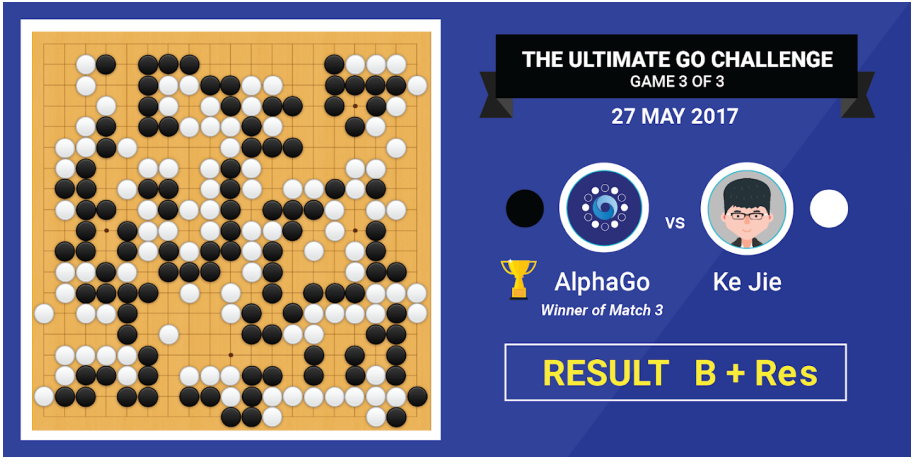
Statistical Inference in Reinforcement Learning

Chengchun Shi
Assistant Professor of Data Science
London School of Economics and Political Science

Developing AI with Reinforcement Learning



Reinforcement Learning Applications

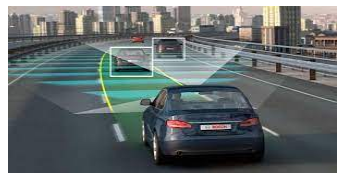
(a) Games

(b) Health Care

(c) Ridesharing

(d) Robotics

(e) Finance



(f) Automated Driving

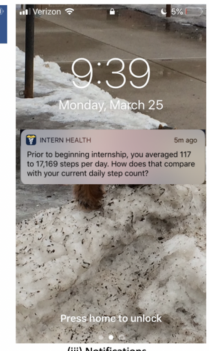
We focus on applications in **mobile health** (mHealth) and **ridesharing**

Applications in mHealth

- Use of cellphones and wearable devices in healthcare
- **Data**: Intern Health Study (NeCamp et al., 2020)
- **Subject**: First-year medical interns working in stressful environments (e.g., long work hours and sleep deprivation)
- **Objective**: Promote physical and mental well-beings
- **Intervention**: Determine whether to send certain text message to a subject

(i) App Dashboard

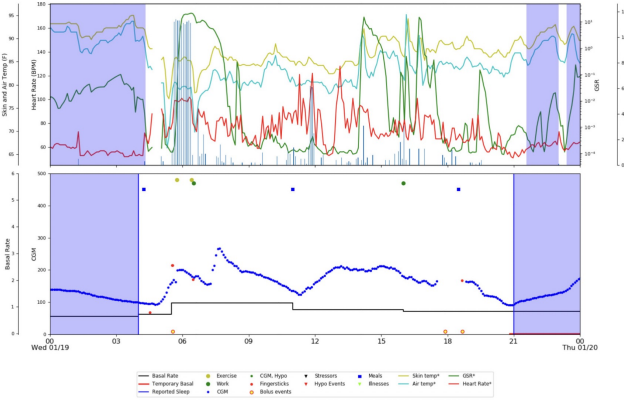
(ii) Mood EMA



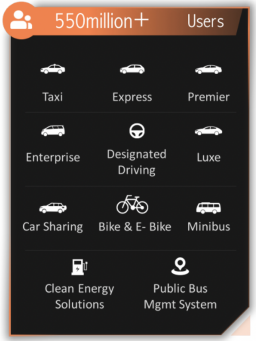
(iii) Notifications

Applications in mHealth (Cont'd)

- Management of **Type-I diabetes**
- **Subject**: Patients with Type-I diabetes
- **Intervention**: Determine whether a patient needs to **inject insulin or not** based on their glucose levels, food intake, exercise intensity
- **Data**: OhioT1DM dataset (Marling and Bunescu, 2018)



Applications in Ridesharing



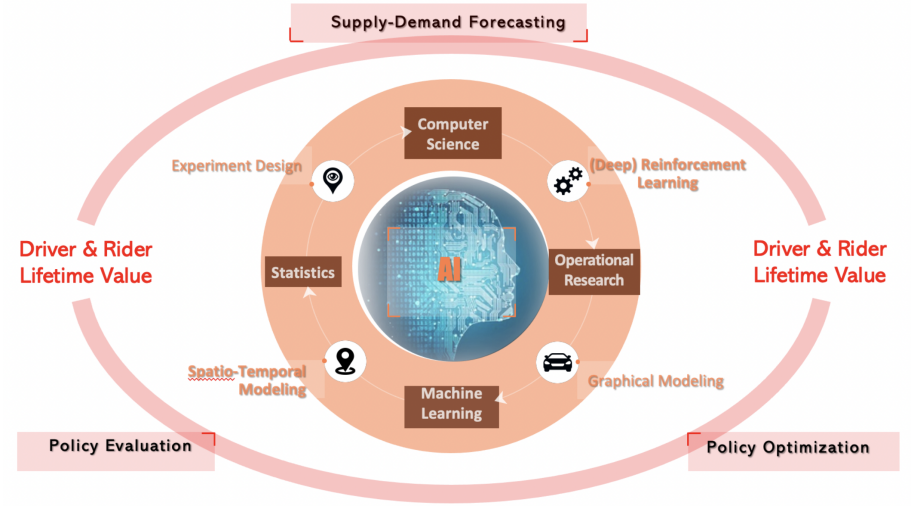
106TB+
vehicle trajectory data/day

4875TB+
data processed/day

40billion+
routing requests/day

15billion+
location points/day

Applications in Ridesharing (Cont'd)



In this talk, we will focus on ...

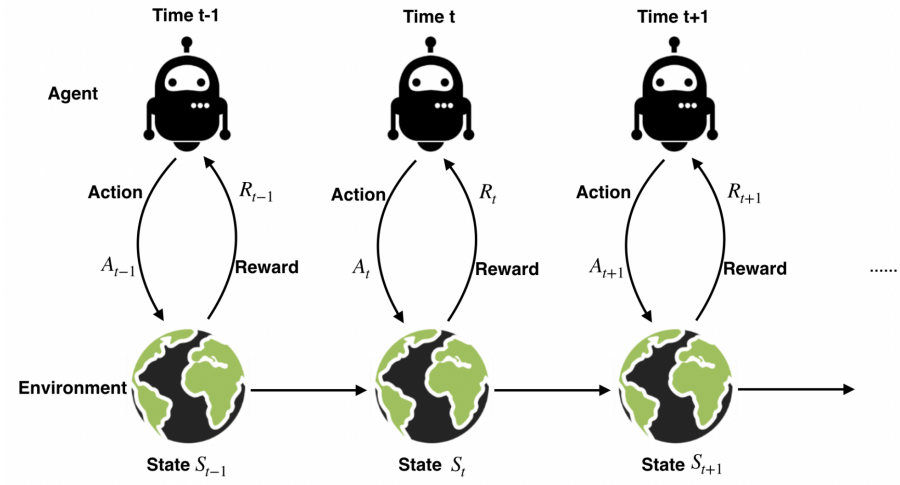
- **Statistical inference** in reinforcement learning (RL)
- Is statistical inference useful for RL?

Project I

Does the Markov Decision Process Fit the Data: Testing for the Markov Property in Sequential Decision Making

Joint work with Runzhe Wan, Wenbin Lu, Rui Song and Ling Leng
——ICML (2020)

Sequential Decision Making



Objective: find an optimal policy that maximizes the cumulative reward

The Agent's Policy

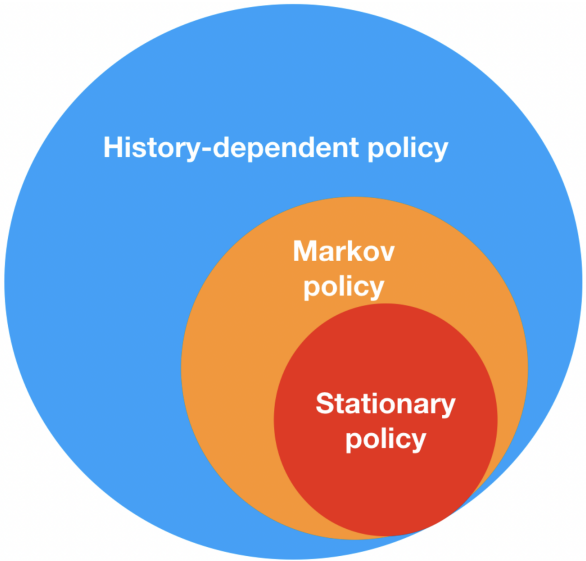
- The agent implements a **mapping** π_t from the observed data to a probability distribution over actions at each time step
- The collection of these mappings $\pi = \{\pi_t\}_t$ is called **the agent's policy**:

$$\pi_t(a|\bar{s}) = \Pr(A_t = a|\bar{S}_t = \bar{s}),$$

where $\bar{S}_t = (S_t, R_{t-1}, A_{t-1}, S_{t-1}, \cdots, R_0, A_0, S_0)$ is the set of **observed data history** up to time t.

- **History-Dependent** Policy: π_t depends on $\bar{S}_t$.
- **Markov** Policy: π_t depends on $\bar{S}_t$ only through S_t.
- **Stationary** Policy: π is Markov & π_t is **homogeneous** in t, i.e., $\pi_0 = \pi_1 = \cdots$.

The Agent's Policy (Cont'd)



Reinforcement Learning

- **RL algorithms**: trust region policy optimization (Schulman et al., 2015), deep Q-network (DQN, Mnih et al., 2015), asynchronous advantage actor-critic (Minh et al., 2016), quantile regression DQN (Dabney et al., 2018).
- **Foundations** of RL:
  - **Markov decision process** (MDP, Puterman, 1994): ensures the optimal policy is *stationary*, and is *not* history-dependent.
  - **Markov assumption** (MA): conditional on the present, the future and the past are independent,

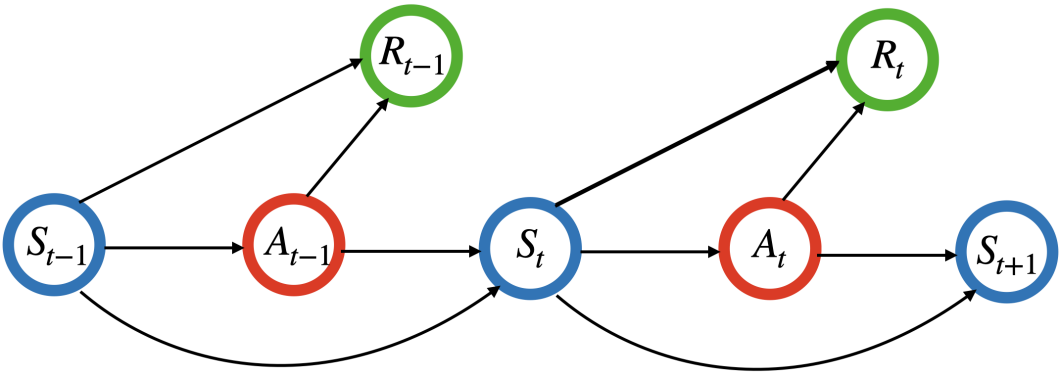
$$S_{t+1}, R_t \perp\!\!\!\perp \{(S_j, A_j, R_j)\}_{j<t} | S_t, A_t.$$

When R_t is a deterministic function of (S_t, A_t, S_{t+1})

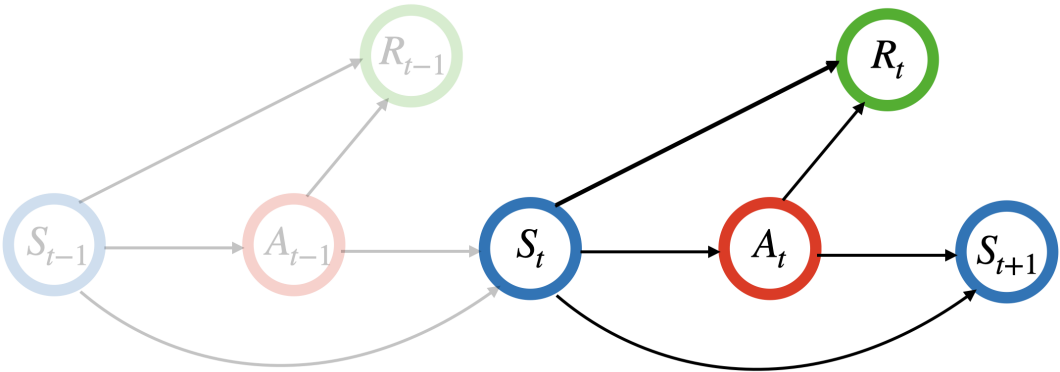
$$S_{t+1} \perp\!\!\!\perp \{(S_j, A_j)\}_{j<t} | S_t, A_t.$$

The Markov transition kernel is homogeneous in time

Markov Assumption



Markov Assumption



RL Models

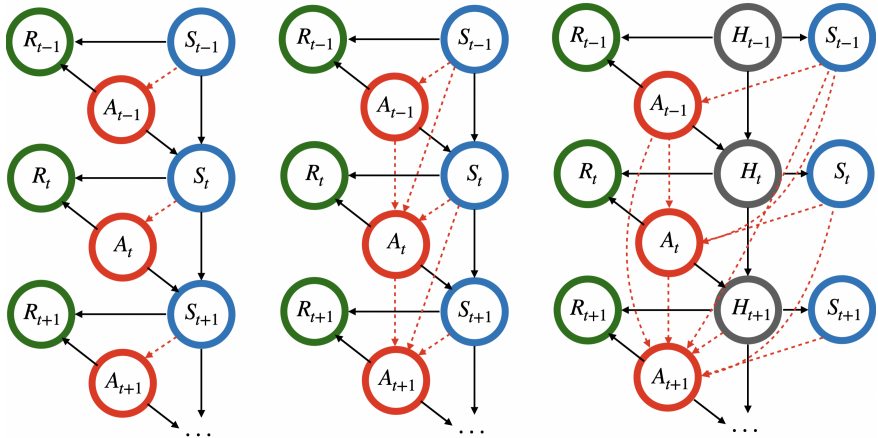


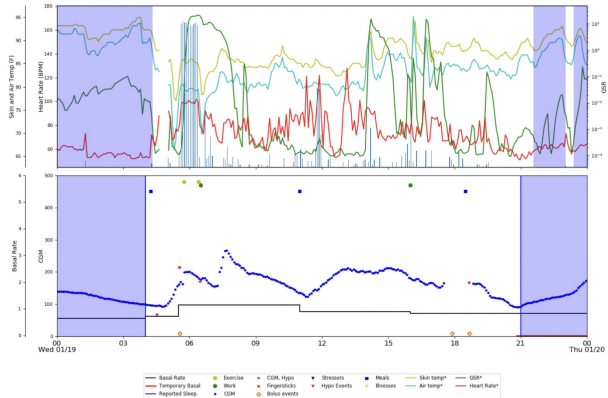
Figure: Causal diagrams for MDPs, HMDPs & POMDPs. The solid lines characterize the relationships among the variables and the dashed lines indicate the information needed to implement the optimal policy. $\{\boldsymbol{H_t}\}_{\boldsymbol{t}}$ denotes latent variables.

Contributions

- **Methodologically**
  - propose a **forward-backward learning** procedure to test MA
  - **first** work on developing consistent tests for MA in RL
  - sequentially apply the proposed test for RL **model selection** (e.g., test kth order MDP for $k = 1, 2, \cdots$)
  - critical to **offline** domains given a historical dataset **without online collection**:
    - For **under-fitted** models, any stationary policy is not optimal
    - For **over-fitted** models, the estimated policy might be very noisy due to the inclusion of many irrelevant lagged variables
- **Empirically**
  - identify the optimal policy in **high-order** MDPs
  - detect **partially observable** MDPs
- **Theoretically**
  - prove our test **controls type-I error** under a **bidirectional** asymptotic framework

Applications in High-Order MDPs

- **Data**: the OhioT1DM dataset
- Measurements for 6 patients with type I diabetes over 8 weeks.
- One-hour interval as a time unit.
- **State**: glucose levels, food intake, exercise intensity
- **Action**: to inject insulin or not.
- **Reward**: the Index of Glycemic Control (Rodbard, 2009).

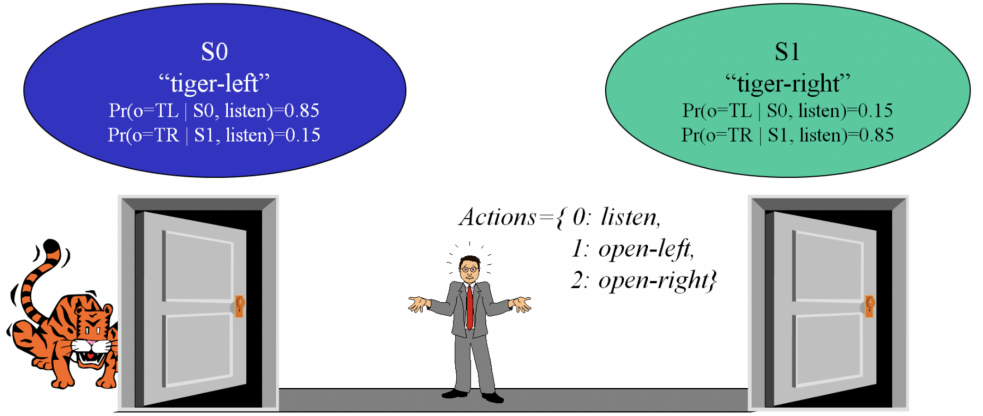


Applications in High-Order MDPs (Cont'd)

- **Analysis I**:
  - sequentially apply our test to determine the order of MDP
  - conclude it is a **fourth-order** MDP
- **Analysis II**:
  - split the data into training/testing samples
  - policy optimization based on **fitted-Q iteration**, by assuming it is a k-th order MDP for $k = 1, \cdots, 10$
  - policy evaluation based on **fitted-Q evaluation**
  - use **random forest** to model the Q-function
  - repeat the above procedure to compute the average value of policies computed under each MDP model assumption

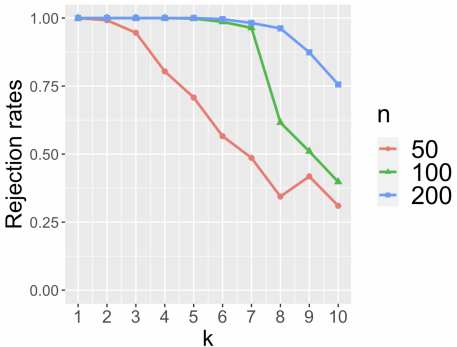
order	1	2	3	4	5	6	7	8	9	10
value	-90.8	-57.5	-63.8	**-52.6**	-56.2	-60.1	-63.7	-54.9	-65.1	-59.6

Applications in Partially Observable MDPs

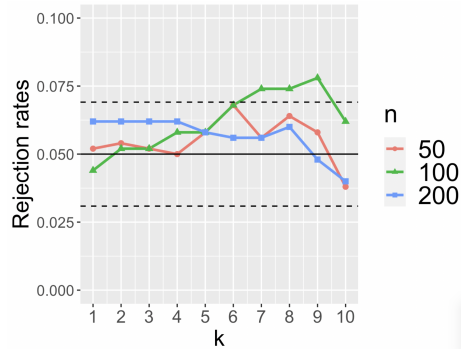


Applications in Partially Observable MDPs (Cont'd)

- Under $\mathcal{H}_1$ (MA is violated, alternative). Significance level = 0.05.



- Under $\mathcal{H}_0$ (MA holds, null). Significance level = 0.05.



Methodology

- **First** work to test MA in RL
- Existing approach in time series: Cheng and Hong (2012)
  - characterize MA based on the notion of **conditional characteristic function** (CCF)
  - use local polynomial regression to estimate CCF
- **Challenge**:
  - develop a valid test for MA in **moderate or high-dimensions**
  - the dimension of the state increases as we concatenate measurements over multiple time points in order to test for a high-order MDP.
- This motivates our **forward-backward learning** procedure.

Methodology (Cont'd)

Some key components of our algorithm:

- To deal with moderate or high-dimensional state space, employ modern machine learning (ML) algorithms to estimate CCF:
  - Learn CCF of $\boldsymbol{S_{t+1}}$ given $\boldsymbol{A_t}$ and $\boldsymbol{S_t}$ (**forward learner**)
  - Learn CCF of ($\boldsymbol{S_t}$, $\boldsymbol{A_t}$) given ($\boldsymbol{S_{t+1}}$, $\boldsymbol{A_{t+1}}$) (**backward learner**)
  - Develop a **random forest**-based algorithm to estimate CCF
  - Borrow ideas from the quantile random forest algorithm (Meinshausen, 2006) to facilitate the computation
- To alleviate the bias of ML algorithms, construct **doubly-robust** test statistics by integrating forward and backward learners;
- To improve the power, consider a **maximum-type** test statistic;
- To control the type-I error, approximate the distribution of our test via **high-dimensional multiplier bootstrap** (Chernozhukov, et al., 2014).

Bidirectional Theory

- $\boldsymbol{N}$ the number of trajectories
- $\boldsymbol{T}$ the number of decision points per trajectory
- **bidirectional asymptotics**: a framework allows either $\boldsymbol{N}$ or $\boldsymbol{T} \to \infty$
- large $\boldsymbol{N}$, small $\boldsymbol{T}$ (Intern Health Study)

- small $\boldsymbol{N}$, large $\boldsymbol{T}$ (OhioT1DM dataset)

- large $\boldsymbol{N}$, large $\boldsymbol{T}$ (games)

Bidirectional Theory (Cont'd)

(C1) Actions are generated by a fixed behavior policy.

(C2) The observed data is exponentially β-mixing.

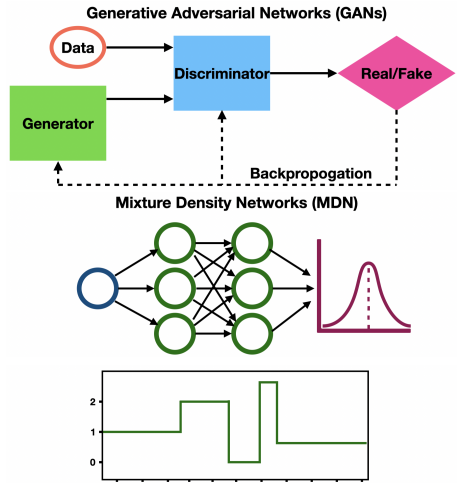
(C3) The ℓ_2 prediction errors of forward and backward learners converge at a rate faster than $(\boldsymbol{NT})^{-\mathbf{1/4}}$.

Theorem

Assume (C1)-(C3) hold. Then under some other mild conditions, our test controls the type-I error asymptotically as either $\boldsymbol{N}$ *or* $\boldsymbol{T}$ *diverges to* ∞.

Some Follow-ups

- **Double GANs** for conditional independence testing (*JMLR, 2021*)
- Testing DAGs via supervised, structural learning and **GANs** (*JASA, revised*)
- Testing Markovanity in time series via **deep generative learning** (*JRSSB, revised*)
  - Derive the convergence rate of **MDN**
- Testing **stationarity** and **changepoint detection** in RL (*AOS, submitted*)
  - Our test helps identify a better policy in the **Intern Health Study**

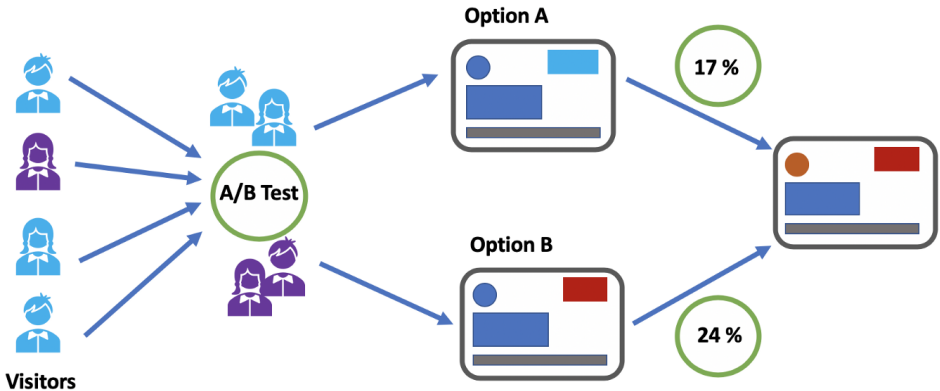


Project II

Dynamic Causal Effects Evaluation in A/B Testing with a Reinforcement Learning Framework

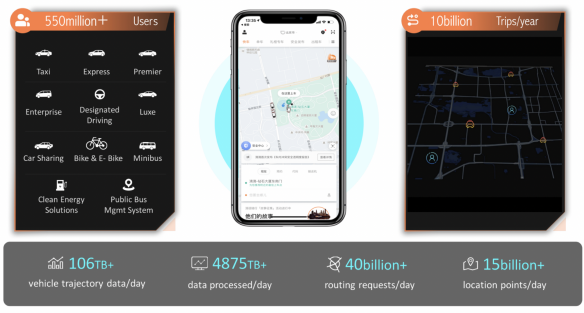
Joint work with Xiaoyu Wang, Shikai Luo, Hongtu Zhu, Jieping Ye and Rui Song
——JASA

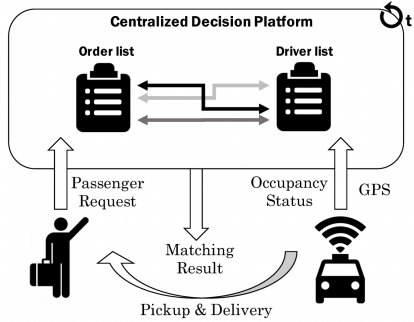
A/B Testing



Taken from
https://towardsdatascience.com/how-to-conduct-a-b-testing-3076074a8458

Motivation: Order Dispatch

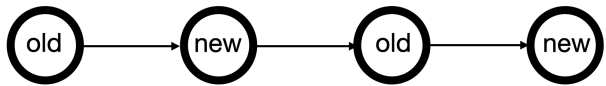




Our project is motivated by the need for comparing the **long-term rewards** of different **order dispatching** policies in **ridesharing platforms**

Challenges

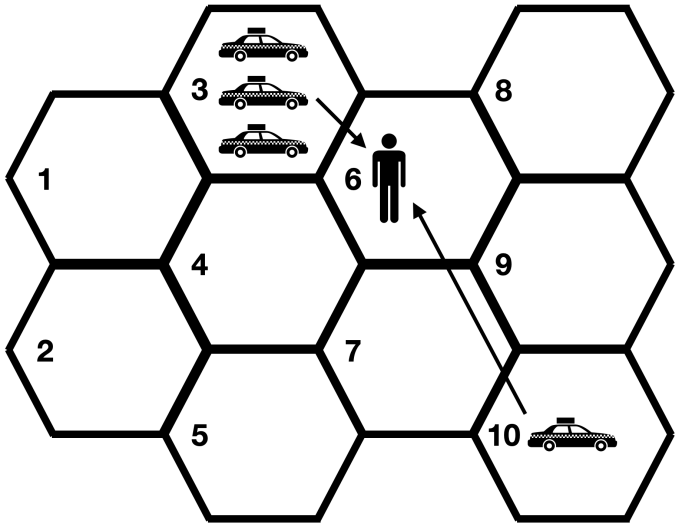
1. The existence of **carryover effects**:
   - Under the alternating-time-interval design



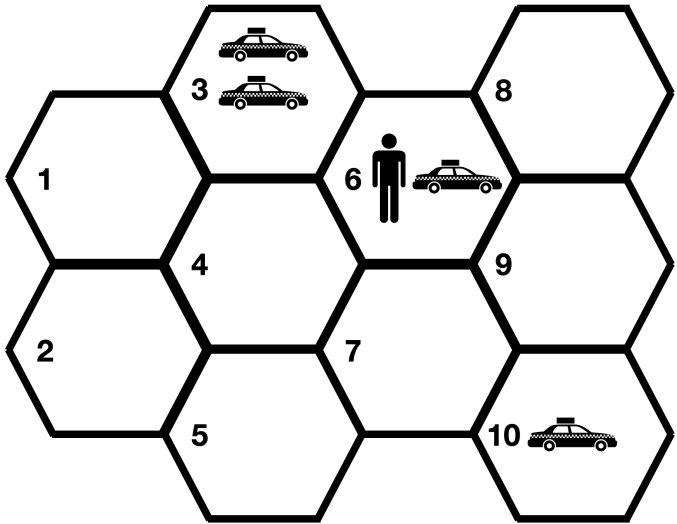
   - Past actions will affect future outcomes
2. The need for **early termination**:
   - Each experiment takes a considerable time (at most 2 weeks)
   - Early termination to save time and budget
3. The need for **adaptive randomization**:
   - Maximize the total reward (e.g., epsilon-greedy)
   - Detect the alternative faster

To our knowledge, **no** existing test has addressed three challenges simultaneously

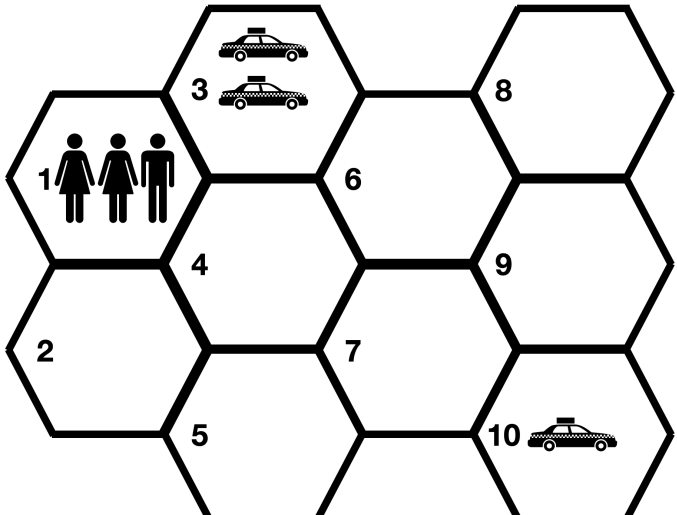
Illustration of the Carryover Effects



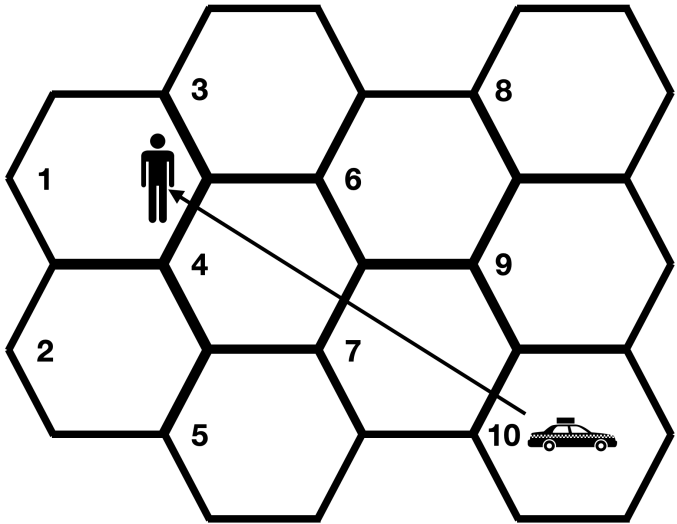
Adopting the Closest Driver Policy



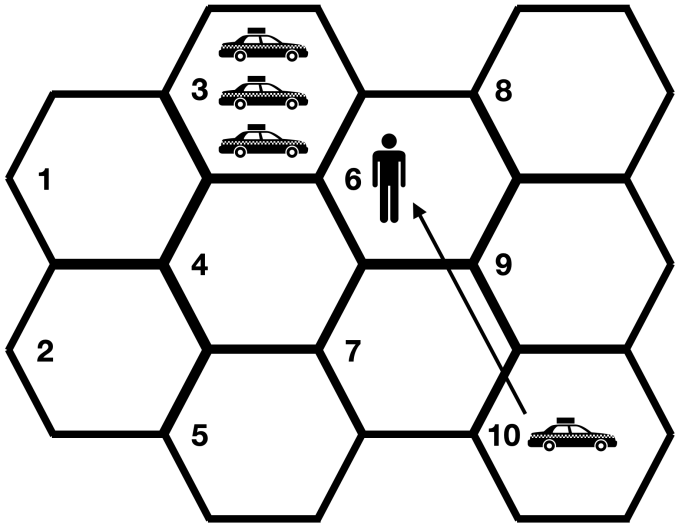
Some Time Later · · ·



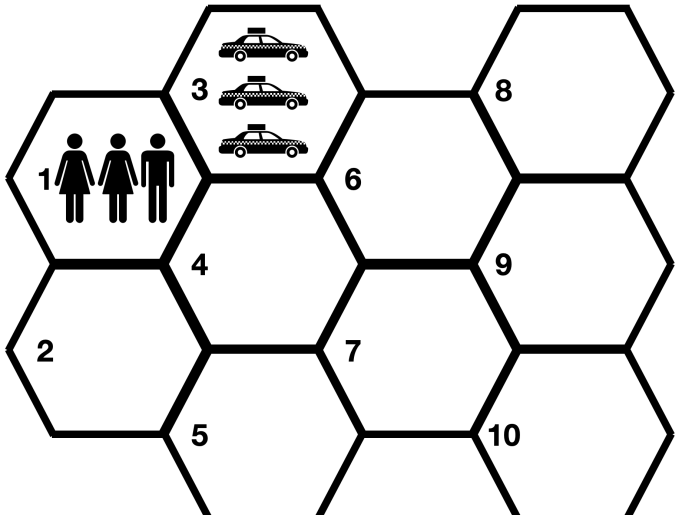
Miss One Order



Consider a Different Action



Able to Match All Orders



Existence of Carryover Effects

past actions $\rightarrow$ distribution of drivers $\rightarrow$ future rewards

Limitations of Existing A/B tests

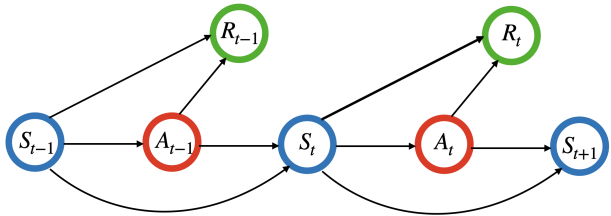
- Most existing tests **cannot** detect carryover effects
- **Example 1**. $S_t \sim N(0, 0.25)$, $R_t = S_t + \delta A_t$
- **Example 2**. $S_t = 0.5 S_{t-1} + \delta A_{t-1} + N(0, 0.25)$, $R_t = S_t$
- $\mathcal{H}_0$: The old policy ($A = 0$) has larger cumulative rewards ($\delta \leq 0$)
- $\mathcal{H}_1$: The new policy ($A = 1$) has larger cumulative rewards ($\delta > 0$)

Table: Powers of t-test, DML-based test (Chernozhukov et al., 2018) and the proposed test with $T = 500, \delta = 0.1$ ($\mathcal{H}_1$ holds in both examples)

Example 1	t-test 0.76	DML-based test **1.00**	our test **0.98**
Example 2	t-test 0.04	DML-based test 0.06	our test **0.73**

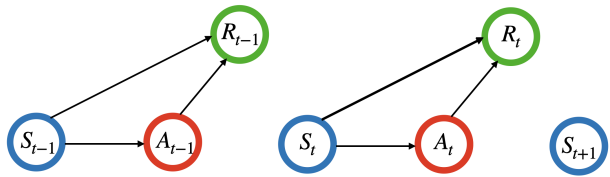
Contributions and Advances of Our Proposal

- Introduce an RL framework for A/B testing



1. A_{t-1} impacts R_t indirectly through its effect on S_t
2. S_t shall include important **mediators** between A_{t-1} and R_t

- Most existing works require the independence assumption



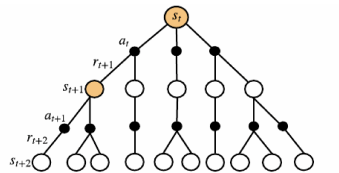
Contributions and Advances (Cont'd)

Propose a test procedure for comparing long-term rewards of two policies

1. allows for **sequential monitoring**
2. allows for **online updating**
3. applicable to a wide range of designs, including the **Markov** design, **alternating-time-interval** design and **adaptive** design

Methodology

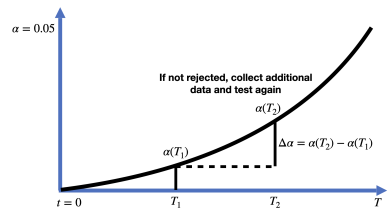
- Apply temporal difference (**TD**) **learning** with **sieve** method for value evaluation



$$TD: v(s_t) \leftarrow v(s_t) + \eta(r_{t+1} + \lambda v(s_{t+1}) - v(s_t))$$

- Provide **uncertainty quantification** (Shi et al., 2022, JRSSB)

- Adopt the α-**spending approach** (Lan & DeMets, 1983) for sequential monitoring



- Develop a **bootstrap-assisted procedure** for determining the stopping boundary[a]

[a] The numerical integration method designed for classical sequential tests is **not** applicable in adaptive design, due to the carryover effects

Theory

Theorem (Validity and Consistency)

Under the Markov, alternating-time-interval or adaptive design, the proposed test can **control type-I error** *and can* **detect** *local alternative hypotheses.*

Theorem (Undersmoothing and Efficiency)

Suppose **sieve** *method is used for function approximation in temporal difference learning.*

1. **Undersmoothing** *is not needed to guarantee that the policy value estimator has a tractable limiting distribution.*
2. *The final policy value estimator is* **semiparametrically efficient**.

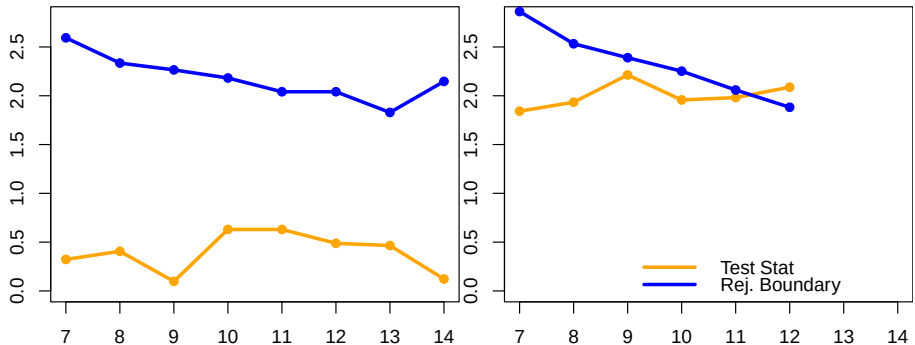
- The bias of the policy value estimator decays at a faster rate than the pointwise bias of the sieve estimator (Shen 1997; Newey et al, 1998)
- The proposed test will **not** be overly sensitive to the number of basis functions
- **Cross-validation** can be employed to select the basis functions

Application to Ridesharing Platform

- **Data**: a given city from December 3rd to 16th (two weeks)
- **30 minutes** as one time unit, sample size = **672**
- **State**:
  1. number of drivers (supply)
  2. number of requests (demand)
  3. supply and demand equilibrium metric (mediator)
- **Action**: new policy $\boldsymbol{A = 1}$ v.s. old $\boldsymbol{A = 0}$
- **Reward**: drivers' income
- The new policy is expected to have **better** performance

Application to Ridesharing Platform (Cont'd)

- The proposed test



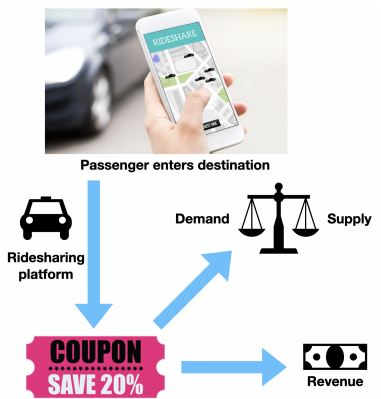
(a) AA Experiment: Day

(b) AB Experiment: Day

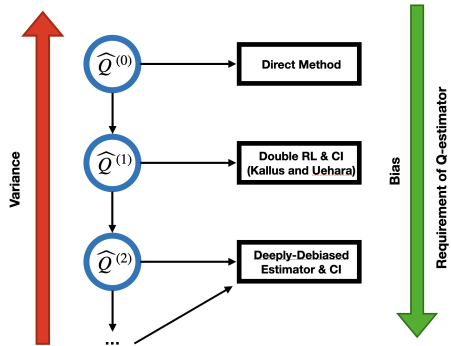
- t-test: **fail** to reject $\mathcal{H}_0$ in A/B experiment with p-value 0.18

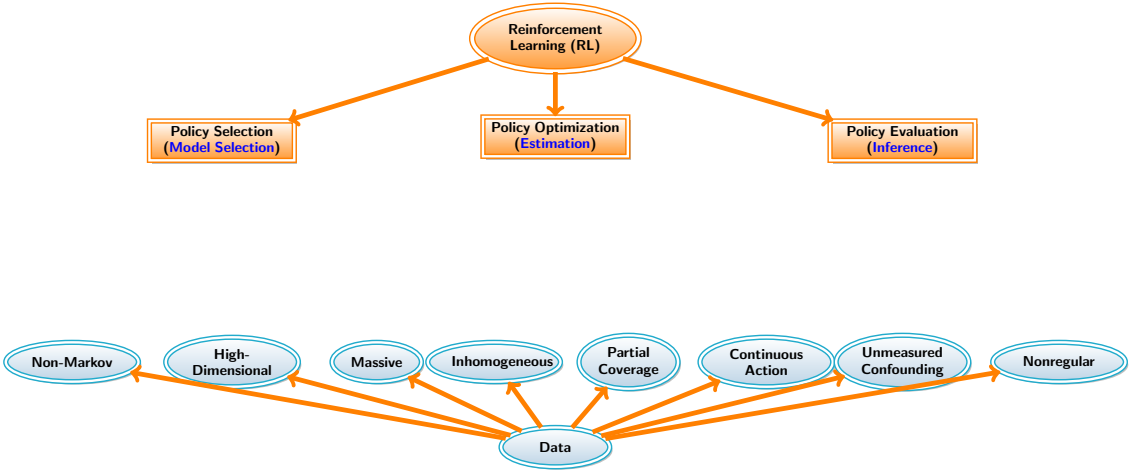
Some Follow-ups

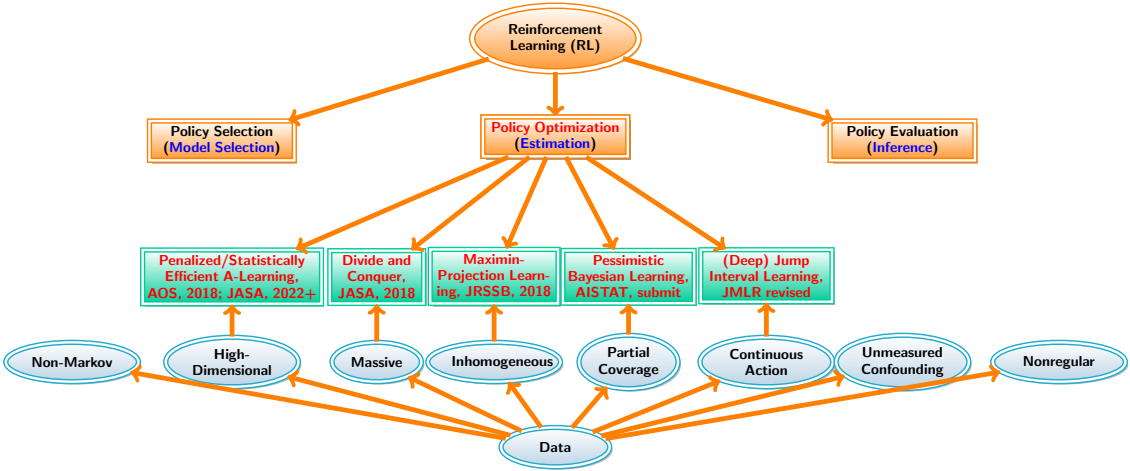
- A **multi-agent RL** framework for policy evaluation (*AOAS, 2022+*)

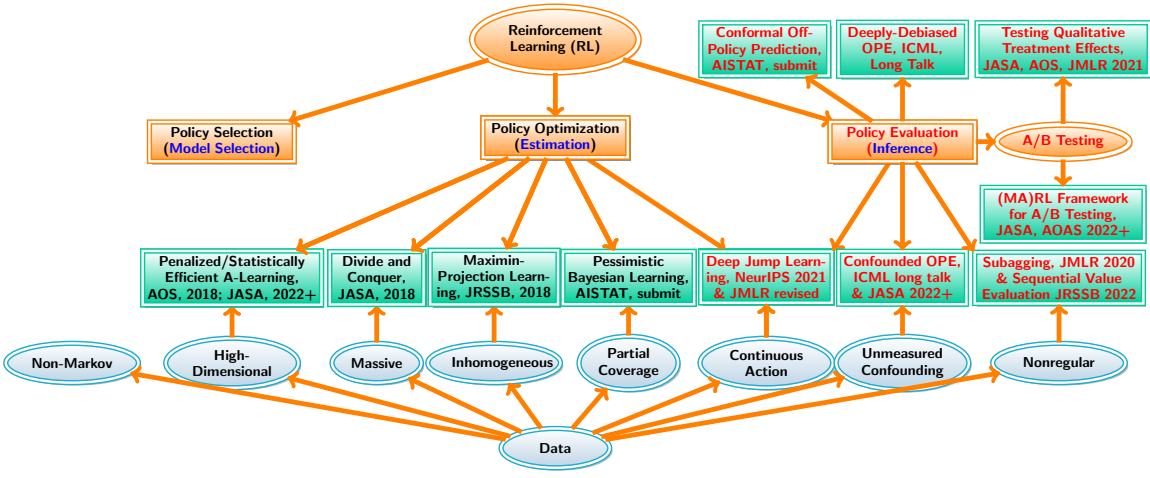


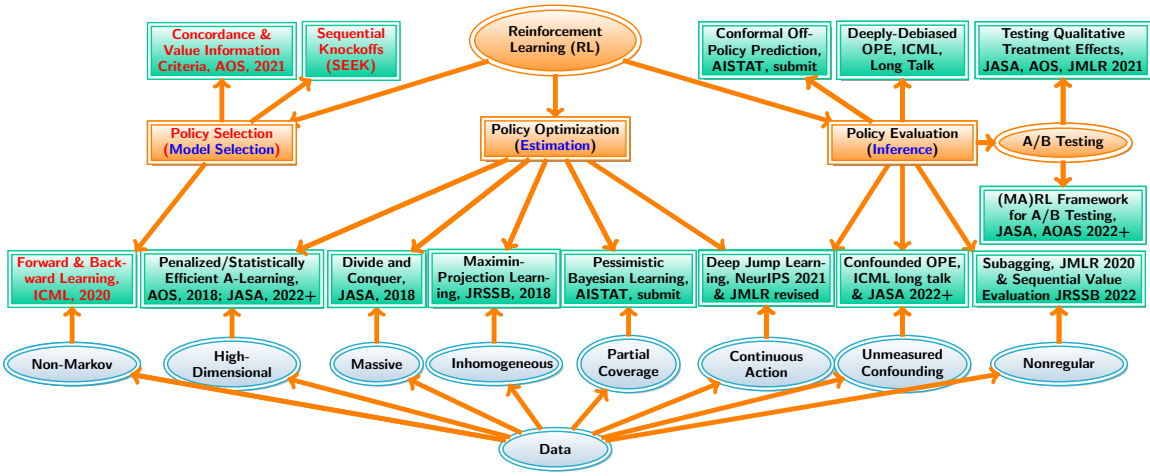
- **Deeply-debiased** off-policy **confidence interval** estimation (*ICML, 2021*, Long Talk, top 3% of submissions)

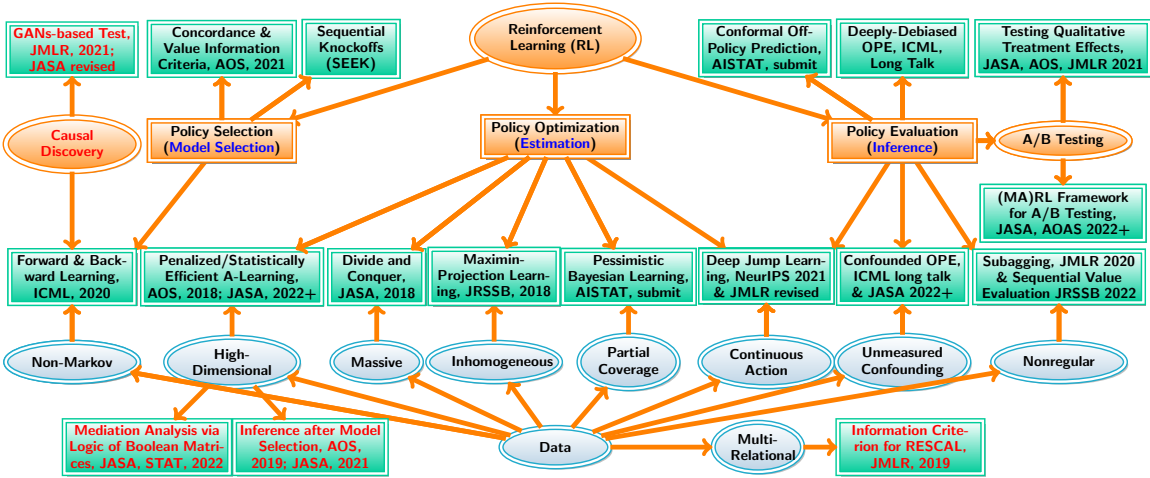



Reinforcement Learning (RL)
Policy Selection (Model Selection)
Policy Optimization (Estimation)
Policy Evaluation (Inference)
Non-Markov
High-Dimensional
Massive
Inhomogeneous
Partial Coverage
Continuous Action
Unmeasured Confounding
Nonregular
Data


Reinforcement Learning (RL)
Policy Selection (Model Selection)
Policy Optimization (Estimation)
Policy Evaluation (Inference)
Penalized/Statistically Efficient A-Learning, AOS, 2018; JASA, 2022+
Divide and Conquer, JASA, 2018
Maximin-Projection Learning, JRSSB, 2018
Pessimistic Bayesian Learning, AISTAT, submit
(Deep) Jump Interval Learning, JMLR revised
Non-Markov
High-Dimensional
Massive
Inhomogeneous
Partial Coverage
Continuous Action
Unmeasured Confounding
Nonregular
Data


Reinforcement Learning (RL)
Conformal Off-Policy Prediction, AISTAT, submit
Deeply-Debiased OPE, ICML, Long Talk
Testing Qualitative Treatment Effects, JASA, AOS, JMLR 2021
Policy Selection (Model Selection)
Policy Optimization (Estimation)
Policy Evaluation (Inference)
A/B Testing
(MA)RL Framework for A/B Testing, JASA, AOAS 2022+
Penalized/Statistically Efficient A-Learning, AOS, 2018; JASA, 2022+
Divide and Conquer, JASA, 2018
Maximin-Projection Learning, JRSSB, 2018
Pessimistic Bayesian Learning, AISTAT, submit
Deep Jump Learning, NeurIPS 2021 & JMLR revised
Confounded OPE, ICML long talk & JASA 2022+
Subagging, JMLR 2020 & Sequential Value Evaluation JRSSB 2022
Non-Markov
High-Dimensional
Massive
Inhomogeneous
Partial Coverage
Continuous Action
Unmeasured Confounding
Nonregular
Data


Concordance & Value Information Criteria, AOS, 2021
Sequential Knockoffs (SEEK)
Reinforcement Learning (RL)
Conformal Off-Policy Prediction, AISTAT, submit
Deeply-Debiased OPE, ICML, Long Talk
Testing Qualitative Treatment Effects, JASA, AOS, JMLR 2021
Policy Selection (Model Selection)
Policy Optimization (Estimation)
Policy Evaluation (Inference)
A/B Testing
(MA)RL Framework for A/B Testing, JASA, AOAS 2022+
Forward & Backward Learning, ICML, 2020
Penalized/Statistically Efficient A-Learning, AOS, 2018; JASA, 2022+
Divide and Conquer, JASA, 2018
Maximin-Projection Learning, JRSSB, 2018
Pessimistic Bayesian Learning, AISTAT, submit
Deep Jump Learning, NeurIPS 2021 & JMLR revised
Confounded OPE, ICML long talk & JASA 2022+
Subagging, JMLR 2020 & Sequential Value Evaluation JRSSB 2022
Non-Markov
High-Dimensional
Massive
Inhomogeneous
Partial Coverage
Continuous Action
Unmeasured Confounding
Nonregular
Data


GANs-based Test, JMLR, 2021; JASA revised
Concordance & Value Information Criteria, AOS, 2021
Sequential Knockoffs (SEEK)
Reinforcement Learning (RL)
Conformal Off-Policy Prediction, AISTAT, submit
Deeply-Debiased OPE, ICML, Long Talk
Testing Qualitative Treatment Effects, JASA, AOS, JMLR 2021
Causal Discovery
Policy Selection (Model Selection)
Policy Optimization (Estimation)
Policy Evaluation (Inference)
A/B Testing
(MA)RL Framework for A/B Testing, JASA, AOAS 2022+
Forward & Backward Learning, ICML, 2020
Penalized/Statistically Efficient A-Learning, AOS, 2018; JASA, 2022+
Divide and Conquer, JASA, 2018
Maximin-Projection Learning, JRSSB, 2018
Pessimistic Bayesian Learning, AISTAT, submit
Deep Jump Learning, NeurIPS 2021 & JMLR revised
Confounded OPE, ICML long talk & JASA 2022+
Subagging, JMLR 2020 & Sequential Value Evaluation JRSSB 2022
Non-Markov
High-Dimensional
Massive
Inhomogeneous
Partial Coverage
Continuous Action
Unmeasured Confounding
Nonregular
Mediation Analysis via Logic of Boolean Matrices, JASA, STAT, 2022
Inference after Model Selection, AOS, 2019; JASA, 2021
Data
Multi-Relational
Information Criterion for RESCAL, JMLR, 2019

Thank You!

☺Papers and softwares can be found on my personal website

`callmespring.github.io`

Appendix: Project I – Forward-Backward Learning

- $\mathcal{H}_0$: For any t, S_{t+1} is independent of the past state-action pairs given (S_t, A_t).
- $\mathcal{H}_1$: There exists some t, S_{t+1} is dependent of past state-action pairs given (S_t, A_t).
- **Forward** CCF: $\varphi^*(\mu|s, a) = \mathbb{E}[\exp(i\mu^\top S_{t+1})|S_t = s, A_t = a]$
- **Backward** CCF: $\psi^*(\nu|s, a) = \mathbb{E}[\exp(i\nu^\top(S_t, A_t))|S_{t+1} = s, A_{t+1} = a]$[1]
- For a given lag $q \geq 1$, μ and ν, define $\Gamma(q, \mu, \nu)$ as

$$\mathbb{E}[\exp(i\mu^\top S_{t+q+1}) - \varphi(\mu|S_{t+q}, A_{t+q})][\exp(i\nu^\top(S_t, A_t)) - \psi(\nu|S_{t+1}, A_{t+1})].$$

- $\Gamma(q, \mu, \nu)$ measures the **weak conditional independence** between S_{t+q+1} and (S_t, A_t) given $S_{t+1}, A_{t+1}, S_{t+q}, A_{t+q}$.
- Given certain **forward learner** $\widehat{\varphi}$ and **backward learner** $\widehat{\psi}$, an empirical estimator $\widehat{\Gamma}(q, \mu, \nu)$ can be constructed to detect the deviation from $\mathcal{H}_0$.
- **Test statistic**: $\max_{b=1,\cdots,B} \max_{q=1,\cdots,Q} \max(|\mathrm{Re}(\widehat{\Gamma}(q, \mu_b, \nu_b))|, |\mathrm{Im}(\widehat{\Gamma}(q, \mu_b, \nu_b))|)$. Page 22

[1] ψ is independent of t when the data generating process is stationary

Project I – Forward-Backward Learning (Cont'd)

- **Fact 1**: **Double robustness**. Under $\mathcal{H}_0$, $\Gamma(\boldsymbol{q}, \boldsymbol{\mu}, \boldsymbol{\nu}) = \mathbf{0}$ when $\varphi = \varphi^*$ or $\psi = \psi^*$.
  - The doubly-robust property offers protection against **model-misspecification** of CCFs.
  - In addition, it guarantees each $\widehat{\Gamma}(\boldsymbol{q}, \boldsymbol{\mu}, \boldsymbol{\nu})$ is **asymptotically normal** even when the estimated CCFs converge slower than the parametric rate.
  - Allow the use model ML algorithms to learn CCFs to handle **high-dimensionality**.

- **Fact 2**: **Martingale structure**. Under $\mathcal{H}_0$, each $\widehat{\Gamma}(\boldsymbol{q}, \boldsymbol{\mu}, \boldsymbol{\nu})$ corresponds to a sum of martingale difference sequence.
  - Allow us to treat the summands as if they were i.i.d so that existing **high-dimensional multiplier bootstrap** methods (Chernozhukov et al., 2013; 2014) are applicable to conduct statistical inference.

Project II – RL for A/B Testing

- (State) value function under a policy π starting from a given initial state s:

$$V^{\pi}(s) = \sum_{t=1}^{+\infty} \gamma^{t} \mathbb{E}^{\pi}(R_t | S_0 = s),$$

  where γ denotes a discounted factor $\in [0, 1)$.

- We focus on two nondynamic policies $\mathbf{1}$, $\mathbf{0}$ that assign action 1 (or 0) all the time:

$$\text{Conditional Average Treatment Effect (CATE)} = V^{1}(s) - V^{0}(s),$$
$$\text{Average Treatment Effect (ATE)} = \int_{s} \text{CATE}(s) \mathbb{G}(ds),$$

  where $\mathbb{G}$ denotes a reference initial state distribution function.

- $\mathcal{H}_0 : \text{ATE} \leq 0$ v.s. $\mathcal{H}_1 : \text{ATE} > 0$

Project II – RL for A/B Testing (Cont'd)

- **Input**: time points $\{T_k\}_{k=1}^K$ when the interim analyses will be conducted, a set of basis functions ϕ, number of bootstrap samples B and an α-spending function $\alpha(\bullet)$.
- **For** $k = 1$ to K:
  - **Step 1**. Online update of the value estimator:
    - **For** $t = T_{k-1}$ to $T_k - 1$, update the **Bellman equation** based on the data tuple (S_t, A_t, R_t, S_{t+1}) and the summary statistics at the $(k-1)$th interim stage
    - Solve the Bellman equation to compute the sieve estimator $\widehat{\beta}$, the value function estimator $\phi^\top(s,a)\widehat{\beta}$ and the ATE estimator $\widehat{\tau}$ as a linear combination of $\widehat{\beta}$
  - **Step 2**. Online update of the variance estimator:
    - **For** $t = T_{k-1}$ to $T_k - 1$, update the covariance matrix estimator of $\widehat{\beta}$ based on the tuple (S_t, A_t, R_t, S_{t+1}), the summary statistics at the $(k-1)$th interim stage and $\widehat{\beta}$
    - Compute the variance estimator of $\widehat{\tau}$ based on the covariance matrix estimator of $\widehat{\beta}$
  - **Step 3**. Bootstrap test statistic:
    - **For** $b = 1, \cdots, B$, update the bootstrapped statistic Z^b based on the (co)variance estimators obtained at step 2 and a random error $e^b \sim N(0, I)$
    - Set the critical value z to the $[\alpha(T_k) - |\mathcal{I}^c|/B]/[1 - |\mathcal{I}^c|/B]$th quantile of $\{Z^b\}_b$
    - Set $\mathcal{I} \to \{b \in \mathcal{I} : Z^b \leq z\}$
  - **Step 4**. Reject $\mathcal{H}_0$ if $\sqrt{T_k}\widehat{\tau}/\widehat{\sigma} > z$

Project II – RL for A/B Testing (Cont'd)

- ATE corresponds to the difference between two **policy value functions**,

$$\text{ATE} = \int_{\boldsymbol{s}} \boldsymbol{V^1}(\boldsymbol{s})\mathbb{G}(d\boldsymbol{s}) - \int_{\boldsymbol{s}} \boldsymbol{V^0}(\boldsymbol{s})\mathbb{G}(d\boldsymbol{s}).$$

- We apply the **sieve** method to estimate $\boldsymbol{V^1}$, $\boldsymbol{V^0}$, and plug-in these estimators to estimate the policy value and the ATE
- In **supervised learning**, when estimating a **smooth** functional of a regression function, plug-in sieve estimator is asymptotically normal and efficient even when a **minimax rate optimal** regression estimator is plugged in (Shen, 1997).
  - Bias of regression estimator provides a **loose** bound for that of the functional estimator
- We extend the aforementioned results to **reinforcement learning**
  - The plug-in sieve estimator equals a **doubly robust** estimator where both nuisance functions are computed via sieve estimation

Project III: Test MA via Deep Generative Learning

- Two key components in **mixture density network** (MDN)
  1. use a **conditional Gaussian mixture model** to approximate the underlying distribution function f
  2. use **deep neural networks** to parametrize the conditional mean and (co)variance functions
- The first work to establish the **convergence rate** of MDN
  1. When f follows a finite conditional Gaussian mixture distribution: $n^{-\beta/(2\beta+d)}$
  2. When f follows an infinite conditional Gaussian mixture model: $n^{-2\beta^2/[(2\beta+d)(3\beta+d)]}$
  3. More generally, when f is Lipschitz continuous: $n^{-\beta^2/[(2\beta+d)(6\beta+d)]}$
- Provide a sharp upper bound on the approximation error of MDN when f follows an infinite conditional Gaussian mixture model (sharper than existing results)

Project IV: Test Stationarity in RL

Table 1. Examples of 6 different groups of notifications.

Notification groups	Life insight	Tip
Mood	Your mood has ranges from 7 to 9 over the past 2 weeks. The average intern's daily mood goes down by 7.5% after intern year begins.	Treat yourself to your favorite meal. You've earned it!
Activity	Prior to beginning internship, you averaged 117 to 17,169 steps per day. How does that compare with your current daily step count?	Exercising releases endorphins which may improve mood. Staying fit and healthy can help increase your energy level.
Sleep	The average nightly sleep duration for an intern is 6 hours 42 minutes. Your average since starting internship is 7 hours 47 minutes.	Try to get 6 to 8 hours of sleep each night if possible. Notice how even small increases in sleep may help you to function at peak capacity & better manage the stresses of internship.

Page 4

Project V: Deeply-debiased OPE

- Evaluate a target policy **offline** using historical data generated from a different behavior policy and construct a **confidence interval** for the target policy's value
- Consider the RL (e.g., MDP) setting
- **Main idea**: Develop a **deeply-debiasing** process using higher order influence function (Robins et al., 2017)
- Deeply-debiasing guarantees the aggregated value estimator's bias is much smaller than its standard deviation so that the resulting confidence interval is valid
- Our proposal is
  - **robust**: more robust than existing doubly robust methods
  - **efficient**: achieve the semiparametric efficiency as doubly robust methods
  - **flexible**: requires much weaker and practically more feasible conditions than doubly robust methods

Project VI: Mediation Analysis via LOGAN

- propose a new testing procedure to evaluate the **individual mediation effect**, while allowing directed paths among the mediators
- construct the test statistic using the **logic of Boolean matrices** $\rightarrow$ establish the proper limiting distribution under the null $\rightarrow$ the asymptotics of the test statistic built on **regular matrix operations** are difficult to establish
- can be naturally coupled with a **screening** procedure $\rightarrow$ help scale down the number of potential paths to a moderate level $\rightarrow$ enhance the **power** of the test
- use **data splitting** to ensure a valid type-I error rate control under minimal conditions on the screening
- devise a **decorrelated estimator** to reduce potential bias induced by high-dimensional mediators
- employ **multiplier bootstrap** to obtain the critical values
- couple with a **multiple testing** procedure for FDR control
- establish the **asymptotic size**, **power**, and **FDR control** in theory

Project VII: Divide and Conquer

- Develop a **massive data** framework for **cubic-rate M-estimators**
- Show the aggregated estimator obtained via **divide and conquer**
  - has faster convergence rate and asymptotic normal distribution
  - more tractable in both computation and inference than the original M-estimator based on the pooled data
- Provide an upper error bound for the **bias** of general cubic-rate estimators
  - different from $\boldsymbol{n}^{-\frac{1}{2}}$ estimators, $\boldsymbol{n}^{-\frac{1}{3}}$ estimators do not have a **linear** representation
  - **KMT approximation** (Komlós et al., 1975) can be used to upper bound the bias. However, the rate of the approximation will depend on the parameter dimension and decays faster as the dimension increases
  - We introduce a **linear perturbation** in the empirical objective function
  - This transforms the problem of quantifying the bias into comparison of the expected supremum of the empirical objective function and that of its limiting Gaussian process
  - Existing techniques are available (Chernozhukov et al., 2013; 2014) to obtain **dimension-free rate**

Project VII: Divide and Conquer (Cont'd)

- Consider a $\boldsymbol{d}$-dimensional cubic-rate M-estimator

$$\widehat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \sum_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{n}} m(\boldsymbol{X_i}; \boldsymbol{\theta}) \equiv \arg\min_{\boldsymbol{\theta}} M_{\boldsymbol{n}}(\boldsymbol{\theta})$$

- Define the following process with a linear drift

$$M_{\boldsymbol{n}}^{\varepsilon, \boldsymbol{a}}(\boldsymbol{\theta}) = M_{\boldsymbol{n}}(\boldsymbol{\theta}) + \varepsilon \boldsymbol{a}^\top (\boldsymbol{\theta} - \boldsymbol{\theta_0}),$$

  for any $\varepsilon \in \mathbb{R}$, $\boldsymbol{a} \in \mathbb{R}^{\boldsymbol{d}}$ and $\boldsymbol{\theta_0}$ denotes the ground truth

- This allows us to upper bound the bias $|\mathbb{E}\boldsymbol{a}^\top(\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta_0})|$ by

$$\max \left[\left| \frac{\mathbb{E} \min_{\boldsymbol{\theta}} M_{\boldsymbol{n}}^{\varepsilon, \boldsymbol{a}}(\boldsymbol{\theta}) - \mathbb{E} \min_{\boldsymbol{\theta}} M_{\boldsymbol{n}}(\boldsymbol{\theta})}{\varepsilon} \right|, \left| \frac{\mathbb{E} \min_{\boldsymbol{\theta}} M_{\boldsymbol{n}}^{-\varepsilon, \boldsymbol{a}}(\boldsymbol{\theta}) - \mathbb{E} \min_{\boldsymbol{\theta}} M_{\boldsymbol{n}}(\boldsymbol{\theta})}{\varepsilon} \right| \right]$$