Statistical Inference in Reinforcement Learning

Chengchun Shi

Associate 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



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



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



Applications in Ridesharing (Cont'd)



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



Does the Markov Decision Process Fit the Data: Testing for the Markov Property in Sequential Decision Making –Joint work with Runzhe Wan, Rui Song, Wenbin Lu and Ling Leng (ICML, 2020)

Testing Markovanity in Time Series via Deep Generative Learning

-Joint with Yunzhe Zhou, Lexin Li and Qiwei Yao (JRSSB, 2023+)

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**:

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

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

- **History-Dependent** Policy: π_t depends on \overline{S}_t .
- Markov Policy: π_t depends on \overline{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)

History-dependent policy



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,

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

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

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

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. In Model III, $\{S_t\}_t$ 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, · · ·)
 - 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, · · · , 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



Reward Function

- Penalty for wrong opening: -100
- Reward for correct opening: +10
- Cost for listening action: -1

Observations

- to hear the tiger on the left (TL)
- to hear the tiger on the right(TR)

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: Chen 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 S_{t+1} given A_t and S_t (forward learner)
 - Learn CCF of (S_t, A_t) given (S_{t+1}, 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

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



• small **N**, large **T** (OhioT1DM dataset)



т

• large **N**, large **T** (games)

- (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 $(NT)^{-1/4}$.

Theorem

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

Some Follow-ups

- **Double GANs** for conditional independence testing (*JMLR*, 2021)
- Testing DAGs via supervised, structural learning and **GANs** (*JASA*, 2023+)
- Testing Markovanity in time series via deep generative learning (*JRSSB*, 2023+)
 - Derive the convergence rate of MDN
- A robust test for the **stationarity** assumption in RL (*ICML*, 2023)
 - Our test helps identify a better policy in the **Intern Health Study**



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 (2023)

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



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. $S_t = 0.5S_{t-1} + \delta A_{t-1} + N(0, 0.25), R_t = S_t$
- $\mathcal{H}_{\mathbf{0}}$: The old policy ($\mathbf{A} = 0$) has larger cumulative rewards ($\delta \leq \mathbf{0}$)
- \mathcal{H}_1 : The new policy $(\mathbf{A}=1)$ has larger cumulative rewards $(\delta > \mathbf{0})$

Table: Powers of t-test, DML-based test (Chernozhukov et al., 2018) and the proposed test with $T = 500, \delta = 0.1$ (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



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

- 1. allows for $\ensuremath{\textbf{sequential monitoring}}$
- 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



• 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

^aThe 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 A = 1 v.s. old A = 0
- Reward: drivers' income
- The new policy is expected to have **better** performance

Application to Ridesharing Platform (Cont'd)

• The proposed test



• 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 A/B testing (AOAS, JRSSB, 2023)



• **Optimal experimental designs** for A/B testing (*NeurIPS, 2023; Wen et al., 2024*)



Figure: Boxplots of the RMSEs of various ATE estimators under different designs in a **real-data-based simulator**

Deeply-Debiased Off-Policy Interval Estimation

joint work with Runzhe Wan, Victor Chernozhukov, and Rui Song ——ICML, 2021 (long talk, top 3% of submissions)

- **Objective**: Evaluate the impact of a target policy **offline** using historical data generated from a different behavior policy and provide rigorous **uncertainty quantification** (healthcare, automated driving, ridesharing, robotics, e.g.)
- Consider the reinforcement learning (e.g., MDP) setting
- Many existing methods focus on providing point estimators
- Main idea: Develop a deeply-debiasing process using higher order influence function (Robins et al., 2017)

Method



47 / 49

Some Follow-ups

 Causal RL: OPE in confounded MDPs/POMDPs (JASA, 2022+; ICML 2022, 2023)



• OPE in doubly inhomogeneous environments (Bian et al., 2023)

 An RL framework for dynamic mediation analysis (ICML, 2023; Luo et al. 2023)



- IDE: $A_1 \rightarrow R_1$
- IME: $A_1 \rightarrow M_1 \rightarrow R_1$
- DDE: $A_0 \rightarrow S_1 \rightarrow R_1$
- DME: $A_0 \rightarrow M_0 \rightarrow S_1 \rightarrow R_1$

Thank You!



