Classic and Influence Majority Rules in Python with Mesa

Emilio Luz-Ricca

Introduction

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

The Model

In writing this code, I followed the approach 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 each agent v, $O_v(0)$ 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 update their opinion using either the CMR or the IMR. In the CMR, agents choose the most popular opinion among their neighbors (all vertices that share an edge with v). In the IMR, the agent weights the opinion of their neighbors by their degree (the "influence" component) and chooses the most frequent weighted opinion. If $\eta(v)$ represents all of the neighbors of a particular vertex 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 many key model elements.

Basic Results

I then tried to reproduce some basic results and visualizations from (*1*). I also created an interactive visualization with `Mesa`'s visualization framework, which is available in the GitHub repository that contains all the code for this project.

I first attempted to reproduce Figure 1 from (*1*), but wasn't able to run simulations at quite the same capacity as they did. Here, $F_0(t)$ represents fraction of agents that hold opinion 0 at time t. As you can

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

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. In (*1*), they use $N = 20000$, $\langle k \rangle = 25$, and run for $T = 60$ steps, but find similar trends.

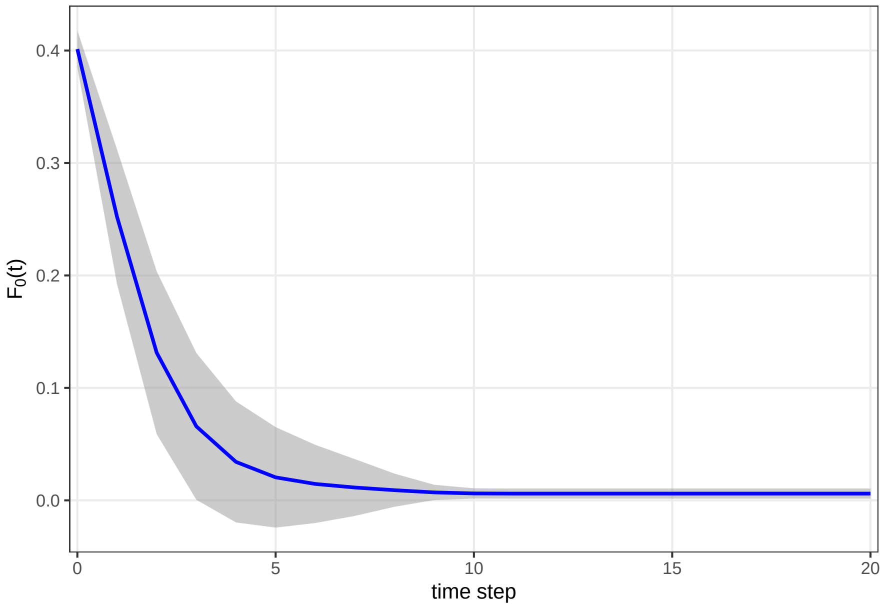

Figure 1: Fraction of 0-opinion agents over time. Results are averaged over 50 model iterations with $\langle k \rangle = 5$ on an Erdős-Renyi network with $N = 1000$. The band represents the standard deviation.

Then, I produced a visualization showing how opinion diversity can be sustained 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. Because $\langle k \rangle$ is small (i.e., the network is *sparse*), the majority opinion has a much harder time propagating across the network than in a denser network. This plot is quite different from those included in the (*1*), both of which are much nicer and admittedly a little more clear. In particular, the graph generation procedure in (*1*) seems to be slightly different than in my implementation (they seem to allow for a wider range of degrees k_v than I do). Still, I think this is a pretty nice result.

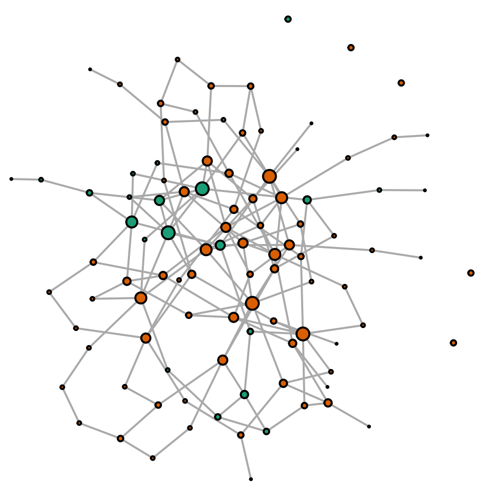

Figure 2: The steady-state ($T = 100$) of a representative simulation. Vertices are scaled by their degree k_v.

Next Steps

There are a few articles that extend these principles quite nicely. In the future, I would like to try to approach models such as those introduced in (2–4). In particular, I'd like to adapt these models to a setting more indicative of interaction on social media, similar to (2), with an emphasis on the formation of extreme communities and the role of social media recommendation algorithms.

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).

2. K. Sasahara, W. Chen, H. Peng, G. L. Ciampaglia, A. Flammini, F. Menczer, Social influence and unfollowing accelerate the emergence of echo chambers. *Journal of Computational Social Science* (2020), doi:10.1007/s42001-020-00084-7.

3. Y. Yu, G. Xiao, G. Li, W. P. Tay, H. F. Teoh, Opinion diversity and community formation in adaptive networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*. **27**, 103115 (2017).

4. V. X. Nguyen, G. Xiao, J. Zhou, G. Li, B. Li, Bias in social interactions and emergence of extremism in complex social networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*. **30**, 103110 (2020).