1.022 Introduction to Network Models

Amir Ajorlou

Laboratory for Information and Decision Systems
Institute for Data, Systems, and Society
Massachusetts Institute of Technology

Lectures 15-17

Positive linear system

- Positive linear system
  - Let $A = [A_{ij}] \in \mathbb{R}^{n \times n}$ be such that $A_{ij} > 0$ for all $1 \leq i, j \leq n$
  - System dynamics:

$$x(k) = Ax(k-1), \quad \text{for} \quad k \geq 1.$$

- Perron-Frobenius Theorem: let $A \in \mathbb{R}^{n \times n}$ be positive
  - Let $\lambda_1, \ldots, \lambda_n$ be eigenvalues such that

$$0 \leq |\lambda_n| \leq |\lambda_{n-1}| \leq \cdots \leq |\lambda_2| \leq |\lambda_1|$$

  - Then, maximum eigenvalue $\lambda_1 > 0$
  - It is unique, i.e. $|\lambda_1| > |\lambda_2|$
  - Corresponding eigenvector, say s_1 is component-wise > 0

Positive linear system

- More generally, we call A positive system if
  - $A \geq 0$ component-wise
  - For some integer $m \geq 1$, $A^m > 0$
  - If eigenvalues of A are λ_i, $1 \leq i \leq n$
  - Then eigenvalues of A^m are λ_i^m, $1 \leq i \leq m$
  - The Perron-Frobenius for A^m implies similar conclusions for A
- Special case of positive systems are Markov chains
  - we consider them next
  - as an important example, we'll consider random walks on graphs

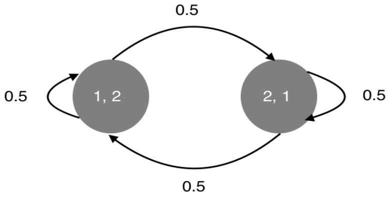
An Example

- Shuffling cards

- A special case of *Overhead* shuffle:
  - choose a card at random from deck and place it on top
- How long does it take for card deck to become random?
  - Any one of 52! orderings of cards is equally likely

An Example

- Markov chain for deck of 2 cards



- Two possible card order: $(1,2)$ or $(2,1)$
- Let X_k denote order of cards at time $k \geq 0$

$$\begin{aligned}\mathbb{P}(X_{k+1} = (1,2)) &= \mathbb{P}(X_k = (1,2) \text{ and card 1 chosen}) + \\ &\qquad \mathbb{P}(X_k = (2,1) \text{ and card 1 chosen}) \\ &= \mathbb{P}(X_k = (1,2)) \times 0.5 + \mathbb{P}(X_k = (2,1)) \times 0.5 \\ &= 0.5\end{aligned}$$

Notations

- Markov chain defined over state space $N = \{1, \ldots, n\}$
  - $X_k \in N$ denote random variable representing state at time $k \geq 0$
  - $P_{ij} = \mathbb{P}(X_{k+1} = j | X_k = i)$ for all $i, j \in N$ and all $k \geq 0$

$$\mathbb{P}(X_{k+1} = i) = \sum_{j \in N} P_{ji} \mathbb{P}(X_k = j)$$

- Let $p(k) = [p_i(k)] \in [0,1]^n$, where $p_i(k) = \mathbb{P}(X_k = i)$

$$p_i(k+1) = \sum_{j \in N} p_j(k) P_{ji}, \text{ for all} i \in N \quad \Leftrightarrow \quad p(k+1)^T = p(k)^T P$$

- $P = [P_{ij}]$: probability transition matrix of Markov chain
  - non-negative: $P \geq 0$
  - row-stochastic: $\sum_{j \in N} P_{ij} = 1$ for all $i \in N$

Stationary distribution

- Markov chain dynamics: $p(k+1) = P^T p(k)$
  - Let the probability transition matrix $P > 0$: positive linear system
  - Perron-Frobenius:
    - P^T has unique real positive largest eigenvalue: $\lambda_{max} = \lambda_1 > 0$
    - Corresponding eigenvector: $P^T p^\star = \lambda_{\max} p^\star$, then $p^\star > 0$.
    - We assume $p^\star$ normalized such that $p_1^\star + \ldots + p_n^\star = 1$.
- We claim $\lambda_{\max} = 1$ and $p(k) \to p^\star$
  - Recall, $\|p(k)\| \to 0$ if $\lambda_{\max} < 1$ or $\|p(k)\| \to \infty$ if $\lambda_{\max} > 1$
  - But $\sum_i p_i(k) = 1$ for all k, since $\sum_i p_i(0) = 1$ and

$$\sum_i p_i(k+1) = p(k+1)^T \mathbf{1} = p(k)^T P \mathbf{1}$$

$$= p(k)^T \mathbf{1} = \sum_i p_i(k) = 1.$$

  - We have used $P\mathbf{1} = \mathbf{1}$
  - Therefore, $\lambda_{\max}$ must be 1 and $p(k) \to c_1 p^\star = p^\star$ (argued before)
  - $c_1 = 1$ since $\sum_i p_i(k) = \sum_i p_i^\star = 1$

Stationary distribution

- Stationary distribution: if $P > 0$, then there exists $p^\star > 0$ such that

$$p^\star = P^T p^\star \quad \Leftrightarrow \quad p_i^\star = \sum_j P_{ji} p_j^\star, \text{for all} i.$$

$$p(k) \stackrel{k\to\infty}{\to} p^\star$$

- Above holds also when $P^k > 0$ for some $k \geq 1$
  - Sufficient *structural* condition: P is irreducible and aperiodic
  - Irreducibility:
    - for each $i \neq j$, there is a positive probability to reach j starting from i
  - Aperiodicity:
    - There is no partition of N so that Markov chain state 'periodically' rotates through those partitions
    - Special case: for each i, $P_{ii} > 0$

Random walk on Graph

- Consider an undirected connected graph G over $N = \{1, \ldots, n\}$
  - It's adjacency matrix A
  - Let k_i be degree of node $i \in N$
- Random walk on G
  - Each time, remain at current node or walk to a random neighbor
  - Precisely, for any $i, j \in N$

$$P_{ij} = \begin{cases} \frac{1}{2} \text{ if } i = j \\ \frac{1}{2k_i} \text{ if } A_{ij} > 0, i \neq j \\ 0 \text{ if } A_{ij} = 0, i \neq j \end{cases}$$

- Does it have stationary distribution? If yes, what is it?

Random walk on Graph

- Answer: Yes, because irreducible and aperiodic.
  - Further, $p_i^\star = k_i/2m$, where m is number of edges
- Why?
  - $P = \frac{1}{2}(I + D^{-1}A)$, $p^\star = \frac{1}{2m}D\mathbf{1}$, where $D = diag(k_i)$, $\mathbf{1} = [1]$

$$
\begin{aligned}
p^{\star,T}P = \frac{1}{2}p^{\star,T}(I + D^{-1}A) &= \frac{1}{2}p^{\star,T} + \frac{1}{2}p^{\star,T}D^{-1}A \\
&= \frac{1}{2}p^{\star,T} + \frac{1}{2m}\mathbf{1}^T A = \frac{1}{2}p^{\star,T} + \frac{1}{4m}(A\mathbf{1})^T, \quad \text{because} \quad A = A^T \\
&= \frac{1}{2}p^{\star,T} + \frac{1}{4m}[k_i]^T = \frac{1}{2}p^{\star,T} + \frac{1}{2}p^{\star,T} = p^{\star,T}.
\end{aligned}
$$

- Stationary distribution of random walk:
  - $p^\star = \frac{1}{2}(I + D^{-1}A)p^\star$
  - $p_i^\star \propto k_i \rightarrow$ *Degree centrality*!

Katz Centrality

- Consider solution of equation

$$\mathbf{v} = \alpha A \mathbf{v} + \beta$$

  for some $\alpha > 0$ and $\beta \in \mathbb{R}^{\mathbf{n}}$
  - Then v_i is called Katz centrality of node i

- Recall
  - Solution exists if
    - $\det(I - \alpha A) \neq 0$
    - equivalently A doesn't have α^{-1} as eigenvalue
  - But dynamically solution is achieved if
    - largest eigenvalue of A is smaller than α^{-1}

- Dynamic range of interest: $0 < \alpha < \lambda_{\max}^{-1}(A)$

Convergence to stationary distribution

- Let $p(k)$ be probability distribution at time k

$$p(k+1) = P^T p(k)$$

- Let $s_1, s_2, \ldots, s_n$ be eigenvectors of P^T
  - with associated eigenvalues $1, \lambda_2, \ldots, \lambda_n$
  - $0 \leq |\lambda_n| \leq \cdots \leq |\lambda_2| < 1$
  - Define spectral gap $g(P) = 1 - |\lambda_2|$

- Then, as argued for linear dynamics, we have

$$p(k) = c_1 s_1 + \sum_{i=2}^{n} \lambda_i^k c_k s_k$$

with some constants $c_1, \ldots, c_n$

Convergence to stationary distribution

– Therefore:

$$\|p(k) - c_1 s_1\| \leq \sum_{i=2}^{n} |\lambda_i|^k |c_i| \|s_i\| \leq (n-1) C |\lambda_2|^k$$

where $C = \max_{i=2}^{n} |c_i| \|s_i\|$

– Subsequently

$$k \geq \frac{\log n + \log C + \log \frac{1}{\varepsilon}}{\log \frac{1}{|\lambda_2|}} \quad \Rightarrow \quad \|p(k) - c_1 \mathbf{1}\| \leq \varepsilon.$$

– The ε-convergence time scales as

$$T_{conv}(\varepsilon) \sim \frac{\log n + \log \frac{1}{\varepsilon}}{\log \frac{1}{|\lambda_2|}}.$$

– Using $\log(1-x) \approx -x$ for $x \in (0,1)$, we get

$$T_{conv}(\varepsilon) \sim \frac{\log n + \log \frac{1}{\varepsilon}}{g(P)}$$

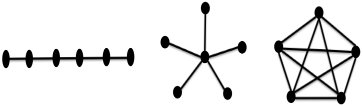
Information spread

- Network graph G over $N = \{1, \ldots, n\}$ nodes, edges E
  - Given information at one of the nodes, spread it to *all* nodes
  - By "Gossiping"
  - How long does it take?

- Gossip dynamics:
  - At each time, each node $i \in N$ does the following:
  - if node i does not have information, nothing to spread or gossip
  - else if it does have information, it sends it to one of it's neighbors
    - let $P_{ij} = \mathbb{P}(i \text{ sends information to } j)$
    - by definition, $\sum_{j \in N} P_{ij} = 1$, and
    - $P_{ij} = 0$ if j is not neighbor of i
  - Example: uniform gossip
    - $P_{ij} = 1/k_i$ for all $(i, j) \in E$

Information spread

- Why study Gossip dynamics
  - This is how socially information spreads
  - More generally, this is how "contact" driven network effect spreads
  - This is how large scale distributed computer systems are built
    - e.g. Cassandra, an Apache Open Source Distributed DataBase
    - used by some of the largest organizations including Netflix, etc.
- Key question
  - How long does it take for all nodes to receive information?
  - How does it depend on the graph structure, P?
  - Let us consider few examples:
    - A path
    - Star graph
    - Complete graph



Information spread and Conductance

- Conductance of $P = [P_{ij}]$ is defined as

$$\Phi(P) = \min_{S \subset N : |S| \leq n/2} \frac{\sum_{i \in S, j \in S^c} P_{ij}}{|S|} \quad (1)$$

- Examples: uniform gossip
  - Path: $\Phi \sim \frac{1}{n}$
  - Star graph: $\Phi \sim \frac{1}{n}$
  - Complete graph: $\Phi \sim \frac{1}{2}$
- How long does it take for all nodes to almost surely receive information?
- A crisp answer

$$T_{spr} \sim \frac{\log n}{\Phi(P)}$$

where $\Phi(P)$ is the conductance of P (and hence graph)

- Spectral gap and conductance:
  - Markov chain can not converge faster than information spread
  - And information spreads in time $\log n/\Phi(P)$
  - That is (ignoring constants)

$$\frac{\log n}{\Phi(P)} \leq \frac{\log n}{g(P)} \quad \Leftrightarrow \quad g(P) \leq \Phi(P)$$

- A remarkable fact known as Cheeger's inequality:

$$\frac{1}{2}\Phi(P)^2 \leq g(P) \leq 2\Phi(P).$$

Distributed computation

- Generic question:
  - Given network G over nodes N with edges E
  - Each node $i \in N$ has information x_i
  - Compute a global function:

$$f(x_1, \ldots, x_n)$$

  - by communicating along the network links
  - processing *local* information at each node continually
  - while keeping *limited* local state at each node

Know your neighbors

- The simplest possible example
  - Estimate number of nodes in the entire network at each node locally
  - there is no globally agreed unique names for each node
  - only local communications are allowed while keeping local state small
- A distributed algorithm
  - Every node generates a random number
    - Node $i \in N$ draws random variable R_i as per an Exponential distribution of mean 1
  - Compute global minimum, $R^\star = \min_{i \in N} R_i$
    - Using *Gossip* mechanism
  - Repeat the above for L times
    - $R^\star_\ell, \; 1 \leq \ell \leq L$ be global minimum computed during these L trials
  - Estimate of number of neighbors: $\hat{n} = \frac{L}{\sum_{\ell=1}^{L} R^\star_\ell}$

Exponential distribution

- Exponential distribution with parameter $\lambda > 0$
  - X be random variable with this distribution: for any $t \geq 0$,

$$\mathbb{P}\big(X > t\big) = \exp\big(-\lambda t\big).$$

- Minimum of exponential random variables
  - Let $X_i,\ i \in N$ be independent random variables
  - Distribution of X_i is Exponential with parameter λ_i, $i \in N$
  - $X^* = \min_{i \in N} X_i$

$$\begin{aligned}
\mathbb{P}\big(X^* > t\big) &= \mathbb{P}\big(\cap_{i \in N} X_i > t\big) \\
&= \prod_{i \in N} \mathbb{P}\big(X_i > t\big) \\
&= \prod_{i \in N} \exp\big(-\lambda_i t\big) \\
&= \exp\big(-\big(\sum_i \lambda_i\big) t\big).
\end{aligned}$$

- Exponential distribution with parameter $\lambda > 0$
  - X be random variable with this distribution: for any $t \geq 0$,

$$\mathbb{P}(X > t) = \exp\big(-\lambda t\big).$$

- Minimum of exponential random variables
  - $X^* = \min_{i \in N} X_i$ has exponential distribution with parameter $\sum_{i \in N} \lambda_i$
- Mean of exponential variable X with parameter $\lambda > 0$

$$\begin{aligned}\mathbb{E}[X] &= \int_0^\infty \mathbb{P}(X > t)dt \\ &= \int_0^\infty \exp\big(-\lambda t\big)dt \\ &= \frac{1}{\lambda}\Big[\exp\big(-\lambda t\big)\Big]_\infty^0 \\ &= \frac{1}{\lambda}.\end{aligned}$$

Exponential distribution

- Back to counting nodes
  - Node i's random number has exponential distribution of parameter 1
  - All nodes computed minimum of these numbers
  - Hence minimum had exponential distribution with parameter n
  - That is, mean of the minimum is $1/n$
  - Averaging over multiple trials gives a good estimation of $1/n$

- How to add up numbers?
  - Node i has a number x_i
  - Node i draws random variable per exponential distribution of parameter x_i
  - Then minimum would have exponential distribution with parameter $\sum_i x_i$
  - Subsequently, algorithm is computing estimation of $\sum_i x_i$

Gossip algorithm for finding minimum

- Gossip algorithm
  - Node $i \in N$ has value R_i and goal is to compute $R^\star = \min_i R_i$
  - Node $i \in N$ keeps an estimate of global minimum, say $\hat{R}_i^\star$
  - Initially, $\hat{R}_i^\star = R_i$ for all $i \in N$
  - Whenever node j contacts i
    - Node j sends $\hat{R}_j^\star$ to i
    - Node i updates $\hat{R}_i^\star = \min\left(\hat{R}_j^\star, \hat{R}_i^\star\right)$

- How long does it take for everyone to know minimum?
  - Suppose $R_1 = R^\star$.
  - Then the spread of minimum obeys exactly same dynamics as spreading information starting with node 1.
  - That is, *information spread* = *minimum computation*!