$$p(red, green) = \frac{n}{100}$$

3. Calculate the true probability on paper. How close is your estimate?

4. Repeat the above steps for 500, 1000, 5000 and 10000 experiments. What do you find?

3 Problems

On paper, calculate the solutions to the following problems (use the lecture notes to help). In each case, check your answer by writing a Matlab program to simulate the problem and estimating the required probability from a very large number of experiments, using the methods from Section 2. You will need to decide whether the problem is equivalent to sampling with replacement or sampling without replacement.

1. A fair coin is tossed four times. What is the probability of getting two heads and two tails (in any order)?

2. A lottery has balls numbered from 1 to 10. Five balls are drawn, and the winner must match all five balls (ordering doesn't matter). What is the probability of winning?

3. Two four sided dice are thrown at the same time. This is repeated three times. What is the probability that a double 4 is thrown at least once out of the three times?

4. (*Optional*) A four-sided die is thrown 6 times. What is the probability of throwing two 4s consecutively? (This is much easier to simulate than to calculate by hand!)