



Each contact j in the neighbors of i infects i based on a simple random trial that occurs with probability $pP(j \text{ is infected at time } t)$. This is a simple exercise in probability.³ What that reference explains is that it is easiest to look at the probability that i is not infected. Which corresponds with *all* of the “infection attempts” failing. We assume these are independent, so the failure to be infected is just the probability

³ See <https://www.khanacademy.org/math/ap-statistics/probability-ap/probability-multiplication-rule/a/probabilities-involving-at-least-one-success>.

$$\prod_{j \in \text{neighbors}(i)} (1 - P(j \text{ is infected at time } t)).$$

This makes sense. If any neighbor is infected with probability 1 and $p = 1$, then you will be infected, so this quantity will be 0 (so there no chance you are not infected.) The probability that i is infected is simply the complement:

$$P(i \text{ is infected at time } t + 1) = 1 - \prod_{j \in \text{neighbors}(i)} (1 - P(j \text{ is infected at time } t)).$$

We then evaluate this for all i .

This describes a very simple evolution in terms of the adjacency matrix A that is easiest to explain in terms of code. Let \mathbf{x} be the vector $P(i \text{ is infected at time } t)$ for all i and \mathbf{y} be the vector $P(i \text{ is infected at time } t + 1)$ for all i . Then

```

1 function evolve(x::Vector, p::Real, A::AbstractMatrix)
2   log_not_infected = log.(1 .- p.*x)
3   y = 1 .- exp.(A*log_not_infected)
4   y = max.(y, x)
5 end

```

Here, we are using the product is the exponentiated sum of logs. Consequently, we can simultaneously evaluate *all* of the probabilities by taking the log and then *summing* using the adjacency matrix. This is because

$$[\mathbf{Ax}]_i = \sum_{j \in \text{neighbors}(i)} x_j.$$

The final max is useful if you have a boundary condition with a set of definitely infected nodes, but this could also be omitted.⁴

⁴ Play around with it and see. It is a model, not a commandment! The idea is to modify it and understand what happens.

AN INTERESTING ASIDE.

When I ran this, initially, I thought this would converge to all probabilities of 1. This does not happen. Instead it converges to a steady state I can't quite explain. A steady state corresponds with

$$\log(1 - P(i)) = \sum_{j \in \text{neighbors}(i)} \log(1 - pP(j)).$$

It is totally unclear to me *why* and *how* this iteration ought to converge and why this fixed point ought to exist. But it does—reliably so!

AN APPROXIMATION

But here is where linear algebra comes into play. Suppose we make the reasonable approximation that $pP(j \text{ is infected at time } t)$ is small. This means the chance of getting this

from an arbitrary interaction is small. This is plausible at the start of an infection. Then that our expression looks like

$$(1 - a) \cdot (1 - b) \cdot (1 - c) \dots$$

If a, b, c are fairly small, then products ab are even smaller, so we could use the approximation:

$$(1 - a) \cdot (1 - b) \cdot (1 - c) \approx 1 - a - b - c.$$

Applied to our expression, this gives: Then note that

$$\prod_{j \in \text{neighbors}(i)} (1 - P(j \text{ is infected at time } t)) \approx 1 - \sum_{j \in \text{neighbors}(i)} P(j \text{ is infected at time } t)$$

This suggests an even simpler iteration.

```

1 function evolve_approx(x::Vector, p::Real, A::AbstractMatrix)
2   y = p.*(A*x)
3 end

```

This is just a repeated matrix vector product! If $\mathbf{x}^{(t)}$ is the set of probabilities from this approximation at the t th step, then

$$\mathbf{x}^{(t+1)} = \rho \mathbf{A} \mathbf{x}^{(t)} = (\rho \mathbf{A})^{t+1} \mathbf{x}^{(0)}$$

where $\mathbf{x}^{(0)}$ is the *start* of everything.

3 FIXING THE PROBLEM

But there is a problem in the above formulation. This was hinted at in the interesting aside. If you get the infection with probability ρ , then over enough time, everyone would become infected. The probabilities in either model above, though, do not go to 1. This is because we forgot a piece: you *infect yourself* based on the probability in the prior iteration.

The adjustment is simple

$$\begin{aligned}
 &P(i \text{ is infected at time } t + 1) \\
 &= \underbrace{\left(1 - \prod_{j \in \text{neighbors}(i)} (1 - P(j \text{ is infected at time } t))\right)}_{\text{infected via neighbors}} \underbrace{(1 - P(i \text{ is infected at time } t))}_{\text{actually infected in the previous step.}}
 \end{aligned}$$

and

```

1 function evolve_with_self(x::Vector, p::Real, A::AbstractMatrix)
2   log_not_infected = log.(1 .- p.*x)
3   y = (1 .- exp.(A*log_not_infected)).*(1 .- x)
4   y = max.(y, x)
5 end

```

In this new model, the probabilities always go to one.⁵

⁵ This is not a complicated argument, but it isn't the focus on this class.

THE APPROXIMATION AGAIN

Let's use that same idea and approximation to understand what will happen when the probabilities are small. Applying this to the adjusted formulation

$$P(i \text{ is infected at time } t + 1) \approx \rho \sum_{j \in \text{neighbors}(i)} P(j \text{ is infected at time } t) + P(i \text{ is infected at time } t).$$

```

1 function evolve_with_self_approx(x::Vector, p::Real, A::AbstractMatrix)
2   y = rho*(A*x) + x
3 end

```

This is also just repeated matrix vector products, but with the matrix $\rho \mathbf{A} + \mathbf{I}$ instead of $\rho \mathbf{A}$. As in, if again $\mathbf{x}^{(t)}$ is the set of probabilities from this approximation at the t th step, then

$$\mathbf{x}^{(t+1)} = (\rho \mathbf{A} + \mathbf{I}) \mathbf{x}^{(t)} = (\rho \mathbf{A} + \mathbf{I})^{t+1} \mathbf{x}^{(0)}$$

where $\mathbf{x}^{(0)}$ is the *start* of everything.

4 THE EIGENANALYSIS

As we will see in this class, the *eigenvectors* of \mathbf{A} determine the behavior of both powers of $\rho\mathbf{A}$ and $(\rho\mathbf{A} + \mathbf{I})$ as the powers get large. This information then suggests how epidemics spread on networks and a variety of other related behaviors.

5 A SLIGHTLY DIFFERENT MODEL

These models are not commandments. They encode slightly different and related ideas. Here's another way to understand this. What we are doing in the first (incorrect) model is evaluating the probability that node i is infected *by neighbors* at time t . Let $n_i^{(t)} = P(i \text{ is infected via neighbors at time } t)$. Then we have

$$n_i^{(t+1)} = 1 - \prod_{j \in \text{neighbors}(i)} (1 - \rho n_j^{(t)}) \approx \rho \sum_{j \in \text{neighbors}(i)} n_j^{(t)}.$$

As a matrix-vector iteration, the approximation is

$$\mathbf{n}^{(t+1)} = \rho\mathbf{A}\mathbf{n}^{(t)} = (\rho\mathbf{A})^{t+1}\mathbf{n}^{(0)}.$$

But what is $\mathbf{n}^{(0)}$, the starting condition? This has to do with what is often called a *boundary condition*. If we are in a scenario like the US, where the virus is *everywhere* then we can reasonably set $\mathbf{n}^{(0)}$ to be a small constant to model the scenario where everyone has some small chance of being infected. Alternatively, if we are in a contact tracing scenario or a test and trace scenario like Purdue is trying to do, we would remove the contacts from the network that we know are infected and look at the probability that. Here, we simply take any nodes *we know* are infected, remove them from the network, but evaluate the probability that they infect their neighbors. For simplicity, suppose there is one node z infected. Then we set $n_j^{(0)} = \rho$ if j is a neighbor of z and 0 otherwise. The matrix \mathbf{A} for this second scenario *does not include* z .

Now, this models transmission, but we know to know infection probabilities. These are just given by

$$P(i \text{ is infected by time } t) = 1 - \prod_{\ell=0}^t (1 - n_i^{(\ell)}) \approx \sum_{\ell=0}^t n_i^{(\ell)}.$$

Let $\mathbf{x}^{(t)}$ be the probability that i is infected by time t above for all nodes. Then we have

$$\mathbf{x}^{(t)} \approx \sum_{\ell=0}^t \mathbf{n}^{(\ell)} \approx \sum_{\ell=0}^t (\rho\mathbf{A})^\ell \mathbf{n}^{(0)}.$$

This last expression is known as a Neumann series, and admits a closed form solution when ρ is sufficiently small.⁶ If ρ is small enough then

$$\sum_{\ell=0}^{\infty} (\rho\mathbf{A})^\ell \mathbf{n}^{(0)} = (\mathbf{I} - \rho\mathbf{A})^{-1} \mathbf{x}^{(0)}.$$

Put another way, your chance of ever being infected via neighbors from a starting set of probabilities is given by the the solution of a linear system of equations whose right hand side is the initial set of probabilities:

$$(\mathbf{I} - \rho\mathbf{A})\mathbf{x}^{(\infty)} = \mathbf{n}^{(0)}.$$

This is something known as Katz centrality [Katz, 1953] that was derived to understand *social structure*. but again shows how simple matrix systems arise in a problem that is relevant to the times!

⁶ Later in class, we'll see that ρ has to be smaller than the largest magnitude eigenvalue of \mathbf{A} .

REFERENCES

- [Berger et al., 2005] N. BERGER, C. BORGS, J. T. CHAYES, and A. SABERI. *On the spread of viruses on the internet*. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 301–310. 2005. Cited on page 5.
- [Ghosh et al., 2014] R. GHOSH, S.-h. TENG, K. LERMAN, and X. YAN. *The interplay between dynamics and networks: Centrality, communities, and cheeger inequality*. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1406–1415. 2014. doi:10.1145/2623330.2623738. Cited on page 5.
- [Katz, 1953] L. KATZ. *A new status index derived from sociometric analysis*. *Psychometrika*, 18 (1), pp. 39–43, 1953. doi:10.1007/BF02289026. Cited on page 4.
- [Larremore et al., 2020] D. B. LARREMORE, B. K. FOSDICK, K. M. BUBAR, S. ZHANG, S. M. KISSLER, C. J. E. METCALF, C. BUCKEE, and Y. GRAD. *Estimating sars-cov-2 seroprevalence and epidemiological parameters with uncertainty from serological surveys*. medRxiv, 2020. arXiv:https://www.medrxiv.org/content/early/2020/06/22/2020.04.15.20067066.full.pdf, doi:10.1101/2020.04.15.20067066. Cited on page 5.

The idea to relate eigenvectors or Katz scores and epidemics arose, most recently, from a Tweet Dan Larremore at UC Boulder sent about how these determine sampling probabilities in one of their COVID testing papers [Larremore et al., 2020]. In other scenarios, these also are called replicator dynamics from Rumi Ghosh [Ghosh et al., 2014], which better model viral phenomenon. There is more history on understanding matrix diffusions with A as *non-conservative* diffusions that *create mass*, but I've forgotten where all I've read about this and how much (if any) I liked explaining in alternative ways. Some earlier references I'm aware of that would have inspired these thoughts are Saberi's work on (computer) viral spreading and eigenvalues [Berger et al., 2005].