Graph Robustness I: Percolation Theory

Geelon So (ags2191)

March 2, 2018

Percolation Theory

*Percolation is one of the simplest models in probability theory which exhibits...*critical phenomena. [S2011]

Percolation Theory

The classical problem: the d-dimensional lattice $\mathbb{Z}^d$ with each edge/bond removed with some probability p:

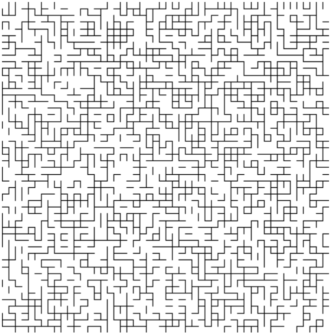

Figure 1: $\mathbb{Z}^2$ with percolation threshold 51%. *Wikipedia*.

Percolation Theory

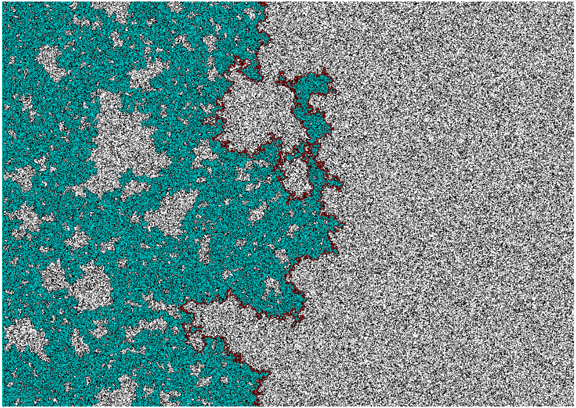

Figure 2: $\mathbb{Z}^2$ with percolation threshold 59.3%. *Wikipedia*.

Node vs. Bond Percolation

We set p to be the probability that a [node/edge] is removed (the probability that a component of the system incurs damage).

Node vs. Bond Percolation

We set p to be the probability that a [node/edge] is removed (the probability that a component of the system incurs damage).

- This is often studied on infinite graphs (like $\mathbb{Z}^2$)
- yet, infinite networks percolation is “self-averaging” [B2018]

Finite Networks

While infinite networks almost never deviate from typical, finite networks may have *large deviations*.

Finite Networks: the typical

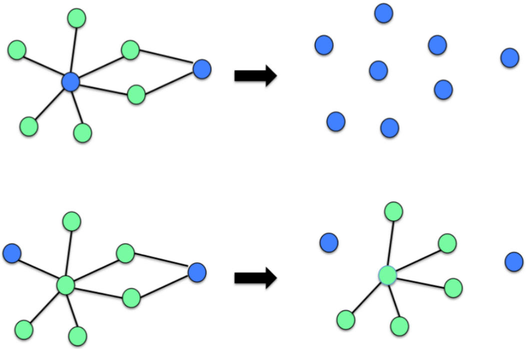

Figure 3: Example of two different types of resulting damage. [B2018]

Finite Networks: the typical

What is the size R of the *giant component* that results after damage to an p-fraction of nodes?

Finite Networks: the typical

We can calculate R using *belief propagation* (BP). The setup:

- Graph is a *locally tree-like* graph with nodes $i = 1, 2, \ldots, N$.

Finite Networks: the typical

We can calculate R using *belief propagation* (BP). The setup:

- Graph is a *locally tree-like* graph with nodes $i = 1, 2, \ldots, N$.
- Each node is assigned a value x_i, where $x_i = 0$ if damaged, $x_i = 1$ if not.

Finite Networks: the typical

We can calculate R using *belief propagation* (BP). The setup:

- Graph is a *locally tree-like* graph with nodes $i = 1, 2, \ldots, N$.
- Each node is assigned a value x_i, where $x_i = 0$ if damaged, $x_i = 1$ if not.
- Each node i passes a message to its neighbor j:

$$\sigma_{i\to j} = x_i \left[1 - \prod_{\ell \in N(i)\setminus j} (1 - \sigma_{\ell\to i})\right].$$

Finite Networks: the typical

We can calculate R using *belief propagation* (BP). The setup:

- Graph is a *locally tree-like* graph with nodes $i = 1, 2, \ldots, N$.
- Each node is assigned a value x_i, where $x_i = 0$ if damaged, $x_i = 1$ if not.
- Each node i passes a message to its neighbor j:

$$\sigma_{i \to j} = x_i \left[1 - \prod_{\ell \in N(i) \setminus j} (1 - \sigma_{\ell \to i}) \right].$$

- Each node computes:

$$\rho_i = x_i \left[1 - \prod_{j \in N(i)} (1 - \sigma_{j \to i}) \right].$$

Finite Networks: the typical

The size of the giant component is:

$$R_{\mathbf{x}} = \sum_{i=1}^{N} \rho_i,$$

where $\mathbf{x} = (x_1, \ldots, x_N)$.

Finite Networks: the typical

The expected size of the giant component is:

$$\hat{R} = \sum_{\mathbf{x}} \Pr(\mathbf{x}) R_{\mathbf{x}},$$

Finite Networks: the typical

The expected size of the giant component is:

$$\hat{R} = \sum_{\mathbf{x}} \Pr(\mathbf{x}) R_{\mathbf{x}},$$

where:

$$\Pr(\mathbf{x}) = \prod_{i=1}^{N} p^{x_i} (1-p)^{1-x_i}.$$

Finite Networks: the typical

Parametrizing the message probability, we can also show that:

$$\hat{R} = \sum_{i=1}^{N} \hat{\rho}_i,$$

where $\hat{\rho}_i$ is the expected computation by node i.

Finite Networks: the typical

This computes the expected damage that results to the network *over all possible damage*; but can we describe the extent that the system will probably be damaged?

Finite Networks: the typical

This computes the expected damage that results to the network *over all possible damage*; but can we describe the extent that the system will probably be damaged?

- that is, what is $\pi(R)$, the probability that the giant component is size R?

Detour: Large Deviation Theory

Large deviation theory formalizes the study of *how different can the behavior of a system be from average*.

- concentration of measures
- e.g. variance

Large Deviation Theory: Example

Toss a fair coin N times.

Large Deviation Theory: Example

Toss a fair coin N times.

- What is the probability that we see heads r-fraction of times?

Large Deviation Theory: Example

Toss a fair coin N times.

- What is the probability that we see heads r-fraction of times?

$$\Pr[R = r] = \sum_{b:R(b)=r} \Pr(b),$$

  - where b is a sequence of N coin tosses,
  - $R(b) = \text{fractions of outcomes that are heads.}$

Large Deviation Theory: Example

$$\Pr\left[R = r\right] = \frac{1}{2^N}\binom{N}{r} = \frac{1}{2^N}\frac{N!}{(rN)!\big[(1-r)N\big]!}$$

Large Deviation Theory: Example

Using Stirling's, $n! \sim n^n e^{-n}$, we get:

$$\Pr[R = r] \sim e^{-nI(r)},$$

where

$$I(r) = \ln 2 + r \ln r + (1 - r) \ln(1 - r).$$

Rate Function

Let A_n be a random variable indexed by $n \in \mathbb{N}$. We say that $\Pr[A_n = r]$ satisfies a *large deviation principle* with *rate* $I(r)$ if the following limit exists:

$$\lim_{n \to \infty} -\frac{1}{n} \ln \Pr[A_n = r] = I(r).$$

That is,

$$\Pr[A_n = r] \sim e^{-nI(r)}.$$

Finite Networks: the atypical

Returning to $\pi(R)$, the probability that the giant component is R, we can express this in terms of the rate function:

$$\pi(R) \sim e^{-NI(R)}.$$

Finite Networks: the atypical

Returning to $\pi(R)$, the probability that the giant component is R, we can express this in terms of the rate function:

$$\pi(R) \sim e^{-NI(R)}.$$

Question: How do you calculate $I(R)$?

Detour: Maximum Entropy Principle

This system has *microstates* and *macrostates*:

- microstates correspond to specific damage to the nodes,

Detour: Maximum Entropy Principle

This system has *microstates* and *macrostates*:

- microstates correspond to specific damage to the nodes,
- macrostates correspond to the number of giant components.

Detour: Maximum Entropy Principle

This system has *microstates* and *macrostates*:

- microstates correspond to specific damage to the nodes,
- macrostates correspond to the number of giant components.

Detour: Maximum Entropy Principle

This system has *microstates* and *macrostates*:

- microstates correspond to specific damage to the nodes,
- macrostates correspond to the number of giant components.

More generally, we can think of macrostates as viewing the microstate of the system through the lens of a few *features*.

Detour: Maximum Entropy Principle

How do the probabilites of the macrostates and microstates relate?

- If we know the distribution of microstates, we can compute the distribution of macrostates (integrate/expectations).

Detour: Maximum Entropy Principle

How do the probabilites of the macrostates and microstates relate?

- If we know the distribution of microstates, we can compute the distribution of macrostates (integrate/expectations).
- If we know the distribution of macrostates, what is the probability distribution on the microstates that introduces the fewest number of assumptions?

Entropy

Question: Consider a coin toss that returns head with probability p. If we don't know p, what distribution of heads–tails "introduces the fewest number of assumptions"?

Entropy

Question: Consider a coin toss that returns head with probability p. If we don't know p, what distribution of heads–tails "introduces the fewest number of assumptions"?

$$(p_H, p_T) = (0.5, 0.5).$$

Entropy

Question: What if $p = 0.01$. That is, 1 out of 100 coin tosses will be heads. Is this 'more' or 'less' random? How many bits does it take on average to represent the outcome of this coin toss?

Entropy

‘More randomness’ means we need more bits to write down the outcome.

Entropy

'More randomness' means we need more bits to write down the outcome. Let's take this as the heuristic definition of entropy of a random variable:

- the fewest number of bits needed to represent the outcome of a random variable in expectation

Entropy

'More randomness' means we need more bits to write down the outcome. Let's take this as the heuristic definition of entropy of a random variable:

- the fewest number of bits needed to represent the outcome of a random variable in expectation
- thus, 'more randomness' $\iff$ 'greater entropy'

Entropy

Reasonable properties of entropy $H(X)$ of a random variable $X \sim \{1, \ldots, n\}$: [D2005]

1. **Expansibility**: if X has distribution $(p_1, \ldots, p_n)$ and Y has distribution $(p_1, \ldots, p_n, 0)$, then $H(X) = H(Y)$.
2. **Symmetry**: $H(X) = H(\sigma(X))$ for permutations $\sigma \in \Sigma_n$.
3. **Additivity**: If X and Y are independent, then $H(X, Y) = H(X) + H(Y)$.
4. **Subadditivity**: $H(X, Y) \leq H(X) + H(Y)$.
5. **Normalization**: $H(\text{fair coin}) = 1$.
6. **Small for small probability**: $\lim_{p \to 0} H(p) = 0$.

Entropy

It turns out that the only possible function that satisfies all these is:

$$H(X) = \mathbb{E}_{x \sim p}\left[-\log p(x)\right].$$

Maximum Entropy Principle

Let X be the microstate of a system. Let $T_i(X)$ for $1 \leq i \leq K$ be the corresponding features.

Maximum Entropy Principle

Let X be the microstate of a system. Let $T_i(X)$ for $1 \leq i \leq K$ be the corresponding features.

- If we know the expectations $\mathbb{E}_X[T_i] = b_i$, what distribution over X introduces the fewest number of assumptions?

Maximum Entropy Principle

Let X be the microstate of a system. Let $T_i(X)$ for $1 \leq i \leq K$ be the corresponding features.

- If we know the expectations $\mathbb{E}_X[T_i] = b_i$, what distribution over X introduces the fewest number of assumptions?
- that is, what distribution $p(X)$ has the greatest entropy while satisfying the constraints in expectations?

Optimization Problem

Want to optimize:

$$\max \sum_x p_x \log \frac{1}{p_x},$$

constrained to:

$$\sum_x p_x T_i(x) = b_i \quad i = 1, \ldots, K,$$

$$\sum_x p_x = 1$$

$$p_x \geq 0$$

Optimization Solution

Using Lagrange multipliers, we get:

$$p(x) = \frac{1}{Z} \cdot \exp\left(\sum_{i=1}^{K} \lambda_i T_i(x)\right) \pi(x),$$

where $\pi(x)$ is the prior distribution and Z is a normalization constant (that depends on the λ's).

Optimization Solution

Using Lagrange multipliers, we get:

$$p(x) = \frac{1}{Z} \cdot \exp\left(\sum_{i=1}^{K} \lambda_i T_i(x)\right) \pi(x),$$

where $\pi(x)$ is the prior distribution and Z is a normalization constant (that depends on the λ's).

- We call $Z(\lambda)$ the *partition function*.

Returning to the Rate Function

Recall that we had $\pi(R) \sim e^{-NI(R)}$. So, we can set the partition function:

$$Z = \sum_x P(x) e^{-\lambda R_x}.$$

Returning to the Rate Function

Recall that we had $\pi(R) \sim e^{-NI(R)}$. So, we can set the partition function:

$$Z = \sum_x P(x) e^{-\lambda R_x}.$$

This is solvable using the Legendre-Fenchel transform given certain conditions. This allowed [B2018] to analyze the behavior of finite networks.

References

[B2018] Bianconi, Ginestra. "Rare events and discontinuous percolation transitions." Physical Review E 97.2 (2018): 022314.

[D2005] Dasgupta, Sanjoy. "Information geometry." Machine Learning Summer School, Chicago 2005. Video.

[H2009] Horak, Danijela, Slobodan Maletić, and Milan Rajković. "Persistent homology of complex networks." Journal of Statistical Mechanics: Theory and Experiment 2009.03 (2009).

[N2017] Otter, Nina, et al. "A roadmap for the computation of persistent homology." EPJ Data Science 6.1 (2017): 17.

[S2011] Steif, Jeffrey E. "A mini course on percolation theory." Jyväskylä lectures in mathematics 3 (2011).

[T2009] Touchette, Hugo. "The large deviation approach to statistical mechanics." Physics Reports 478.1-3 (2009): 1-69.