Classic and Influence Majority Rules in Python with Mesa

Emilio Luz-Ricca

Introduction

This past winter break, I have spent some time reading about opinion dynamics on networks and agent-based modeling (ABM). I have also begun to acquaint myself with the python ABM framework `Mesa`. I thought that it might be nice to put everything together in a short write-up explaining the methodology and explaining some basic results.[1]

The Basic Model

In writing this code, I followed the methodology of (*1*). They define two types of majority rules: the classic majority rule (CMR) and the influence majority rule (IMR). The ABM for both models contains N agents and T time steps. Additionally, each agent is placed at a vertex in a graph $G = (V, E)$, where each $v \in V$ is a vertex in G and E is the collection of edges. At each time $t \leq T$, the opinion for the agent at node v is $O_v(t) \in \{0, 1\}$. For all agents, $O_v(0)$ is randomly chosen from $\{0, 1\}$ with equal probability.

In each subsequent time step, the agents are activated in a random order (`Mesa`'s `RandomActivation()` scheduler). When an agent is activated, they assume a new opinion using either the CMR or the IMR. In the CMR, agents choose the most frequent opinion amongst their neighbors. In the IMR, the agent weights the opinion of their neighbors by their degree and chooses the most frequent weighted opinion. If $\eta(v)$ represents all of the neighbors of a particular vertex, i.e., all of the vertices that share an edge with v, and $k_v = |\eta(v)|$ is the degree of v then the CMR and IMR can be represented mathematically as:

$$\begin{aligned}
\text{CMR}: \quad & O_v(t+1) = \text{argmax}_{o \in \{0,1\}} |\{u \mid O_u(t) = o,\ u \in \eta(v)\}| \\
\text{IMR}: \quad & O_v(t+1) = \text{argmax}_{o \in \{0,1\}} \sum_{w \in \{u \mid O_u(t) = o,\ u \in \eta(v)\}} k_w
\end{aligned}$$

While I won't attempt to pursue a mathematical analysis of the dynamics of this system, I find that the more ways that an agent-based model is explained (in words, mathematically, in pictures, etc.), the better. Additionally, the notation introduced in (*1*) provides a compact method for describing key model elements.

Basic Results

I then tried to produce some basic results and visualizations. I also created an interactive visualization with `Mesa`'s visuaization framework, which is available at **GITHUB LINK**.

I attempted to reproduce the first plot from (*1*), but wasn't able to run simulations at quite the same capacity as they did. In this plot, $F_0(t)$ is the fraction of agents that hold opinion 0 at time t. As you can see, opinion 0 begins as the minority opinion ($F_0(0) = 0.4$) by design to show how a significantly minority opinion is quickly overtaken when the average degree, $\langle k \rangle$, is sufficiently large.

[1] Consequently, this was also a good opportunity to play around with the new R Markdown features.

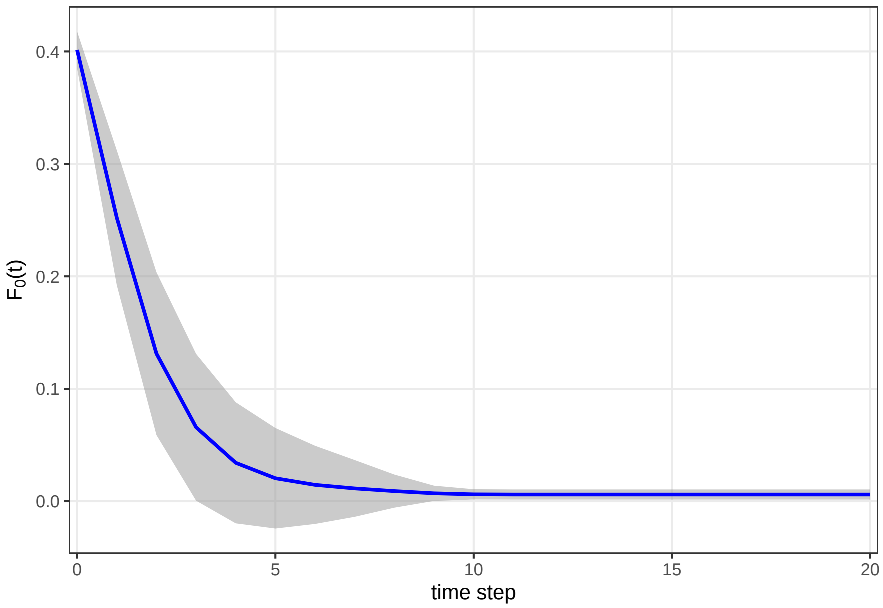
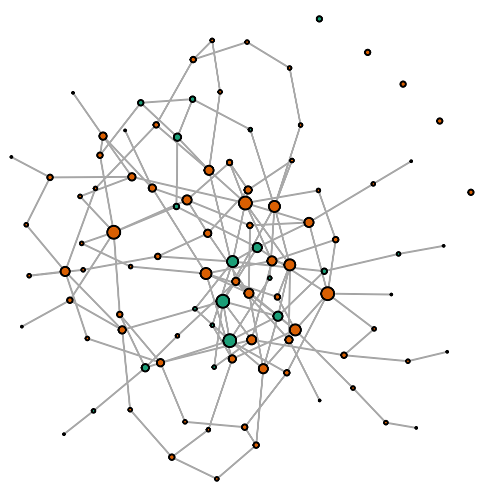


Figure 1: Fraction of 0-opinion agents over time. Results are averaged over 50 model iterations with $\langle k \rangle = 5$ on an Erdős-Renyi network with $N = 1000$. The band represents the standard deviation.

Finally, I produced a visualization showing how opinion diversity continue even in the steady-state under the CMR. In this particular simulation, I used $N = 100$ and $\langle k \rangle = 3$, which is also quite different from the paper. This simulation ran for $T = 100$ steps, which is more than enough steps to reach a steady-state. This plot is quite different from those included in the paper, which are much nicer and admittedly a little more clear. However, for my first time interacting with network visualization tools in `R` (I used the `igraph` package), I think this is a pretty nice result.



References

1. V. X. Nguyen, G. Xiao, X.-J. Xu, Q. Wu, C.-Y. Xia, Dynamics of opinion formation under majority rules on complex social networks. *Scientific Reports.* **10**, 456 (2020).