

Does the Markov decision process fit the data

—Testing for the Markov property in sequential decision making

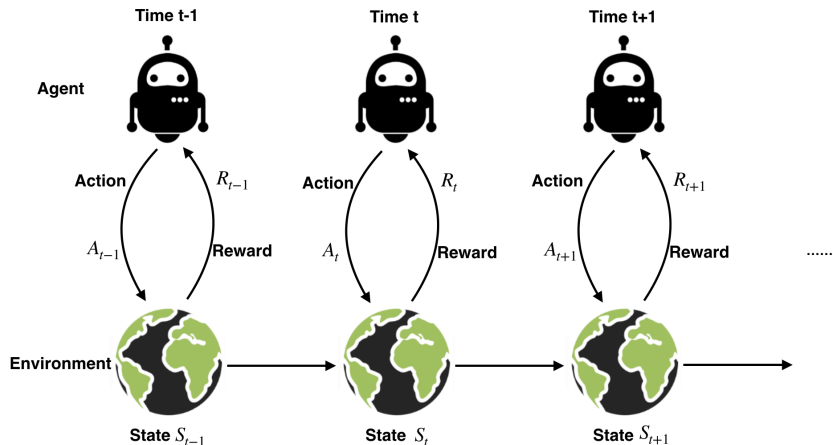
Chengchun Shi¹ and Runzhe Wan² and Rui Song² and
Wenbin Lu² and Ling Leng³

¹London School of Economics and Political Science

²North Carolina State University

³Amazon

Sequential decision making



Objective: find an optimal policy that maximizes the cumulative reward

Reinforcement learning (RL)

- **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.
 - π_t^{opt} depends only on $S_t \cup \{(S_j, A_j)\}_{j < t}$ only through S_t ;
 - $\pi_t^{opt} = \pi^{opt}$ for any t .
 - **Markov assumption** (MA): conditional on the present, the future and the past are independent,

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

The Markov transition kernel is homogeneous in time.

RL models

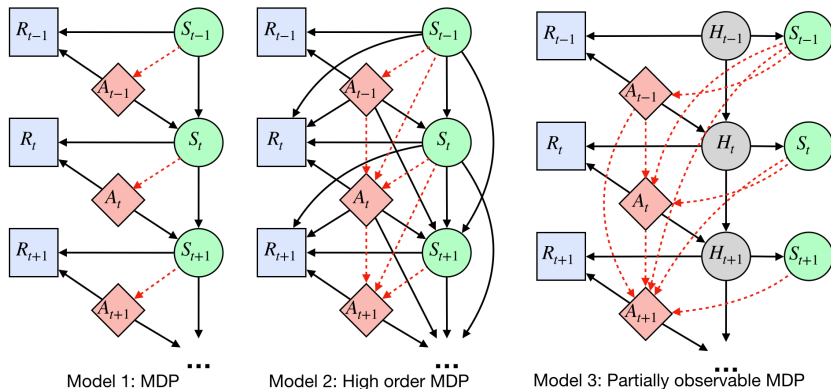


Figure: Causal diagrams for MDPs, HMDPs and POMDPs. The solid lines represent the causal relationships and the dashed lines indicate the information needed to implement the optimal policy. $\{H_t\}_t$ denotes latent variables.

Our 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**:
 - 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**

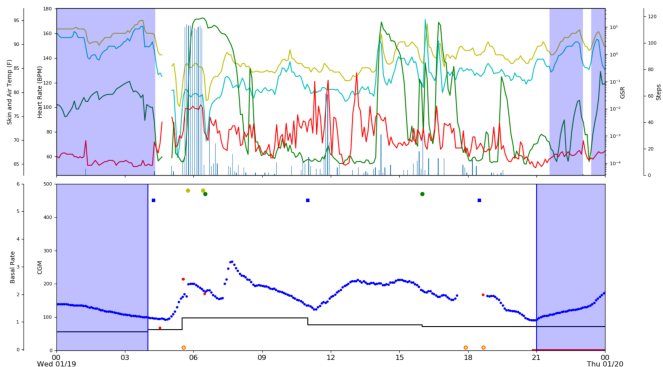
- identify the optimal policy in **high-order** MDPs;
- detect **partially observable** MDPs.

- **Theoretically**

- prove our test **controls type-I error** under a **bidirectional** asymptotic framework.

Applications in high-order MDPs

- **Data:** the OhioT1DM dataset (Marling & Bunescu, 2018).
 - Measurements for 6 patients with type I diabetes over 8 weeks.
 - One-hour interval as a time unit.
 - **State:** patients' time-varying variables, e.g., glucose levels.
 - **Action:** to inject insulin or not.
 - **Reward:** the Index of Glycemic Control (Rodbard, 2009).



Applications in high-order MDPs (Cont'd)

- **Analysis I:**

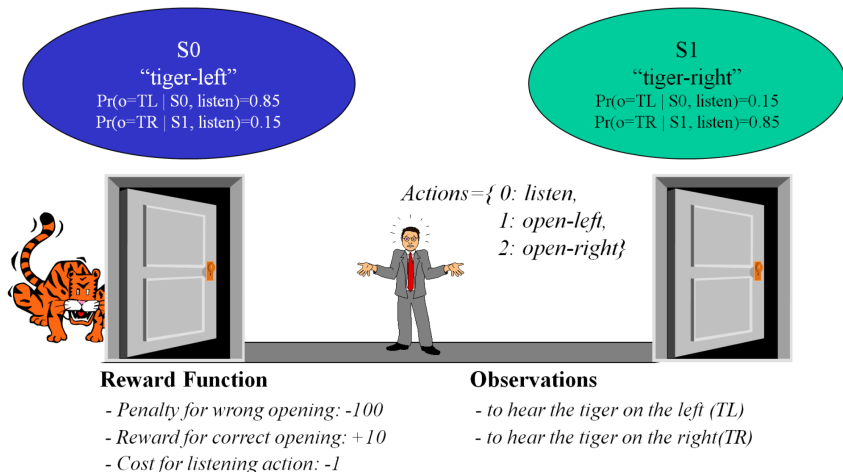
- sequentially apply our test to determine the order of MDP;
- conclude it is a **fourth-order** MDP.

- **Analysis II:**

- split the data into training/testing samples;
- policy optimization based on fitted-Q iteration (Ernst et al., 2005), by assuming it is a k -th order MDP for $k = 1, \dots, 10$;
- policy evaluation based on fitted-Q evaluation (Le et al., 2019);
- 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.

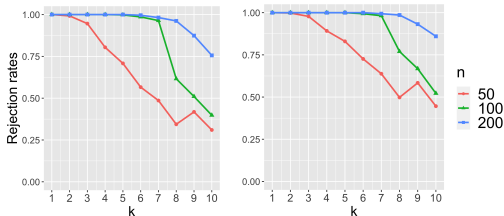
order	1	2	3	4	5	6	7	8	9	10
value	-90.8	-57.5	-63.8	-52.6	-56.2	-60.1	-63.7	-54.9	-65.1	-59.6

Applications in partially observable MDPs

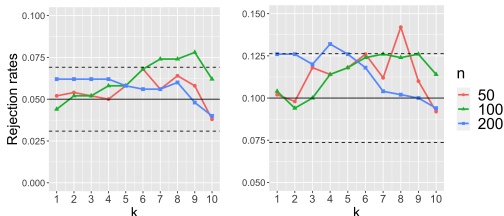


Applications in partially observable MDPs (Cont'd)

- Empirical rejection rates under the alternative hypothesis (MA is violated). $\alpha = (0.05, 0.1)$ from left to right.



- Empirical rejection rates under the null hypothesis (MA holds). $\alpha = (0.05, 0.1)$ from left to right.



Forward-backward learning

- Challenge:
 - develop a valid test for MA in **moderate or high-dimensions** (no existing method works well);
 - 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.

Forward-backward learning (Cont'd)

Some key components of our algorithm:

- Characterize MA based on the notion of **conditional characteristic function** (CCF);
- 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** estimating equations by integrating forward and backward learners;
- To improve the power, construct a **maximum-type** test statistic;
- To control the type-I error, approximate the distribution of our test via **multiplier bootstrap**.

Bidirectional theory

- N the number of trajectories;
- T the number of decision points in each trajectory;
- bidirectional asymptotics: a framework where either N or T grows to ∞ ;
- large T , small N (mobile health)



- large N , small T (some medical studies)



- large N , large T (games)

Bidirectional theory (cont'd)

- (C1) Actions are generated by a fixed behavior policy.
- (C2) The process $\{S_t\}_{t \geq 0}$ 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 ∞ .

Thanks!

The paper is accepted at ICML 2020.

Preprint <https://arxiv.org/pdf/2006.02615.pdf>,

Python code TestMDP <https://github.com/RunzheStat/TestMDP>