ONLINE LEARNING IN GAMES

REGRET, EQUILIBRIUM, AND THE ROAD AHEAD

Panayotis Mertikopoulos

French National Center for Scientific Research (CNRS)

Laboratoire d'Informatique de Grenoble (LIG)

⟨ ROADEF 2022 | Lyon | March 4, 2022 ⟩

Outline

1 Motivation & background

2 Online learning

3 Learning in games

Motivation

The basic premise

Receive a question: *Will it rain tomorrow?*

Motivation

The basic premise

Receive a question: *Will it rain tomorrow?*

Predict/Act: Yes/Take umbrella

Motivation

The basic premise

Receive a question: *Will it rain tomorrow?*

Predict/Act: Yes/Take umbrella

Outcome revealed: 

Motivation

The basic premise

Receive a question: *Will it rain tomorrow?*

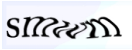
Predict/Act: Yes/Take umbrella

Outcome revealed: 

Obtain payoff: Dry

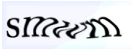
Motivation

The basic premise

Receive a question:	*Will it rain tomorrow?*	
Predict/Act:	Yes/Take umbrella	smum
Outcome revealed:		smwm
Obtain payoff:	Dry	3/4

Motivation

The basic premise

Receive a question:	*Will it rain tomorrow?*		*Won $1,000?*
Predict/Act:	Yes/Take umbrella	smum	Spam
Outcome revealed:		smwm	True!
Obtain payoff:	Dry	3/4	AUGGGH!

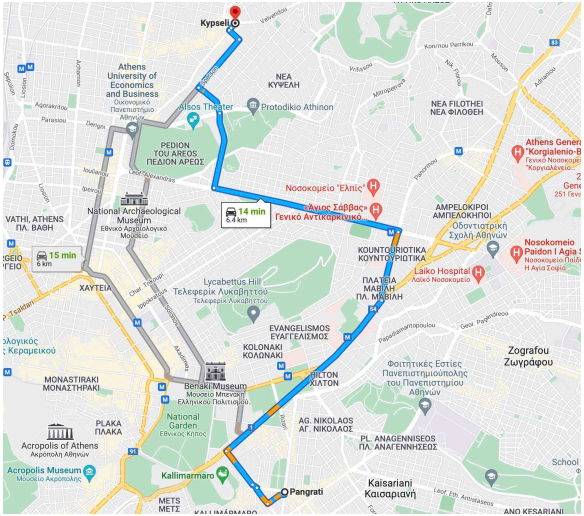
Motivation

The basic premise

Receive a question:	*Will it rain tomorrow?*		*Won $1,000?*
Predict/Act:	Yes/Take umbrella	smum	Spam
Outcome revealed:		smwm	True!
Obtain payoff:	Dry	3/4	AUGGGH!

Goal: get better by learning from previous actions

Game of roads



Athens at a glance

- $3,754,000$ people
- $937,000$ daily trips
- Up to 10^4 trips/min
- 1393 nodes
- 5429 edges
- $1,360,000$ O/D pairs
- $\approx 7 * 10^{18}$ paths

A very large game!

Online learning

A generic **online decision process:**

---

repeat

At each epoch t

- Choose ***action*** # single- / multi-player
- Receive ***reward*** # endogenous / exogenous
- Get ***feedback*** (maybe) # full info / oracle / payoff-based

until end

Online learning

A generic **online decision process:**

---

repeat

At each epoch t

- Choose ***action*** # single- / multi-player
- Receive ***reward*** # endogenous / exogenous
- Get ***feedback*** (maybe) # full info / oracle / payoff-based

until end

Defining elements

- **Time:** continuous or discrete?
- **Players:** continuous or finite?
- **Actions:** continuous or finite?
- **Reward mechanism:** endogenous or exogenous (determined by other players or by "Nature")?
- **Feedback:** observe other actions / other rewards / only received?

Online learning

A generic **online decision process:**

---

repeat

At each epoch t

- Choose ***action*** # single- / multi-player
- Receive ***reward*** # endogenous / exogenous
- Get ***feedback*** (maybe) # full info / oracle / payoff-based

until end

Defining elements

- **Time:** ~~continuous~~ or discrete
- **Players:** ~~continuous~~ or finite
- **Actions:** ~~continuous~~ or finite
- **Reward mechanism:** endogenous or exogenous (determined by other players or by "Nature")?
- **Feedback:** observe other actions / other rewards / only received?

Online vs. multi-agent learning

How are payoffs generated?

Online vs. multi-agent learning

How are payoffs generated?

- *Part I: Online viewpoint (exogenous)*
  - Single, focal agent
  - Different payoff vector encountered at each stage
  - **Agnostic:** no assumptions on reward-generating mechanism (dispassionate Nature)

Online vs. multi-agent learning

How are payoffs generated?

- ***Part I: Online viewpoint (exogenous)***
  - Single, focal agent
  - Different payoff vector encountered at each stage
  - **Agnostic:** no assumptions on reward-generating mechanism (dispassionate Nature)
- ***Part II: Multi-agent viewpoint (endogenous)***
  - Several agents
  - Individual payoff functions depend on actions of other agents
  - **Game-theoretic:** underlying mechanism is a (finite) game

Online vs. multi-agent learning

How are payoffs generated?

- ***Part I: Online viewpoint (exogenous)***
  - Single, focal agent
  - Different payoff vector encountered at each stage
  - **Agnostic:** no assumptions on reward-generating mechanism (dispassionate Nature)
- ***Part II: Multi-agent viewpoint (endogenous)***
  - Several agents
  - Individual payoff functions depend on actions of other agents
  - **Game-theoretic:** underlying mechanism is a (finite) game

What is the interplay between online and multi-agent learning?

Outline

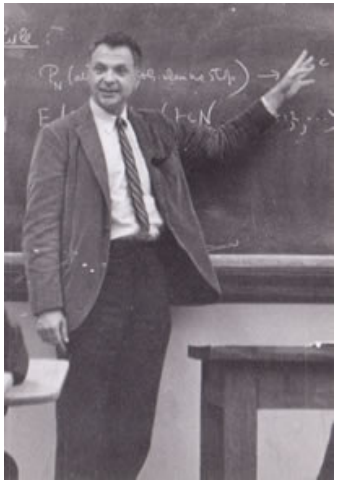
1 Motivation & background

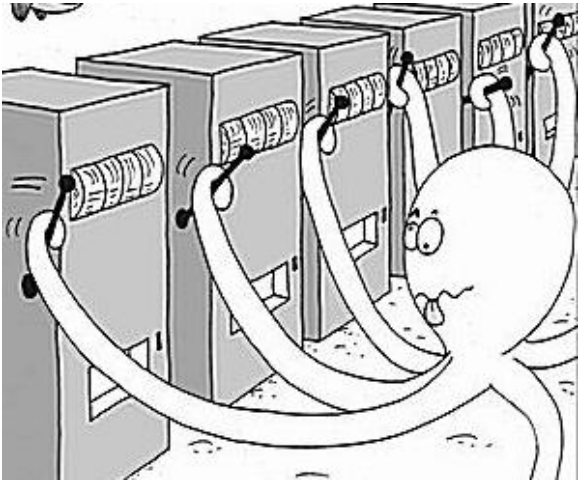
2 Online learning

3 Learning in games

Multi-armed bandits

Robbins' multi-armed bandit problem: **how to play in a (rigged) casino?**





Sequence of events

Require: finite set of actions $\mathcal{A}$; sequence of payoff vectors $v_t \in \mathbb{R}^{\mathcal{A}}$, $t = 1, 2, \ldots$

repeat

At each epoch $t = 1, 2, \ldots$ **do**

Play ***action*** $a_t \in \mathcal{A}$ according to ***mixed strategy*** $x_t \in \mathcal{X} := \Delta(\mathcal{A})$ # action selection

Encounter ***payoff vector*** v_t and receive ***reward*** $u_t(a_t) = v_{a_t,t}$ # payoff function $u_t(a) = v_{a,t}$

Observe **full payoff vector** v_t # payoff-relevant information

until end

Defining elements

- **Time:** discrete
- **Players:** single
- **Actions:** finite
- **Payoffs:** exogenous
- **Feedback:** depends (**full** or partial information, ...)

Sequence of events

Require: finite set of actions $\mathcal{A}$; sequence of payoff vectors $v_t \in \mathbb{R}^{\mathcal{A}}$, $t = 1, 2, \ldots$

repeat

At each epoch $t = 1, 2, \ldots$ **do**

Play ***action*** $a_t \in \mathcal{A}$ according to ***mixed strategy*** $x_t \in \mathcal{X} := \Delta(\mathcal{A})$ # action selection

Encounter ***payoff vector*** v_t and receive ***reward*** $u_t(a_t) = v_{a_t,t}$ # payoff function $u_t(a) = v_{a,t}$

Observe realized payoff $\hat{u}_t = u_t(a_t)$ # payoff-relevant information

until end

Defining elements

- **Time:** discrete
- **Players:** single
- **Actions:** finite
- **Payoffs:** exogenous
- **Feedback:** depends (full or partial information, ...)

Regret

Performance of a policy x_t measured by the agent's **regret**

$$u_t(p) - u_t(x_t)$$

Regret

Performance of a policy x_t measured by the agent's **regret**

$$\sum_{t=1}^{T}[u_t(p) - u_t(x_t)]$$

Regret

Performance of a policy x_t measured by the agent's **regret**

$$\max_{p \in \mathcal{X}} \sum_{t=1}^{T} [u_t(p) - u_t(x_t)]$$

Regret

Performance of a policy x_t measured by the agent's **regret**

$$\mathrm{Reg}(T) = \max_{p \in \mathcal{X}} \sum_{t=1}^{T} [u_t(p) - u_t(x_t)] = \max_{p \in \mathcal{X}} \sum_{t=1}^{T} \langle v_t, p - x_t \rangle$$

Regret

Performance of a policy x_t measured by the agent's **regret**

$$\mathrm{Reg}(T) = \max_{p \in \mathcal{X}} \sum_{t=1}^{T} [u_t(p) - u_t(x_t)] = \max_{p \in \mathcal{X}} \sum_{t=1}^{T} \langle v_t, p - x_t \rangle$$

or, against a **fixed comparator** $p \in \mathcal{X}$:

$$\mathrm{Reg}_p(T) = \sum_{t=1}^{T} [u_t(p) - u_t(x_t)]$$

Regret

Performance of a policy x_t measured by the agent's **regret**

$$\mathrm{Reg}(T) = \max_{p\in\mathcal{X}} \sum_{t=1}^{T} [u_t(p) - u_t(x_t)] = \max_{p\in\mathcal{X}} \sum_{t=1}^{T} \langle v_t, p - x_t \rangle$$

or, against a **fixed comparator** $p \in \mathcal{X}$:

$$\mathrm{Reg}_p(T) = \sum_{t=1}^{T} [u_t(p) - u_t(x_t)]$$

No regret: $\mathrm{Reg}(T) = \max_p \mathrm{Reg}_p(T) = o(T)$ [the smaller the better]

"The chosen policy is as good as the best fixed strategy in hindsight."

Regret

Performance of a policy x_t measured by the agent's **regret**

$$\mathrm{Reg}(T) = \max_{p\in\mathcal{X}} \sum_{t=1}^{T} [u_t(p) - u_t(x_t)] = \max_{p\in\mathcal{X}} \sum_{t=1}^{T} \langle v_t, p - x_t \rangle$$

or, against a **fixed comparator** $p \in \mathcal{X}$:

$$\mathrm{Reg}_p(T) = \sum_{t=1}^{T} [u_t(p) - u_t(x_t)]$$

No regret: $\mathrm{Reg}(T) = \max_p \mathrm{Reg}_p(T) = o(T)$ [the smaller the better]

"The chosen policy is as good as the best fixed strategy in hindsight."

Prolific literature:

- Economics [Hannan, 1957; Erev & Roth, 1998; Fudenberg & Levine, 1998; Hart & Mas-Colell, 2000]
- Optimization [Blackwell, 1956; Shalev-Shwartz, 2011; Bubeck & Cesa-Bianchi, 2012]
- Computer science [Vovk, 1990; Littlestone & Warmuth, 1994; Auer et al., 1995; Lattimore & Szepesvári, 2020]

Feedback model

Different types of feedback (from best to worst):

- **Full, exact information**: observe entire payoff vector v_t
- **Full, inexact information**: observe noisy estimate $\hat{v}_t$ of v_t
- **Partial information / Bandit:** only chosen component $u_t(a_t) = v_{a_t,t}$

Feedback model

Different types of feedback (from best to worst):

- **Full, exact information**: observe entire payoff vector v_t
- **Full, inexact information**: observe noisy estimate $\hat{v}_t$ of v_t
- **Partial information / Bandit:** only chosen component $u_t(a_t) = v_{a_t,t}$

Black-box model

A ***stochastic first-order oracle (SFO)*** model of v_t is a random vector of the form

$$\hat{v}_t = v_t + U_t + b_t \qquad \text{(SFO)}$$

where U_t is **zero-mean** and $b_t = \mathbb{E}[\hat{v}_t \mid \mathcal{F}_t] - v(x_t)$ is the **bias** of $\hat{v}_t$

Assumptions

- ***Bias:*** $\|b_t\| \leq B_t$
- ***Variance:*** $\mathbb{E}[\|U_t\|^2 \mid \mathcal{F}_t] \leq \sigma_t^2$
- ***Second moment:*** $\mathbb{E}[\|\hat{v}_t\|^2 \mid \mathcal{F}_t] \leq M_t^2$

Reconstructing payoffs

Definition (Importance weighted estimators)

The ***importance weighted estimator*** of a vector $v \in \mathbb{R}^{\mathcal{A}}$ given a mixed strategy $x \in \Delta(\mathcal{A})$ is defined as

$$\hat{v}_a = \frac{\mathbb{1}_a}{x_a} v_a = \begin{cases} v_a / x_a & \text{if } a \text{ is drawn } (a = \hat{a}) \\ 0 & \text{otherwise} \quad (a \neq \hat{a}) \end{cases} \tag{IWE}$$

IWE as a black-box oracle

- ***Unbiased:***

$$\mathbb{E}[\hat{v}_a] = v_a$$

- ***Second moment:***

$$\mathbb{E}[\hat{v}_a^2] = \frac{v_a^2}{x_a}$$

Be the leader / Follow the leader

If v_t is known at action time, the agent can ***be the leader (BTL)***

$$x_t \in \arg\max_{x \in \mathcal{X}} \sum_{s=1}^{t} \langle v_s, x \rangle \qquad \text{(BTL)}$$

Regret of BTL

➥ Under (BTL), the learner incurs $\mathrm{Reg}(T) = 0$.

Be the leader / Follow the leader

If v_t is known at action time, the agent can ***be the leader (BTL)***

$$x_t \in \arg\max_{x \in \mathcal{X}} \sum_{s=1}^{t} \langle v_s, x \rangle \qquad \text{(BTL)}$$

Regret of BTL

➥ Under (BTL), the learner incurs $\mathrm{Reg}(T) = 0$.

More realistically, if v_t is only observed **after** playing, the agent can instead ***follow the leader (FTL)***

$$x_{t+1} \in \arg\max_{x \in \mathcal{X}} \sum_{s=1}^{t} \langle v_s, x \rangle \qquad \text{(FTL)}$$

Be the leader / Follow the leader

If v_t is known at action time, the agent can ***be the leader (BTL)***

$$x_t \in \arg\max_{x \in \mathcal{X}} \sum_{s=1}^{t} \langle v_s, x \rangle \qquad \text{(BTL)}$$

Regret of BTL

➥ Under (BTL), the learner incurs $\mathrm{Reg}(T) = 0$.

More realistically, if v_t is only observed **after** playing, the agent can instead ***follow the leader (FTL)***

$$x_{t+1} \in \arg\max_{x \in \mathcal{X}} \sum_{s=1}^{t} \langle v_s, x \rangle \qquad \text{(FTL)}$$

Does (FTL) lead to no regret?

The failure of FTL

Playing against ***omniscient adversary:***

- FTL outputs $a_t \in \{0,1\}$
- Adversary sets $v_t(a) = |a_t - a| \mod 2$
- Aggregate incurred loss:

$$\sum_{t=1}^{T} |1| = T$$

- One ***constant policy*** (0 or 1) would have done at most $T/2$ mistakes in hindsight
- $\Omega(T)$ **gap from best fixed comparator** [Cover 1965]

Follow the regularized leader

Add a fictitious "zero-day bonus" $\Longrightarrow$ ***follow the regularized leader (FTRL)***

$$x_{t+1} = \arg\max_{x \in \mathcal{X}} \left\{ \sum_{s=1}^{t} \langle v_s, x \rangle - \underbrace{\lambda h(x)}_{\text{"day zero"}} \right\} \tag{FTRL}$$

[Algorithm due to Shalev-Shwartz & Singer, 2006; Shalev-Shwartz, 2011]

- The ***regularization function*** $h : \mathcal{X} \to \mathbb{R}$ is continuous and strongly convex
- The ***regularization weight*** $\lambda > 0$ can be tuned at will by the learner

Main idea: Regularization $\Longrightarrow$ Stability $\Longrightarrow$ Less regret

Examples

Example 1: Ridge regularization

- **Regularizer:**

$$h(x) = \tfrac{1}{2}\|x\|^2$$

- **Algorithm:**

$$x_{t+1} = \arg\max_{x \in \mathcal{X}} \left\{ \sum_{s=1}^{t} \langle v_s, x \rangle - \frac{\lambda}{2}\|x\|^2 \right\} = \mathrm{proj}_{\mathcal{X}}\left(\sum_{s=1}^{t} v_s \right)$$

Examples

Example 1: Ridge regularization

- **Regularizer:**

$$h(x) = \tfrac{1}{2}\|x\|^2$$

- **Algorithm:**

$$x_{t+1} = \arg\max_{x \in \mathcal{X}} \left\{ \sum_{s=1}^{t} \langle v_s, x \rangle - \frac{\lambda}{2}\|x\|^2 \right\} = \mathrm{proj}_{\mathcal{X}}\left(\sum_{s=1}^{t} v_s \right)$$

Example 2: Entropic regularization

- **Regularizer:**

$$h(x) = \sum_{a \in \mathcal{A}} x_a \log x_a$$

- **Algorithm:**

$$x_{t+1} = \arg\max_{x \in \mathcal{X}} \left\{ \sum_{s=1}^{t} \langle v_s, x \rangle - \lambda \sum_{a \in \mathcal{A}} x_a \log x_a \right\} = \frac{\exp(\sum_{s=1}^{t} v_{a,s}/\lambda)}{\sum_{a' \in \mathcal{A}} \exp(\sum_{s=1}^{t} v_{a',s}/\lambda)}$$

The exponential weights algorithm

Algorithm Exponential weights with oracle feedback (ExpWeight)

Require: set of actions $\mathcal{A}$; sequence of payoff vectors v_t and SFO feedback $\hat{v}_t$, $t = 1, 2, \ldots$

Initialize: $y_1 \in \mathbb{R}^{\mathcal{A}}$

for all $t = 1, 2, \ldots$ **do**

- **set** $x_t \leftarrow \Lambda(y_t)$ # mixed strategy
- **play** $a_t \sim x_t$ and **receive** $u_t(a_t) = v_{a_t,t}$ # choose action / get payoff
- **observe** $\hat{v}_t \in \mathbb{R}^{\mathcal{A}}$ # receive feedback
- **set** $y_{t+1} \leftarrow y_t + \gamma_t \hat{v}_t$ # update scores

end for

Basic idea:

- Score actions by aggregating payoff-relevant feedback
- Choose actions with probability exponentially proportional to their scores
- Rinse / repeat

Regret guarantees of FTRL

Theorem (Shalev-Shwartz & Singer, 2007; Shalev-Shwartz, 2011)

➥ **Assume:**

- *SFO feedback, unbiased and bounded in mean square* ($B_t = 0, \sup_t M_t < M$)
- $\lambda = M\sqrt{T/(2KH)}$ *with* $H = \max h - \min h$

✍ ***Then:*** *FTRL enjoys the bound*

$$\overline{\mathrm{Reg}}(T) \leq 2M\sqrt{(H/K)T} = \mathcal{O}(\sqrt{T})$$

Regret guarantees of FTRL

Theorem (Shalev-Shwartz & Singer, 2007; Shalev-Shwartz, 2011)

➥ **Assume:**

- *SFO feedback, unbiased and bounded in mean square* ($B_t = 0, \sup_t M_t < M$)
- $\lambda = M\sqrt{T/(2KH)}$ *with* $H = \max h - \min h$

✍ ***Then:*** *FTRL enjoys the bound*

$$\overline{\mathrm{Reg}}(T) \leq 2M\sqrt{(H/K)T} = \mathcal{O}(\sqrt{T})$$

Observe:

- Dependence in T cannot be improved [Nesterov, 2004; Abernethy et al., 2008; Bubeck, 2015]
- Can extend to unknown T via ***doubling trick*** [Cesa-Bianchi & Lugosi, 2006]

Are all regularizers created equal?

Full info guarantees

[assume $0 \le v_{a,t} \le 1$ for simplicity]

- **Euclidean regularization**
  - L^2-norm bound $M = |\mathcal{A}|^{1/2}$; strong convexity modulus $K = 1$; $H \le 1/2$
  - Optimal tuning gives

$$\overline{\mathrm{Reg}}(T) \le 2\sqrt{|\mathcal{A}| \cdot T}$$

Are all regularizers created equal?

Full info guarantees

[assume $0 \le v_{a,t} \le 1$ for simplicity]

- **Euclidean regularization**
  - L^2-norm bound $M = |\mathcal{A}|^{1/2}$; strong convexity modulus $K = 1$; $H \le 1/2$
  - Optimal tuning gives
$$\overline{\mathrm{Reg}}(T) \le 2\sqrt{|\mathcal{A}| \cdot T}$$
- **ExpWeight**
  - L^∞-norm bound $M = 1$; strong convexity modulus $K = 1$; $H = \log|\mathcal{A}|$
  - Optimal tuning gives
$$\overline{\mathrm{Reg}}(T) \le 2\sqrt{\log|\mathcal{A}| \cdot T}$$
  - **Huge reduction in dimensionality!**

Are all regularizers created equal?

Full info guarantees

[assume $0 \le v_{a,t} \le 1$ for simplicity]

- **Euclidean regularization**
  - L^2-norm bound $M = |\mathcal{A}|^{1/2}$; strong convexity modulus $K = 1$; $H \le 1/2$
  - Optimal tuning gives
$$\overline{\mathrm{Reg}}(T) \le 2\sqrt{|\mathcal{A}| \cdot T}$$
- **ExpWeight**
  - L^∞-norm bound $M = 1$; strong convexity modulus $K = 1$; $H = \log|\mathcal{A}|$
  - Optimal tuning gives
$$\overline{\mathrm{Reg}}(T) \le 2\sqrt{\log|\mathcal{A}| \cdot T}$$
  - **Huge reduction in dimensionality!**

Bandit guarantees

[ExpWeight + IWE = EXP3]

- EXP3 guarantees [Auer et al., 1995]
$$\overline{\mathrm{Reg}}(T) \le \sqrt{2|\mathcal{A}|\log|\mathcal{A}| \cdot T}$$
- This can be reduced to $\mathcal{O}(\sqrt{|\mathcal{A}|T})$ but no further [Audibert & Bubeck, 2010]

Outline

1 Motivation & background

2 Online learning

3 Learning in games

Game-theoretic learning

- **Multiple agents**, individual objectives
- Payoffs determined by actions of **all** agents
- Agents receive payoffs, **adjust actions**, and the process repeats

Game-theoretic learning

- **Multiple agents**, individual objectives

  [Select a route from home to work]

- Payoffs determined by actions of **all** agents

  [Encounter other commuters on the road]

- Agents receive payoffs, **adjust actions**, and the process repeats

  [Update road choice tomorrow]

Game-theoretic learning

- **Multiple agents**, individual objectives

  [Select a route from home to work]

- Payoffs determined by actions of **all** agents

  [Encounter other commuters on the road]

- Agents receive payoffs, **adjust actions**, and the process repeats

  [Update road choice tomorrow]

Does no-regret learning lead to stable / rational outcomes?

Some basics

What's in a game?

A ***game in normal form*** is a collection of three basic elements:

1. A set of ***players*** $\mathcal{N}$
2. A set of ***actions*** (or ***pure strategies***) $\mathcal{A}_i$ per player $i \in \mathcal{N}$
3. An ensemble of ***payoff functions*** $u_i \colon \prod_j \mathcal{A}_j \to \mathbb{R}$ per player $i \in \mathcal{N}$

Some basics

What's in a game?

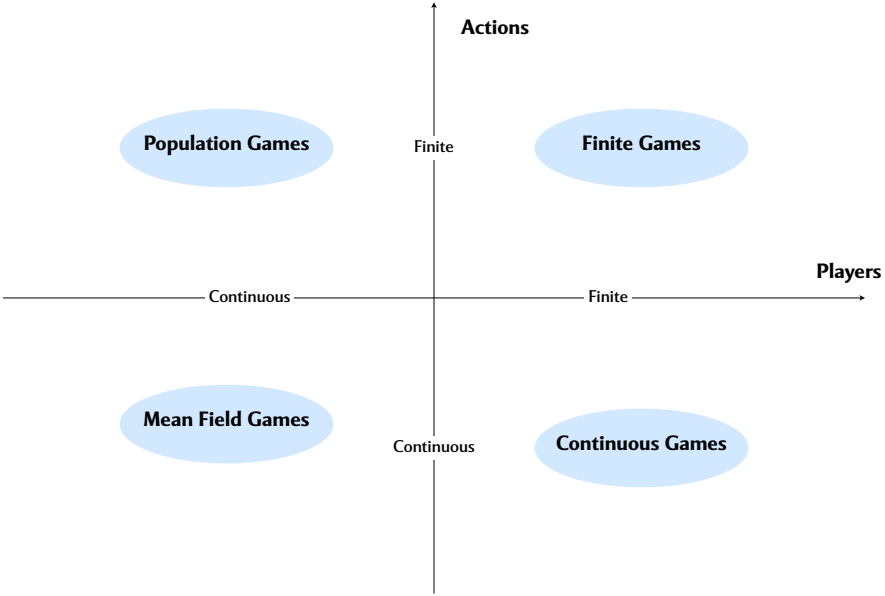
A ***game in normal form*** is a collection of three basic elements:

1. A set of ***players*** $\mathcal{N}$
2. A set of ***actions*** (or ***pure strategies***) $\mathcal{A}_i$ per player $i \in \mathcal{N}$
3. An ensemble of ***payoff functions*** $u_i \colon \prod_j \mathcal{A}_j \to \mathbb{R}$ per player $i \in \mathcal{N}$

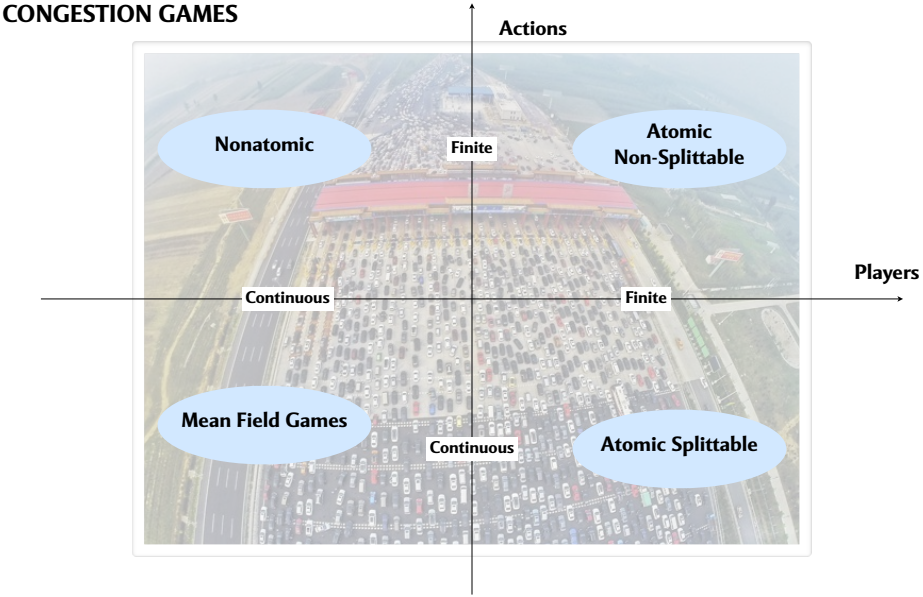
Important:

- Player set: atomic vs. nonatomic
- Action sets: finite vs. continuous; shared vs. individual; ...
- **NB:** do not mix game classes!

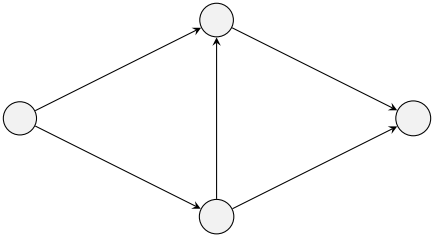
Taxonomy


Actions
Finite
Population Games
Finite Games
Players
Continuous
Finite
Mean Field Games
Continuous
Continuous Games

Taxonomy

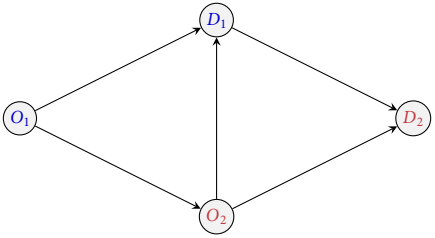

CONGESTION GAMES
Actions
Players
Nonatomic
Finite
Atomic
Non-Splittable
Continuous
Finite
Mean Field Games
Continuous
Atomic Splittable

Congestion games



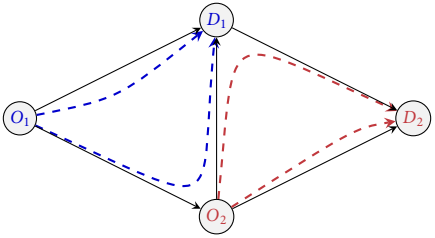
- ***Network:*** multigraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

Congestion games



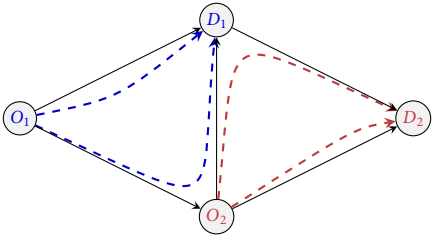
- ***Network:*** multigraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- ***O/D pairs*** $i \in \mathcal{N}$**:** origin O_i sends m_i units of traffic to destination D_i [atomic, non-splittable]

Congestion games



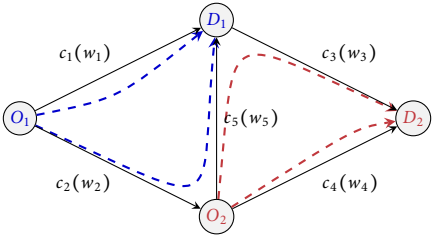
- ***Network:*** multigraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- ***O/D pairs*** $i \in \mathcal{N}$**:** origin O_i sends m_i units of traffic to destination D_i [atomic, non-splittable]
- ***Set of routes*** $\mathcal{A}_i$**:** (sub)set of paths joining $O_i \rightsquigarrow D_i$ [not necessarily *all* paths]

Congestion games



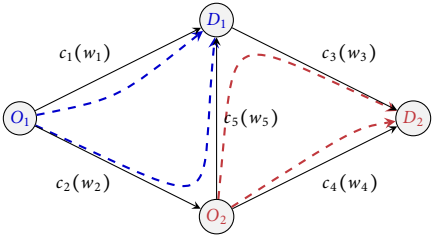
- ***Network:*** multigraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- ***O/D pairs*** $i \in \mathcal{N}$**:** origin O_i sends m_i units of traffic to destination D_i [atomic, non-splittable]
- ***Set of routes*** $\mathcal{A}_i$**:** (sub)set of paths joining $O_i \rightsquigarrow D_i$ [not necessarily *all* paths]
- ***Routing choice*** $a_i \in \mathcal{A}_i \rightsquigarrow$ congestion load of m_i units along each edge $e \in a_i$ [congestion elements]
- ***Load*** $w_e = \sum_{a_i \ni e} m_i$**:** total congestion load on edge e [congestion mechanism]

Congestion games



- ***Network:*** multigraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- ***O/D pairs*** $i \in \mathcal{N}$**:** origin O_i sends m_i units of traffic to destination D_i [atomic, non-splittable]
- ***Set of routes*** $\mathcal{A}_i$**:** (sub)set of paths joining $O_i \rightsquigarrow D_i$ [not necessarily *all* paths]
- ***Routing choice*** $a_i \in \mathcal{A}_i \rightsquigarrow$ congestion load of m_i units along each edge $e \in a_i$ [congestion elements]
- ***Load*** $w_e = \sum_{a_i \ni e} m_i$**:** total congestion load on edge e [congestion mechanism]
- ***Edge cost function*** $c_e(w_e)$**:** cost along edge e when edge load is w_e [congestion cost]
- ***Path cost:*** $c_a(f) = \sum_{e \in a} c_e(w_e)$ [aggregate cost]

Congestion games



- ***Network:*** multigraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- ***O/D pairs*** $i \in \mathcal{N}$**:** origin O_i sends m_i units of traffic to destination D_i [atomic, non-splittable]
- ***Set of routes*** $\mathcal{A}_i$**:** (sub)set of paths joining $O_i \rightsquigarrow D_i$ [not necessarily *all* paths]
- ***Routing choice*** $a_i \in \mathcal{A}_i \rightsquigarrow$ congestion load of m_i units along each edge $e \in a_i$ [congestion elements]
- ***Load*** $w_e = \sum_{a_i \ni e} m_i$**:** total congestion load on edge e [congestion mechanism]
- ***Edge cost function*** $c_e(w_e)$**:** cost along edge e when edge load is w_e [congestion cost]
- ***Path cost:*** $c_a(f) = \sum_{e \in a} c_e(w_e)$ [aggregate cost]
- **Congestion game:** $\Gamma = (\mathcal{G}, \mathcal{N}, \{m_i\}_{i \in \mathcal{N}}, \{\mathcal{A}_i\}_{i \in \mathcal{N}}, \{c_e\}_{e \in \mathcal{E}})$

Finite games

Finite games:

[sometimes known as *(poly)matrix games*]

- Finite set of ***players*** $\mathcal{N} = \{1, \dots, N\}$
- Finite set of ***actions*** (or "***pure strategies***") $\mathcal{A}_i = \{1, \dots, A_i\}$ per player
- Action profile $a = (a_1, \dots, a_N) \in \mathcal{A} := \prod_i \mathcal{A}_i$
- Payoffs given by ***payoff functions*** $u_i : \mathcal{A} \to \mathbb{R}$

$$u_i(a) \equiv u_i(a_1, \dots, a_N) \equiv u_i(a_i; a_{-i})$$

- ***Pure payoff vector*** of player i:

$$v_i(a) = (u_i(a'_i; a_{-i}))_{a'_i \in \mathcal{A}_i}$$

Finite games

Finite games:

[sometimes known as *(poly)matrix games*]

- Finite set of ***players*** $\mathcal{N} = \{1, \dots, N\}$
- Finite set of ***actions*** (or "***pure strategies***") $\mathcal{A}_i = \{1, \dots, A_i\}$ per player
- Action profile $a = (a_1, \dots, a_N) \in \mathcal{A} := \prod_i \mathcal{A}_i$
- Payoffs given by ***payoff functions*** $u_i \colon \mathcal{A} \to \mathbb{R}$

$$u_i(a) \equiv u_i(a_1, \dots, a_N) \equiv u_i(a_i; a_{-i})$$

- ***Pure payoff vector*** of player i:

$$v_i(a) = (u_i(a_i'; a_{-i}))_{a_i' \in \mathcal{A}_i}$$

Mixed extensions of finite games:

- Players, actions and payoffs as above
- ***Mixed strategy*** of player i:

$$x_i = (x_{ia})_{a \in \mathcal{A}_i} \in \Delta(\mathcal{A}_i) =: \mathcal{X}_i$$

- ***Mixed payoff*** of player i

$$u_i(x) = \mathbb{E}_{a \sim x}\, u_i(a) = \sum_{a_1 \in \mathcal{A}_1} \dots \sum_{a_N \in \mathcal{A}_N} x_{1,a_1} \cdots x_{N,a_N}\, u_i(a_1, \dots, a_N)$$

- ***Mixed payoff vector*** of player i:

$$v_i(x) = (u_i(a_i; x_{-i}))_{a_i \in \mathcal{A}_i}$$

Nash equilibrium

Nash equilibrium (Nash, 1950, 1951)

"No player has an incentive to deviate from their chosen strategy if other players don't"

$$u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*) \quad \text{for all } x_i \in \mathcal{X}_i,\ i \in \mathcal{N} \tag{NE}$$

Nash equilibrium

Nash equilibrium (Nash, 1950, 1951)

"No player has an incentive to deviate from their chosen strategy if other players don't"

$$u_i(x_i^*; x_{-i}^*) \geq u_i(x_i; x_{-i}^*) \quad \text{for all } x_i \in \mathcal{X}_i,\ i \in \mathcal{N} \tag{NE}$$

Nash equilibria always exist *but can they be learned?*

No regret and equilibrium

No-regret learning converges to equilibrium!

No regret and equilibrium

Under no-regret learning, **empirical frequencies** converge to equilibrium

No regret and equilibrium

Under no-regret learning, empirical frequencies of play converge to coarse correlated equilibrium

No regret and equilibrium

Under no-regret learning, **empirical frequencies of play** converge to **coarse correlated** equilibrium ¯_(ツ)_/¯

No regret and equilibrium

Under no-regret learning, **empirical frequencies of play** converge to **coarse correlated** equilibrium ¯_(ツ)_/¯

✗ **Very weak notion of "convergence"**

$\rightsquigarrow$ stray arbitrarily far from equilibrium infinitely often

[Hart & Mas-Colell, 2000, 2003]

No regret and equilibrium

Under no-regret learning, **empirical frequencies of play** converge to **coarse correlated** equilibrium ¯_(ツ)_/¯

✗ **Very weak notion of "convergence"**

⇝ stray arbitrarily far from equilibrium infinitely often

[Hart & Mas-Colell, 2000, 2003]

✗ **Very weak notion of "equilibrium"**

⇝ may assign positive weight only to **strictly dominated** strategies

[Viossat & Zapechelnyuk, 2013]

Exponential weights redux

Algorithm Exponential weights in discrete time (ExpWeight)

Require: finite game $\Gamma \equiv \Gamma(\mathcal{N}, \mathcal{A}, u)$; payoff vector oracle $\hat{v}$

Initialize: $y_i \in \mathbb{R}^{\mathcal{A}_i}$, $i = 1, \dots, N$

for all $t = 1, 2, \dots$ all players $i \in \mathcal{N}$ **do simultaneously**

set $x_{i,t} \propto \exp(y_{i,t})$ # mixed strategy

play $a_{i,t} \sim x_{i,t}$ # choose action

get estimate $\hat{v}_{i,t}$ of $v_i(x_t)$ # oracle feedback

set $y_{i,t+1} \leftarrow y_{i,t} + \gamma_t \hat{v}_{i,t}$ # update scores

end for

Basic idea:

- Score actions by aggregating payoff vector estimates provided by oracle
- Choose actions with probability exponentially proportional to their scores
- Rinse / repeat

NB: extends similarly to all FTRL policies

Model 1: ExpWeight with mixed vector feedback

If players observe ***mixed payoff vectors:***

$$\hat{v}_{i,t} = v_i(x_{i,t}; x_{-i,t})$$

Oracle features:

- ***Deterministic:*** no randomness!
- ***Bias:*** $B_t = 0$
- ***Variance:*** $\sigma_t = 0$
- ***Second moment:*** $M_t = \mathcal{O}(1)$

Model 2: ExpWeight with pure vector feedback

If players observe ***pure payoff vectors:***

$$\hat{v}_{i,t} = v_i(a_{i,t}; a_{-i,t})$$

Oracle features:

- ***Stochastic:*** random action selection
- ***Bias:*** $B_t = 0$
- ***Variance:*** $\sigma_t = \mathcal{O}(1)$
- ***Second moment:*** $M_t = \mathcal{O}(1)$

NB: this algorithm is known as as **Hedge** [Auer et al., 1995, 2002b,a]

Model 3: ExpWeight with bandit feedback

If players observe ***realized payoffs only:***

$$\hat{v}_{i,t} = \frac{u_i(a_{i,t}; a_{-i,t})}{\mathbb{P}(a_{i,t} = a_i)} e_{a_{i,t}}$$

Oracle features:

- ***Stochastic:*** random action selection
- ***Bias:*** $B_t = 0$
- ***Variance:*** $\sigma_t = \mathcal{O}(1/x_{ia_i,t})$
- ***Second moment:*** $M_t = \mathcal{O}(1/x_{ia_i,t})$

NB: this algorithm is known as as **EXP3** [Auer et al., 1995, 2002b,a]

Model 4: ExpWeight with bandit feedback

If players observe ***realized payoffs only:***

$$\hat{v}_{i,t} = \frac{u_i(a_{i,t}; a_{-i,t})}{\mathbb{P}(a_{i,t} = a_i)} e_{a_{i,t}}$$

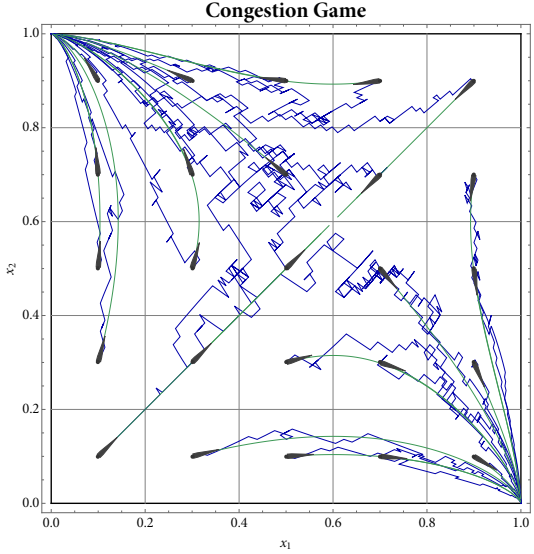
Oracle features:

- ***Stochastic:*** random action selection
- ***Explicit exploration:*** draw $a_{i,t} \sim x_{i,t}$ with prob. $1 - \delta_t$, otherwise uniformly
- ***Bias:*** $B_t = \mathcal{O}(\delta_t)$
- ***Variance:*** $\sigma_t = \mathcal{O}(1/\delta_t^2)$
- ***Second moment:*** $M_t = \mathcal{O}(1/\delta_t^2)$

NB: this algorithm is known as as **EXP3 WITH EXPLICIT EXPLORATION** [Shalev-Swartz, 2011; Lattimore & Szepesvári, 2020]

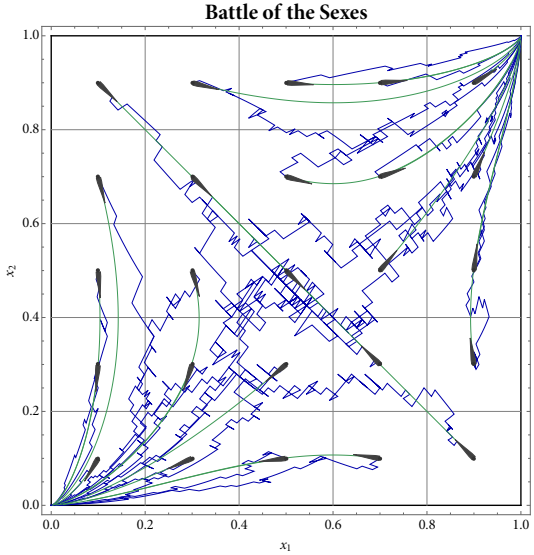
Visualization

What does the sequence of play look like?



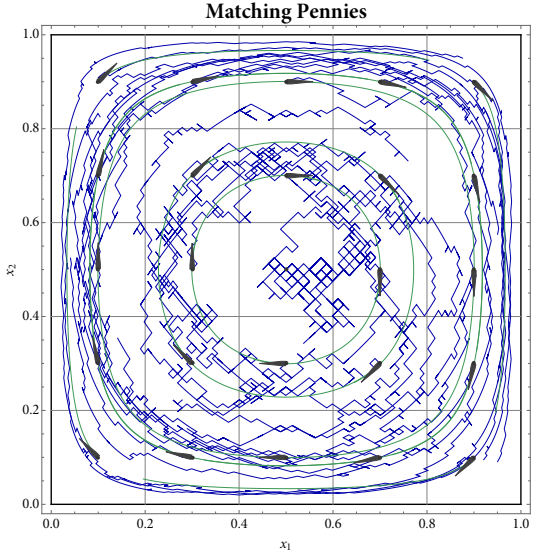
Visualization

What does the sequence of play look like?



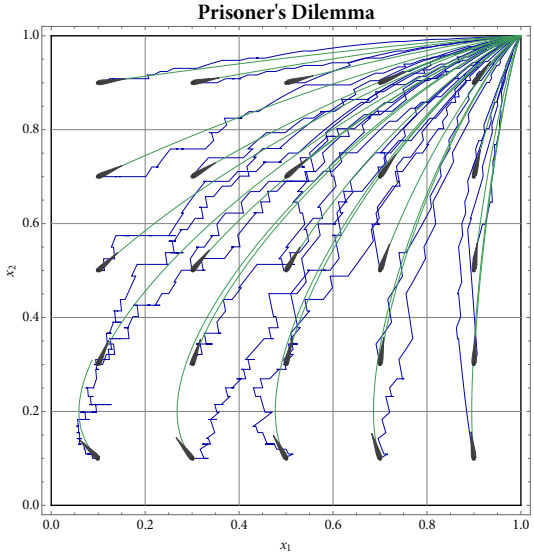
Visualization

What does the sequence of play look like?



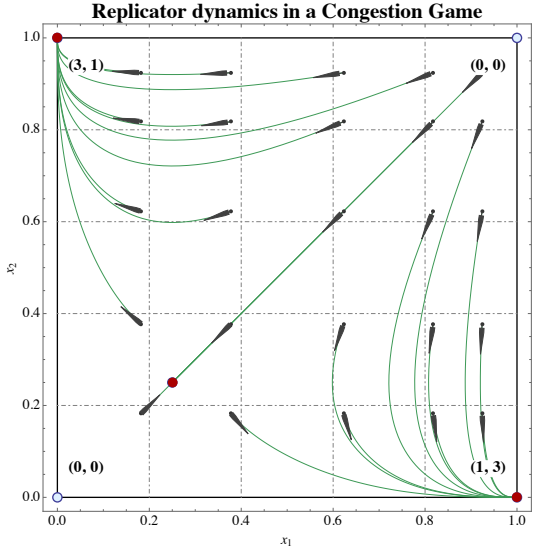
Visualization

What does the sequence of play look like?



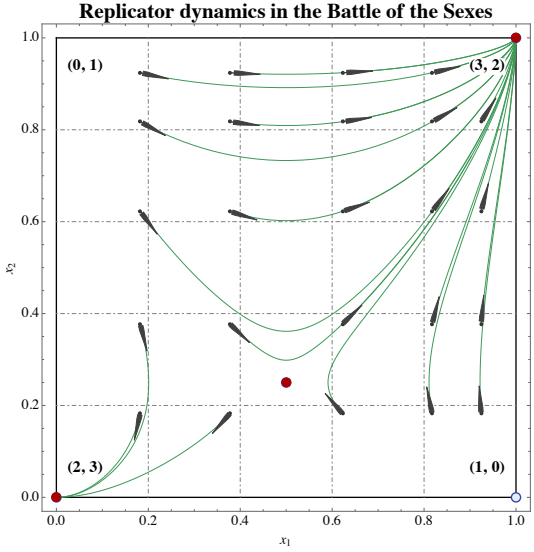
Phase portraits

Mean evolution follows the **replicator dynamics:** $\dot{x}_{ia} = x_{ia}\left[v_{ia}(x) - u_i(x)\right]$ [Taylor & Jonker, 1978]



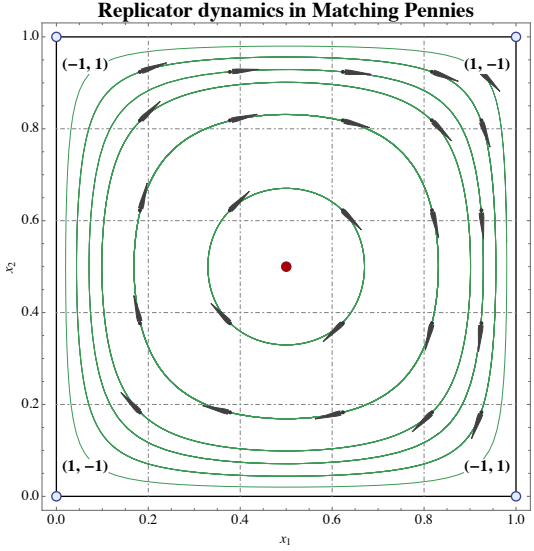
Phase portraits

Mean evolution follows the **replicator dynamics:** $\dot{x}_{ia} = x_{ia}\left[v_{ia}(x) - u_i(x)\right]$ [Taylor & Jonker, 1978]



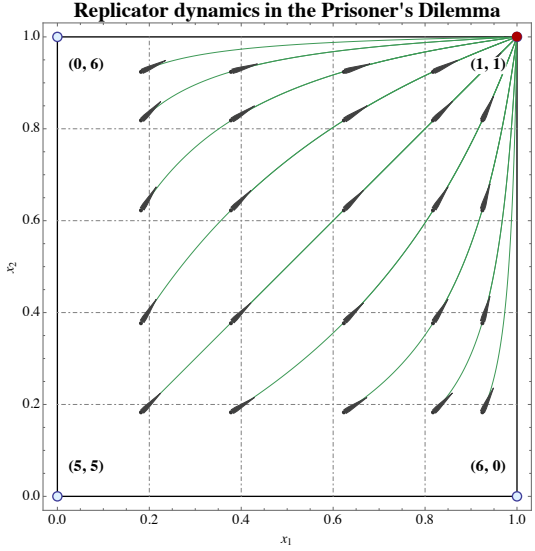
Phase portraits

Mean evolution follows the **replicator dynamics:** $\dot{x}_{ia} = x_{ia}\left[v_{ia}(x) - u_i(x)\right]$ [Taylor & Jonker, 1978]



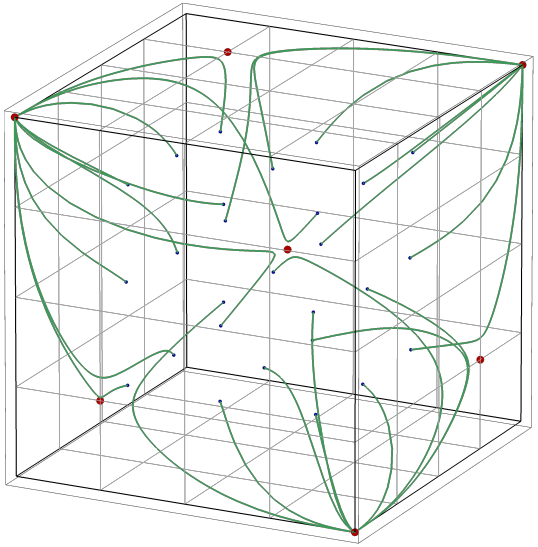
Phase portraits

Mean evolution follows the **replicator dynamics:** $\dot{x}_{ia} = x_{ia}\left[v_{ia}(x) - u_i(x)\right]$ [Taylor & Jonker, 1978]



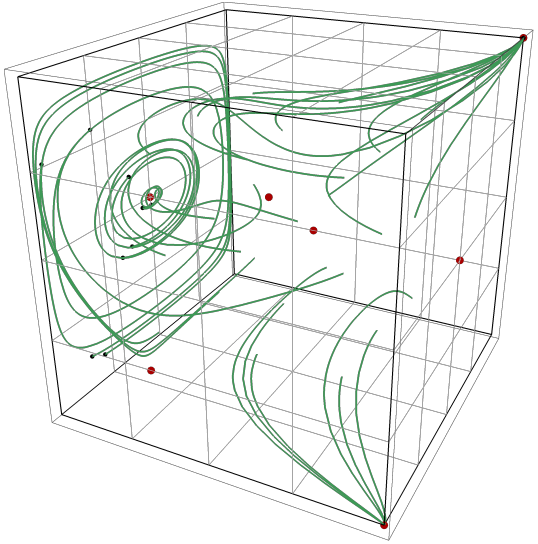
Phase portraits

Mean evolution follows the **replicator dynamics:** $\dot{x}_{ia} = x_{ia}\left[v_{ia}(x) - u_i(x)\right]$ [Taylor & Jonker, 1978]



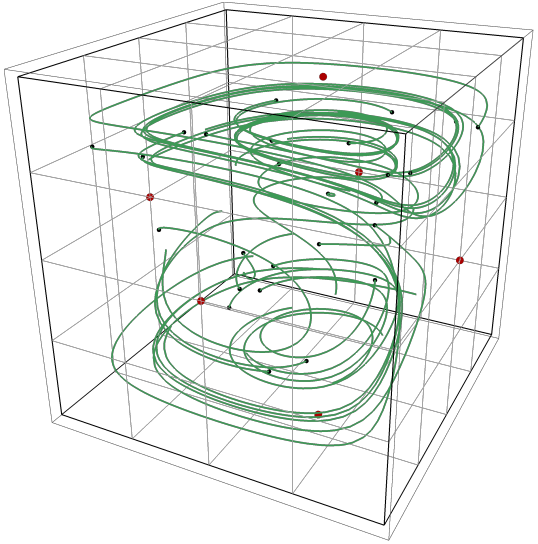
Phase portraits

Mean evolution follows the **replicator dynamics:** $\dot{x}_{ia} = x_{ia}\left[v_{ia}(x) - u_i(x)\right]$ [Taylor & Jonker, 1978]



Phase portraits

Mean evolution follows the **replicator dynamics:** $\dot{x}_{ia} = x_{ia}\left[v_{ia}(x) - u_i(x)\right]$ [Taylor & Jonker, 1978]



Rationality analysis

Are game-theoretic solution concepts consistent with the players' learning dynamics?

- Are Nash equilibria stationary? [Yes and no]
- Are they ***stable?*** [Not always]
- Are they ***attracting?*** [Not always]
- Do the dynamics always converge? [Not always]
- What other behaviors can we observe?
- ...

Notions of stability

Are all stationary points created equal?

Definition (Stochastic stability)

$x^* \in \mathcal{X}$ is ***stochastically stable*** under x_t if, for every confidence level $\delta > 0$ and every neighborhood $\mathcal{U}$ of x^*, there exists a neighborhood $\mathcal{U}_1$ of x^* such that

$$\mathbb{P}(x_t \in \mathcal{U} \text{ for all } t = 1, 2, \dots \mid x_1 \in \mathcal{U}_1) \geq 1 - \delta$$

[**Intuition:** with high probability, if x_t starts near x^*, it remains nearby]

Notions of stability

Are all stationary points created equal?

Definition (Stochastic stability)

$x^* \in \mathcal{X}$ is ***stochastically stable*** under x_t if, for every confidence level $\delta > 0$ and every neighborhood $\mathcal{U}$ of x^*, there exists a neighborhood $\mathcal{U}_1$ of x^* such that

$$\mathbb{P}(x_t \in \mathcal{U} \text{ for all } t = 1, 2, \dots \mid x_1 \in \mathcal{U}_1) \geq 1 - \delta$$

[**Intuition:** with high probability, if x_t starts near x^*, it remains nearby]

Definition (Stochastic asymptotic stability)

- $x^* \in \mathcal{X}$ is ***attracting*** if, for every confidence level $\delta > 0$, there exists a neighborhood $\mathcal{U}_1$ of x^* such that

$$\mathbb{P}(x_t \to x^* \text{ as } t \to \infty \mid x_1 \in \mathcal{U}_1) \geq 1 - \delta$$

- $x^* \in \mathcal{X}$ is ***stochastically asymptotically stable*** if it is stochastically stable and attracting.

[**Intuition:** with arbitrarily high probability, if x_t starts near x^* then, it remains nearby and eventually converges to x^*]

Towards a "folk theorem" for learning in games

Theorem

➥ ***Assume:*** *all players* (FTRL) *with step-size* γ_t *and oracle parameters* b_t *(bias) and* U_t *(noise) such that:*

(A1) $\gamma_t > 0$ *and* $\sum_t \gamma_t = \infty$

(A2) $b_t \to 0$

(A3) $\mathbb{E}[\|U_t\|^q] \leq \sigma_t^q$ *for some* $q > 2$

(A4) $\sum_{s=1}^t \gamma_s^{1+q/2} \sigma_s^q \big/ [\sum_{s=1}^t \gamma_s]^{1+\alpha q}$ *is summmable for some* $\alpha \in (0,1)$

Towards a "folk theorem" for learning in games

Theorem

➥ ***Assume:*** *all players* (FTRL) *with step-size* γ_t *and oracle parameters* b_t *(bias) and* U_t *(noise) such that:*

(A1) $\gamma_t = \gamma/t^p$ *for some* $p \in [0,1]$

(A2) $b_t = \mathcal{O}(1/t^b)$ *for some* $b > 0$

(A3) $\mathbb{E}[\|U_t\|^q] = \mathcal{O}(1/t^r)$ *for some* $q > 2, r < 1/2$

Towards a "folk theorem" for learning in games

Theorem

➥ ***Assume:*** *all players* (FTRL) *with step-size* γ_t *and oracle parameters* b_t *(bias) and* U_t *(noise) such that:*

(A1) $\gamma_t = \gamma/t^p$ *for some* $p \in [0,1]$

(A2) $b_t = \mathcal{O}(1/t^b)$ *for some* $b > 0$

(A3) $\mathbb{E}[\|U_t\|^q] = \mathcal{O}(1/t^r)$ *for some* $q > 2$, $r < 1/2$

✍ ***Then:*** *the sequence* x_t *generated by* (FTRL) *enjoys the following properties*

(P1) *If* x_t *converges, its limit is a Nash equilibrium* [M & Zhou, 2019]

(P2) *If* x^* *is stochastically stable, it is a Nash equilibrium* [Giannou et al., 2021a]

(P3) x^* *is stochastically asymptotically stable if and only if it is a strict Nash equilibrium* [Giannou et al., 2021b]

(P4) *If* $p > 1/2$ *and* Γ *is a congestion game, then* x_t *converges to a Nash equilibrium (a.s.)* [Cohen et al., 2017]

Towards a "folk theorem" for learning in games

Theorem

➥ ***Assume:*** *all players* (FTRL) *with step-size* γ_t *and oracle parameters* b_t *(bias) and* U_t *(noise) such that:*

(A1) $\gamma_t = \gamma/t^p$ *for some* $p \in [0,1]$

(A2) $b_t = \mathcal{O}(1/t^b)$ *for some* $b > 0$

(A3) $\mathbb{E}[\|U_t\|^q] = \mathcal{O}(1/t^r)$ *for some* $q > 2$, $r < 1/2$

✍ ***Then:*** *the sequence* x_t *generated by* (FTRL) *enjoys the following properties*

(P1) *If* x_t *converges, its limit is a Nash equilibrium* [M & Zhou, 2019]

(P2) *If* x^* *is stochastically stable, it is a Nash equilibrium* [Giannou et al., 2021a]

(P3) x^* *is stochastically asymptotically stable if and only if it is a strict Nash equilibrium* [Giannou et al., 2021b]

(P4) *If* $p > 1/2$ *and* Γ *is a congestion game, then* x_t *converges to a Nash equilibrium (a.s.)* [Cohen et al., 2017]

Notes:

✍ Learning analogue of the "folk theorem" of evolutionary game theory [Hofbauer & Sigmund, 2003]

✗ Converses to (P1) and (P2) do not hold!

Rate of convergence

Theorem (Giannou et al., 2021b)

➥ ***Assume:*** *all players* (FTRL) *with step-size* γ_t *and oracle parameters* b_t *(bias) and* U_t *(noise) such that:*

(A1) $\gamma_t = \gamma/t^p$ *for some* $p \in [0,1]$

(A2) $b_t = \mathcal{O}(1/t^b)$ *for some* $b > 0$

(A3) $\mathbb{E}[\|U_t\|^q] = \mathcal{O}(1/t^r)$ *for some* $q > 2$, $r < 1/2$

✍ ***Then:*** *if* x^* *is a strict Nash equilibrium and* x_t *converges to* x^*, *then*

$$\|x_t - x^*\|_1 \leq \sum_{a \notin \operatorname{supp}(x^*)} \phi\left(A - B\sum_{s=1}^{t}\gamma_s\right)$$

where

- $A, B > 0$ *are initialization- and game-dependent constants*
- *The* **rate function** ϕ *is defined as* $\phi = (\theta')^{-1}$ *where* θ *is the method's regularizer*
  - ***For exponential weights:*** $\phi(z) = \exp(z) \implies$ ***geometric convergence*** *in* $S_t = \sum_{s=1}^{t}\gamma_s$
  - ***For projection dynamics:*** $\phi(z) = [z]_+ \implies$ ***convergence in a finite number of iterations!***

Overview

I. Online learning

- ***Exogenous*** reward mechanism

✓ Efficient regret minimization via ***exponential weights*** $\rightsquigarrow \mathrm{Reg}(T) = \mathcal{O}(\sqrt{T})$

✗ Feedback plays a **crucial** role: performance decreases by a factor of $|\mathcal{A}|$ in the bandit case

✓ Possibility to exploit ***combinatorial structures*** [György et al., 2007; Cesa-Bianchi & Lugosi, 2012]

II. Learning in games

- ***Endogenous*** reward generation

✗ Regret minimization $\not\Longrightarrow$ convergence to Nash equilibrium

✓ Convergence in congestion games

✓ Robust convergence $\Longleftrightarrow$ Strict Nash equilibria

- Feedback less crucial but guarantees less explicit

Overview

I. Online learning

- ***Exogenous*** reward mechanism

✓ Efficient regret minimization via ***exponential weights*** $\rightsquigarrow \mathrm{Reg}(T) = \mathcal{O}(\sqrt{T})$

✗ Feedback plays a **crucial** role: performance decreases by a factor of $|\mathcal{A}|$ in the bandit case

✓ Possibility to exploit ***combinatorial structures*** [György et al., 2007; Cesa-Bianchi & Lugosi, 2012]

II. Learning in games

- ***Endogenous*** reward generation

✗ Regret minimization $\not\Longrightarrow$ convergence to Nash equilibrium

✓ Convergence in congestion games

✓ Robust convergence $\Longleftrightarrow$ Strict Nash equilibria

- Feedback less crucial but guarantees less explicit

Many open questions

- Other limit behaviors?
- Explicit rates?
- Robustness to delays / corruptions / ...

References I

Abernethy, J., Bartlett, P. L., Rakhlin, A., and Tewari, A. Optimal strategies and minimax lower bounds for online convex games. In *COLT '08: Proceedings of the 21st Annual Conference on Learning Theory*, 2008.

Audibert, J.-Y. and Bubeck, S. Regret bounds and minimax policies under partial monitoring. *Journal of Machine Learning Research*, 11: 2635-2686, 2010.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. Gambling in a rigged casino: The adversarial multi-armed bandit problem. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, 1995.

Auer, P., Cesa-Bianchi, N., and Fischer, P. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47:235-256, 2002a.

Auer, P., Cesa-Bianchi, N., Freund, Y., and Schapire, R. E. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1): 48-77, 2002b.

Blackwell, D. An analog of the minimax theorem for vector payoffs. *Pacific Journal of Mathematics*, 6:1-8, 1956.

Bubeck, S. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3-4):231-358, 2015.

Bubeck, S. and Cesa-Bianchi, N. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1-122, 2012.

Cesa-Bianchi, N. and Lugosi, G. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Cesa-Bianchi, N. and Lugosi, G. Combinatorial bandits. *Journal of Computer and System Sciences*, 78:1404-1422, 2012.

Cohen, J., Héliou, A., and Mertikopoulos, P. Learning with bandit feedback in potential games. In *NIPS '17: Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017.

References II

Cover, T. M. Behavior of sequential predictors of binary sequences. In *Transactions of the Fourth Prague Conference on Information Theory, Statistical Decision Functions, and Random Processes*, pp. 263-272, 1965.

Erev, I. and Roth, A. E. Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. *American Economic Review*, 88:848-881, 1998.

Fudenberg, D. and Levine, D. K. *The Theory of Learning in Games*, volume 2 of *Economic learning and social evolution*. MIT Press, Cambridge, MA, 1998.

Giannou, A., Vlatakis-Gkaragkounis, E. V., and Mertikopoulos, P. Survival of the strictest: Stable and unstable equilibria under regularized learning with partial information. In *COLT '21: Proceedings of the 34th Annual Conference on Learning Theory*, 2021a.

Giannou, A., Vlatakis-Gkaragkounis, E. V., and Mertikopoulos, P. The convergence rate of regularized learning in games: From bandits and uncertainty to optimism and beyond. In *NeurIPS '21: Proceedings of the 35th International Conference on Neural Information Processing Systems*, 2021b.

György, A., Linder, T., Lugosi, G., and Ottucsák, G. The online shortest path problem under partial monitoring. *Journal of Machine Learning Research*, 8:2369-2403, 2007.

Hannan, J. Approximation to Bayes risk in repeated play. In Dresher, M., Tucker, A. W., and Wolfe, P. (eds.), *Contributions to the Theory of Games, Volume III*, volume 39 of *Annals of Mathematics Studies*, pp. 97-139. Princeton University Press, Princeton, NJ, 1957.

Hart, S. and Mas-Colell, A. A simple adaptive procedure leading to correlated equilibrium. *Econometrica*, 68(5):1127-1150, September 2000.

Hart, S. and Mas-Colell, A. Uncoupled dynamics do not lead to Nash equilibrium. *American Economic Review*, 93(5):1830-1836, 2003.

Hofbauer, J. and Sigmund, K. Evolutionary game dynamics. *Bulletin of the American Mathematical Society*, 40(4):479-519, July 2003.

Lattimore, T. and Szepesvári, C. *Bandit Algorithms*. Cambridge University Press, Cambridge, UK, 2020.

References III

Littlestone, N. and Warmuth, M. K. The weighted majority algorithm. *Information and Computation*, 108(2):212–261, 1994.

Mertikopoulos, P. and Zhou, Z. Learning in games with continuous action sets and unknown payoff functions. *Mathematical Programming*, 173 (1-2):465–507, January 2019.

Nash, J. F. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences of the USA*, 36:48–49, 1950.

Nash, J. F. Non-cooperative games. *The Annals of Mathematics*, 54(2):286–295, September 1951.

Nesterov, Y. *Introductory Lectures on Convex Optimization: A Basic Course*. Number 87 in Applied Optimization. Kluwer Academic Publishers, 2004.

Shalev-Shwartz, S. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2011.

Shalev-Shwartz, S. and Singer, Y. Convex repeated games and Fenchel duality. In *NIPS' 06: Proceedings of the 19th Annual Conference on Neural Information Processing Systems*, pp. 1265–1272. MIT Press, 2006.

Shalev-Shwartz, S. and Singer, Y. A primal-dual perspective of online learning algorithms. *Journal of Machine Learning Research*, 69(2):115–142, 2007.

Taylor, P. D. and Jonker, L. B. Evolutionary stable strategies and game dynamics. *Mathematical Biosciences*, 40(1-2):145–156, 1978.

Viossat, Y. and Zapechelnyuk, A. No-regret dynamics and fictitious play. *Journal of Economic Theory*, 148(2):825–842, March 2013.

Vovk, V. G. Aggregating strategies. In *COLT '90: Proceedings of the 3rd Workshop on Computational Learning Theory*, pp. 371–383, 1990.