Research Centre for Information Systems Engineering (LIRIS), KU Leuven, Leuven, Belgium

KU LEUVEN

On Timing Process Interventions for Prescriptive Process Monitoring

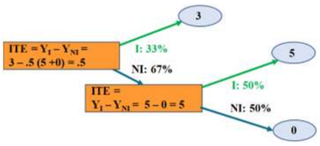
Hans Weytjens, Wouter Verbeke, Jochen De Weerdt

Introduction

- From predictive to prescriptive process monitoring: optimization, rather than mere prediction, of processes delivers tangible benefits
- Causal inference (CI) and Reinforcement Learning (RL) emerge as state-of-art methods in literature. No comparison exists
- Nearly all existing research based on work on real, offline data. Lack of counterfactuals prevents accurate evaluation of tested methods
- We compare CI to RL using synthetic datasets

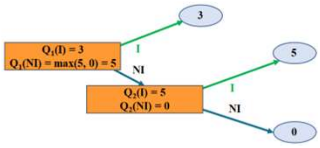
1. Causal Inference (CI) and Reinforcement Learning (RL)

CI



- ITE: individual treatment effect
- Policy requires threshold
- At the first event, the prediction model for the outcome for non-intervention (Y_{NI}) will observe two different outcomes (5 and 0) and **summarize** those to 2.5
- This value depends on the loss function and the samples distribution (% in the graph) in the training set which itself depends on the **data-gathering** policy

RL



- Q-learning: Q value learned for each state-action combination
- No thresholds required
- At the first event, RL selects the **maximum** of the two Q values (5 and 0).
- **Online** learning (exploration) ensures independence of data-gathering policy
- **Offline** learning requires modifications, e.g.,: - process-aware model
  - predictive model
  - synthetic data
  - nearest-neigbor algorithms

2. Experimental Setup

Problem setting

- Process interventions: binary, one-off actions

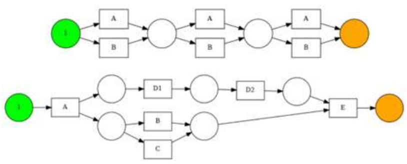
Implementation details

- Deep RL Q-table approximated by neural network
- Perfect policy (manually) computed as well
- Metric used is **uplift**: difference between the process outcomes implementing the policy and not intervening at all
- Assumption: independent processes

Synthetic data from two generative processes

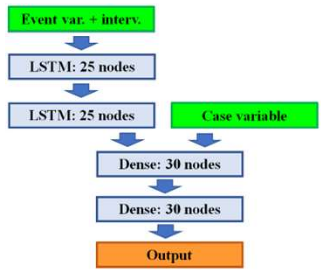
Process_1:

Process_2:



- Datasets for CI 10,000 samples, largely exceeding state space size to offset offline handicap
- Test set 1,000 samples, with ALL counterfactuals computed

Identical neural network for CI and RL



3. Results

Results

		Uplift		Computational effort		
		Mean	StDev.	Unit	Mean	StDev.
Process_1	CI	1,526	36.8	epochs	188	27.6
	RL	1,616	5.0	transitions	6,000	2,549
	perfect	*1,651*				
	RCT	*-515*				
Process_2	CI	1,682	203.1	epochs	212	51.7
	RL	1,806	34.7	transitions	26,800	10,628
	perfect	*1,845*				
	RCT	*-50,336*				

- RL outperforms CI:
  1) Higher scores
  2) More robust (lower standard devs)
- RL nearly reaches results perfect policy (within <3%)
- CI also much better than random policy (RCT)
- RL very computing-intensive (epochs similar to transitions)

Discussion

Causal Inference
- Sequential aspect of processes poses problem and leads to results depending on data-gathering policy
- CI limited to problems with few (or only one) successive actions
- Alternative: indirect CI: predict suffixes first, then outcomes: compounds errors but overcomes fallacies above

Reinforcement Learning
- Online RL theoretically converges to optimal policy
- Depends on strong online assumption
- Reward specification: how to include subordinate goals, human intuition, etc.?
- Techniques for offline RL exist

Real-World Implemenatation
- Slow (low initial exploration rate) RL implementation should not be riskier than CI
- Reward specification: deviation from proposed policy during implementation is possible (e.g., human factor, danger avoidance)

Conclusion

- Online RL better and more robust than CI; however, requires online environment
- CI viable, simple alternative for problems with few successive actions

Future work

- CI-offline RL comparison (with synthetic data for accurate evaluations)
- RL for more complex problems: successive and multi-class actions
- Lifting independence assumption
- Direct CL

This work is unpublished as of today and was submitted to the CAiSE 2023 conference in paper form