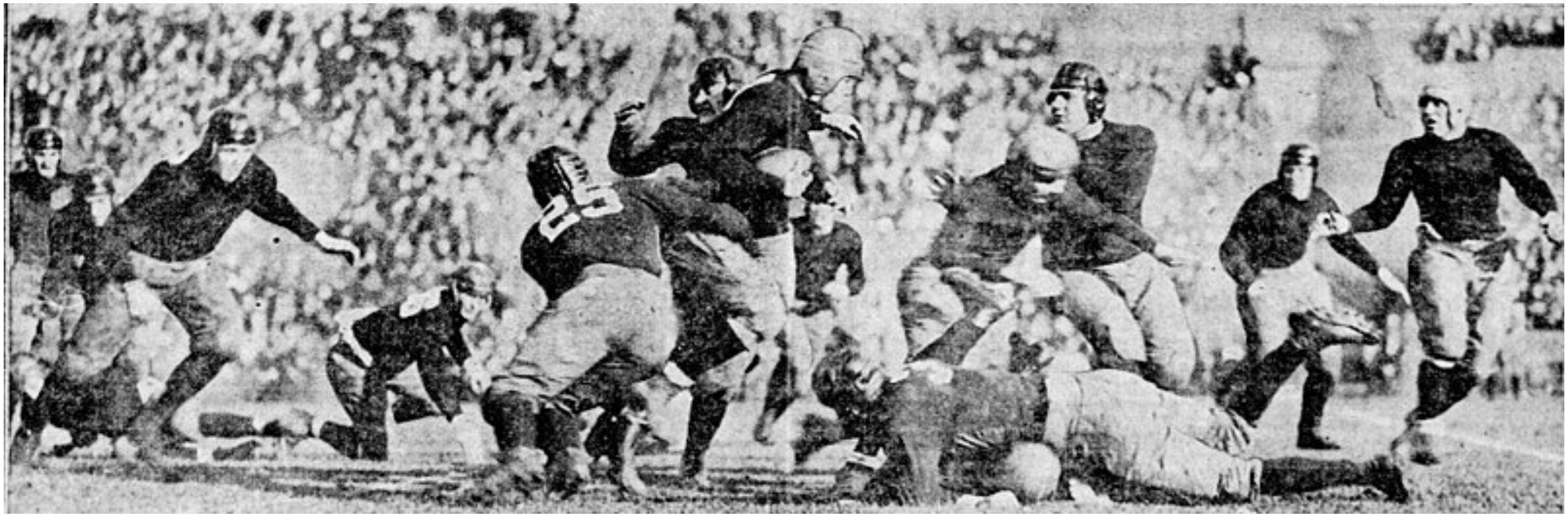
CS 440/ECE448 Lecture 32: Adversarial Learning & Mechanism Design

Mark Hasegawa-Johnson

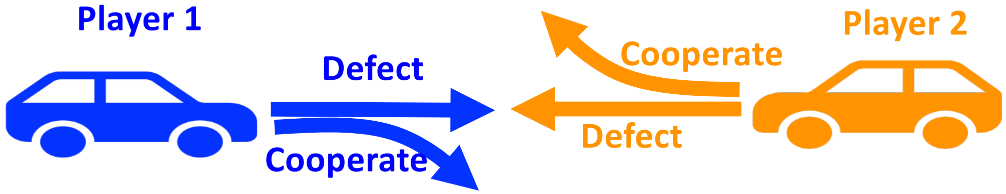


Public domain. https://commons.wikimedia.org/wiki/File:Indiana_In_Early_Fight,_But_Succumbs_To_Chicago,_November_10,_1923_(cropped).jpg

Outline

- Stability of a Nash equilibrium
- The Paparazzi game: Games with no stable equilibria
- Generative adversarial network
- Mechanism design: encourage desirable behavior
- Mechanism design: gather information

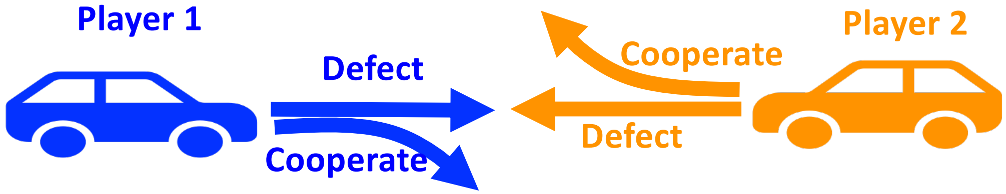
Review: Game of chicken



		Bob	
		Defect	Cooperate
Alice	Defect	-10, -10	2, -1
	Cooperate	-1, 2	1, 1

- Remember the game of chicken? It has three Nash equilibria
  - Alice defects, Bob cooperates. Rewards: $r_A = 2, r_B = -1$.
  - Alice cooperates, Bob defects. Rewards: $r_A = -1, r_B = 2$.
  - Each of them defects with probability $\frac{1}{10}$, independently at random. Rewards: $E[r_A] = E[r_B] = \frac{8}{10}$.
- The mixed equilibrium is unstable:
  - If either player changes their action probabilities just a little bit, then the other player's actions no longer have equal expected reward.

Notation for mixed strategies



		Bob	
		Defect	Cooperate
Alice	Defect	-10 -10	-1 2
	Cooperate	2 -1	1 1

- Let's say that each player chooses an action according to these probabilities:

$$\boldsymbol{p}_A = \begin{bmatrix} P(A=d) \\ P(A=c) \end{bmatrix} = \begin{bmatrix} \dfrac{1}{1+e^{z_A}} \\ \dfrac{1}{1+e^{-z_A}} \end{bmatrix}, \qquad \boldsymbol{p}_B = \begin{bmatrix} P(B=d) \\ P(B=c) \end{bmatrix} = \begin{bmatrix} \dfrac{1}{1+e^{z_B}} \\ \dfrac{1}{1+e^{-z_B}} \end{bmatrix}$$

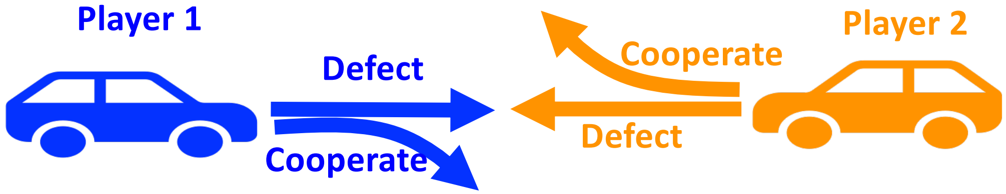
- …and gets these rewards as a result:

$$\boldsymbol{R}_A = \begin{bmatrix} -10 & -1 \\ 2 & 1 \end{bmatrix}, \qquad \boldsymbol{R}_B = \begin{bmatrix} -10 & 2 \\ -1 & 1 \end{bmatrix}$$

- Using this notation, the expected rewards are:

$$E[r_A] = \boldsymbol{p}_A^T \boldsymbol{R}_A \boldsymbol{p}_B, \qquad E[r_B] = \boldsymbol{p}_A^T \boldsymbol{R}_B \boldsymbol{p}_B,$$

Notation for mixed equilibrium



Bob \ Alice	Defect	Cooperate
Defect	Alice: -10, Bob: -10	Alice: -1, Bob: 2
Cooperate	Alice: 2, Bob: -1	Alice: 1, Bob: 1

- A mixed Nash equilibrium is a pair of strategies (z_A, z_B) such that neither player can improve their expected reward by unilaterally changing, strategy, i.e.,

$$\frac{\partial E[r_A]}{\partial z_A} \leq 0, \qquad \frac{\partial E[r_B]}{\partial z_B} \leq 0,$$

- For the game of chicken, this is true:

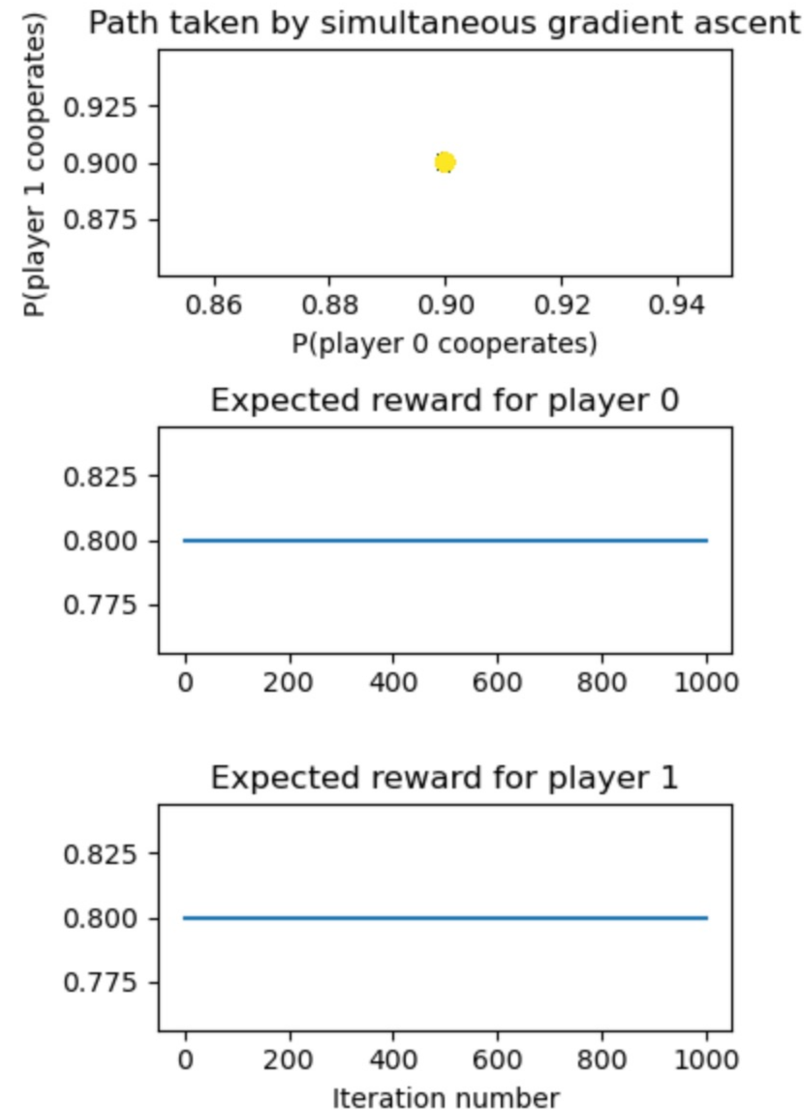
$$\frac{\partial E[r_B]}{\partial z_B} = \boldsymbol{p}_A^T \boldsymbol{R}_B \frac{\partial \boldsymbol{p}_B}{\partial z_B} = \left[\frac{1}{10}, \frac{9}{10}\right] \begin{bmatrix} -10 & -1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} -\frac{9}{100} \\ \frac{9}{100} \end{bmatrix} = 0$$

Simultaneous gradient ascent

We can prove that a mixed equilibrium is really an "equilibrium" by using simultaneous gradient ascent:

$$z_A = z_A + \eta \frac{\partial E[r_A]}{\partial z_A}$$
$$z_B = z_B + \eta \frac{\partial E[r_B]}{\partial z_B}$$

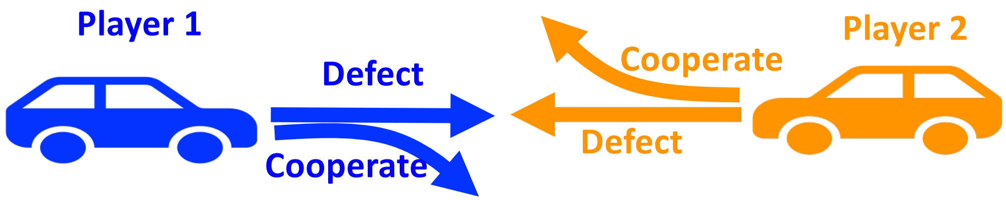
- An equilibrium is a pair of strategies (z_A, z_B) that neither player has any reason to change.
- When we perform gradient ascent, we find the neither player changes!



Try the quiz!

Try the quiz!

What does it mean to be unstable?



Bob \ Alice	Defect	Cooperate
Defect	-10 (Bob), -10 (Alice)	2 (Bob), -1 (Alice)
Cooperate	-1 (Bob), 2 (Alice)	1 (Bob), 1 (Alice)

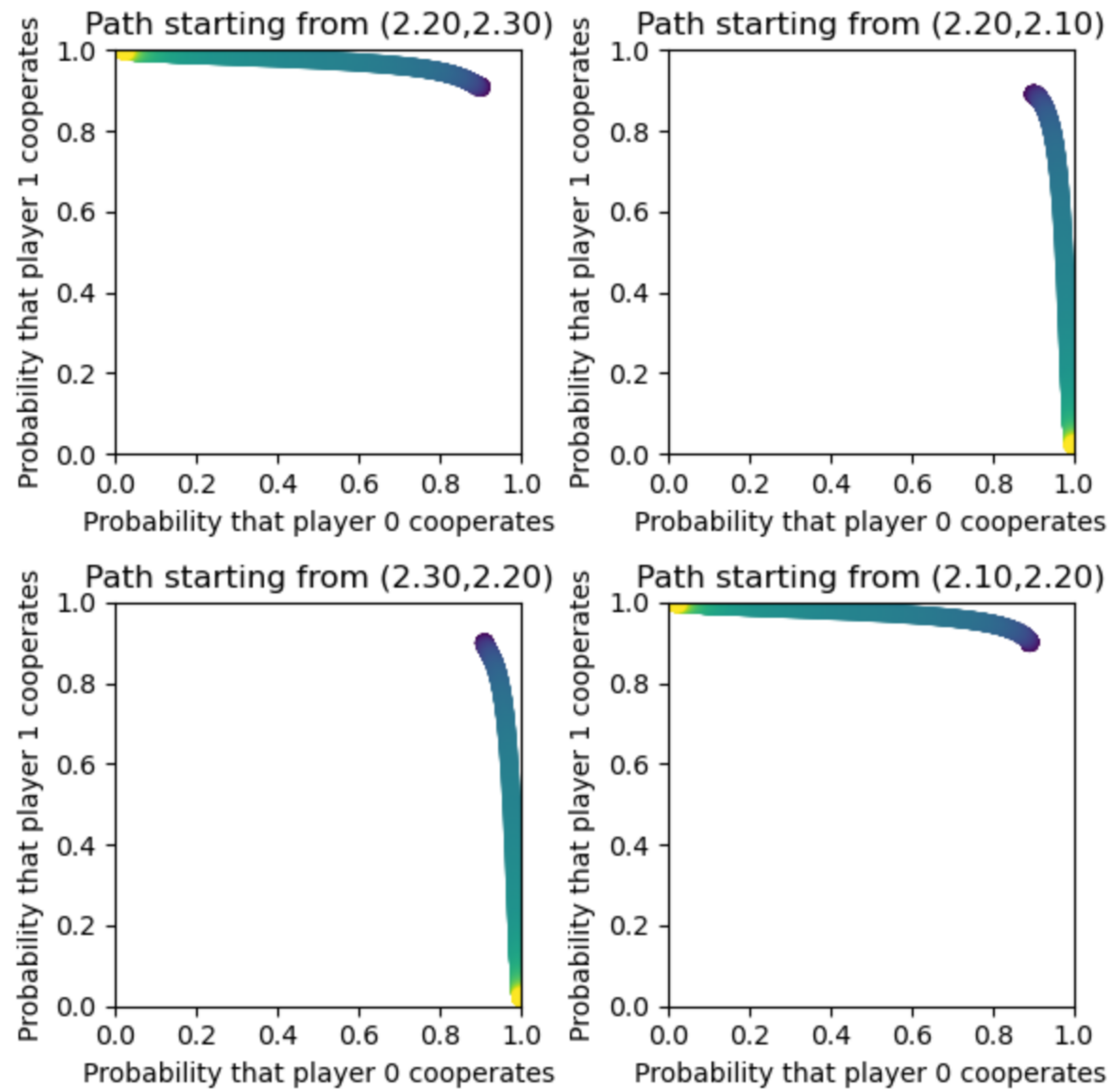
An equilibrium is unstable if a small change in the equilibrium results in a situation that causes the players to move even farther away from equilibrium. For example, suppose that Alice decides to cooperate less often, $P(A=c)=\frac{8}{10}$ instead of $\frac{9}{10}$. Then

$$\frac{\partial E[r_B]}{\partial z_B} = \boldsymbol{p}_A^T \boldsymbol{R}_B \frac{\partial \boldsymbol{p}_B}{\partial z_B} = \left[\frac{2}{10}, \frac{8}{10}\right] \begin{bmatrix} -10 & -1 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} -\frac{9}{100} \\ \frac{9}{100} \end{bmatrix} = +\frac{18}{1000}$$

Since $\frac{\partial E[r_B]}{\partial z_B}$ is positive, it is rational for Bob to increase $P(B=c)$. In response, Alice further decreases $P(A=c)$, until eventually $P(B=c)=1$ and $P(A=c)=0$.

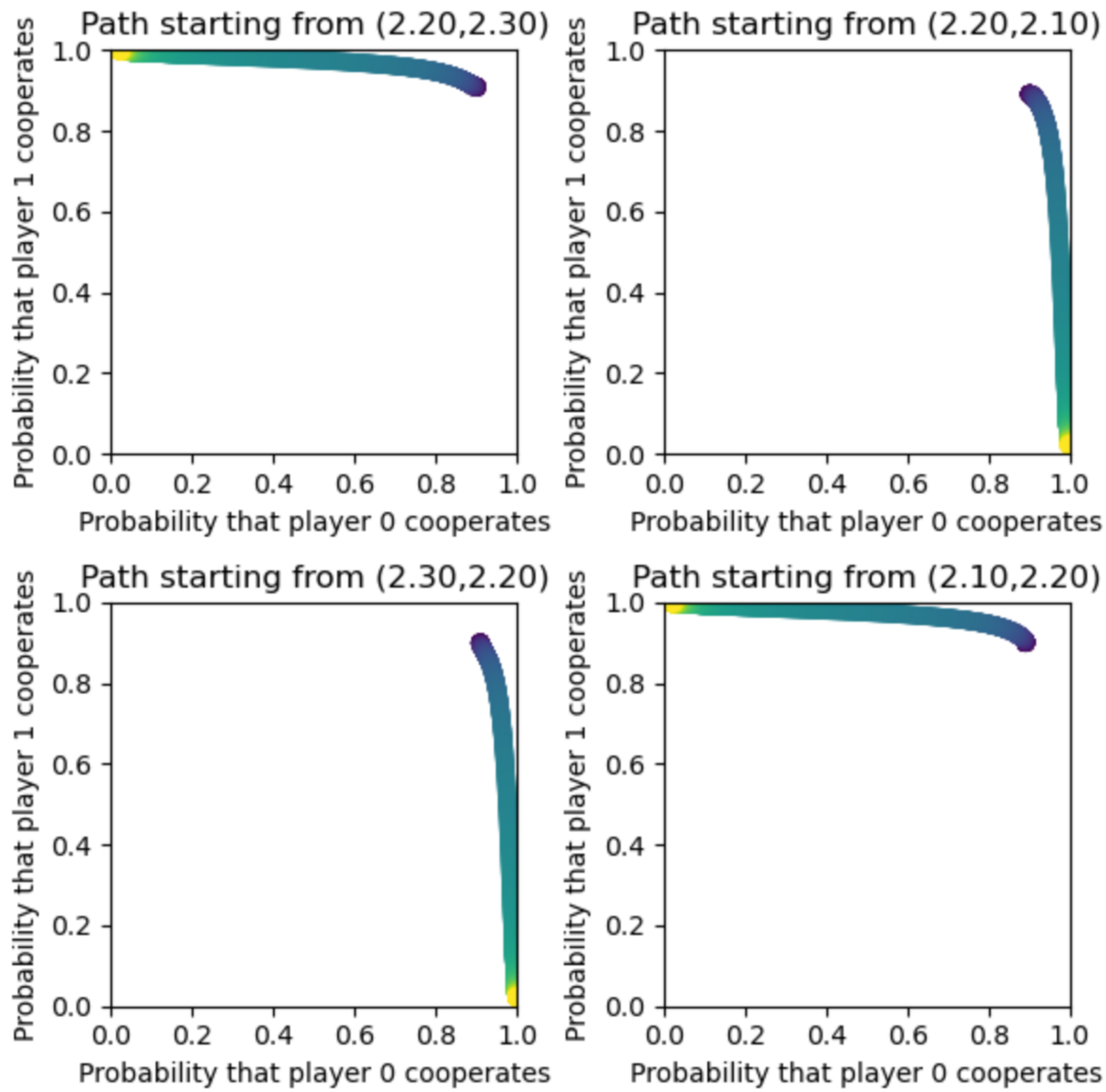
Testing stability using simultaneous gradient ascent

If a Nash equilibrium is unstable, it will not be reached by simultaneous gradient ascent starting from any nearby starting point.



Why is it unstable?

- If Alice cooperates with probability even slightly more than 0.9, then Bob gets better reward by always defecting -> converge to the (C,D) equilibrium.
- If Alice cooperates with probability even slightly less than 0.9, then Bob gets better reward by always cooperating -> converge to the (D,C) equilibrium.



Outline

- Stability of a Nash equilibrium
- The Paparazzi game: Games with no stable equilibria
- Generative adversarial network
- Mechanism design: encourage desirable behavior
- Mechanism design: gather information

The Paparazzi game

- Alice is a famous movie star. Her agent announces that she will be at Illini Union signing autographs all day, but secretly, she might go Grainger to get some work done.
- Bob is paparazzi. His job is to get Alice's photograph.
- If Alice and Bob are in the same location, Alice loses (-1), Bob wins (+1)
- If they are in different locations, Alice wins (+1), Bob loses (-1)

Alice \ Bob	Defect	Cooperate
Defect	-1, 1	1, -1
Cooperate	1, -1	-1, 1

The Paparazzi game

- Alice's strategy is

$$\boldsymbol{p}_A = \begin{bmatrix} P(A = d) \\ P(A = c) \end{bmatrix} = \begin{bmatrix} 1 - \sigma(z_A) \\ \sigma(z_A) \end{bmatrix}$$

- Bob's strategy is

$$\boldsymbol{p}_B = \begin{bmatrix} P(B = d) \\ P(B = c) \end{bmatrix} = \begin{bmatrix} 1 - \sigma(z_B) \\ \sigma(z_B) \end{bmatrix}$$

Alice \ Bob	Defect	Cooperate
Defect	-1 / 1	1 / -1
Cooperate	1 / -1	-1 / 1

The Paparazzi game

Alice \ Bob	Defect	Cooperate
Defect	-1 (Alice), 1 (Bob)	1 (Alice), -1 (Bob)
Cooperate	1 (Alice), -1 (Bob)	-1 (Alice), 1 (Bob)

- Alice's strategy is $\boldsymbol{p}_A = \begin{bmatrix} 1-\sigma(z_A) \\ \sigma(z_A) \end{bmatrix}$

- Bob's strategy is $\boldsymbol{p}_B = \begin{bmatrix} 1-\sigma(z_B) \\ \sigma(z_B) \end{bmatrix}$

- Alice's expected reward is

$$E[r_A] = \boldsymbol{p}_A^T \boldsymbol{R}_A \boldsymbol{p}_B = [1-\sigma(z_A), \sigma(z_A)] \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1-\sigma(z_B) \\ \sigma(z_B) \end{bmatrix}$$

- Bob's expected reward is

$$E[r_B] = \boldsymbol{p}_A^T \boldsymbol{R}_B \boldsymbol{p}_B = [1-\sigma(z_A), \sigma(z_A)] \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1-\sigma(z_B) \\ \sigma(z_B) \end{bmatrix}$$

The Nash Equilibrium

		Bob: Defect	Bob: Cooperate
Alice	Defect	-1, 1	1, -1
Alice	Cooperate	1, -1	-1, 1

- Alice's expected reward is

$$E[r_A] = \boldsymbol{p}_A^T \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \boldsymbol{p}_B$$

Bob's expected reward is

$$E[r_B] = \boldsymbol{p}_A^T \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \boldsymbol{p}_B$$

- The Nash equilibrium is:

$$\boldsymbol{p}_A = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}, \qquad \boldsymbol{p}_B = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$$

…you can verify that this is a Nash equilibrium by noticing that

- If $\boldsymbol{p}_A = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$, then Bob has no preference between cooperating and defecting, so he can choose at random.
- If $\boldsymbol{p}_B = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}$, then Alice has no preference, and can choose at random.

Simultaneous gradient ascent

- Suppose both Alice and Bob are using mixed strategies:

$$\boldsymbol{p}_A = \begin{bmatrix} 1-\sigma(z_A) \\ \sigma(z_A) \end{bmatrix}, \qquad \boldsymbol{p}_B = \begin{bmatrix} 1-\sigma(z_B) \\ \sigma(z_B) \end{bmatrix}$$

- On successive days, they each try to improve their strategies using gradient ascent:

$$\begin{bmatrix} z_A \\ z_B \end{bmatrix} \leftarrow \begin{bmatrix} z_A \\ z_B \end{bmatrix} + \eta \begin{bmatrix} \frac{\partial E[r_A]}{\partial z_A} \\ \frac{\partial E[r_B]}{\partial z_B} \end{bmatrix} = \begin{bmatrix} z_A \\ z_B \end{bmatrix} + \eta \begin{bmatrix} \left(\frac{\partial \boldsymbol{p}_A}{\partial z_A}\right)^T \boldsymbol{R}_A \boldsymbol{p}_B \\ \boldsymbol{p}_A^T \boldsymbol{R}_B \frac{\partial \boldsymbol{p}_B}{\partial z_B} \end{bmatrix}$$

- If you start at exactly the equilibrium, $\boldsymbol{p}_A^T = \left[\frac{1}{2}, \frac{1}{2}\right]$, $\boldsymbol{p}_B^T = \left[\frac{1}{2}, \frac{1}{2}\right]$, then gradient ascent will stay there. But this is an unstable equilibrium…

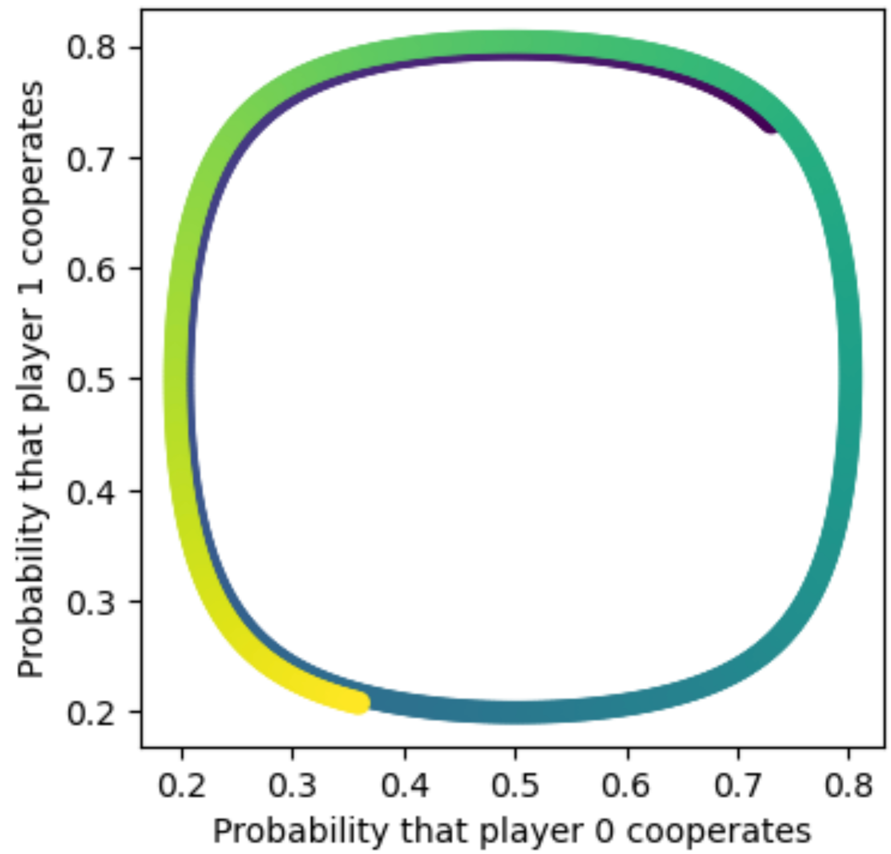
Simultaneous gradient ascent

- Surprisingly, simultaneous gradient ascent fails.
- The graph at right is the sequence of vectors

$$\begin{bmatrix} P(A=c) \\ P(B=c) \end{bmatrix} = \begin{bmatrix} 1/(1+e^{-z_A}) \\ 1/(1+e^{-z_B}) \end{bmatrix}$$

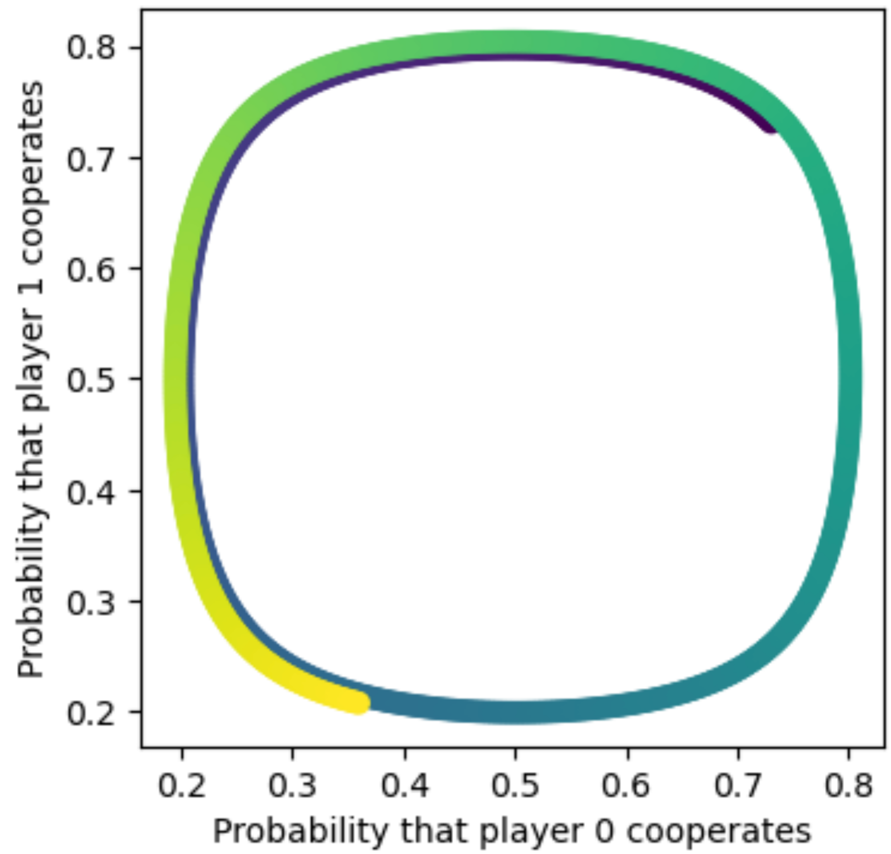
…obtained using

$$\begin{bmatrix} z_A \\ z_B \end{bmatrix} \leftarrow \begin{bmatrix} z_A \\ z_B \end{bmatrix} + \eta \begin{bmatrix} \frac{\partial E[r_A]}{\partial z_A} \\ \frac{\partial E[r_B]}{\partial z_B} \end{bmatrix}$$



Simultaneous gradient ascent

- Why does it never converge?
- If Alice and Bob are in the same location, then Alice goes elsewhere
- If Alice and Bob are in different locations, then Bob follows Alice
- … and so on, forever.



Wait--- Doesn't gradient ascent converge?

- Yes. Gradient ascent always converges. But gradient ascent means that both z_A and z_B are chasing after the SAME goal. For example, if they're both trying to improve Alice's day, then the result would converge:

$$\begin{bmatrix} z_A \\ z_B \end{bmatrix} \leftarrow \begin{bmatrix} z_A \\ z_B \end{bmatrix} + \eta \begin{bmatrix} \frac{\partial E[r_A]}{\partial z_A} \\ \frac{\partial E[r_A]}{\partial z_B} \end{bmatrix}$$

- …but if Alice is trying to improve her day, and Bob is trying to improve HIS day, then it might never converge:

$$\begin{bmatrix} z_A \\ z_B \end{bmatrix} \leftarrow \begin{bmatrix} z_A \\ z_B \end{bmatrix} + \eta \begin{bmatrix} \frac{\partial E[r_A]}{\partial z_A} \\ \frac{\partial E[r_B]}{\partial z_B} \end{bmatrix}$$

Outline

- Stability of a Nash equilibrium
- The Paparazzi game: Games with no stable equilibria
- Generative adversarial network
- Mechanism design: encourage desirable behavior
- Mechanism design: gather information

Unsupervised learning

Given $\mathcal{D} = \{x_1, \ldots, x_n\}$, learn G so that $P(G = x) \approx P(X = x)$.

- Maximum likelihood: unseen cases have probability zero

$$P(G = x) = \frac{\text{\# times } x \text{ occurs in } \mathcal{D}}{n}$$

- Laplace smoothing: unseen cases all have the same probability

$$P(G) = \frac{k + \text{\# times } x \text{ occurs in } \mathcal{D}}{\sum_{x \in \mathcal{X}} (k + \text{\# times } x \text{ occurs in } \mathcal{D})}$$

Unsupervised learning

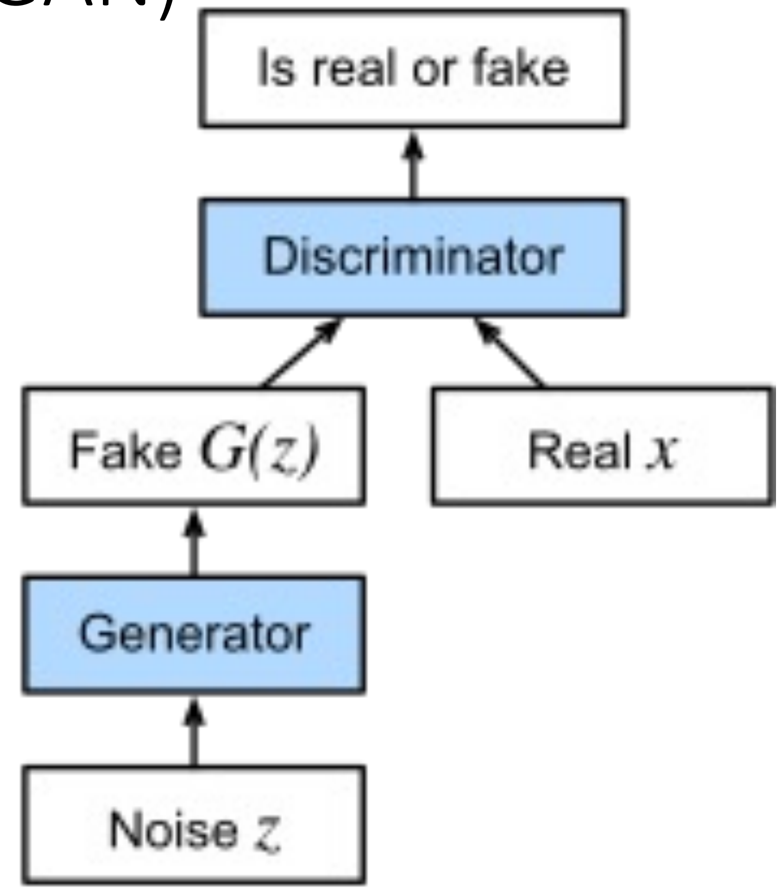
Neither maximum likelihood nor Laplace smoothing is very good for complex random variables. For example, suppose $\mathcal{X}$ is the set of all face images, and we want to train a neural network G so that $P(G = x) \approx P(X = x)$. We would prefer a network to generate images like the one on left, not the one on right:

https://commons.wikimedia.org/wiki/File:Outdoors-man-portrait_(cropped).jpg

https://en.wikipedia.org/wiki/File:Pablo_Picasso,_1910,_Woman_with_Mustard_Pot_(La_Femme_au_pot_de_moutarde),_oil_on_canvas,_73_x_60_cm,_Gemeentemuseum,_The_Hague._Exhibited_at_the_Armory_Show,_New_York,_Chicago,_Boston_1913.jpg

Generative adversarial network (GAN)

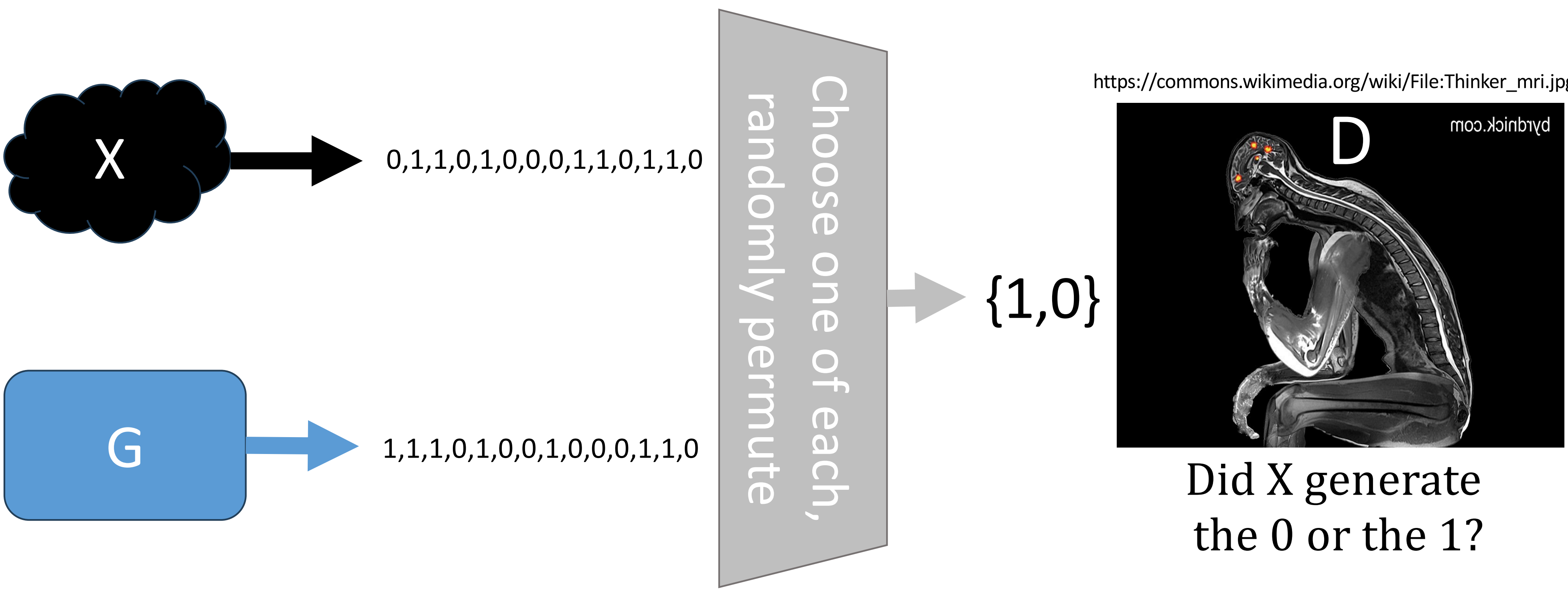
- A generative adversarial network is composed of two networks, a generator (G) and a discriminator (D)
- The generator is trained so that $P(G = x) \approx P(X = x)$, where X is some type of data in the real world
- The discriminator tries to tell the difference between G and X
- If the discriminator can tell the difference, then the discriminator wins
- If the discriminator can't tell the difference, then the generator wins



https://commons.wikimedia.org/wiki/File:Generative_adversarial_network.svg

GAN as a game

- X is a random bit
- G must generate one bit without seeing X
- D gets to see X and G, and needs to decide which one is X



GAN as a game

- If X and G are the same, all rewards are zero.
- If X and G differ, and D can tell which one is X, then D gets rewarded, G gets penalized.
- If X and G differ, and D is incorrect, then D gets penalized, and G gets rewarded.

X=0 (Discriminator \ Generator)	G=0	G=1
D=0	0 / 0	1 / -1
D=1	0 / 0	-1 / 1

X=1 (Discriminator \ Generator)	G=0	G=1
D=0	-1 / 1	0 / 0
D=1	1 / -1	0 / 0

Outcome probabilities

Suppose, independent of one another,

- $X = 1$ with probability P_X
- $G = 1$ with probability P_G

The rewards are all based on the difference between the probabilities of these two rectangles:

$$P(X = 0, G = 1) - P(X = 1, G = 0)$$
$$= P_G(1 - P_X) - P_X(1 - P_G)$$
$$= P_G - P_X$$

X=0

Discriminator / Generator	G=0	G=1
D=0	0 / 0	1 / -1
D=1	0 / 0	-1 / 1

X=1

Discriminator / Generator	G=0	G=1
D=0	-1 / 1	0 / 0
D=1	1 / -1	0 / 0

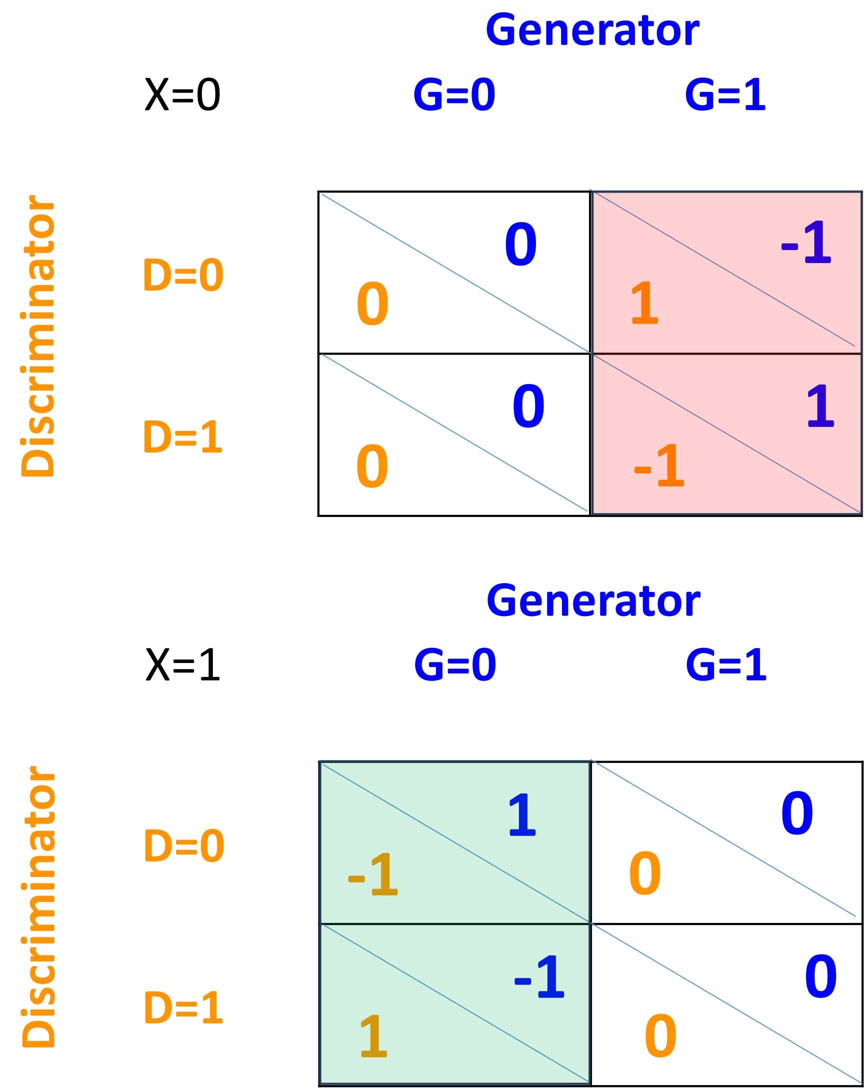
Expected rewards

If the discriminator chooses to say that $D = 0$ is the truth, the expected rewards are

$$E[r_D(X, G, 0)] = P_G - P_X$$
$$E[r_G(X, G, 0)] = -(P_G - P_X)$$

If the discriminator chooses to say that $D = 1$ is the truth, the expected rewards are

$$E[r_D(X, G, 1)] = P_X - P_G$$
$$E[r_G(X, G, 1)] = -(P_X - P_G)$$



Rational behavior

The **discriminator** should maximize its expected reward, so it should always choose:

- Always choose $D = 0$ if $P_G > P_X$
- Always choose $D = 1$ if $P_G < P_X$
- Choose with 50/50 probability if $P_G = P_X$

The **generator** should maximize its expected reward, so it should choose:

- Always generate $G = 0$ if $P(D = 0) > 0.5$
- Always generate $G = 1$ if $P(D = 1) > 0.5$
- Generate with exactly $P_G = P_X$ if $P(D = 1) = 0.5$

Generator

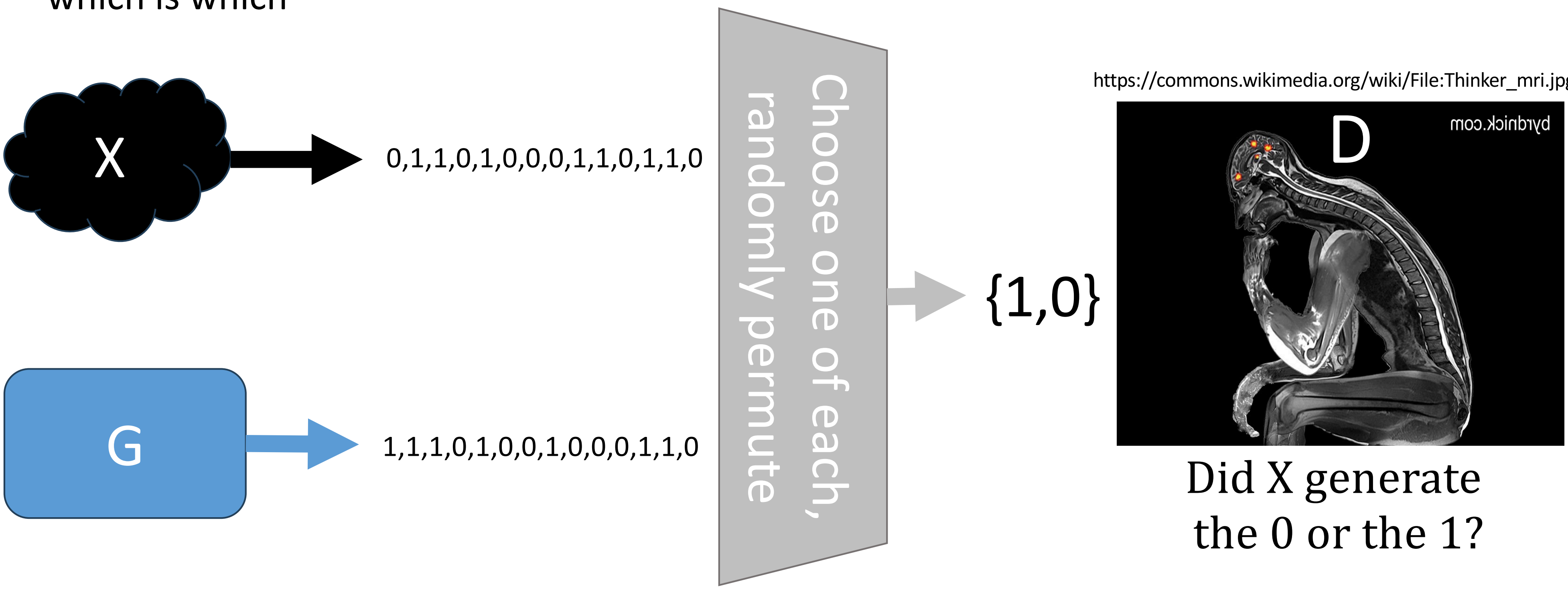
X=0 (Discriminator)	G=0	G=1
D=0	0 / 0	1 / -1
D=1	0 / 0	-1 / 1

Generator

X=1 (Discriminator)	G=0	G=1
D=0	-1 / 1	0 / 0
D=1	1 / -1	0 / 0

Nash equilibrium: Given the other player's behavior, neither player has a reason to change their strategy.

- The generator tries to match the data distribution as exactly as possible
- The discriminator has no choice but to choose uniformly at random, since it doesn't know which is which



GAN: Unstable Nash equilibrium

- Notice that gains for the GAN are asymmetric:
  - Whenever the generator wins, the discriminator loses
  - Whenever the discriminator wins, the generator loses
- For this reason, the equilibrium is unstable, just like the paparazzi game! --- GANs can be very hard to train
- Some possibilities:
  - Force the players to alternate their updates so it becomes a minimax game like chess or go (this makes convergence weird, but methods exist)
  - Add extra terms to the loss functions to help convergence (one method, called “symplectic loss,” is modeled after the dynamics of a decaying satellite orbit)

Outline

- Stability of a Nash equilibrium
- The Paparazzi game: Games with no stable equilibria
- Generative adversarial network
- Mechanism design: encourage desirable behavior
- Mechanism design: gather information

Mechanism design

- Using game theory, we can predict how rational agents will behave
- Suppose we want them to behave in a particular way
- Can we change the rules of the game to get the desired behavior?

Example: Mixed equilibrium

- Suppose we want to Alice and Bob to choose actions with action probabilities given by the vectors $\boldsymbol{p}_A, \boldsymbol{p}_B$.
- Suppose the reward matrices are initialized to

$$\boldsymbol{Q}_A = \begin{bmatrix} q_{A00} & q_{A01} \\ q_{A10} & q_{A11} \end{bmatrix}, \boldsymbol{Q}_B = \begin{bmatrix} q_{B00} & q_{B01} \\ q_{B10} & q_{B11} \end{bmatrix}$$

- Suppose we want to change $\boldsymbol{Q}_A$, $\boldsymbol{Q}_B$ to some new set of reward matrices $\boldsymbol{R}_A$, $\boldsymbol{R}_B$ so that $(\boldsymbol{p}_A, \boldsymbol{p}_B)$ is a Nash equilibrium.
- What is the smallest modification that will make $(\boldsymbol{p}_A, \boldsymbol{p}_B)$ a Nash equilibrium?

		Bob	
		Defect	Cooperate
Alice	Defect	q_{A00} / q_{B00}	q_{A01} / q_{B01}
	Cooperate	q_{A10} / q_{B10}	q_{A11} / q_{B10}

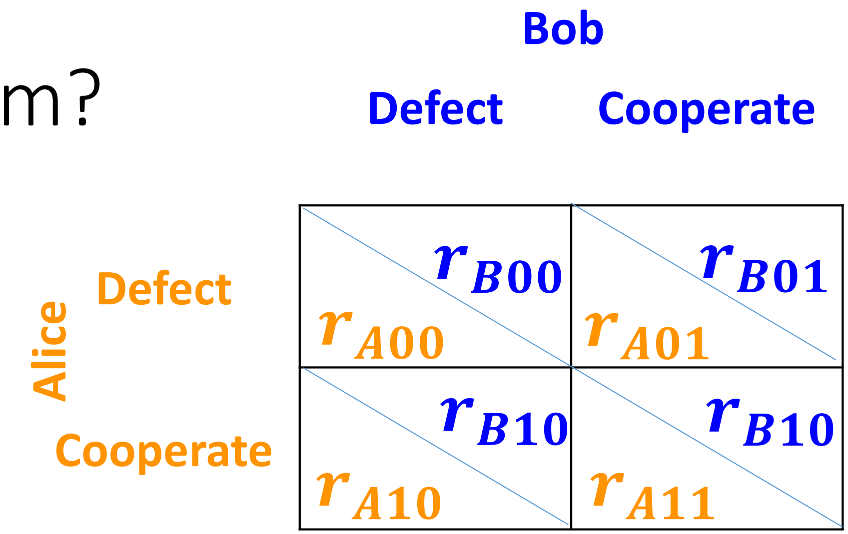
How do we know if it’s equilibrium?

- (p_A, p_B) is a Nash equilibrium if

$$\boldsymbol{p}_A^T \boldsymbol{R}_B \begin{bmatrix} -1 \\ 1 \end{bmatrix} = 0$$
$$[-1,1]\boldsymbol{R}_A \boldsymbol{p}_B = 0$$

- We want to choose $\boldsymbol{R}_A$, $\boldsymbol{R}_B$ that are close to $\boldsymbol{Q}_A$, $\boldsymbol{Q}_B$, but that make those equations true.

- How can we do that?



Solution using gradient descent

One way we can solve this problem is by starting with $\boldsymbol{R}_A = \boldsymbol{Q}_A, \boldsymbol{R}_B = \boldsymbol{Q}_B$, and then gradually improving the fit to the desired Nash equilibrium.

Define the loss to be:

$$\mathcal{L} = \frac{1}{2}\left(\boldsymbol{p}_A^T \boldsymbol{R}_B \begin{bmatrix} -1 \\ 1 \end{bmatrix}\right)^2 + \frac{1}{2}\left([-1,1]\boldsymbol{R}_A \boldsymbol{p}_B\right)^2$$

Learn $\boldsymbol{R}_A$ and $\boldsymbol{R}_B$ using gradient descent with some step size η:

$$\boldsymbol{R}_A \leftarrow \boldsymbol{R}_A - \eta \frac{\partial \mathcal{L}}{\partial \boldsymbol{R}_A}$$

$$\boldsymbol{R}_B \leftarrow \boldsymbol{R}_B - \eta \frac{\partial \mathcal{L}}{\partial \boldsymbol{R}_B}$$

… until we reach $\mathcal{L} = 0$.

Other types of mechanism design

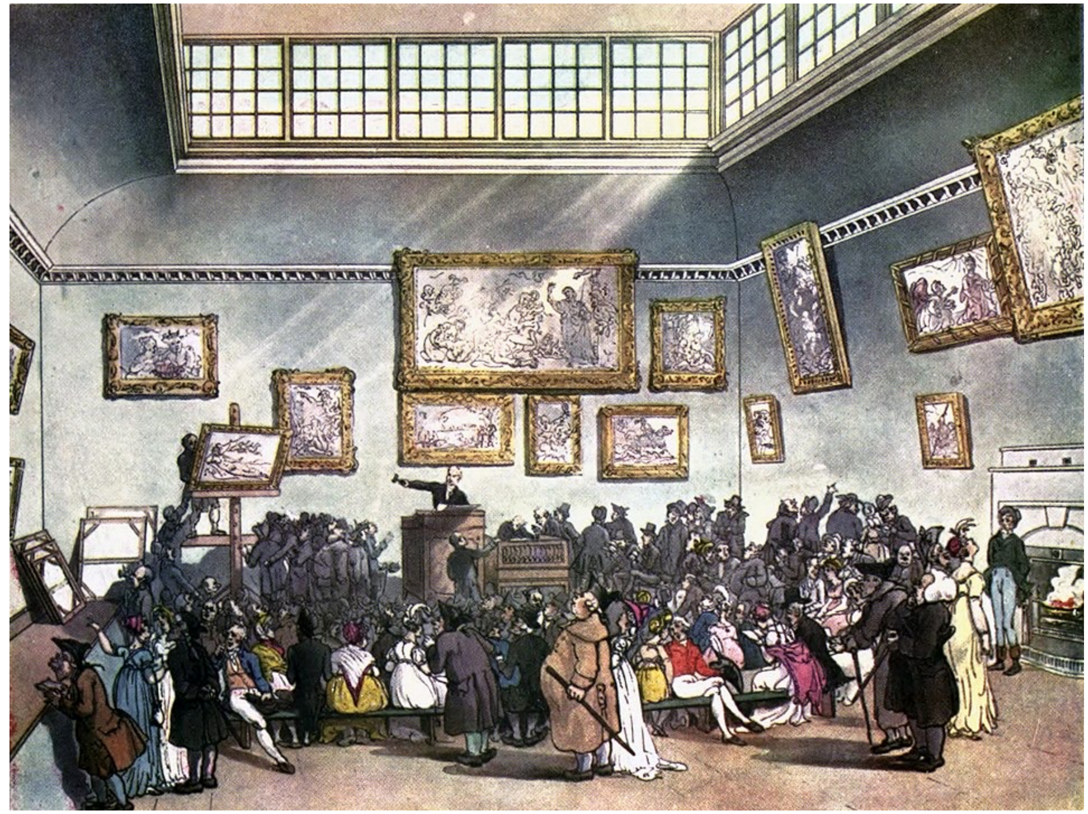
- The "prisoner's dilemma" was designed by the police so that "always defect" is the dominant strategy for both players. Can we design, instead, a strategy so that "always cooperate" is the dominant strategy?
- In the game of chicken, the mixed Nash equilibrium has a positive expected reward for both players, but is hard to achieve in practice, because it is unstable. Can we $\boldsymbol{R}_A$ and $\boldsymbol{R}_B$ to make the mixed equilibrium stable?

Outline

- Stability of a Nash equilibrium
- The Paparazzi game: Games with no stable equilibria
- Generative adversarial network
- Mechanism design: encourage desirable behavior
- Mechanism design: gather information

The auction game

- The object being auctioned is worth v_i to the i^{th} player
- The i^{th} player offers to pay b_i for the item (”bid”)
- If the i^{th} player’s bid is accepted, they can make an amount of money equal to $r_i(\text{win}) = v_i - b_i$
- If not, the amount of money they make is $r_i(\text{lose}) = 0$



https://commons.wikimedia.org/wiki/File:Microcosm_of_London_Plate_006_-_Auction_Room,_Christie%27s_(colour).jpg

Nash equilibrium of a classic auction

- Suppose there are only two players. Player 1's expected reward is

$$E[r_1] = P(b_1 > b_0)r_1(\text{win}) + P(b_1 \le b_0)r_1(\text{lose})$$
$$= P(b_1 > b_0)(v_1 - b_1)$$

- The rational bid is

$$b_1^* = \underset{b_1}{\operatorname{argmax}}\, P(b_1 > b_0)(v_1 - b_1)$$

…which depends on the probability distribution $P(b_0)$, but is always $b_1^* < v_1$.

Recurring auction: Knowledge is worth more than money

- Some resources (oil, advertising) are sold by the same organization once per day (or once per minute)
- The auctioneer wants to know how much the resource is worth
- …and is willing to sacrifice a little revenue to find out

https://commons.wikimedia.org/wiki/File:The_Ladies%27_home_journal_(1948)_(14785694143).jpg

Vickrey auction (second-price auction)

- Player 1 wins the auction if $b_1 > b_0$, but ***only pays the auctioneer $\boldsymbol{b_0}$ dollars, not $\boldsymbol{b_1}$***. Their expected reward is therefore

$$E[r_1] = P(b_1 > b_0) r_A(\text{win}) + P(b_1 \le b_0) r_1(\text{lose})$$
$$= P(b_1 > b_0)(v_1 - b_0)$$

- Their rational bid is

$$b_1^* = \underset{b_1}{\operatorname{argmax}}\, P(b_1 > b_0)(v_1 - b_0)$$

…which should be larger than b_0 whenever $v_1 > b_0$, but smaller than b_0 whenever $v_1 < b_0$. In other words,

$$b_1^* = v_1$$

- Auctioneer learns each player's true valuation of the resource.

Summary

- Simultaneous gradient ascent:

$$\begin{bmatrix} z_A \\ z_B \end{bmatrix} \leftarrow \begin{bmatrix} z_A \\ z_B \end{bmatrix} + \eta \begin{bmatrix} \dfrac{\partial E[r_A]}{\partial z_A} \\ \dfrac{\partial E[r_B]}{\partial z_B} \end{bmatrix}$$

- Nash equilibrium: $\frac{\partial E[r_A]}{\partial z_A} = \frac{\partial E[r_B]}{\partial z_B} = 0$
- Every game has a Nash equilibrium, but not every game has a stable Nash equilibrium!
- Mechanism design: Adjust $\boldsymbol{R}_A$ and $\boldsymbol{R}_B$ to get desired player behavior