TASK

If you break a stick uniformly in two places, you will be left with three segments. Write an algorithm for computing the probability that the three segments form a triangle. This algorithm is supposed to employ Metropolis - Hastings ideas and serve as an independent verification of our theoretical calculations.

SOLUTION

First, we note that the probability can be calculated quite easily on a piece of paper...

Let A and B be the two break points falling on a stick of length L. To distinguish between the break points, we will order then chronologically, A being the older one. Two cases are to be considered:

CASE 1: 0 <= A <= B <= L;

and

CASE 2: 0 <= A < B <= L.

The three pieces will form a triangle if none is longer than the sum of the others. In terms of A and B, the conditions are the following.

CASE 1:

A < L - A	<=========>	A < L/2;
L - B < B	<=========>	B > L/2;
B - A < A + (L - A)	<=========>	B < A + L/2.

CASE 2: B < L/2, A > L/2 and A < B + L/2.

Now we are capable of plotting the acceptable region on the plane. We see that it consists of two small triangles: one triangle corresponds to case 1 and the other triangle corresponds to case 2. The area of the acceptable region can be calculated as

1/2 * (L/2)^2 + 1/2 * (L/2)^2.

The space of all elementary possibilities is the square [0,L]*[0,L]. It has the area of L^2. Since (A,B) are uniformly distributed on [0,L]*[0,L], the probability of the three segments forming a triangle equals

(the area of the acceptable region) / (the area of the space of all elementary possibilities) =

= (1/2 * (L/2)^2 + 1/2 * (L/2)^2) / (L^2) = 1/4.

~~~~~~

Next, we are going to build an algorithm to verify our theoretical result...

Now let LF = min(A,B) be the left break point and RT = max(A,B) be the right breakpoint. It is a straightforward exercise to determine conditional and marginal distributions of LF and RT. First we focus on marginal distributions:

Fm_LF(x) = P(LF <= x) = 1 - P(LF > x) = 1 - P(min(A,B) > x) = 1 - P(A>x,B>x) =
~~~~~~

= 1 - P(A > x)*P(B > x) = 1 - ((1-x)/L)^2.

Fm_RT(y) = P(RT <= y) = P(max(A,B) <= y) = P(A <= y, B <=y) = P(A <= y) * P(B <=y) =

= (y/L)^2.

And now we are ready to calculate conditional distributions. For any x <= y,

Fc_LF(x | y) = P(LF <= x | RT = y) = P(min(A,B) <= x | max(A,B) = y) =

= 1/2 * P(min(A,B) <= x | max(A,B) = y, A < B) + 1/2 * P(min(A,B) <= x | max(A,B) = y, A <= B) =

= 1/2 * P(A <= x | A < y) + 1/2 * P(B <= x | B <= y) = x/y.

Similarly, for any x <= y,

Fc_RT(y | x) = P(RT <= y | LF = x) = 1 - P(RT > y | LF = x) = 1 - (1-y)/(1-x).

Using functions Fc_LF() and Fc_RT(), random variables LF and RT can be simulated one from the other. Here we employ the rule:

if F(x) is a cumulative distribution function (cdf) of a given distribution, then random variable F_{-1}(U) has this distribution, where U is uniformly distributed on [0,1]. **(***)**

NOTE: of course, we did not have to derive conditional distributions of TF and RT to simulate the three random segments of the line. We could have easily simulated the marginals of A and B and seen if the three segments form a triangle. Focusing on LF and RT was necessitated by the requirement to use **Metropolis algorithm**.

~~~~~~

The algorithm below employs **Gibbs sampling**, which says: to simulate a joint distribution of (LF,RT), we can simulate LF given RT and RT given LF long enough.

```
% INITIALIZATION
Counter = 0
Random.Seed(0)

for(S = 1:Sample_Number)
          % SIMULATING INITIAL VALUES OF Z AND W
          U = Simulated_Uniform(0,1)
          LF = Fm^{-1}_LF(U)        % Using the marginal cdf of LF and rule (***) to simulate LF.
          U = Simulated_Uniform(0,1)
          RT = Fm^{-1}_RT(U)      % Using the marginal cdf of RT and rule (***) to simulate RT.

          for(iter = 1:(Burn.In+1))
             % THE MAGIC OF GIBBS SAMPLING.

             % Randomly selecting LF or RT.
             U = Simulated_Uniform(0,1)
             if( U <= 1/2 )
                       U = Simulated_Uniform(0,1)
                       LF = Fc^{-1}_LF(U | RT)   % Simulating LF using its conditional cdf
                                   % and the current value of RT.
             else
```
~~~~~~

```
                U = Simulated_Uniform(0,1)
                RT = Fc^{-1}_RT(U | LF)   % Simulating RT using its conditional cdf
                                          % and the current value of LF.
            end
        end

        % CHECKING IF ONE CAN MAKE A TRIANGLE
        % OUT OF THE SIMULATED SEGMENTS
        if(LF < L/2 & RT > L/2 & RT < LF + L/2)
            Counter = Counter + 1
        end
end

Prob_Of_Triangle = Counter / Sample_Number.
```

~~~~~~

The proposed computational algorithm uses Gibbs sampling. So how is our work related to the ideas of Metropolis?... It turns out that the employed version of Gibbs sampling is a particular case of the Metropolis-Hastings algorithm. Let us denote W = (LF,RT).

- Gibbs sampling simulates a Markov chain of different realizations of W in multiple steps (just like in the Metropolis-Hastings algorithm).

- At each step we have a current value of W and propose a new value W' [just like in Metropolis].

- We propose the new value W' with the proposal density Q(w' | w), which is based on the following two-stage procedure. First, a single dimension i of W is chosen randomly. Second, the proposed value W' is identical to W, except for its value along the i-dimension W_i (which is either LF or RT). W_i is sampled from the conditional distribution P(W_i | W_{-i}), where W_{-i} is the other dimension (if W_i = LF, then W_{-i} = RT, and the other way around). Therefore

  Q(W' | W) = P(W'_i | W_{-i}).

- The new value is accepted with probability

  (P(W') * Q(W | W')) / (P(W) * Q(W' | W))

  (just like in the Metropolis-Hastings algorithm). We note that, due to the specific construction of Q(w,w'), the acceptance probability equals

  (P(W') * Q(W | W')) / (P(W) * Q(W' | W)) =
  = (P(W') * P(W_i | W'_{-i})) / (P(W) * P(W'_i | W_{-i})) =
  = (P(W_{-i}) * P(W'_i | W_{-i}) * P(W_i | W'_{-i})) /
  / (P(W'_{-i}) * P(W_i | W'_{-i}) * P(W'_i | W_{-i})) =
  = P(W_{-i}) / P(W'_{-i}) **= 1**.

  So we always accept the new realization W'.

~~~~~~