Dongyu Lang

24174288

Stat 133

BML Model

Frist, I would like to talk about my simulation first. The topic that I simulate about is that at what density there is a free flow or a traffic jam and how many steps it takes to arrive at a gridlock and does scale of matrix have any effect on the result.

The simulate functions in bml_simulation.r file are bml.gridlock and bml.gridlock1. I will explain my function as below. For bml.gridlock, the idea is that if the matrix at process N is the same as process N-1, then we can safely claim that the matrix hit gridlock. However, what about the state of free flow? We can know the exact combination of how the matrix would come out with a given number of zero; thus, if our steps exceed the combination number, we know that at the extreme case, the simulation runs at every combination of the matrix, thus if the steps exceed the combination, it's free flow, because the next step should be the same as one of the combinations; that is, there should be a repeated form after the step. I we call the total combination steps the maximum step in the following passage. The maximum step can be calculated as $\binom{r\,c}{r\,c\,(1-p)} \times 2^{r\,c\,p}$. We can further decrease the maximum step by dividing 4, since by logic, we know that if the blue car goes north is the same as the blue car goes south. Thus, there are four ways to represent one result; that is 1W2N, 1E2N, 1W2S and 1E2S (1 represents red, 2 represents blue car, N, E, W and S represents directions).

However, the maximum step is still too large to calculate, even if it is the most accurate result. In bml.gridlock1, which is my second simulation function, I create a more sensible maximum step. I set the maximum step as 2500. The reason is that suppose every entry in matrix is a block. Let's suppose that it takes a car 20s to move to the other entry, that is 20s as one step. Thus, 3500 represents 13 hours. If traffic still could not reach gridlock in 13 hours, we are safely to claim that it is free flow, or the model is not practical at all.

I'll answer the first question as follow. Let's use four scenarios to understand the question. I create 30*30, 20*20, 10*10 and 5*5 matrix to help me solve this problem. I create p=80%, 70%, 60%, 50%, 40% and 30% in each scenario. Each case will be tested for 30 times and saved in a R file as a list. The R.Data files for all are 3030, 2020, 1010 and 55 in names. In each file, there are 6 lists, with a, b, c, d, e and f to represent from 80% to 30%.

Table for 30*30(ff represents number of free flow, mean represents mean step to hit gridlock)

80%		70%		60%		50%		40%		30%
Mean	ff	Mean	ff	Mean	ff	Mean	ff	Mean	ff	ff
36.97	0	62.67	0	86.26	0	349.27	0	840.25	26	30

Table for 20*20

80%		70%		60%		50%		40%		30%
Mean	ff	Mean	ff	Mean	ff	Mean	ff	Mean	ff	ff
27.4	0	49.63	0	93.93	0	373.79	2	110	29	30

Table for 10*10

80%		70%		60%		50%		40%		30%
Mean	ff	Mean	ff	Mean	ff	Mean	ff	Mean	ff	ff
17.5	0	29.67	0	84.93	27	282.4	18	1085.5	28	30

Table for 5*5

80%		70%		60%		50%		40%	30%
Mean	ff	Mean	ff	Mean	ff	Mean	ff	ff	ff
7.64	5	10.8	15	10.4	25	17.3	27	30	30

We can answer all of the three questions above. First, is the scale of the matrix matter? From the table, we can see that the smaller the scale, the higher the chance that it flows freely in high percentage.

Secondly, the steps to observe free flows or traffic jams are all in the saved R file. The interesting thing is that for smaller scale, it usually takes fewer steps to hit gridlock than the steps for larger scale. The reason may be that for larger scale, there is more chance for the whole matrix to move.

Thirdly, in general, it is safe to say that around and below 45%, the traffic can eventually flow freely in a given 2500 steps, while for a larger scale, it hits gridlock when p is over 65% percent. However, for smaller scale, like 5*5, it should be like p over 80% for more. For a mixture of free flow and traffic jam, it usually starts at p less than 60% to have some cases to be free flow and ends at p over 40% to have all the cars in traffic jam mode. For smaller scale, like 5*5, the pattern doesn't hold.