Learning, Society & Causality

An Excursion in Theory Research

Nirjhar Das
MTech (Res.), CSA
Advisor: Prof Siddharth Barman

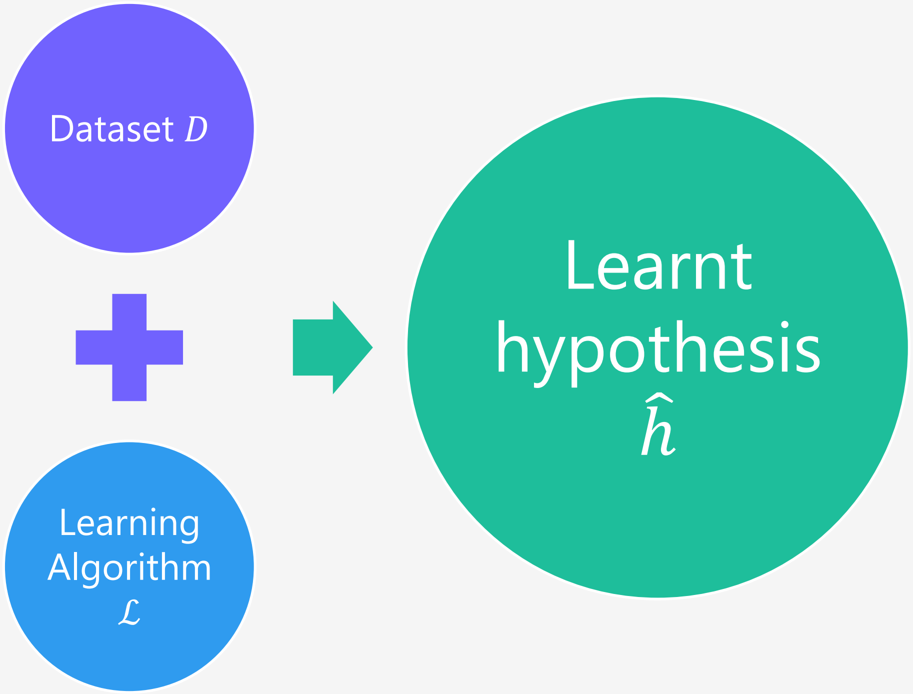
Contents

Learning Theory

Learning Theory: Supervised Learning

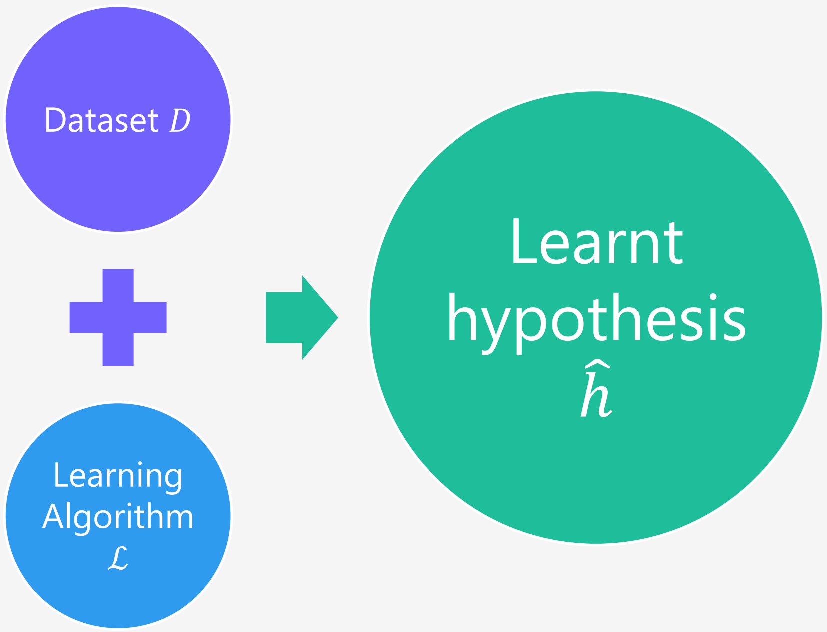
- Hypothesis class $\mathcal{H}$
- Hypothesis $h \in \mathcal{H},\ h: \mathcal{X} \rightarrow \mathcal{Y}$
- *True* hypothesis $h^* \in \mathcal{H}$
- Dataset $D = \{(X_i, Y_i)\}_{i=1}^{n}$, $X_i \sim \mathcal{D}$, iid, $Y_i = h^*(X_i)$
- Learning algorithm $\mathcal{L}: D \rightarrow \mathcal{H}$, let its output be $\hat{h}$
- **Question:** How *close* can we $\hat{h}$ get to h^*, given

  that number of samples is finite?



Learning Theory: Supervised Learning

Question: How *close* can we $\hat{h}$ get to h^*, given that number of samples is finite?

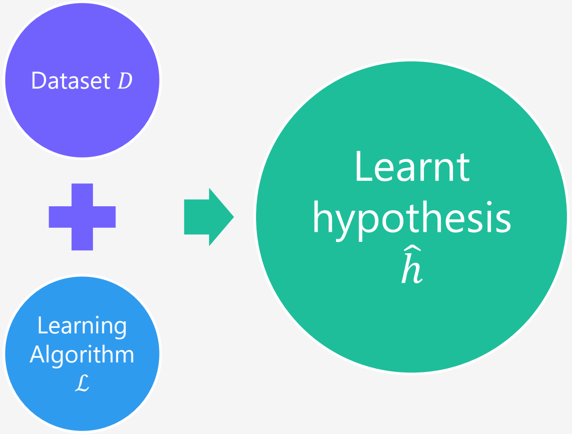


Why should we study this question?

- Characterizes the relation between complexity of the hypothesis class and the number of samples
- Has guided many practical choices in deep learning
- Gives provable guarantees

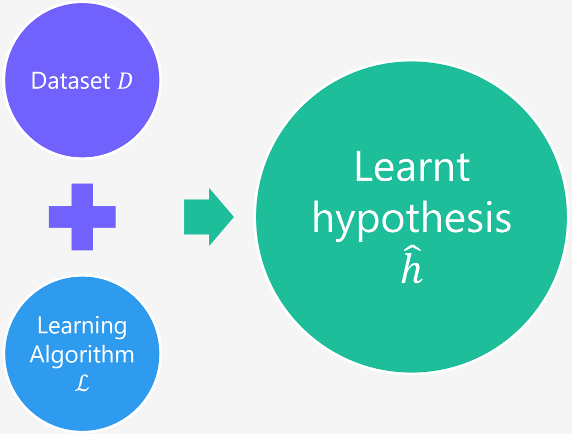
Learning Theory: Supervised Learning

Question: How *close* can we $\hat{h}$ get to h^*, given that number of samples is finite?



Learning Theory: Supervised Learning

Question: How *close* can we $\hat{h}$ get to h^*, given that number of samples is finite?

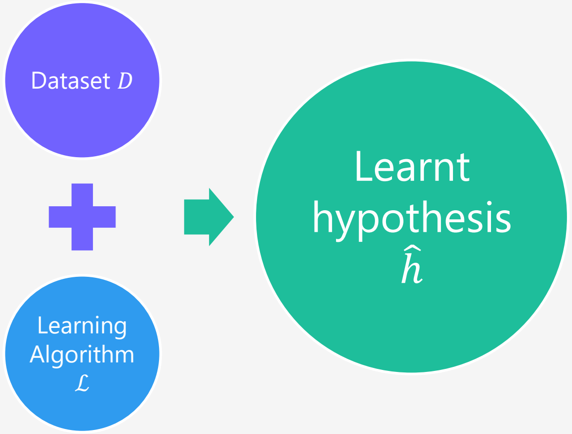


Reformulate the question:

Given $\varepsilon > 0$ and $0 < \delta < 1$, if there is a learning algorithm $\mathcal{L}$ such that the learnt hypothesis $\hat{h}$ is consistent, that is, $\hat{h}(X_i) = Y_i = h^*(X_i)$ for all $i = 1, \dots, n$, then what should be the minimum value of n such that $err_{\mathcal{D}}(\hat{h}) \coloneqq \mathbb{P}_{X \sim \mathcal{D}}\left[\hat{h}(X) \neq h^*(X)\right] < \varepsilon$ with prob. at least $1 - \delta$?

Learning Theory: Supervised Learning

Question: How *close* can we $\hat{h}$ get to h^*, given that number of samples is finite?

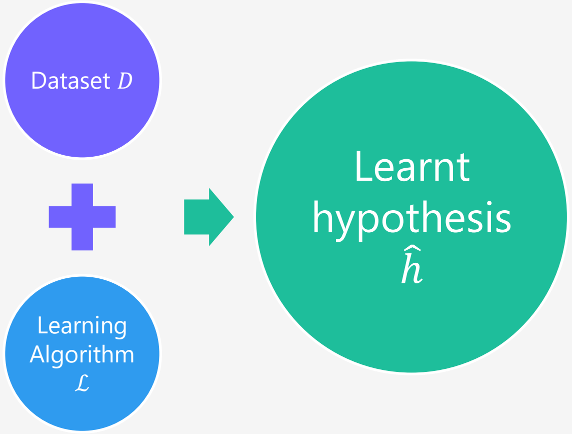


Reformulate the question:

Given $\varepsilon > 0$ and $0 < \delta < 1$, if there is a learning algorithm $\mathcal{L}$ such that the learnt hypothesis $\hat{h}$ is consistent, that is, $\hat{h}(X_i) = Y_i = h^*(X_i)$ for all $i = 1, \dots, n$, then what should be the minimum value of n such that $err_{\mathcal{D}}(\hat{h}) \coloneqq \mathbb{P}_{X\sim\mathcal{D}}\left[\hat{h}(X) \neq h^*(X)\right] < \varepsilon$ with prob. at least $1 - \delta$?

Learning Theory: Supervised Learning

Question: How *close* can we $\hat{h}$ get to h^*, given that number of samples is finite?

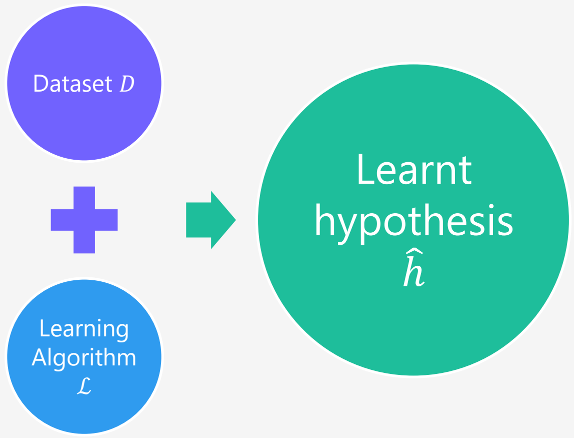


Reformulate the question:

Given $\varepsilon > 0$ and $0 < \delta < 1$, if there is a learning algorithm $\mathcal{L}$ such that the learnt hypothesis $\hat{h}$ is consistent, that is, $\hat{h}(X_i) = Y_i = h^*(X_i)$ for all $i = 1, \dots, n$, then what should be the minimum value of n such that $err_{\mathcal{D}}(\hat{h}) \coloneqq \mathbb{P}_{X\sim\mathcal{D}}\left[\hat{h}(X) \neq h^*(X)\right] < \varepsilon$ with prob. at least $1 - \delta$?

Learning Theory: Supervised Learning

Question: How *close* can we $\hat{h}$ get to h^*, given that number of samples is finite?



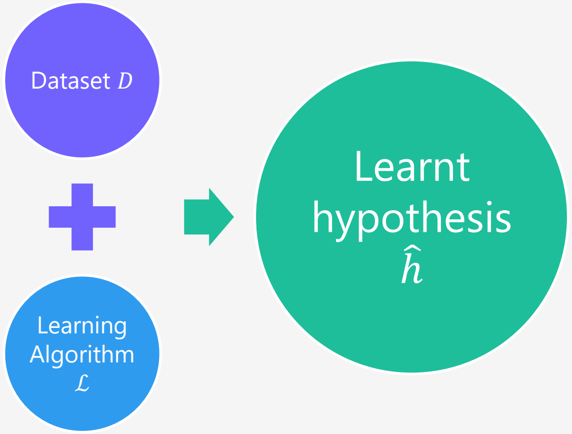
Reformulate the question:

Given $\varepsilon > 0$ and $0 < \delta < 1$, if there is a learning algorithm $\mathcal{L}$ such that the learnt hypothesis $\hat{h}$ is consistent, that is, $\hat{h}(X_i) = Y_i = h^*(X_i)$ for all $i = 1, \dots, n$, then what should be the minimum value of n such that

$err_{\mathcal{D}}(\hat{h}) \coloneqq \mathbb{P}_{X \sim \mathcal{D}}[\hat{h}(X) \neq h^*(X)] < \varepsilon$ with prob. at least $1 - \delta$?

Learning Theory: Supervised Learning

Question: How *close* can we $\hat{h}$ get to h^*, given that number of samples is finite?



Reformulate the question:

Given $\varepsilon > 0$ and $0 < \delta < 1$, if there is a learning algorithm $\mathcal{L}$ such that the learnt hypothesis $\hat{h}$ is consistent, that is, $\hat{h}(X_i) = Y_i = h^*(X_i)$ for all $i = 1, \dots, n$, then what should be the minimum value of n such that $err_{\mathcal{D}}(\hat{h}) \coloneqq \mathbb{P}_{X \sim \mathcal{D}}\left[\hat{h}(X) \neq h^*(X)\right] < \varepsilon$ with prob. at least $1 - \delta$?

Probably Approximately Correct (PAC) Learning Framework

Learning Theory: Supervised Learning

Learning Theory: Supervised Learning

- Since our learning algorithm $\mathcal{L}$ returned a consistent hypothesis $\hat{h}$, we have
$$err_D(\hat{h}) := \frac{1}{n}\sum_{i=1}^{n} 1\{\hat{h}(X_i) \neq h^*(X_i)\} = 0$$

- What is the probability of this happening?
$$\mathbb{P}\left[err_D(\hat{h}) = 0\right] = \mathbb{P}\left[\forall\ 1 \le i \le n: \hat{h}(X_i) = h^*(X_i)\right]$$
$$= \prod_{i=1}^{n} \mathbb{P}\left[\hat{h}(X_i) = h^*(X_i)\right] = \prod_{i=1}^{n}\left(1 - \mathbb{P}_{X\sim\mathcal{D}}\left[\hat{h}(X_i) \neq h^*(X_i)\right]\right) = \left(1 - err_{\mathcal{D}}(\hat{h})\right)^n$$

- Thus, if $err_{\mathcal{D}}(h) \ge \varepsilon$, then, $\mathbb{P}\left[err_D(\hat{h}) = 0\right] \le (1-\varepsilon)^n \le e^{-n\varepsilon}$

- Finally, we don't know what $\hat{h}$ can be, so we do a worst-case bound via Union Bound:
$$\mathbb{P}[\exists\ h \in\ \mathcal{H}: err_D(h) = 0\ and\ err_{\mathcal{D}}(h) \ge \varepsilon] \le \sum_{h\in\mathcal{H}} \mathbb{P}[err_D(h) = 0\ and\ err_{\mathcal{D}}(h) \ge \varepsilon] \le |\mathcal{H}|e^{-n\varepsilon}$$

- Setting this probability to less than δ, we obtain: $|\mathcal{H}|e^{-n\varepsilon} \le \delta \Rightarrow n \ge \frac{1}{\varepsilon}\log\left(\frac{|\mathcal{H}|}{\delta}\right)$

Learning Theory: Supervised Learning

- Since our learning algorithm $\mathcal{L}$ returned a consistent hypothesis $\hat{h}$, we have

$$err_D\left(\hat{h}\right) := \frac{1}{n}\sum_{i=1}^{n} 1\left\{\hat{h}(X_i) \neq h^*(X_i)\right\} = 0$$

- What is the probability of this happening?

$$\mathbb{P}\left[err_D\left(\hat{h}\right) = 0\right] = \mathbb{P}\left[\forall\, 1 \leq i \leq n: \hat{h}(X_i) = h^*(X_i)\right]$$
$$= \prod_{i=1}^{n} \mathbb{P}\left[\hat{h}(X_i) = h^*(X_i)\right] = \prod_{i=1}^{n}\left(1 - \mathbb{P}_{X\sim\mathcal{D}}\left[\hat{h}(X_i) \neq h^*(X_i)\right]\right) = \left(1 - err_{\mathcal{D}}(\hat{h})\right)^n$$

- Thus, if $err_{\mathcal{D}}(h) \geq \varepsilon$, then, $\mathbb{P}\left[err_{\mathcal{D}}\left(\hat{h}\right) = 0\right] \leq (1-\varepsilon)^n \leq e^{-n\varepsilon}$

- Finally, we don't know what $\hat{h}$ can be, so we do a worst-case bound via Union Bound:

$$\mathbb{P}[\exists\, h \in \mathcal{H}: err_D(h) = 0 \text{ } and \text{ } err_{\mathcal{D}}(h) \geq \varepsilon] \leq \sum_{h\in\mathcal{H}} \mathbb{P}[err_D(h) = 0 \text{ } and \text{ } err_{\mathcal{D}}(h) \geq \varepsilon] \leq |\mathcal{H}|e^{-n\varepsilon}$$

- Setting this probability to less than δ, we obtain: $|\mathcal{H}|e^{-n\varepsilon} \leq \delta \Rightarrow n \geq \frac{1}{\varepsilon}\log\left(\frac{|\mathcal{H}|}{\delta}\right)$

Learning Theory: Supervised Learning

- Since our learning algorithm $\mathcal{L}$ returned a consistent hypothesis $\hat{h}$, we have

$$err_D(\hat{h}) := \frac{1}{n}\sum_{i=1}^{n} 1\{\hat{h}(X_i) \neq h^*(X_i)\} = 0$$

- What is the probability of this happening?

$$\mathbb{P}\left[err_D(\hat{h}) = 0\right] = \mathbb{P}\left[\forall\, 1 \le i \le n : \hat{h}(X_i) = h^*(X_i)\right]$$
$$= \prod_{i=1}^{n} \mathbb{P}\left[\hat{h}(X_i) = h^*(X_i)\right] = \prod_{i=1}^{n}\left(1 - \mathbb{P}_{X\sim\mathcal{D}}\left[\hat{h}(X_i) \neq h^*(X_i)\right]\right) = \left(1 - err_{\mathcal{D}}(\hat{h})\right)^n$$

- Thus, if $err_{\mathcal{D}}(h) \ge \varepsilon$, then, $\mathbb{P}\left[err_{\mathcal{D}}(\hat{h}) = 0\right] \le (1-\varepsilon)^n \le e^{-n\varepsilon}$

- Finally, we don't know what $\hat{h}$ can be, so we do a worst-case bound via Union Bound:

$$\mathbb{P}[\exists\, h \in \mathcal{H} : err_D(h) = 0 \; and \; err_{\mathcal{D}}(h) \ge \varepsilon] \le \sum_{h \in \mathcal{H}} \mathbb{P}[err_D(h) = 0 \; and \; err_{\mathcal{D}}(h) \ge \varepsilon] \le |\mathcal{H}| e^{-n\varepsilon}$$

- Setting this probability to less than δ, we obtain: $|\mathcal{H}| e^{-n\varepsilon} \le \delta \Rightarrow n \ge \frac{1}{\varepsilon}\log\left(\frac{|\mathcal{H}|}{\delta}\right)$

Learning Theory: Supervised Learning

- Since our learning algorithm $\mathcal{L}$ returned a consistent hypothesis $\hat{h}$, we have
$$err_D(\hat{h}) := \frac{1}{n}\sum_{i=1}^{n} 1\{\hat{h}(X_i) \neq h^*(X_i)\} = 0$$

- What is the probability of this happening?
$$\mathbb{P}\left[err_D(\hat{h}) = 0\right] = \mathbb{P}\left[\forall\, 1 \leq i \leq n: \hat{h}(X_i) = h^*(X_i)\right]$$
$$= \prod_{i=1}^{n} \mathbb{P}\left[\hat{h}(X_i) = h^*(X_i)\right] = \prod_{i=1}^{n}\left(1 - \mathbb{P}_{X\sim\mathcal{D}}\left[\hat{h}(X_i) \neq h^*(X_i)\right]\right) = \left(1 - err_{\mathcal{D}}(\hat{h})\right)^n$$

- Thus, if $err_{\mathcal{D}}(h) \geq \varepsilon$, then, $\mathbb{P}\left[err_{\mathcal{D}}(\hat{h}) = 0\right] \leq (1-\varepsilon)^n \leq e^{-n\varepsilon}$

- Finally, we don't know what $\hat{h}$ can be, so we do a worst-case bound via Union Bound:
$$\mathbb{P}[\exists\, h \in \mathcal{H}: err_D(h) = 0 \; and \; err_{\mathcal{D}}(h) \geq \varepsilon] \leq \sum_{h\in\mathcal{H}} \mathbb{P}[err_D(h) = 0 \; and \; err_{\mathcal{D}}(h) \geq \varepsilon] \leq |\mathcal{H}|e^{-n\varepsilon}$$

- Setting this probability to less than δ, we obtain: $|\mathcal{H}|e^{-n\varepsilon} \leq \delta \Rightarrow n \geq \frac{1}{\varepsilon}\log\left(\frac{|\mathcal{H}|}{\delta}\right)$

Learning Theory: Supervised Learning

- Since our learning algorithm $\mathcal{L}$ returned a consistent hypothesis $\hat{h}$, we have

$$err_D(\hat{h}) := \frac{1}{n}\sum_{i=1}^{n} 1\{\hat{h}(X_i) \neq h^*(X_i)\} = 0$$

- What is the probability of this happening?

$$\mathbb{P}[err_D(\hat{h}) = 0] = \mathbb{P}[\forall\, 1 \le i \le n : \hat{h}(X_i) = h^*(X_i)]$$

$$= \prod_{i=1}^{n} \mathbb{P}[\hat{h}(X_i) = h^*(X_i)] = \prod_{i=1}^{n} \left(1 - \mathbb{P}_{X\sim\mathcal{D}}[\hat{h}(X_i) \neq h^*(X_i)]\right) = \left(1 - err_{\mathcal{D}}(\hat{h})\right)^n$$

- Thus, if $err_{\mathcal{D}}(h) \ge \varepsilon$, then, $\mathbb{P}[err_{\mathcal{D}}(\hat{h}) = 0] \le (1-\varepsilon)^n \le e^{-n\varepsilon}$

- Finally, we don't know what $\hat{h}$ can be, so we do a worst-case bound via Union Bound:

$$\mathbb{P}[\exists\, h \in \mathcal{H} : err_D(h) = 0 \;and\; err_{\mathcal{D}}(h) \ge \varepsilon] \le \sum_{h\in\mathcal{H}} \mathbb{P}[err_D(h) = 0 \;and\; err_{\mathcal{D}}(h) \ge \varepsilon] \le |\mathcal{H}|e^{-n\varepsilon}$$

- Setting this probability to less than δ, we obtain: $|\mathcal{H}|e^{-n\varepsilon} \le \delta \Rightarrow n \ge \frac{1}{\varepsilon}\log\left(\frac{|\mathcal{H}|}{\delta}\right)$

Learning Theory: Supervised Learning

PAC Learning 1st Result: $\boldsymbol{n \geq \frac{1}{\varepsilon} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$

- What if the hypothesis class is not finite? The above bound is vacuous.
- Problem step is the Union Bound:

$$\mathbb{P}[\exists\, h \in \mathcal{H}: err_D(h) = 0 \; and \; err_{\mathcal{D}}(h) \geq \varepsilon] \leq \sum_{h \in \mathcal{H}} \mathbb{P}[err_D(h) = 0 \; and \; err_{\mathcal{D}}(h) \geq \varepsilon] \leq |\mathcal{H}| e^{-n\varepsilon}$$

- We need a better characterization of the complexity of $\mathcal{H}$.

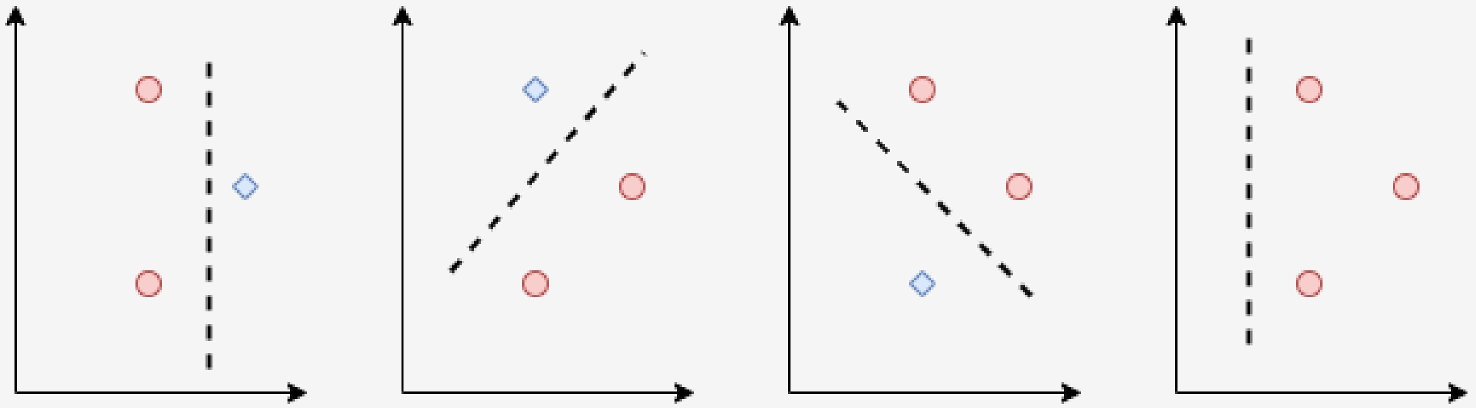
Learning Theory: Supervised Learning

PAC Learning 1st Result:

$$\boldsymbol{n \geq \frac{1}{\varepsilon} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$$

VC Dimension

- Labelling: A labelling of a set S is just an assignment of 0 or 1 to each of its elements.
- Shattering: Given a set $\mathcal{X}$ and a hypothesis class $\mathcal{H}$, we say a subset $S \subseteq \mathcal{X}$ is shattered by $\mathcal{H}$ if there exists a hypothesis h for every labelling of S such that $h(x) = y$ for all $x \in S$ and y is its label.
- VC Dimension $d(\mathcal{H}, \mathcal{X})$ of a hypothesis class $\mathcal{H}$ with respect to set $\mathcal{X}$ is the size of largest subset of $\mathcal{X}$ that can be shattered by $\mathcal{H}$.



Learning Theory: Supervised Learning

PAC Learning
1st Result:

$$\boldsymbol{n \geq \frac{1}{\varepsilon} \log\left(\frac{|\mathcal{H}|}{\delta}\right)}$$

VC Dimension

Theorem: *Let d be the VC dimension of a hypothesis class $\mathcal{H}$ with respect to set $\mathcal{X}$. Then, any consistent learning algorithm outputs an $(\varepsilon,\ \delta)$-PAC hypothesis when the number of training samples m satisfies*:

$$m \geq C\left(\frac{1}{\varepsilon}\log\frac{1}{\delta} + \frac{d}{\varepsilon}\log\frac{1}{\varepsilon}\right)$$

Sauer's Lemma (Informal): *If the VC dimension of of a hypothesis class $\mathcal{H}$ with respect to set $\mathcal{X}$, then, using all hypotheses from $\mathcal{H}$, the number of distinct functions $f: \mathcal{X} \rightarrow \{0,1\}$ is at most $O(|\mathcal{X}|^d)$.*

Learning Theory: Supervised Learning

PAC Learning 2nd Result: $m \geq C\left(\frac{1}{\varepsilon}\log\frac{1}{\delta} + \frac{d}{\varepsilon}\log\frac{1}{\varepsilon}\right)$

Some Interesting VC-dimension Results

- Only hypothesis classes with finite VC dimension can *generalize*, that is, the test and train error rates will converge to the same value.
- The VC dimension of any arbitrary feed-forward NN with linear threshold activation ($sign(w^T x)$) consisting of N weights has VC dimension $O(N \log N)$.
- Suppose $\mathcal{H} = \{h: h(x) = w_0 + w_1\sigma(a_1 x) + w_2\sigma(a_2 x),\ \ w_0, w_1, w_2, a_1, a_2 \in \mathbb{R}\}$ where $\sigma(x) = \frac{1}{1+e^{-x}} + cx^3 e^{-x^3} \sin x$ for some small c. Then, VC dimension of $\mathcal{H}$ is ∞!
- If a neural network has d parameters and performs up to t operations on the input to generate the final output value, then that neural network has VC-Dim of $O(t^2 d^2\ polylog(t, d))$ where operations are $+, -, \times, \div, \exp(\cdot), <, >, \leq, \geq, =, \neq$.

Learning Theory: Supervised Learning

What's New?

- Hanneke, Steve, Kasper Green Larsen, and Nikita Zhivotovskiy. "Revisiting Agnostic PAC Learning." *arXiv preprint arXiv:2407.19777* (2024). --- this paper proves that the empirical risk minimization (ERM), that is, minimizing training error, is not optimal for agnostic learning. Several open questions are posed: (i) Are classifiers that are optimal realizable PAC learners also optimal agnostic PAC learners? (ii) Computationally efficient agnostic PAC learners

- Brand, Cornelius, Robert Ganian, and Kirill Simonov. "A parameterized theory of PAC learning." *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 37. No. 6. 2023. --- this paper is a first step towards characterizing efficient PAC learnability. In traditional complexity theory, the idea of parameterized complexity has resulted into new insights, namely Fixed Parameter Tractability (FPT). Looking at PAC learning problems from a similar lens can lead us to deep insights into the limits of learnability.

- Zeng, Shiwei, and Jie Shen. "Efficient PAC learning from the crowd with pairwise comparisons." *International Conference on Machine Learning*. PMLR, 2022. --- this paper merges social choice with PAC learning to tackle a very important practical setting. Several extensions can be made: (i) Can the crowd-workers be modelled better, e.g. more the workload, lesser the accuracy? (ii) Consider other query models like different costs for different workers

Learning Theory: Bandits

- Online Learning Problem: Dynamic decision-making
- State-less Environment
- Explore-Exploit Trade-off
- Optimize what you actually care about: Reward

Learning Theory: Bandits

- Online Learning Problem: Dynamic decision-making
- State-less Environment
- Explore-Exploit Trade-off
- Optimize what you actually care about: Reward

Applications

- Recommender Systems
- Drug Trials
- Online Advertising

Learning Theory: Bandits

- Online Learning Problem: Dynamic decision-making
- State-less Environment
- Explore-Exploit Trade-off
- Optimize what you actually care about: Reward

Applications

- Recommender Systems

- Drug Trials
- Online Advertising

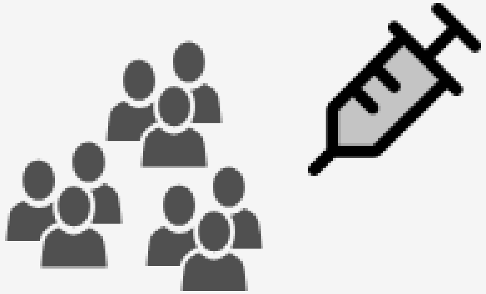
Learning Theory: Bandits

- Online Learning Problem: Dynamic decision-making
- State-less Environment
- Explore-Exploit Trade-off
- Optimize what you actually care about: Reward

Applications

- Recommender Systems

- Drug Trials



- Online Advertising

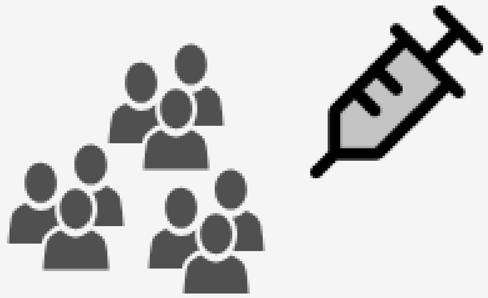
Learning Theory: Bandits

- Online Learning Problem: Dynamic decision-making
- State-less Environment
- Explore-Exploit Trade-off
- Optimize what you actually care about: Reward

Applications

- Recommender Systems

- Drug Trials



- Online Advertising

Learning Theory: Bandits

- K arms, each with distribution $\mathcal{P}_i, i = 1, \dots, K$
- Mean reward of an arm i is μ_i
- Number of rounds of interaction is T
- If we knew the best arm i^*, we would just pull it for all rounds
- What is the price of lack of information?

$$Regret(T) \coloneqq \sum_{t=1}^{T} \mu_{i^*} - \sum_{t=1}^{T} \mu_{i_t}$$

Learning Theory: Bandits

$$Regret(T) \coloneqq \sum_{t=1}^{T} \mu_{i^*} - \sum_{t=1}^{T} \mu_{i_t}$$

Regret vs. Maximum Likelihood

1. Naïve way: Sample each arm $T/2K$ times and fit a model, that is, for $i = 1, \dots, K$, $\hat{\mu}_i = \frac{2K}{T}\sum_{t=1}^{T/2} r_t \cdot 1\{i_t = i\}$ and then play $\hat{\imath} = argmax_{i=1,\dots,K}\, \hat{\mu}_i$ for the remaining $T/2$ rounds

2. However, in use cases, often our goal is to maximize the profit, not the MLE

3. Moreover, in real life problems, there is a cost associated with collecting good quality data for MLE, which is captured by the regret definition

Learning Theory: Bandits

Optimism in the Face of Uncertainty: The UCB Algorithm

$$Regret(T) := \sum_{t=1}^{T} \mu_{i^*} - \sum_{t=1}^{T} \mu_{i_t}$$

Algorithm

1. Input arms $1, \dots, K$ and the number of rounds T
2. Play each arm once and observer reward r_i for every arm i.
3. Set $T_i = 1$ and $\hat{\mu}_i = r_i \quad \forall\, i = 1, \dots, K$ // Set the no. of times an arm has been pulled and its empirical mean reward
4. For $t = K + 1, \dots, T$ do:
5. Play arm $i_t = \arg\max_{i=1,\dots,K} \hat{\mu}_i + \sqrt{\frac{4}{T_i} \log^{+}\left(\frac{T}{KT_i}\right)}$; Observe reward r_t.
6. Update $T_{i_t} = T_{i_t} + 1$ // Update the number of times arm pulled so far
7. Update $\hat{\mu}_{i_t} = \frac{1}{T_{i_t}} \sum_{s=1}^{t} r_s \cdot 1\{i_s = i_t\}$ // Update the empirical mean reward of the arm played

Here, $\log^{+}(x) = \log(\max\{1, x\})$.

Learning Theory: Bandits

$$Regret(T) := \sum_{t=1}^{T} \mu_{i^*} - \sum_{t=1}^{T} \mu_{i_t}$$

Optimism in the Face of Uncertainty: The UCB Algorithm

Key Idea: Play arm $i_t = \arg\max_{i=1,\ldots,K} \hat{\mu}_i + \sqrt{\frac{\mathbf{4}}{\boldsymbol{T_i}} \mathbf{log}^+\left(\frac{\boldsymbol{T}}{\boldsymbol{KT_i}}\right)}$

Theorem: *For the UCB Algorithm,* $\mathbb{E}[Regret(T)] \leq 39\sqrt{KT} + K$

Learning Theory: Bandits

Generalizations of the K-armed Bandit Problem

1. Linear Contextual Bandits:
   a. At each round $t = 1, \dots, T$, a set of vectors $\mathcal{X}_t = \{x_{1,t}, x_{2,t}, \dots, x_{K,t}\}$ are presented
   b. For a vector $x_{i,t}$, the random reward is $r = \langle x_{i,t}, \theta^* \rangle + \eta$, where θ^* is unknown but fixed and η is zero-mean (Gaussian) noise
   c. $Regret(T) \coloneqq \sum_{t=1}^{T} \max_{i=1,\dots,K} \langle x_{i,t}, \theta^* \rangle - \sum_{t=1}^{T} \langle x_{i_t,t}, \theta^* \rangle$
2. Generalized Linear Contextual Bandits:
   a. Arms are same as linear bandits. Parameter θ^* is unknown but fixed
   b. For a vector $x_{i,t}$, the random reward is r is sampled acc. to the PDF $\exp\left(-r\langle x_{i,t}, \theta^* \rangle + b\left(\langle x_{i,t}, \theta^* \rangle\right) + c(r)\right)$
   c. $Regret(T) \coloneqq \sum_{t=1}^{T} \max_{i=1,\dots,K} \mu(\langle x_{i,t}, \theta^* \rangle) - \sum_{t=1}^{T} \mu(\langle x_{i_t,t}, \theta^* \rangle)$, where $\mu\left(\langle x_{i,t}, \theta^* \rangle\right) = \mathbb{E}\left[r \middle| x_{i,t}, \theta^*\right] = \dot{b}\left(\langle x_{i,t}, \theta^* \rangle\right)$

Learning Theory: Bandits

What’s New?

1. Sawarni, Ayush, Nirjhar Das, Gaurav Sinha, and Siddharth Barman. “Generalized Linear Bandits with Limited Adaptivity." *arXiv preprint arXiv:2404.06831* (2024). (to appear in NeurIPS 2024) --- in this paper we closed a major open problem in generalized linear bandit showing optimal regret and efficient computation. However, following questions remain (i) An algorithm that does not require the knowledge of a certain instance-dependent parameter (ii) An algorithm efficient in the number of arms (important for practice).

2. Lattimore, Tor. "Bandit convex optimisation." *arXiv preprint arXiv:2402.06535* (2024). --- this beautiful monograph introduces the problem and discusses the state-of-the-art results. There are lots of open questions in this area!

3. Maiti, Arnab, Ross Boczar, Kevin Jamieson, and Lillian Ratliff. "Near-Optimal Pure Exploration in Matrix Games: A Generalization of Stochastic Bandits & Dueling Bandits." In *International Conference on Artificial Intelligence and Statistics*, pp. 2602-2610. PMLR, 2024. --- this paper gives the sample complexity of finding pure strategy Nash equilibrium in 2-player zero sum games. Recently, a lot of algorithmic game theory problems have been looked at as stochastic models where bandit techniques become essential.

AGT & ComSoc

AGT & ComSoc: Nash Equilibrium

An Example: Prisoner's Dilemma

Prisoner A \ Prisoner B	Stay Silent	Betray
Stay Silent	-1, -1	-3, 0
Betray	0, -3	-2, -2

AGT & ComSoc: Nash Equilibrium

An Example: Prisoner's Dilemma

Prisoner A \ Prisoner B	Stay Silent	Betray
Stay Silent	-1, -1	-3, 0
Betray	0, -3	-2, -2

Normal Form Games

- Players $N = \{1,2,\ldots,n\}$
- Each player chooses action $a_i \in A_i$
  - Action profile $\vec{a} = (a_1, a_2, \ldots, a_n) \in \mathcal{A} = A_1 \times A_2 \times \cdots \times A_n$
  - $\vec{a}_{-i} = (a_1, a_2, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n)$
- Utility function of player $i \in N$ is $u_i : \mathcal{A} \to \mathbb{R}$. Thus, for action profile $\vec{a}$, player i gets $u_i(\vec{a})$

AGT & ComSoc: Nash Equilibrium

An Example: Prisoner's Dilemma

Prisoner A \ Prisoner B	Stay Silent	Betray
Stay Silent	-1, -1	-3, 0
Betray	0, -3	-2, -2

AGT & ComSoc: Nash Equilibrium

An Example: Prisoner's Dilemma

Prisoner A \ Prisoner B	Stay Silent	Betray
Stay Silent	-1, -1	-3, 0
Betray	0, -3	-2, -2

Some Strategies

- Pareto Optimal: A strategy is Pareto Optimal if the utility of a player cannot be increased without decreasing the utility of some other player. (Silent, Silent) is Pareto Optimal.
- Dominant Strategy: A dominant strategy for player $i \in N$ is a strategy $a_i \in A_i$ such that
$$u_i(a_i, \vec{a}_{-i}) \geq u_i(a_i', \vec{a}_{-i}) \quad \forall\, \vec{a}_{-i} \in A_{-i}$$
- Dominant Strategy Equilibrium: If all players play their dominant strategy. (Betray, Betray)
- Nash Equilibrium: An action profile $\vec{a}$ is a Nash equilibrium if for all $i \in N$ and $a_i' \in A_i$,
$$u_i(\vec{a}) \geq u_i(a_i', \vec{a}_{-i})$$

  (Betray, Betray) is the NE.

AGT & ComSoc: Nash Equilibrium

An Example: Prisoner's Dilemma

Prisoner B / Prisoner A	Stay Silent	Betray
Stay Silent	-1, -1	-3, 0
Betray	0, -3	-2, -2

Some Strategies

- Pareto Optimal: A strategy is Pareto Optimal if the utility of a player cannot be increased without decreasing the utility of some other player. (Silent, Silent) is Pareto Optimal.
- Dominant Strategy: A dominant strategy for player $i \in N$ is a strategy $a_i \in A_i$ such that
$$u_i(a_i, \vec{a}_{-i}) \geq u_i(a_i', \vec{a}_{-i}) \quad \forall\, \vec{a}_{-i} \in A_{-i}$$
- Dominant Strategy Equilibrium: If all players play their dominant strategy. (Betray, Betray)
- Nash Equilibrium: An action profile $\vec{a}$ is a Nash equilibrium if for all $i \in N$ and $a_i' \in A_i$,
$$u_i(\vec{a}) \geq u_i(a_i', \vec{a}_{-i})$$

(Betray, Betray) is the NE.

AGT & ComSoc: Nash Equilibrium

An Example: Prisoner's Dilemma

Prisoner A \ Prisoner B	Stay Silent	Betray
Stay Silent	-1, -1	-3, 0
Betray	0, -3	-2, -2

Some Strategies

- Pareto Optimal: A strategy is Pareto Optimal if the utility of a player cannot be increased without decreasing the utility of some other player. (Silent, Silent) is Pareto Optimal.
- Dominant Strategy: A dominant strategy for player $i \in N$ is a strategy $a_i \in A_i$ such that
$$u_i(a_i, \vec{a}_{-i}) \geq u_i(a_i', \vec{a}_{-i}) \quad \forall\, \vec{a}_{-i} \in A_{-i}$$
- Dominant Strategy Equilibrium: If all players play their dominant strategy. (Betray, Betray)
- Nash Equilibrium: An action profile $\vec{a}$ is a Nash equilibrium if for all $i \in N$ and $a_i' \in A_i$,
$$u_i(\vec{a}) \geq u_i(a_i', \vec{a}_{-i})$$

  (Betray, Betray) is the NE.

AGT & ComSoc: Nash Equilibrium

An Example: Prisoner's Dilemma

Prisoner A \ Prisoner B	Stay Silent	Betray
Stay Silent	-1, -1	-3, 0
Betray	0, -3	-2, -2

Some Strategies

- Pareto Optimal: A strategy is Pareto Optimal if the utility of a player cannot be increased without decreasing the utility of some other player. (Silent, Silent) is Pareto Optimal.
- Dominant Strategy: A dominant strategy for player $i \in N$ is a strategy $a_i \in A_i$ such that
$$u_i(a_i, \vec{a}_{-i}) \geq u_i(a_i', \vec{a}_{-i}) \quad \forall\, \vec{a}_{-i} \in A_{-i}$$
- Dominant Strategy Equilibrium: If all players play their dominant strategy. (Betray, Betray)
- Nash Equilibrium: An action profile $\vec{a}$ is a Nash equilibrium if for all $i \in N$ and $a_i' \in A_i$,
$$u_i(\vec{a}) \geq u_i(a_i', \vec{a}_{-i})$$

  (Betray, Betray) is the NE.

AGT & ComSoc: Nash Equilibrium

- Pure Strategy: Action chosen deterministically
- Mixed Strategy: Action sampled from a probability distribution over all actions

Theorem (Nash 1951): *For all finite normal form games, a mixed strategy equilibrium always exists.*

Nash Equilibrium basically says that in such an equilibrium, no individual can benefit by deviating. Thus, all individuals continue to act according to the equilibrium action profile.

Hence, finding Nash Equilibrium in competitive games guarantees a player that no one will deviate from such an equilibrium.

AGT & ComSoc: Nash Equilibrium

Brouwer Fixed Point Theorem: *If $f: P \to P$ is a continuous map over a convex compact domain P, then there exists a point $x \in P$ such that $f(x) = x$.*

Proof of Nash: For a player $i \in N$, denote by $p_i(a)$ the probability of playing action $a \in A_i$. Denote the mixed strategy by $p \in P$.

Define for all players $i \in N$:- $\alpha_{i,a}(p) = \max\{0,\ u_i(a, p_{-i}) - u_i(p)\} \quad \forall\, a \in A_i$

Consider the function $f: P \to P$, and let $f(p) = p'$ such that,

$$p_i'(a) = \frac{p_i(a) + \alpha_{i,a}(p)}{1 + \sum_{a' \in A_i} \alpha_{(i,a')}(p)} \qquad \forall\, i \in N \text{ and } a \in A_i$$

The domain P is convex and compact and f is continuous. Thus, there exists $p_0 \in P$ such that $f(p^0) = p^0$.

Next, we show that p^0 is a Nash Equilibrium.

AGT & ComSoc: Nash Equilibrium

Proof of Nash *(cont'd)*: $\alpha_{i,a}(p) = \max\{0,\ u_i(a, p_{-i}) - u_i(p)\} \quad \forall\, a \in A_i$

$$p_i'(a) = \frac{p_i(a) + \alpha_{i,a}(p)}{1 + \sum_{a' \in A_i} \alpha_{(i,a')}(p)} \qquad \forall\, i \in N \text{ and } a \in A_i$$

First note that if p^0 is a NE, then $u_i\left(a, p_{-i}^0\right) \le u_i(p^0)$, hence $\alpha_{i,a}(p^0) = 0$ which implies $p^0 = p^{0\prime}$.

Now, since $\sum_{a \in A_i} p_i^0(a) \cdot u_i\left(a, p_{-i}^0\right) = u_i(p^0)$, there exists at least one $a' \in A_i$ such that $u_i\left(a', p_{-i}^0\right) \le u_i(p^0)$.

For this action, $\alpha_{i,a'}(p^0) = 0$, thus, $p_i^{0\prime}(a') = \frac{p_i^0(a')}{1 + \sum_{b \in A_i} \alpha_{i,b}(p^0)}$, but since p^0 is a fixed point, $p_i^{0\prime}(a') = p_i^0(a')$.

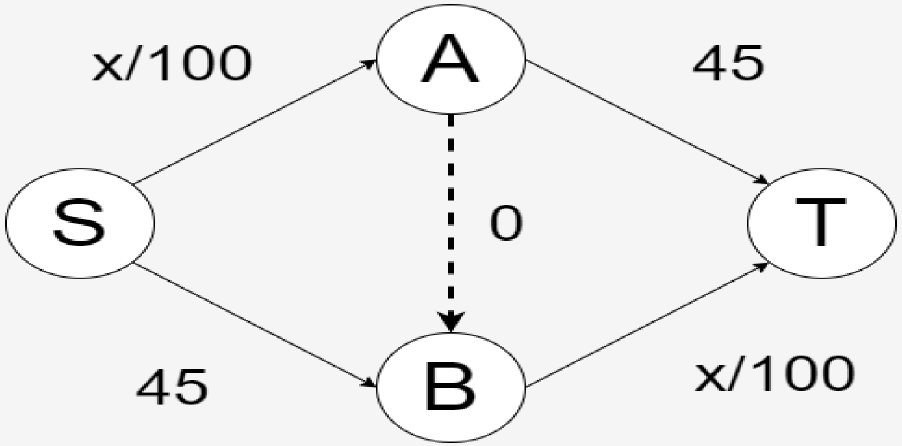
Hence, $\sum_{b \in A_i} \alpha_{i,b}(p^0) = 0$ and since all $\alpha_{i,b}(p^0) \ge 0$, we must have $\alpha_{i,b}(p^0) = 0$, which means

$$u_i\left(b, p_{-i}^0\right) - u_i(p^0) \le 0 \qquad \forall\, b \in A_i$$

Thus p^0 is a NE.

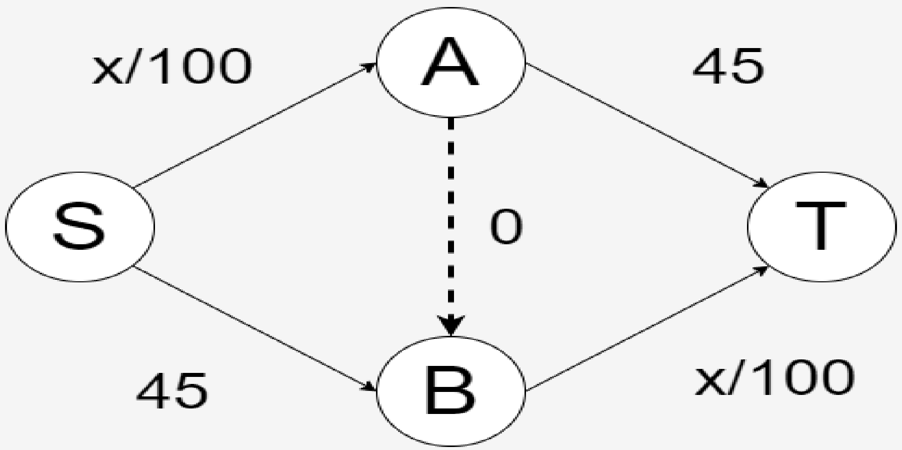
AGT & ComSoc: Nash Equilibrium

Braess Paradox:



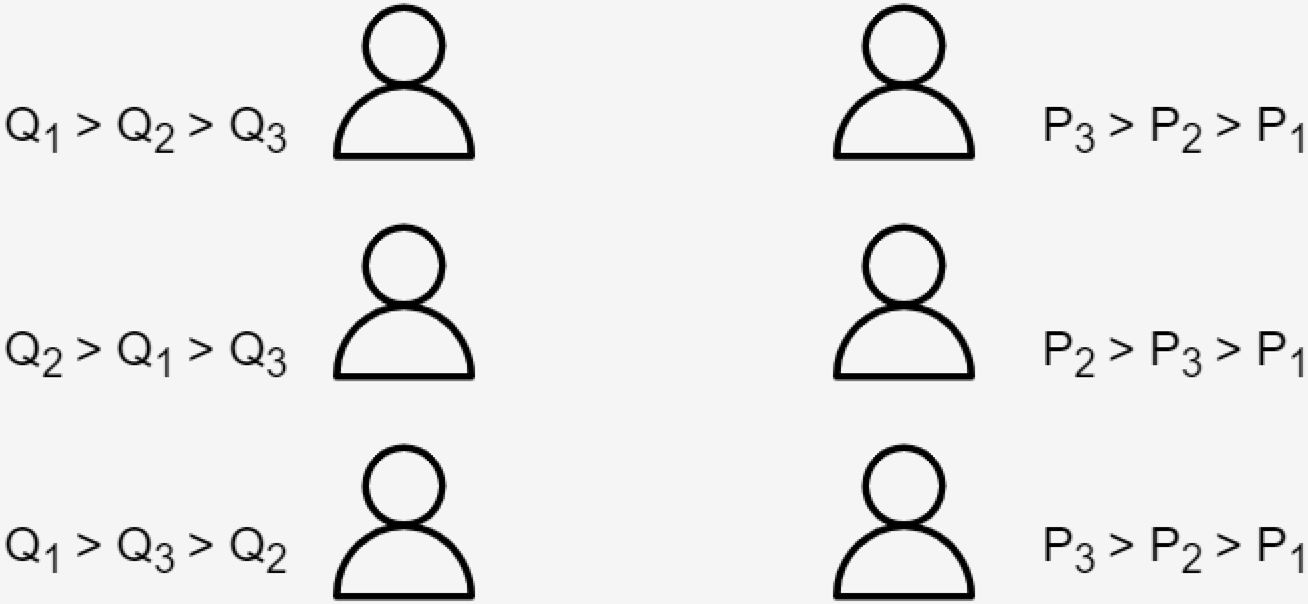
AGT & ComSoc: Nash Equilibrium

Braess Paradox:



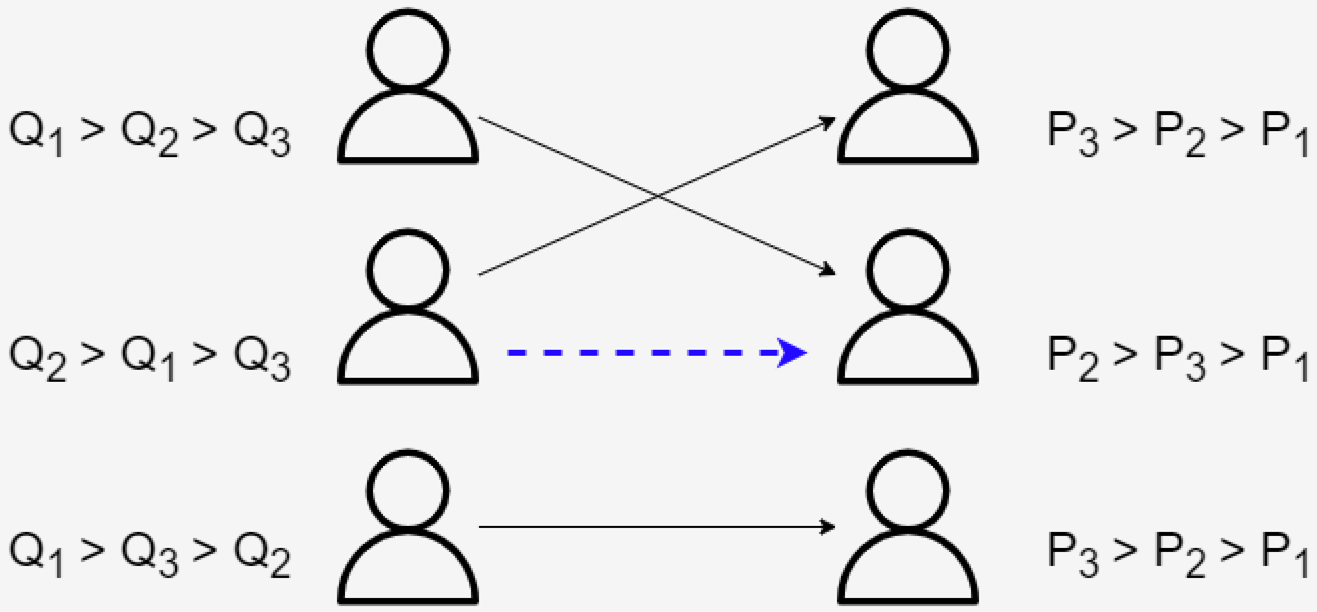
- Consider 4000 cars: 2000 on S-A-T and 2000 on S-B-T is an NE. Travel time is 65 min.
- New road A-B is added whose travel time is 0. The new NE is 4000 cars via S-A-B-T. Travel time is 80 min.

AGT & ComSoc: Stable Matching



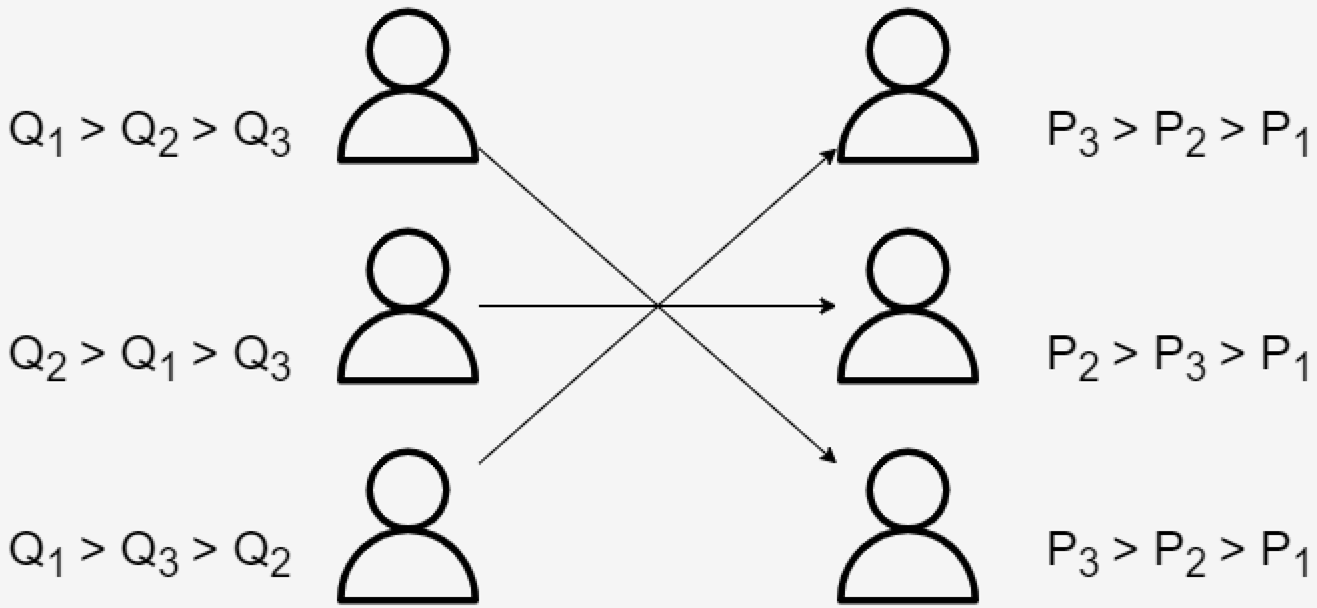
- Two groups P and Q
- Every p in P has a preference ordering over Q
- Every q in Q has a preference ordering over P
- If p prefers q over their current match and q prefers p over their current match, then the matching is *unstable*.

AGT & ComSoc: Stable Matching



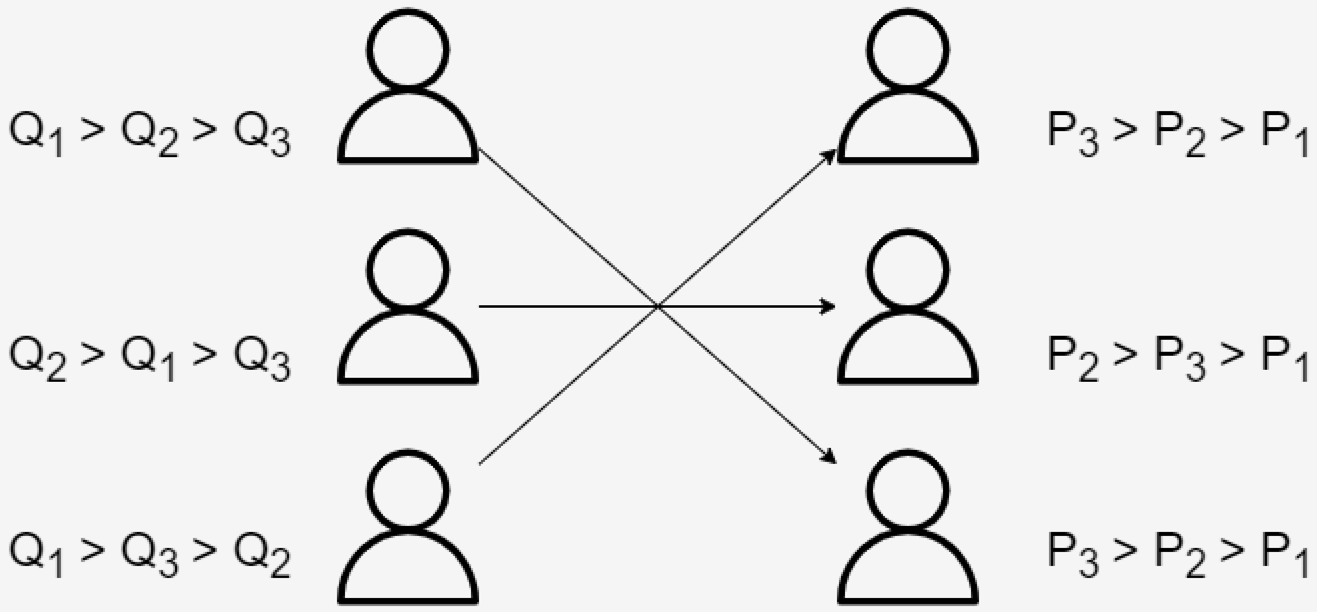
- Two groups P and Q
- Every p in P has a preference ordering over Q
- Every q in Q has a preference ordering over P
- If p prefers q over their current match and q prefers p over their current match, then the matching is *unstable*.

AGT & ComSoc: Stable Matching



- Two groups P and Q
- Every p in P has a preference ordering over Q
- Every q in Q has a preference ordering over P
- If p prefers q over their current match and q prefers p over their current match, then the matching is *unstable*.

AGT & ComSoc: Stable Matching



- Two groups P and Q
- Every p in P has a preference ordering over Q
- Every q in Q has a preference ordering over P
- If p prefers q over their current match and q prefers p over their current match, then the matching is *unstable*.

- **Does stable matching always exist?**
- **If it exists, can it be found in poly-time?**

AGT & ComSoc: Stable Matching

- **Does stable matching always exist?**
- **If it exists, can it be found in poly-time?**

AGT & ComSoc: Stable Matching

- **Does stable matching always exist?**
- **If it exists, can it be found in poly-time?**

- Yes! To both the questions!
- Seminal work by Gale and Shapley in 1962
- Application in Residence Matching for doctors
- 2012 Nobel Prize in Economics to Shapley!

AGT & ComSoc: Stable Matching

- **Does stable matching always exist?**
- **If it exists, can it be found in poly-time?**

- Yes! To both the questions!
- Seminal work by Gale and Shapley in 1962
- Application in Residence Matching for doctors
- 2012 Nobel Prize in Economics to Shapley!

AGT & ComSoc: Stable Matching

- **Does stable matching always exist?**
- **If it exists, can it be found in poly-time?**

- Yes! To both the questions!
- Seminal work by Gale and Shapley in 1962
- Application in Residence Matching for doctors
- 2012 Nobel Prize in Economics to Shapley!

AGT & ComSoc: Stable Matching

- **Does stable matching always exist?**
- **If it exists, can it be found in poly-time?**

- Yes! To both the questions!
- Seminal work by Gale and Shapley in 1962
- Application in Residence Matching for doctors
- 2012 Nobel Prize in Economics to Shapley!

AGT & ComSoc: Stable Matching

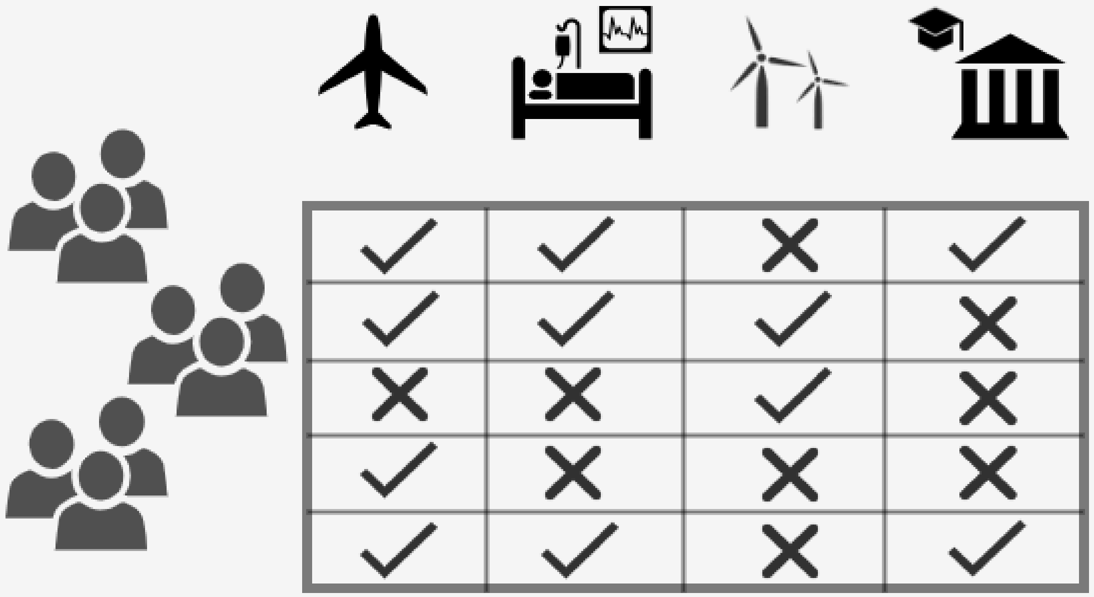
- **Does stable matching always exist?**
- **If it exists, can it be found in poly-time?**

- Yes! To both the questions!
- Seminal work by Gale and Shapley in 1962
- Application in Residence Matching for doctors
- 2012 Nobel Prize in Economics to Shapley!

Deferred Acceptance Algorithm

1. Input P, Q and the preference orderings
2. Initially everyone is unmatched
3. All unmatched p in P proposes to their highest preferred member in Q
4. If a q in Q receives a proposal from some p who is more preferred by q than their current engagement, q gets engaged with p
5. Else q rejects their proposals
6. All p who made the proposals but got rejected strike out the respective q's from their preference list
7. If no p is rejected, then stop, else go back to step 3

AGT & ComSoc: Participatory Budgeting



- n Voters, m Projects, Budget K
- Every project $j \in [m]$ has cost c_j
- Voters cast vote indicating which projects out of $[m]$ they like: $A_i \subseteq [m]$
- Need to output a budget-feasible allocation $W \subseteq [m]$, that is, $\sum_{j \in W} c_j \leq K$
- Objective is to make a *welfarist* allocation

AGT & ComSoc: Participatory Budgeting

Core of the PB

A (budget-feasible) allocation $W \subseteq [m]$ is said to be in the core if

- for all $S \subseteq [n]$ and $T \subseteq [m]$ such that $\sum_{j \in T} c_j \leq \frac{|S|}{n} \cdot K$
- there exists a voter $v^* \in S$ such that $|A_{v^*} \cap T| < |A_{v^*} \cap W|$

Weak Core: Instead of strict inequality in point 2 above, we have less-than-or-equal

AGT & ComSoc: Participatory Budgeting

Core of the PB

A (budget-feasible) allocation $W \subseteq [m]$ is said to be in the core if

- for all $S \subseteq [n]$ and $T \subseteq [m]$ such that $\sum_{j \in T} c_j \leq \frac{|S|}{n} \cdot K$
- there exists a voter $v^* \in S$ such that $|A_{v^*} \cap T| < |A_{v^*} \cap W|$

Weak Core: Instead of strict inequality in point 2 above, we have less-than-or-equal

Current State of Knowledge

1. There are instances for no allocation is in the core.
2. No instance is known where the weak core is empty. Does the weak core always exist?
3. If the weak core exists, then can it be found in poly-time?
4. Can *approximations* of the weak core be found in poly-time?
5. In practical datasets, greedy allocation is observed to be in the weak core! Explain this phenomenon.

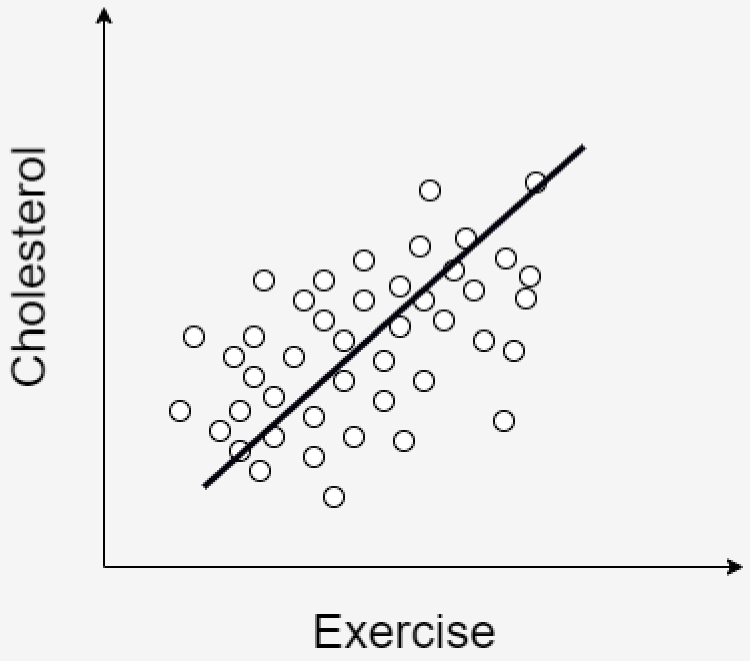
AGT & ComSoc

What’s new?

- Participatory Budgeting & Multi-winner voting --- lots of open questions. Take your pick!
- Algorithmic Game Theory --- newer models with much more realistic assumptions, computational hardness of known existential results, stochastic models for repeated games

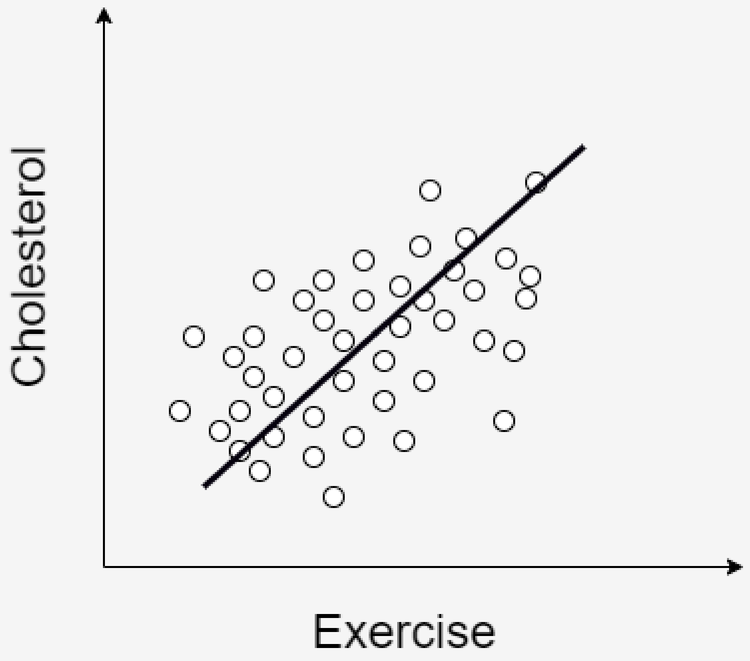
Causality

Causality: Simpson's Paradox



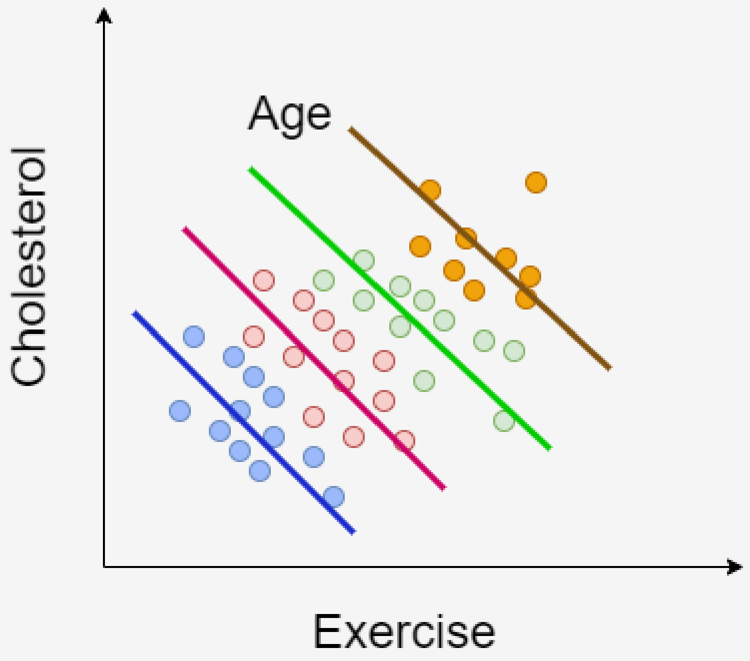
Causality: Simpson's Paradox

- Does exercise increase cholesterol?
- Introduce a new variable
- Can repeat this process again and again
- Is there any hope?



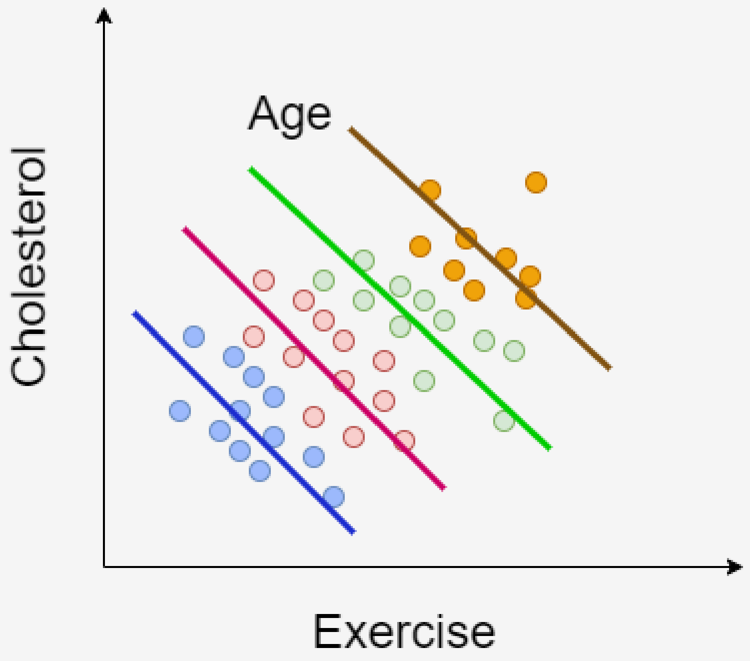
Causality: Simpson's Paradox

- Does exercise increase cholesterol?
- Introduce a new variable
- Can repeat this process again and again
- Is there any hope?



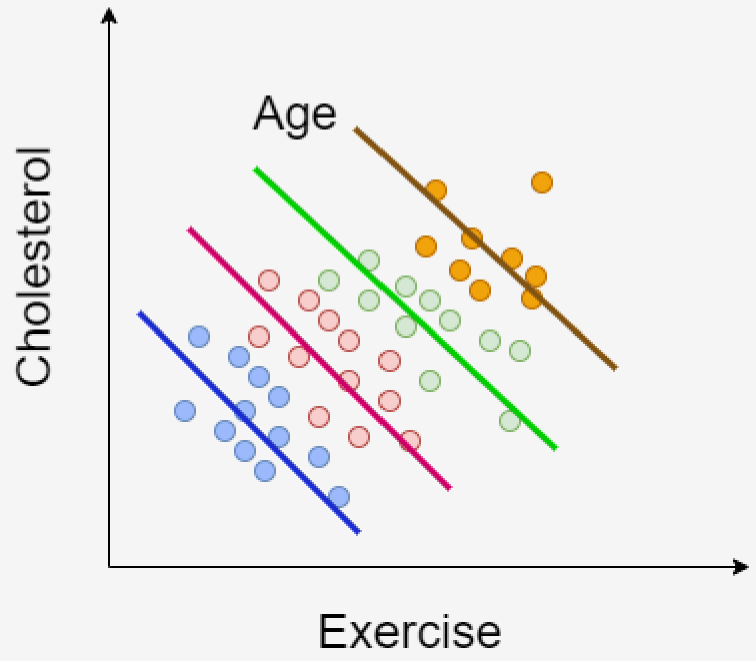
Causality: Simpson's Paradox

- Does exercise increase cholesterol?
- Introduce a new variable
- Can repeat this process again and again
- Is there any hope?



Causality: Simpson's Paradox

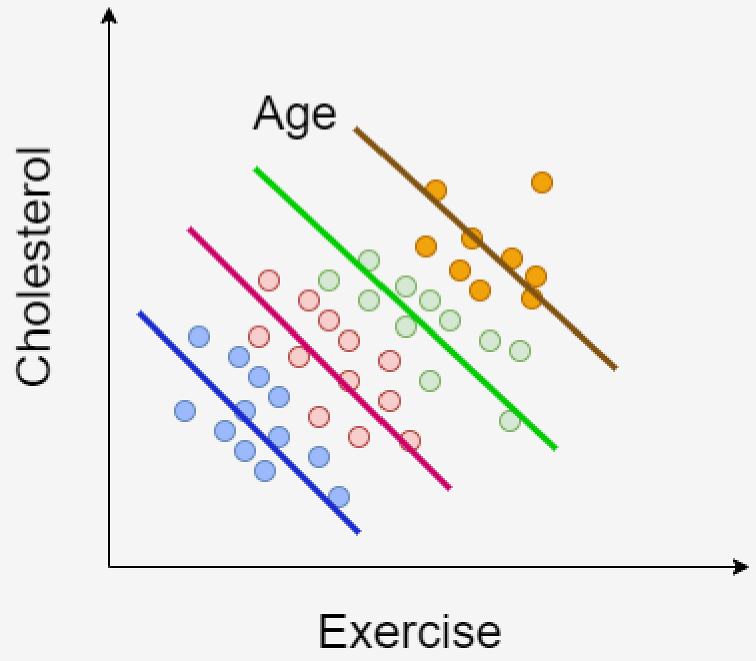
- Does exercise increase cholesterol?
- Introduce a new variable
- Can repeat this process again and again
- Is there any hope?



Causality: Simpson's Paradox

- Does exercise increase cholesterol?
- Introduce a new variable
- Can repeat this process again and again
- Is there any hope?

- Correlation is not causation
- Causal relationships are more stable
- Paradox arises because our brain is hard-wired to find causation
- Should be careful while making assumptions about the data

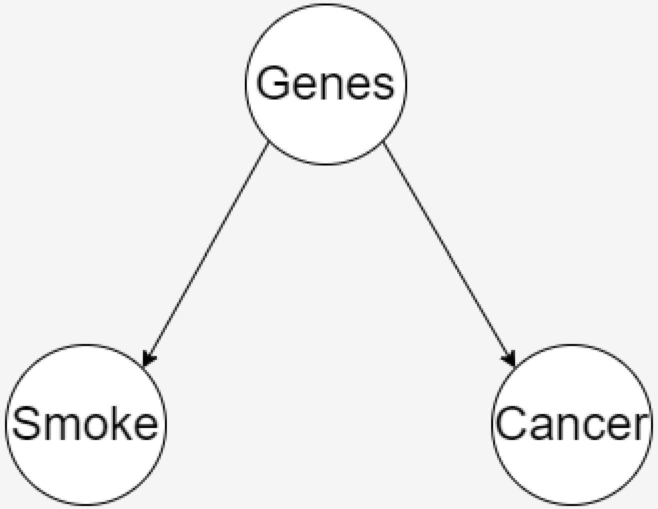


Causality: Fisher's Argument

Does smoking *cause* cancer?

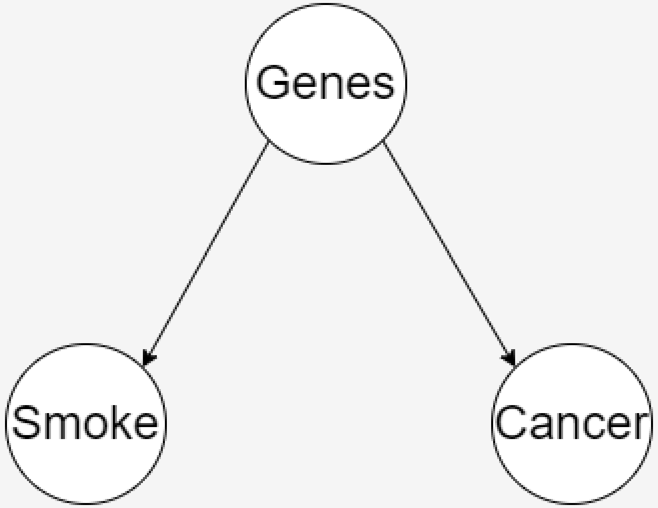
Causality: Fisher's Argument

Does smoking *cause* cancer?



Causality: Fisher's Argument

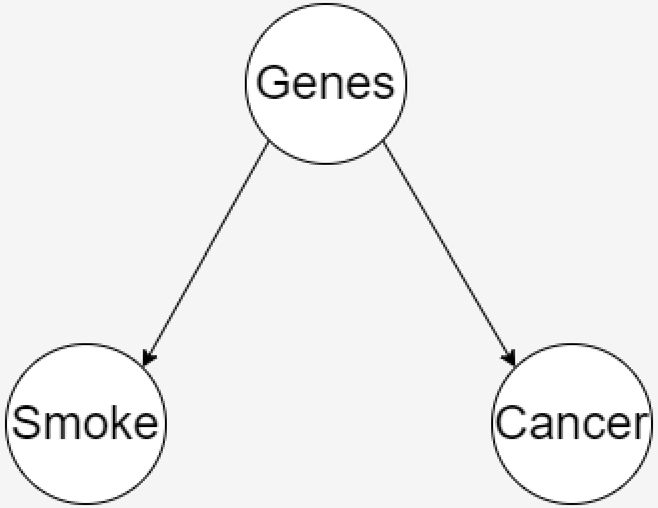
Does smoking *cause* cancer?



Way out?

Causality: Fisher's Argument

Does smoking *cause* cancer?



Way out?

Recall that causal relationships are more stable.

Randomly split people into 2 groups

Make one group smoke, another group not smoke

Observe the rates of cancer in these groups

Causality: Counterfactuals

This is not possible in practice

We need to somehow measure the effect of smoking on chance of cancer based on observational data

Ask the question: *Had this person been a non-smoker, what is the probability that the person would still get cancer, given everything else was same?* Area is called individual treatment effect estimation

If we are interested in aggregate effects, we study *average treatment effect*

Causality: Counterfactuals

Recall that causal relationships are more stable.

Randomly split people into 2 groups

Make one group smoke, another group not smoke

Observe the rates of cancer in these groups

This is not possible in practice

We need to somehow measure the effect of smoking on chance of cancer based on observational data

Ask the question: *Had this person been a non-smoker, what is the probability that the person would still get cancer, given everything else was same?* Area is called individual treatment effect estimation

If we are interested in aggregate effects, we study *average treatment effect*

Causality

What's new?

- Lots of open problems in classical causality
  - The identification problem: given only the observational distribution, can be identify the effect of an intervention?
  - Finite sample guarantees on these problems?
  - Designing estimators for counterfactuals from observation
- Causal Fairness
  - Can we take a causal look into fairness of decision-making?
  - Can we test for fairness efficiently if we know the causal structure?

Thank you for your time!

Feel free to reach me: nirjhardas@iisc.ac.in