Simulating our Initial States

July 4, 2013

Abstract

Generating an initial state value when given the pi matrix. The pi matrix is a vector that represents the probability mass function for the initial states

1 Introduction

These notes are a description of how we can use Matlab to establish an initial state when we are already given values for the pi matrix.

These notes also provide additional explanation and context for the matlab codes written on July 3rd with Todd: Generate RandomOutcome

2 Some Background about the Goal

2.1 pi matrix

We are playing god and we first want to generate a set of data. Initially this will be a set of values at discrete time points: t=1, t=2, ... t=n

- Here t=1 will represent the initial time point, when we are in our initial state

To generate this set of data, we first need a pi matrix. this matrix will tell us the probability that we are initially in a given state.

- example: we have two possible states: state 1 and state 2

  - what is the probability that when our experiment begins, we are in either state 1 or state 2?
  - the pi matrix can give us this probability.

- lets start by saying that the pi matrix is already given to us.
  - example: the probability of initially being in state 1 is equal to 1/4 (i.e. pi(1)=1/4)
  - as a result, pi(2)=3/4, (the probability that we are initially in state 2 at time 1 is 3/4
  - Reminder: the pmf vector (the sum of the probabilities of all possible states) must add up to 1

- now, we are god and we want to simulate data. so how do we randomly generate our initial state?

3 Generating the initial state when pi matrix is given

3.1 using rand in matlab

lets use rand to generate values for a random variable Z

- we will define Z as a continuous random variable that is uniform on the 0,1 line.
  - this means that all values between 0 and 1 are equally likely

next we can evaluate the probability density function for our uniform random variable

- lets evaluate the probability that any given value of Z (randomly generated by matlab) falls within a given interval.
  - because this is a PDF, we know that the entire area under the curve must sum to 1
  - we also know the end points (a,b) of our uniform random variable: U(a,b). Because b-a (1-0)= 1, we know that the height of the curve (1/b-a) must also equal 1.

  - Therefore we can define the probability that a randomly generated value of Z falls within a given interval.

Remember we are using this strategy to establish our initial state. Therefore, we want to define our intervals on Z so that the PDF (probability that the random variable will fall within a particular region. Defined as the area under the curve for a particular interval) in each interval is equal to our given pi matrix probabilities (for this example we are using 1/4 and 3/4.)

- So we want to have two intervals on our continuous uniform random variable. we want the area under the curve of one interval to equal 1/4, and the other area under the other interval of the curve to equal 3/4.
  - (note: the number of defined intervals will equal the number of initial states. And the sum of all integrals (i.e sum of the probabilities for all defined intervals) must add to 1.
- so we want one of our integrals (PDF value for a given interval) to equal 1/4. as a result we must set our (a,b) endpoints so that the area under the curve between those two points is equivalent to 1/4. Because we know that the height of the curve is 1, we know that b-a must equal 1/4.
  - So we can say that the probability that Z falls in this interval is 1/4. and therefore we can say that if Z falls in this interval, we are in state 1
- Similarly, for state 2, where the initial probability of being in that state is 3/4,we can set the endpoints of our interval such that b-a must equal 3/4. So b=1 and a=1/4. (remember b and a are values taken by our random variable Z which is generated using the rand command in matlab

Therefore, one easy way to use matlab to randomly establish our initial state at t=1 is to set the interval for state 1 as follows:

- Z is between 0 and 1/4. In other words:
  - Z is greater than or equal to zero and
  - Z is less than 1/4

Then we know that for any value of Z between 0 and 1/4, the probability of this value is 1/4.

So based on all of this, and given an initial pi matrix PMF, we can use matlab to randomly generate our initial state.

- we can say that Z=rand. thus when we call Z, matlab will generate a random number between 0 and 1. All values are equally likely in this interval.
- we can then say that if Z is between 0 and 1/4, then x(1)=1, where x(1) represents our initial state at time t=1.
- else if Z is between 1/4 and 1, then x(1)=2 (state 2).

So essentially this is what is happening:

1. we use our given pi matrix to establish intervals on the (0,1) line. The PDF for each interval corresponds to the PMF values provided by the pi matrix.
2. we then ask matlab to randomly generate a value between 0 and 1 for our random variable Z. All values in the interval between 0 and 1 are equally likely
3. we then evaluate which interval our Z value lies in. The PDF of that interval is used to define our initial state. In other words, if Z lies in an interval where the PDF of that interval is 1/4, and the initial probability of being in state 1 is equal to 1/4, then we can say that at our initial time point, t=1, we are in state 1