Introduction to Network Epistemology

JAMES OWEN WEATHERALL

LOGIC AND PHILOSOPHY OF SCIENCE, UC IRVINE

Social Learning

Humans are social learners

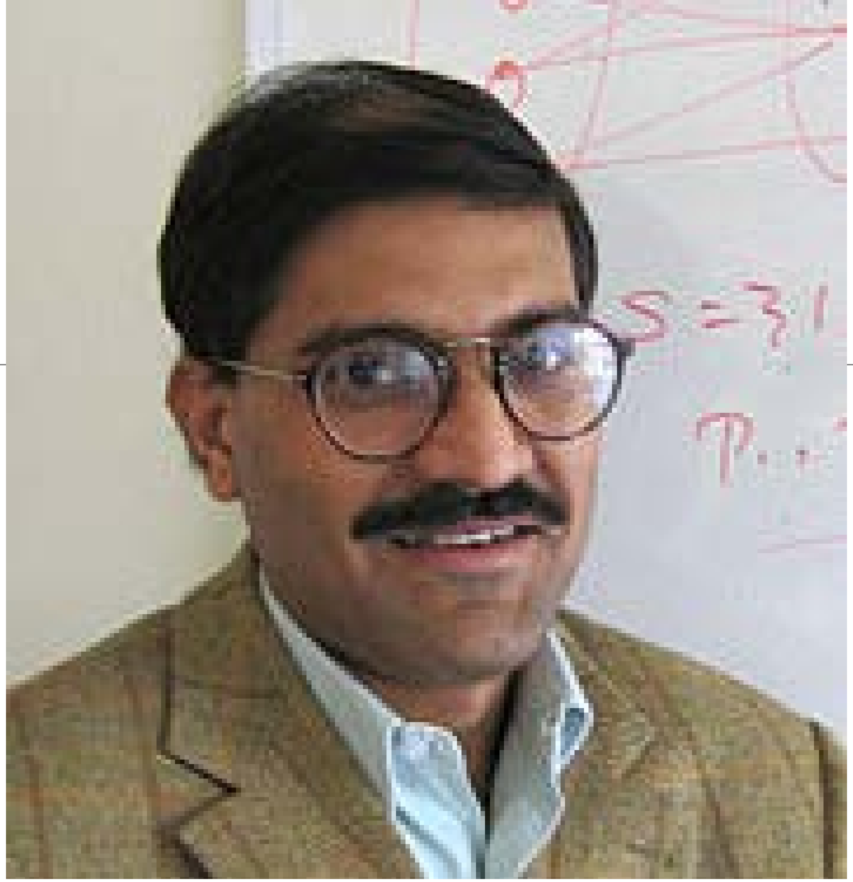
Most of what we know, we learn from others– either by testimony, instruction, or observation of what works for others

We also rely on others to make mistakes or try bad ideas

This is true in ordinary life and also in science

No individual can conduct all the experiments, perform all the analyses, etc. needed to develop all scientific knowledge

For these reasons, it is important to study how social structure influences learning



NETWORK EPISTEMOLOGY

Overview

- Model Description
- Representing Beliefs
- Results

Overview

- **Model Description**
- Representing Beliefs
- Results

Network Epistemology

The network epistemology framework models communities of learners solving a decision problem through trial and error

Agents perform actions that they expect to maximize their payoffs

They learn from the results of their actions and then try again

These agents are represented as nodes on a network

In addition to their own results, agents observe the result of their neighbors' actions and learn from them

Network epistemology and the world

This framework captures one aspect of social learning: learning from others' successes and failures, thereby increasing total data

It also captures the idea that we learn from others' "mistakes"

Different agents in this model can pursue different solutions to the decision problem, depending on what they individually believe best

By observing others' behaviors and their consequences, agents learn about the success rate of actions they do not perform themselves

Agents can change their minds in light of when others succeed

Two-armed bandit

We consider a *two-armed bandit* problem

Agents decide between two actions A and B

Each action pays 1 if it succeeds

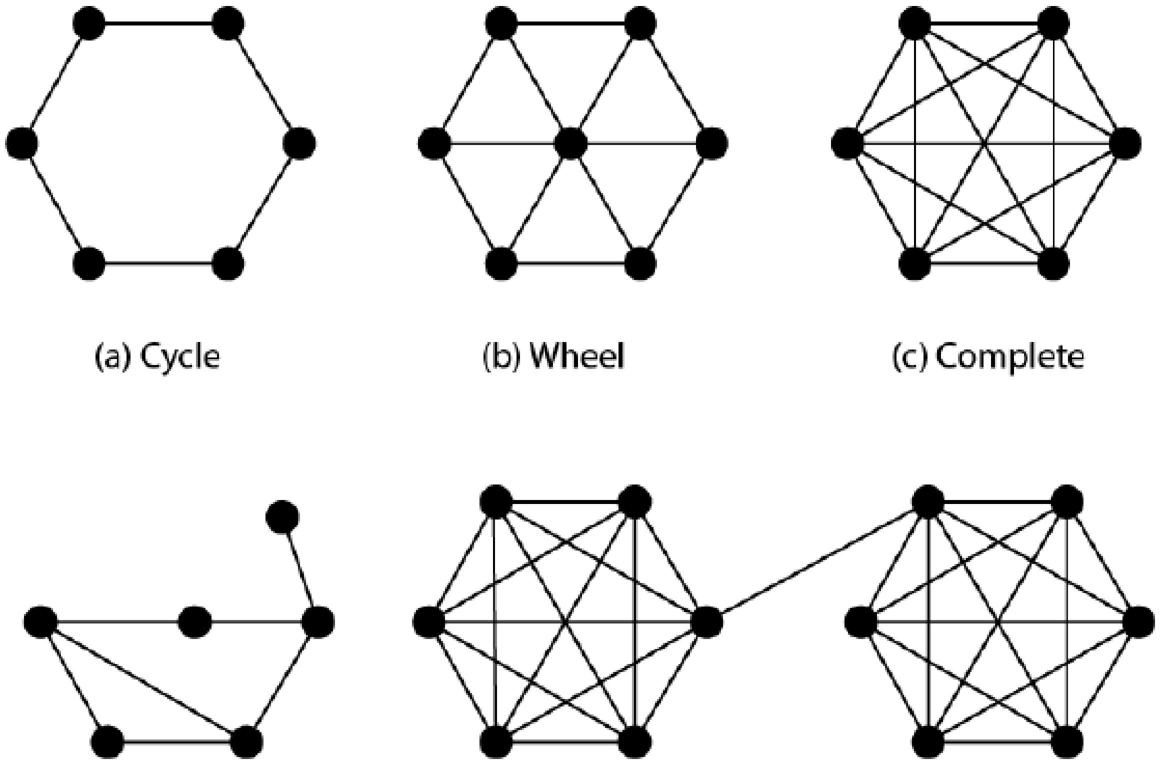
Success is a Markov process, i.e., probability of success is independent of past attempts

A is successful with p(A) = .5

B is successful with p(B) = .5 + e

Agents are unsure about which arm is better

Networks



(a) Cycle (b) Wheel (c) Complete

(d) Random (e) Clumpy

Agents are arranged on a network

Nodes represent agents; edges are connections

Agents only learn from adjacent nodes (neighbors)

Basic Dynamics

The model proceeds in rounds (discrete time steps)

Agents start with a random credence about which arm is better

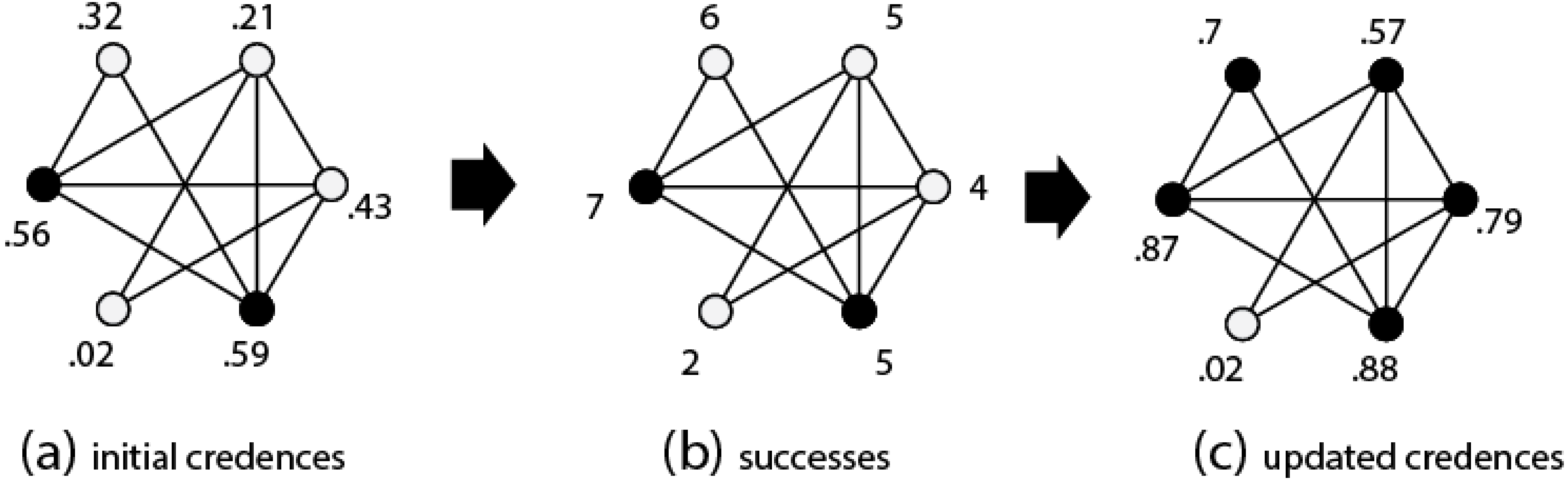
(More on this soon)

In each round agents take the action that they think is better, some number of times *n*

They update their credences, using Bayes rule, based on their result

(Results are of the form: k successes in n attempts)

In addition, they update on results gathered by neighbors


.32
.21
.56
.43
.02
.59
6
5
7
4
2
5
.7
.57
.87
.79
.02
.88
(a) initial credences
(b) successes
(c) updated credences

Overview

- Model Description
- **Representing Beliefs**
- Results

Representing Beliefs, Simple Way

Suppose, first, that agents **know** the success rate of action A is 50%, i.e., their credence that arm A has success rate 50% is 1

Then performing action A is **uninformative**. It does not affect belief

In addition, assume that agents know that the probability that arm B succeeds is either $p(B) = .5 + e$ or $.5 – e$, for **fixed known e**

In this case, agent states can be represented by a single number, C, representing their credence that $p(B) = .5 + e$

Agents believe B is better just in case $C > .5$

Dynamics

On this version of the model, agents are initialized with random credence C drawn from the interval (0,1)

On average, half of agents initially perform action A and half action B

Agents apply Bayes theorem to update on results of neighbors (themselves included) performing action B

Agents assume the result of each instance of action B is a coin flip with probability P(B).

In more detail

Agents assume results are drawn from a binomial distribution:

$$\text{C}(k|B) = \binom{n}{k} P(B)^k (1 - P(B))^{n-k}$$

$$\text{C}(k|not\ B) = \binom{n}{k} (1 - P(B))^k (P(B))^{n-k}$$

If a neighbor gets k successes in n pulls, agent credences update as:

$$C_f(B) = C_i(B|k) = {}^{C_i(B)C(k|B)}/_{C(k)}$$

where $C(k) = C(B)C(k|B) + (1 - C(B))C(k|not\ B)$

Advantages and Disadvantages

The simple model has some advantages:

1. Each player’s state is represented by a single number
2. Accuracy / success is easy to measure: C(B) -> 1
3. Simple model dynamics

It also has disadvantages:

1. Nothing is learned from performing action A
2. Requires strong assumptions about what agents know

Representing Beliefs, Sophisticated Way

Suppose agents know only that results are drawn from a binomial distribution, but do not know the probabilities P(B) or P(A)

We no longer assume that P(B) is fixed to have only two values, or that P(A) is known, so both actions are informative

In this case, we represent each agent's credences as **beta distributions** over values for P(A) and P(B)

These distributions determine expected values for each action, and they also measure agents' uncertainty

Beta distributions

A beta distribution is a continuous probability distribution over the interval (0,1) determined by two parameters, a and b

This is a natural choice because the beta distribution is the conjugate distribution to the binomial distribution

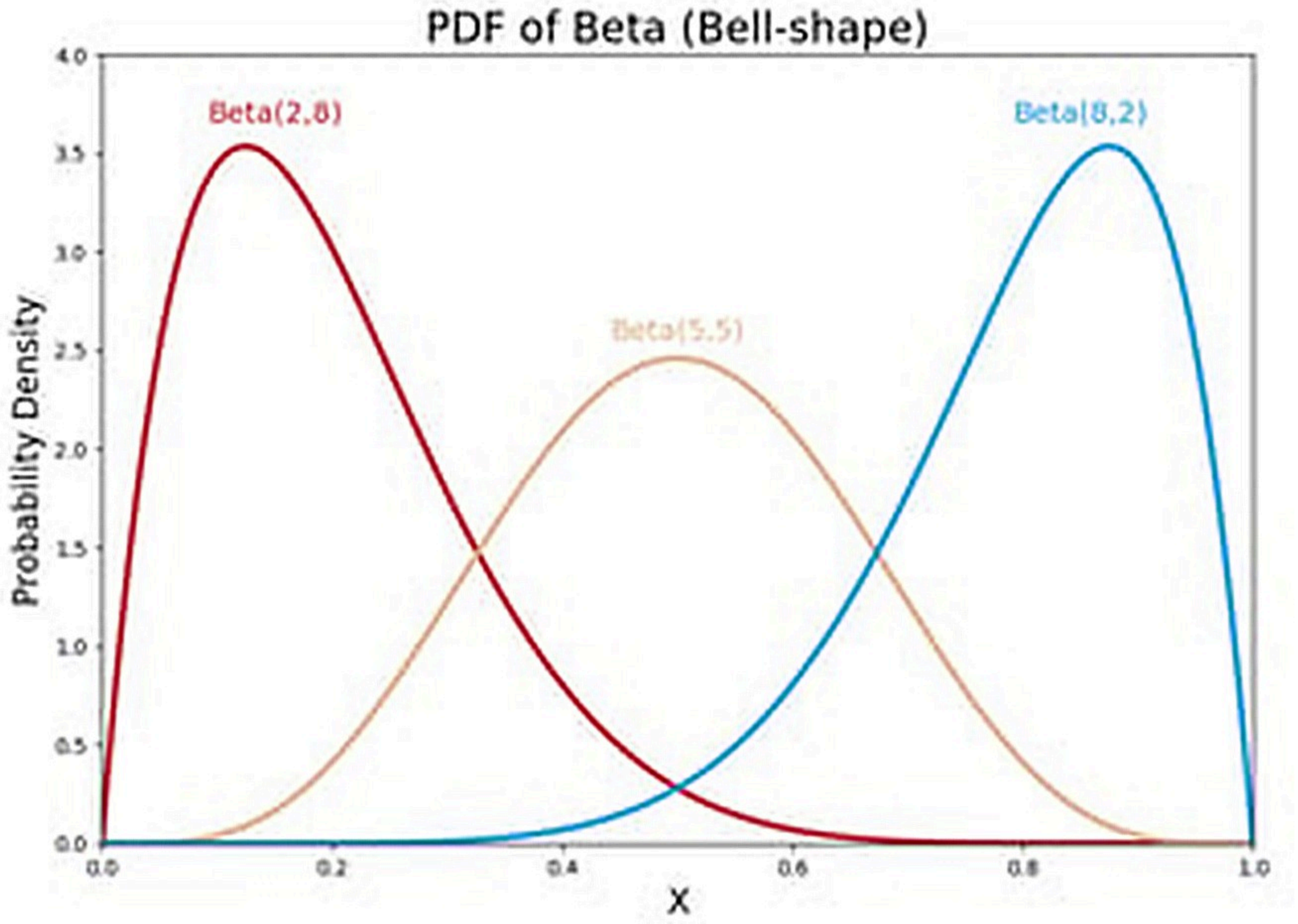
This means that since the likelihood function is binomial, beta distributed priors transform to beta distributed posteriors

The parameters can be interpreted as “number of successes (+1)” and “number of failures (+1)”

Beta PDF

$$PDF(x; a, b) = B(a, b)\, x^{a-1}(1-x)^{b-1}$$

B(a,b) is a constant to normalize the integral



Properties of Beta Distributions

The mean, or expected, value of p for parameters (a,b) is $\frac{a}{a+b}$

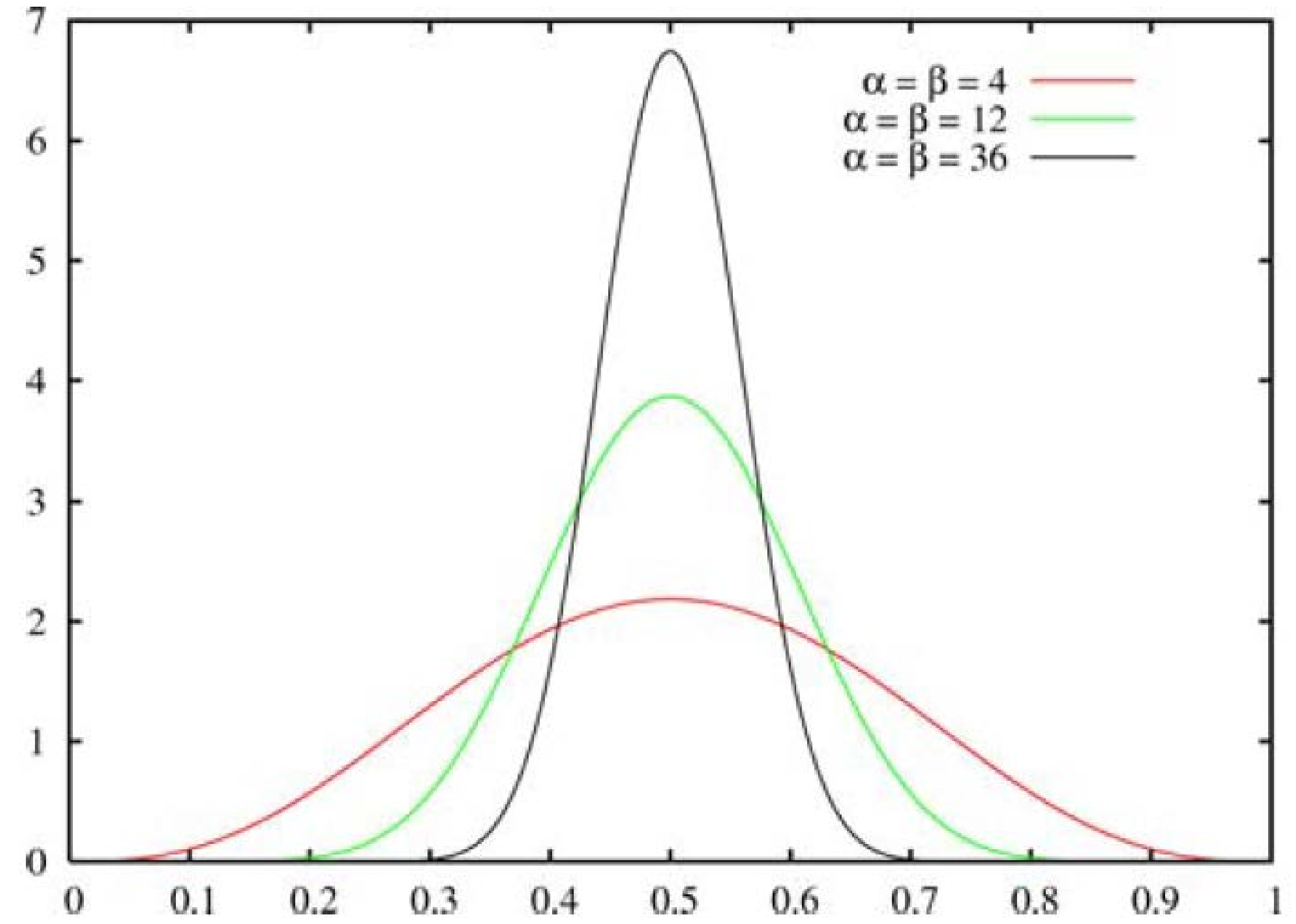
Variance is $\frac{ab}{(a+b)^2(a+b+1)}$

Bayesian updating on a new trial with k successes in n attempts is given by a -> a + k and b-> b + (n - k)

For small values of a and b, the distribution has high variance

For small a,b, small changes in a and b can impact the expected value

As a and b increase, the distribution becomes narrowly peaked around the mean value, total accumulated successes over attempts


α = β = 4
α = β = 12
α = β = 36
0
1
2
3
4
5
6
7
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

Dynamics

On this version of the model, agents are initialized with random values of (a,b) for each arm, A and B

Agents perform the action they assign the highest expected value

Agents use Bayesian updating based on their results and those of their neighbors

Agents treat each action as a draw from a binomial distribution with unknown probability of success

(Compatible with their credence functions)

Overview

- Model Description
- Representing Beliefs
- **Results**

Single Agent Bayesian Lock-In

First, consider what happens with a single agent and this setup

In the simple model, if the agent begins with $C < .5$, they never change because they never perform the informative action

If they begin with $C > .5$, stochastic effects can push their credence below .5 so they switch to the uninformative action

So with random initial credences, they fail > 50% of the time

In the sophisticated model, a similar effect occurs, where the agent never switches to action B

Consensus

For the simple model, **consensus** means all agents perform action A or all perform B and have arbitrarily high credence that B is better

In the sophisticated model, this means all agents perform the same action, with arbitrarily large a,b and mean close to P(A) or P(B)

In the simple model, consensus around A is stable

Otherwise, consensus is unstable, because stochastic effects can push agents to the wrong action, though only with probability -> 0

Agents in this model **always reach consensus**, eventually (theorem)

Benefits of Social Learning

Agents **usually** (>50 % for virtually all model parameters except network size 1) reach consensus that action B is the better action

Thus, social learning **improves outcomes** compared to single agents

The reason is **transient diversity of beliefs** leading to diverse data

Different agents attempt different actions, generating data about both A and B and reducing lock-in to the false action

Results are broadly similar across the two representations of belief except that A is not stable in the sophisticated model

False Consensus

In some runs of the model, the entire network converges to action A

This is due to premature lock in, where all agents stop performing B

Several factors make this more likely:

- Problem difficulty. As e decreases, it becomes harder for agents to distinguish the actions
- Bad methods. As n decreases, stochasticity of results increases
- Network connectivity. As networks become more connected, accuracy decreases (Zollman effect)

Zollman Effect

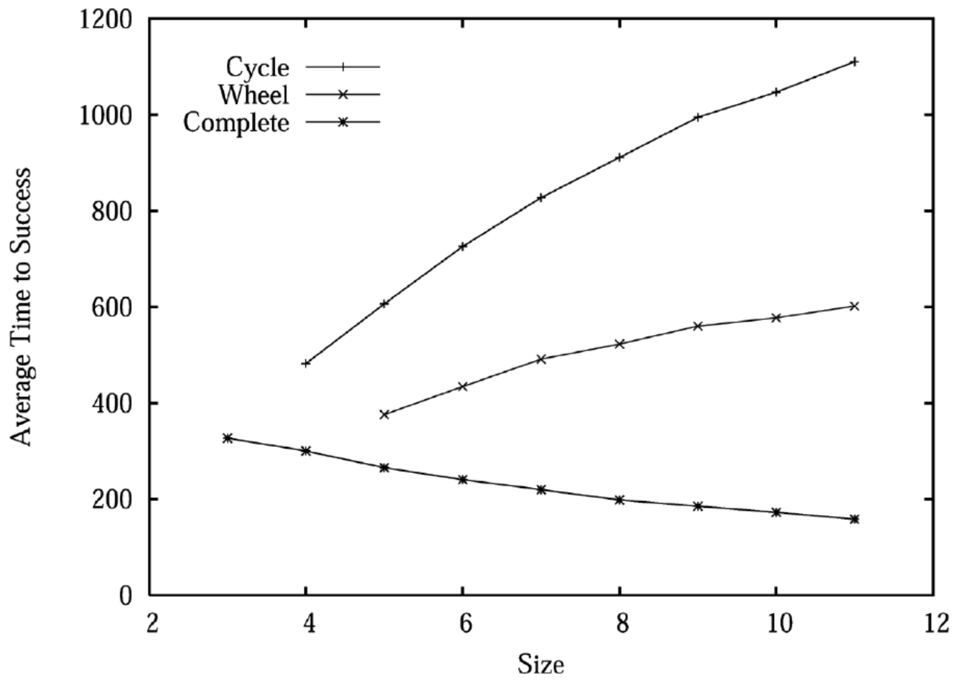
The Zollman Effect is driven by an accuracy / speed trade-off

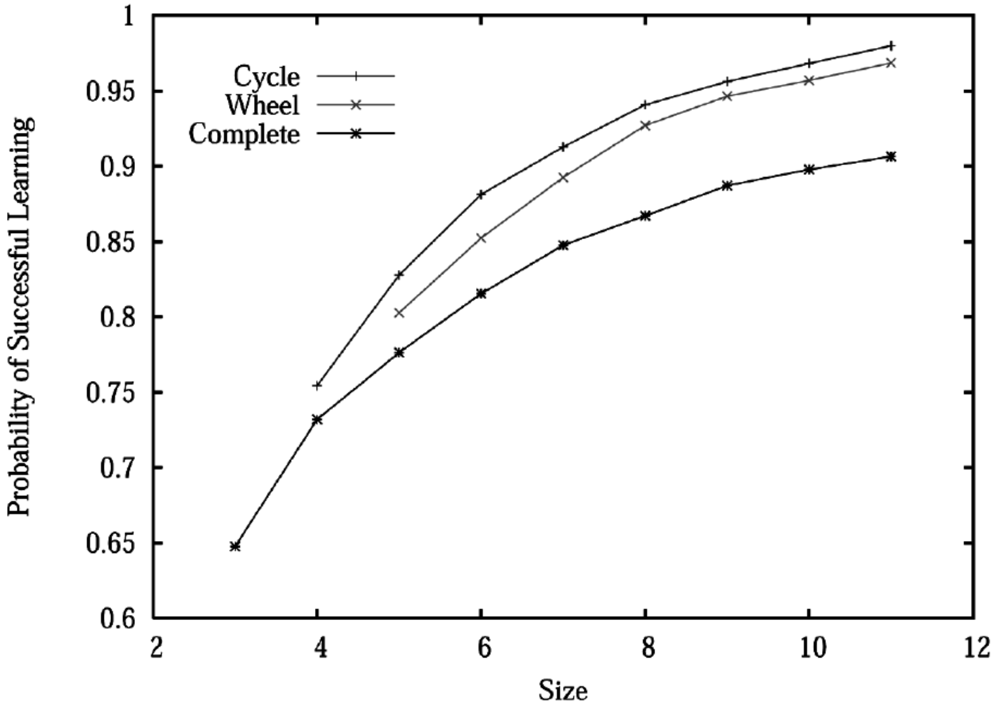
In more connected networks, agents share data more widely and thus all agents learn more quickly

This means stochastic effects pushing towards action A can spread, increasing the chances of lock-in

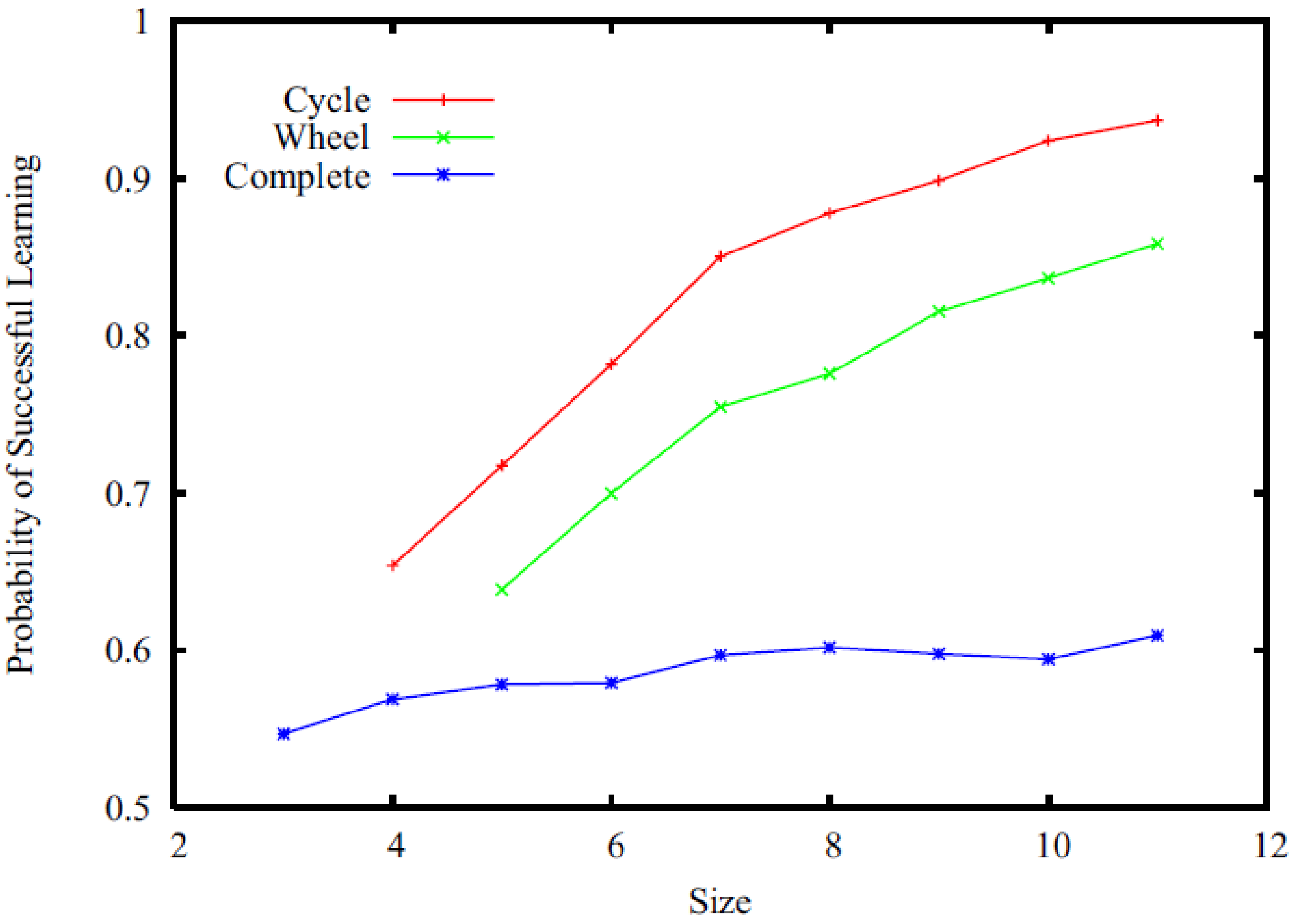
Generally, connectivity suppresses transient diversity of belief

Other mechanisms that increase speed have similar effects



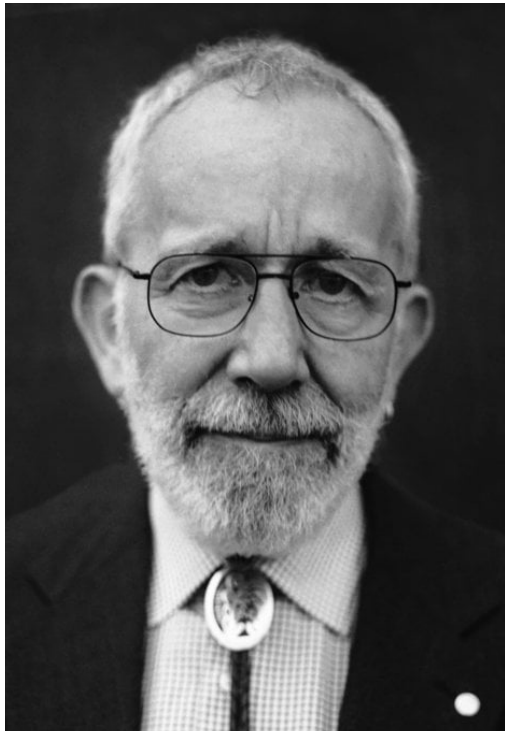


From Zollman (2007), using simple model



From Zollman (2010), using sophisticated model

2005 Nobel Prize



In the mid-1980s, Robin Warren and Barry Marshall showed that PUD is caused by h. pylori

Other researchers thought this impossible

Prior work had “shown” that bacteria could not live in the gut

Remarks and Limitations

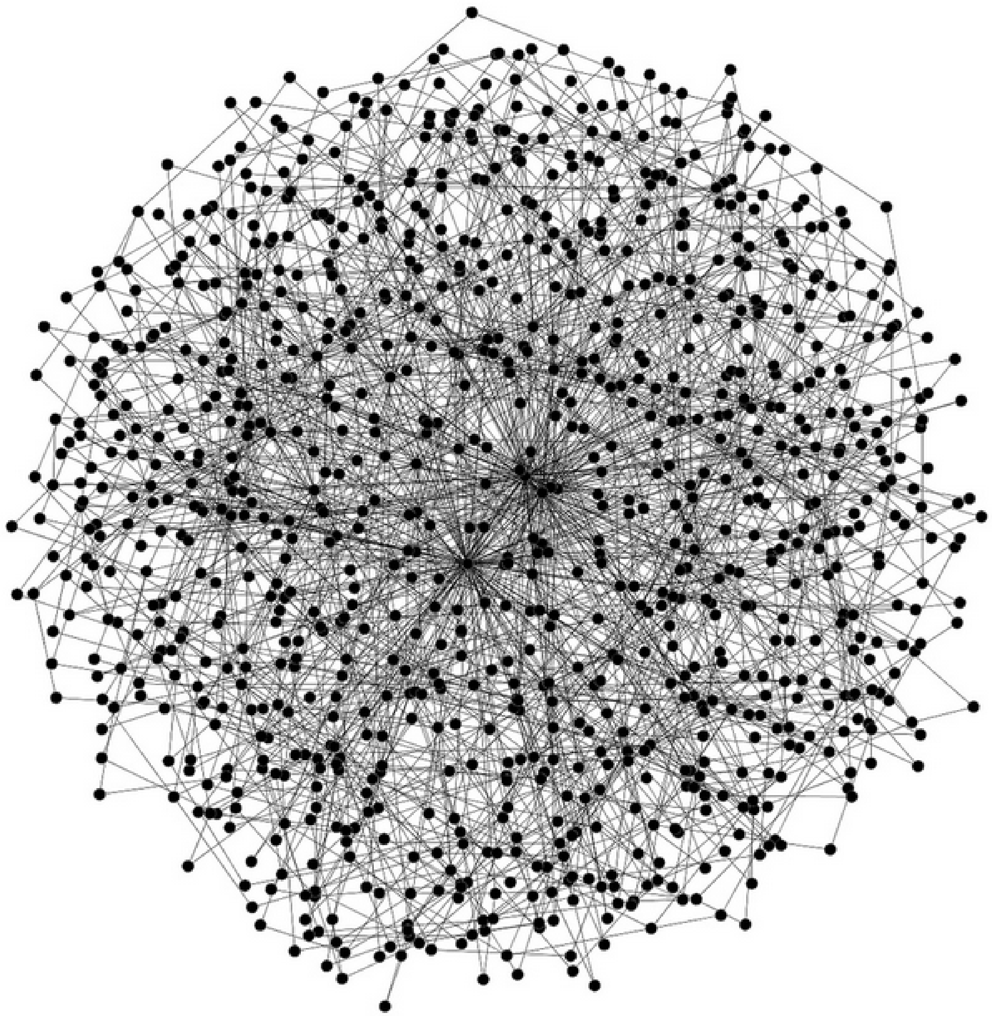
The Zollman Effect has been observed in many different models and in different modeling frameworks (e.g., NK Landscape models)

Even so, it depends on certain modeling assumptions

For instance, for easy problems (large e), large networks, and good methods (large n), the Zollman effect becomes very weak

Additionally, lock-in depends on agent myopia

If agents have “shaky hands” or adopt “epsilon greedy” strategies, lock-in does not occur and true consensus always emerges



Thank you!