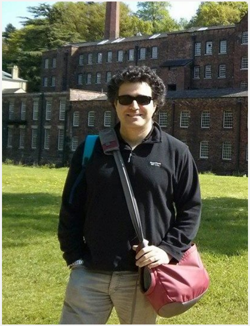
Department of Economics
Bar-Ilan University
ISF Grants: 1626/18 and 448/22

Mentors and Recombinators: Multi-Dimensional Social Learning

Ziv Hellman
(joint with Srinivas Arigapudi, Omer Edhan, and Yuval Heller)

Corvinus, 2022

Co-Authors



Introduction

Ben Zoma said: Who is wise? One who learns from every one, as it is written (Psalms 119:99): ‘From all of my teachers I have gained understanding’.

Ethics of the Fathers, Chapter 4

- Start with review of the standard model in the literature

Introduction

Ben Zoma said: Who is wise? One who learns from every one, as it is written (Psalms 119:99): ‘From all of my teachers I have gained understanding’.

Ethics of the Fathers,
Chapter 4

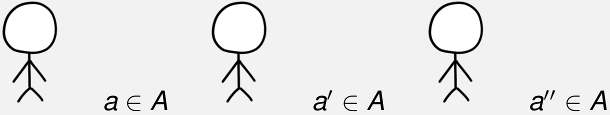
- Start with review of the standard model in the literature
- A continuum of players of mass one is presumed

Introduction

Ben Zoma said: Who is wise? One who learns from every one, as it is written (Psalms 119:99): ‘From all of my teachers I have gained understanding’.

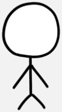
Ethics of the Fathers, Chapter 4

- Start with review of the standard model in the literature
- A continuum of players of mass one is presumed
- There is a set A of actions
- Each player is associated with an action in A, which is that player’s *type*

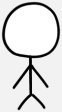


Population State

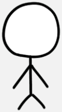
- We want to track over time the state of the population
- By this we mean the frequency distribution of the types: what percentage of players is of type *a*, what percentage of type *a*′ etc



a : 25%



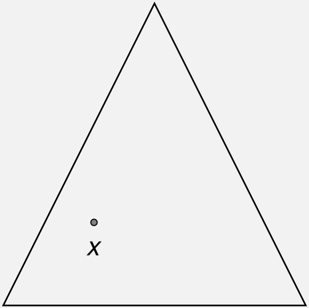
a' : 25%



a'' : 50%

Population State: A Bit More Formally

- The state space is the simplex $\Delta(A)$
- A *population state* $x \in \Delta(A)$ is a distribution of types
- $x(a)$ denotes the frequency of players of type a



Payoff

- Next, we need to specify what causes changes of the state of the population and how it changes
- We suppose that a player of type a gains a payoff $u(a) \in \mathbb{R}^+$, simply by virtue of the type a

Payoff

- However, that payoff to type *a* may depend on the entire state *x* of the population

Payoff

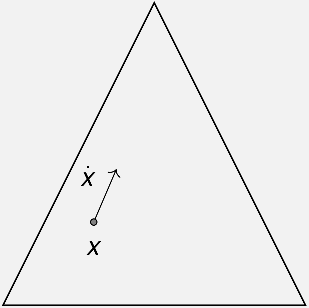
- However, that payoff to type *a* may depend on the entire state *x* of the population
- For example, if I am of type rabbit and nearly everyone else is a rabbit, my payoff will be high
- But if there are many hungry wolves my expected payoff may not be so good
- Hence we write payoff function $u_x(a) \in \mathbb{R}^+$, for $a \in A$ and $x \in \Delta(A)$, with mean payoff $u_x := \sum_{a \in A} x(a) u_x(a)$

Population Dynamic

- The dynamic focus is on the change over time, denoted $\dot{x} = \frac{dx}{dt}$, of the population state x
- This is determined by an equation of motion $\dot{x} = \Phi_u(x)$, as a function of the payoff function u and the current state x
- Different functions Φ_u yield different dynamic behaviours

Population Dynamic, summary

- Payoff function $u_x(a) \in \mathbb{R}^+$, for $a \in A$ and $x \in \Delta(A)$
- An equation of motion is given: $\dot{x} = \Phi_u(x)$, depending on u
- This determines a vector field on $\Delta(A)$
- Integrating the vector field gives population trajectories



Canonical Example

- Canonical example of a vector field equation of motion: the *replicator equation*

$$\frac{\dot{x}(a)}{x(a)} = \frac{1}{u_x}(u_x(a) - u_x)$$

- This equation works for modelling:
- Asexual reproduction
- Social learning of newly born agents from mentors
- Social learning of existing agents by imitating others
- Many more models ...

Payoff Monotonicity

- Under the replicator a state x^* is stationary iff $x^*(a) = x^*(a')$ for all a, a'
- In other words, the dynamic continues until every type gets the same u payoff, and then stops
- Dominated types go extinct asymptotically
- More generally, $\dot{x} = \Phi_u(x)$ is *payoff monotonic* if

$$\frac{\dot{x}(a)}{x(a)} > \frac{\dot{x}(a')}{x(a')} \Longleftrightarrow u_x(a) > u_x(a')$$

- Payoff monotonic dynamics satisfy many of the same results as the replicator dynamics

One Dimensional

- The standard story motivating the replicator is one dimensional:
- An agent imitates another agent and copies a *single* action
- The greater the payoff $u_x(a)$ of an action a, the likelier it will be copied
- Thus 'more successful actions replicate and spread'

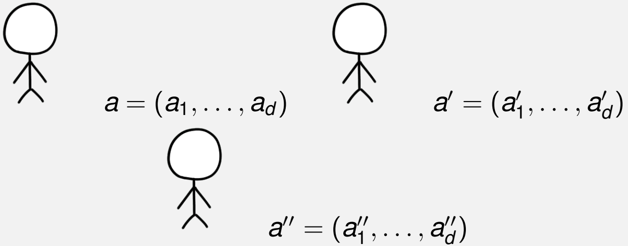
$$\dot{x}(a) = \frac{x(a)}{u_x}(u_x(a) - u_x)$$

Multi Dimensional

- But we are all composed of ensembles of many traits, not only one trait
- For example, how we speak, what we read, how we dress, what we study etc
- We may imitate one person in what to read, imitate another in how we dress, imitate a third in vocabulary, etc
- One might say that each of us has a DNA of traits

Multi Dimensional

- We extend the model, with a set $D = \{1, \ldots, |D|\}$ of *dimensions* of behaviours
- Each $d \in D$ is associated with a set A_d of *traits*
- $A = \times A_d$
- Each $a \in A$ is now a D-tuple, i.e., $a = (a_1, ..., a_{|D|})$
- We can speak of a trait a_d of the d-th dimension within a type a, i.e., $a_d \in a$



Marginal Frequency

- We already have the frequency $x(a)$ of a type a
- Now we also have the *marginal* frequency of a *trait* $a_d \in A_d$

$$x(a_d) := \sum_{a_{-d} \in A_{-d}} x(a_d, a_{-d})$$

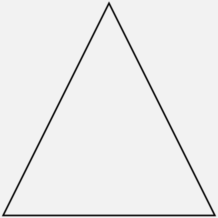
where as standard write $A_{-d} := \prod_{d' \neq d} A_{d'}$

- For each $a_d \in \text{supp}_d(x)$, define the marginal (or mean) payoff of trait a_d:

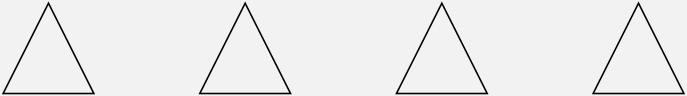
$$u_x(a_d) := \frac{1}{x(a_d)} \sum_{a_{-d} \in A_{-d}} x(a_d, a_{-d}) \cdot u_x(a_d, a_{-d})$$

Several Simplices

At the level of the types, we have a single simplex, $\Delta(A)$

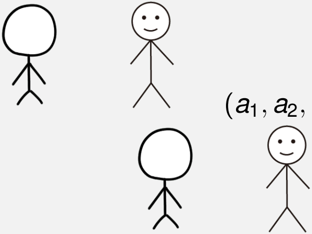


At the level of the traits, we have a cross product of simplices, $\Delta(A_1) \times \ldots \times \Delta(A_{|D|})$



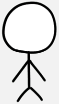
Multi Mentors

- In the *combinator* model, a new born independently samples $|D|$ mentors (with type a selected proportional to $u_x(a)$ and $x(a)$)



(a_1, a_2, a_3) (b_1, b_2, b_3) (c_1, c_2, c_3)

- For each $d \in D$ the new born imitates the d-th trait of the d-th mentor



(a_1, b_2, c_3)

- This models learning from different teachers or different influencers in life

Combinator

- This yields the *combinator* equation of motion:

$$\dot{x}(a) = \prod_{a_d \in a} \frac{x(a_d)u_x(a_d)}{u_x} - x(a)$$

- the first component is the inflow of new agents learning from $|D|$ mentors,
- the second term is the outflow of dying agents

Recombinator

- More generally, let $r \in [0, 1]$
- Suppose that a proportion r of the population samples multiple mentors by the combinator and $1 - r$ selects only one mentor, as in the replicator
- Call the resulting dynamic the *recombinator equation*:

$$\dot{x}(a) = (1 - r)\frac{x(a)u_x(a)}{u_x} + r \prod_{a_d \in a} \frac{x(a_d)u_x(a_d)}{u_x} - x(a)$$

- the first component is the inflow of new agents who imitate a single mentor,
- the second component is the inflow of new agents learning from $|D|$ mentors,
- the last term is the outflow of dying agents

Is there something new here?

Is there something new here?

- The resulting dynamic is quite different from the well studied replicator, in several ways

Forward Invariance

- Under the replicator dynamics the support of any state remains identical along trajectories at all finite times
- this may not hold true (at time $t > 0$) for the recombinator
- because new players may mix together traits from different mentors to create new types that previously did not exist
- This is a source of discontinuity in the model, complicating matters

Payoff Non-Monotonicity

- The recombinator dynamics may violate payoff monotonicity
- Does this mean at stationary states, types can have different payoffs?
- Yes!
- In fact, surprisingly, dominated types might survive
- How can we make sense of what is going on here?

A More General View

- A function $f : A \times \Delta(A) \to [0,1]$ is a *simplicial function* if $\sum_{a \in A} f(a,x) = 1$ for each $x \in \Delta(A)$
- Let $f_1, \ldots, f_k : A \times \Delta(A) \to [0,1]$ be simplicial functions defining k vector fields in the simplices
- Let $\vec{r} = r_1, \ldots r_k \in [0,1]$ satisfy $\sum_{i=1}^{k} r_i = 1$, so we have a convex combination $\sum_{i=1}^{k} r_i f_i(a,x)$
- A population *recombinator dynamic* is

$$\dot{x}(a) = \sum_{i=1}^{k} r_i f_i(a,x) - x(a) \tag{1}$$

- Divide it all by $x(a)$

$$\frac{\dot{x}(a)}{x(a)} = \sum_{i=1}^{k} \frac{r_i f_i(a,x)}{x(a)} - 1$$

$\vec{r}$-Payoff

- Define the $\vec{r}$-payoff of action $a \in \text{supp}(x)$ at x to be the $\vec{r}$ convex combination of $f_i(a,x)/x(a)$ i.e.:

$$\zeta_x^{\vec{r}}(a) := \sum_{i=1}^{m} r_i \frac{f_i(a,x)}{x(a)}.$$

- so

$$\frac{\dot{x}(a)}{x(a)} = \zeta_x^{\vec{r}}(a) - 1$$

Proposition

1. *$\vec{r}$-payoff monotonicity:* $\zeta_x^{\vec{r}}(a) > \zeta_x^{\vec{r}}(a') \Leftrightarrow \frac{\dot{x}(a)}{x(a)} > \frac{\dot{x}(a')}{x(a')}$ *for each state* $x \in \Delta(A)$ *and for each pair of types* $a, a' \in \text{supp}(x)$.
2. *A state x is stationary if and only if* $\zeta_x^{\vec{r}}(a) = 1$ *for all* $a \in \text{supp}(x)$.

Recombinator Payoff Monotonicity

- The recombinator is a special case of the above convex combination of simplicial vector fields (for $a \in \text{supp}(x)$)

$$\frac{\dot{x}(a)}{x(a)} = (1-r)\frac{u_x(a)}{u_x} + r\frac{\prod_{a_d \in a} x(a_d)}{x(a)} \prod_{a_d \in a} \frac{u_x(a_d)}{u_x} - 1$$

- In this special case,

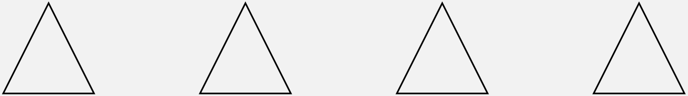
$$\zeta_x^r(a) := (1-r)\frac{u_x(a)}{u_x} + r\frac{\prod_{a_d \in a} x(a_d)}{x(a)} \prod_{a_d \in a} \frac{u_x(a_d)}{u_x}$$

- Conclusion: there *is* payoff monotonicity with respect to ζ_x^r, and x is stationary iff $\zeta_x^{\vec{r}}(a) = 1$ for all $a \in \text{supp}(x)$

Recombinator Payoff Monotonicity

- So the basic payoff function *u* is not monotonic
- But switching attention to the payoff of the $\zeta^{\vec{r}}$ vector field restores payoff monotonicity

- Next we shift attention to the payoffs of the traits *within* each dimension;
- Move focus from the big simplex $\Delta(A)$ to the simplices of the constituent trait dimensions, $\Delta(A_i)$



Partnership

- Focus on a particular dimension or simplex d, and a particular trait $a_d \in A_d$
- Let $f(a, x)$ be a simplicial function
- We want to measure the 'strength' of a_d relative to other traits in the same dimension d
- Think of $a_{-d} \in A_{-d}$ as a set of possible 'partners' of a_d
- When a_d works with those partners the payoff is $f(a_d, a_{-d}, x)$
- The measure of the relative strength of a_d will be the sum of its payoffs for *all* possible partners

Marginal Functions

- Define the *marginal function* of a simplicial function f to be

$$F(a_d, x) = \frac{1}{x(a_d)} \sum_{a'_{-d} \in A_{-d}} f(a_d, a'_{-d}, x)$$

- This enables us to compare $F(a_d, x)$ with $F(a'_d, x)$ for any pair of traits $a_d, a'_d \in A_d$
- f is *marginal payoff increasing* if $u_x(a_d) > u_x(\widehat{a}_d)$ always implies $F(a_d, x) > F(\widehat{a}_d, x)$

Back to Payoff Monotonicity

- Let $\{f_i\}$ be simplicial functions and $\dot{x}(a) = \sum_{i=1}^{k} r_i f_i(a, x) - x(a)$, then for any trait a_d

$$\frac{\dot{x}(a_d)}{x(a_d)} = \sum_{i=1}^{k} r_i F_i(a_d, x) - 1$$

Theorem

If the simplicial functions are marginal payoff increasing then $u_x(a_d) = u_x(a'_d) \Leftrightarrow \frac{\dot{x}(a_d)}{x(a_d)} > \frac{\dot{x}(a'_d)}{x(a'_d)}$,
And x *is a stationary state if and only if* $u_x(a_d) = u_x(a'_d)$ *for each pair of traits* $a_d, a'_d \in supp_d(x)$

Back to the Recombinator

- The recombinator function can be shown to be composed of marginal payoff increasing simplicial functions
- In fact, x is a stationary point of the recombinator if and only if $u_x(a_d) = u_x(a'_d) = u_x$ for all traits

The Traits Perspective

- From the perspective of each *single* dimension, all the traits in that dimension are competing with each other in a dynamic similar to the replicator: the higher $u_x(a_d)$, the greater the relative growth of a_d
- also, the mean payoff of the traits within each dimension equals the mean population payoff
- Put together: we get several inter-related ‘traits games’ (one per dimension) that are all payoff monotonic

The Selfish Trait

- This is an indication that it is the games played by the traits that counts more than that of the types:
- The trajectory continues until stationarity when the payoffs of the traits are equalised, not the types
- This is reminiscent of the concept of the ‘selfish gene’, but here it is the ‘selfish trait’
- We think that we (ourselves, our types) are at the centre of things, when it is really the traits and the competition between them that are running things!

Work in Progress: Asymptotic Stability

- We also characterise asymptotic and Lyapunov stability
- But I have run out of time...

Questions?