Off-policy Model-based Learning under Unknown Factored Dynamics

Assaf Hallak IFOGPH@GMAIL.COM
François Schnitzler FRANCOIS@EE.TECHNION.AC.IL
Timothy Mann MANN@EE.TECHNION.AC.IL
Shie Mannor SHIE@EE.TECHNION.AC.IL
Technion, Haifa, Israel

Abstract

Off-policy learning in dynamic decision problems is essential for providing strong evidence that a new policy is better than the one in use. But how can we prove superiority without testing the new policy? To answer this question, we introduce the G-SCOPE algorithm that evaluates a new policy based on data generated by the existing policy. Our algorithm is both computationally and sample efficient because it greedily learns to exploit factored structure in the dynamics of the environment. We present a finite sample analysis of our approach and show through experiments that the algorithm scales well on high-dimensional problems with few samples.

1. Introduction

Reinforcement Learning (RL) algorithms learn to maximize rewards by analyzing past experience with an unknown environment. Most RL algorithms assume that they can choose which actions to explore to learn quickly. However, this assumption leaves RL algorithms incompatible with many real-world business applications.

To understand why, consider the problem of on-line advertising: Each customer is successively presented with one of several advertisements. The advertiser's goal is to maximize the probability that a user will click on an ad. This probability is called the Click Through Rate (CTR, Richardson et al. 2007). A marketing strategy, called a policy, chooses which ads to display to each customer. However, testing new policies could lose money for the company. Therefore, management would not allow a new policy to be tested unless there is strong evidence that the policy is not worse than the company's existing policy. In other words, we would like to estimate the CTR of other strategies using only data obtained from the company's existing policy. In general, the problem of determining a policy's value from data generated by another policy is called *off-policy evaluation*, where the policy that generates the data is called the *behavior policy*, and the policy we are trying to evaluate is called the *target policy*. This problem may be the primary reason batch RL algorithms are hardly used in applications, despite the maturity of the field.

A simple approach to off-policy evaluation is given by the MFMC algorithm Fonteneau et al. (2010), which constructs complete trajectories for the target policy by concatenating partial trajectories generated by the behavior policy. However, this approach may require a large number of samples to construct complete trajectories. One may think that the number of samples is of little importance, since Internet technology companies have access to billions of transactions. Unfortunately, the dimensionality of real-world problems is generally large (up to millions of dimensions) and the events they want to predict can have extremely small probability of occurring. Thus, sample efficient off-policy evaluation is paramount.

An alternative way of looking at the problem is through counterfactual (CF) analysis Bottou et al. (2013). Given the outcome of an experiment, CF analysis is a framework for reasoning about what would have happened if some aspect of the experiment was different. In this paper, we focus on the question: what would have been the expected reward received for executing the target policy rather than the behavior policy? One approach that falls naturally into the CF framework is Importance Sampling (IS) Bottou et al. (2013); Li et al. (2014). IS methods evaluate the target policy by weighting rewards received by the behavior policy. The weights are determined by the probability that the target policy would perform the same action as the one prescribed by the behavior policy. Unfortunately, IS methods suffer from high variance and typically assume that the behavior policy visits every state that the target policy visits with nonzero probability.

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

Even if this assumption holds, IS methods are not able to exploit structure in the environment because their estimators do not create a compact model of the environment. Exploiting this structure could drastically improve the quality of off-policy evaluation with small sample sizes (relative to the dimension of the state-space). Indeed, there is broad empirical support that model-based methods are more sample efficient than model-free methods Hester & Stone (2009); Jong & Stone (2007). However, one broad class of compact models are Factored-state Markov Decision Processes (FMDPs, Kearns & Koller 1999; Strehl et al. 2007; Chakraborty & Stone 2011). An FMDP model can often be learned with a number of samples *logarithmic* in the total number of states, if the structure is known. Unfortunately, inferring the structure of an FMDP is generally computationally intractable for FMDPs with high-dimensional state-spaces Chakraborty & Stone (2011), and in real-world problems the structure is rarely known in advance.

Ideally, we would like to apply model-based methods to off-policy evaluation because they are generally more sample efficient than model-free methods such as `MFMC` and IS. In addition, we want to use algorithms that are computationally tractable. To this end, we introduce `G-SCOPE`, which learns the structure of an FMDP greedily. `G-SCOPE` is both sample efficient and computationally scalable. Although `G-SCOPE` does not always learn the true structure, we provide theoretical analysis relating the number of samples to the error in evaluating the target policy. Furthermore, our experimental analysis demonstrates that `G-SCOPE` is significantly more sample efficient than model-free methods.

The main contributions of this paper are:

- a novel, scalable method for off-policy evaluation that exploits unknown structure,
- a finite sample analysis of this method, and
- a demonstration through experiments that this approach is sample efficient.

2. Background

We consider dynamics that can be represented by a Markov Decision Process (MDPs; Puterman 2009):

Definition 1. *A Markov Decision Process (MDP) is a tuple* $(S, A, P(s'|s,a), R(s,a), \rho)$ *where* S *is the state space,* A *is the action space,* P *represents the transition probabilities from every state-action pair to another state,* R *represents the reward function fitting each state-action pair with a random real number, and* ρ *is a distribution over the initial state of the process.*

We denote by π a Markov policy that maps states to a distribution over actions. The process horizon is T, and applying a policy for T steps starting from $s_0 \sim \rho$ results in a cumulative reward known as the value function: $V^\pi(s_0) = \mathbb{E}\left[\sum_{t=0}^{T-1} R(s_t, a_t)|s_0, \pi\right]$, where the expectation is taken with respect to P, R and π. We assume R is known and immediate rewards are bounded in $[0, 1]$.

The system dynamics is as follows: First, an initial state s_0 is sampled from ρ. Then, for each time step $t = 0, \ldots, T-1$, an action a_t is sampled according to the policy $\pi(s_t)$, a reward r_t is awarded according to $R(s_t, a_t)$ and the next state s_{t+1} is sampled by $\Pr(\cdot|s_t, a_t)$. The quantity of interest is the expected policy value $\nu^\pi = \rho^\top V^\pi$.

2.1. Off-Policy Evaluation

We consider the finite horizon batch setup. Given are H trajectories of length T sampled from an MDP with an initial state distribution ρ and behavior policy π_b. The off-policy evaluation problem is to estimate the T-step value of a target policy π (different from π_b). For the target policy π, we aim to minimize the difference between the true and estimated policy value:

$$|\nu^\pi - \hat{\nu}^\pi|. \tag{1}$$

2.2. Factored MDPs

Suppose the state space can be decomposed into D discrete values. We denote the i^{th} variable of $\underline{X}$ by $\underline{X}(i)$, and for a given subset of indices $\Psi \subseteq [D] \triangleq \{1, 2, .., D\}$, let $\underline{X}(\Psi)$ be the subset of corresponding variables $\{\underline{X}(i)\}_{i\in\Psi}$. We define a factored MDP, similar to Guestrin et al. 2003:

Definition 2. *A Factored MDP (FMDP) is an MDP* (S, A, P, R, ρ) *such that the state* $\underline{X} \in S$ *is composed of a set of* D *variables* $\{\underline{X}(i)\}_{i=1}^D$*, where each variable can take values from a finite domain, such that the probability of the next state* $\underline{Y}$ *given that action* a *is performed in state* $\underline{X}$ *satisfies*

$$\Pr(\underline{Y}|\underline{X}, a) = \prod_{i=1}^{D} \Pr(\underline{Y}(i)|\underline{X}, a) \ . \tag{2}$$

For simplicity, we assume that all variables lie in the same domain Γ, i.e., $\underline{X} \in \Gamma^D$, where Γ is a finite set. Furthermore, each variable in the next state $\underline{Y}(i)$ only depends on a subset of variables $\underline{X}(\Phi_i)$ where $\Phi_i \subseteq [D]$. The indices in Φ_i are called the parents of i. When the size of the parent sets are smaller than D, then the FMDP can be represented more compactly:

$$\Pr(\underline{Y}|\underline{X}, a) = \prod_{i=1}^{D} \Pr(\underline{Y}(i)|\underline{X}(\Phi_i), a) \ . \tag{3}$$

For a subset of indices $\Psi \subseteq [D]$, a realization-action pair $(v, a) \in \Gamma^{|\Psi|} \times A$ is a specific instantiation of values for the corresponding variables $\underline{X}(\Psi), a$. We denote by $F_i = \Gamma^{|\Phi_i|} \times A$ the set of all realization-action pairs for the parents of node i, and mark $\Lambda = \bigcup_{i=1}^{D} F_i$. Finally, denote by $\Psi \subseteq [D]$ a subset of indices and by $v \in \Gamma^{|\Psi|}$ a realization of the corresponding variables:

$$\Pr(\underline{Y}(i)|\underline{X}(\Psi) = v, a) \triangleq \frac{\sum_{t=1}^{T} \Pr(\underline{Y}(i), \underline{X}(\Psi) = v, a, t)}{\sum_{t=1}^{T} \Pr(\underline{X}(\Psi) = v, a, t)}$$

$$\widehat{\Pr}(\underline{Y}(i) = y|\underline{X}(\Psi) = v, a) \triangleq \frac{n(y, v, a)}{n(v, a)} , \quad (4)$$

where the probabilities in the right term of the first equation are conditioned on the behavior policy π_b omitted for brevity. Note that if $\Psi \supseteq \Phi_i$ then $\Pr(Y(i)|X(\Psi) = v, a) = \Pr(Y(i)|X(\Phi_i) = v(\Phi_i), a)$, and the policy dependency cancels out.

2.3. Previous Work

Previous works on FMDPs focus on finding the optimal policy. Early works assumed the dependency structure is known Guestrin et al. (2002); Kearns & Koller (1999). Degris et al. (2006) proposed a general framework for iteratively learning the dependency structure (this work falls within this framework), yet no theoretical results were presented for their approach. SLF-Rmax Strehl et al. (2007), Met-Rmax Diuk et al. (2009) and LSE-Rmax Chakraborty & Stone (2011) are algorithms for learning the complete structure. Only the first two require as input the in-degree of the DBN structure. The sample complexity of these algorithms is *exponential* in the number of parents. Finally, learning the structure of DBNs with no related reward is in itself an active research topic Friedman et al. (1998); Trabelsi et al. (2013).

There has also been increasing interest in the RL community regarding the topic of off-policy evaluation. Works focusing on model-based approaches mainly provide bounds on the value function estimation error. For example, the simulation lemma Kearns & Singh (2002) can be used to provide sample complexity bounds on such errors. On the other hand, model free approaches suggest estimators while trying to reduce the bias. Precup (2000) presents several methods based on applying importance sampling on eligibility traces, along with an empirical comparison; Thomas et al. (2015) had analyzed bounds on the estimation error for this method. A different approach was suggested by Fonteneau et al. (2010): evaluate the policy by generating artificial trajectories - a concatenation of one-step transitions from observed trajectories. The main problem of these approaches besides the computational cost is that a substantial amount of data required to generate reasonable artificial trajectories.

3. Algorithm

In general, inferring the structure of an FMDP is exponential in D Strehl et al. (2007). Instead, we propose a naive greedy algorithm which under some assumptions can be shown to provide small estimation error on the transition function (G-SCOPE - Algorithm 1).

Algorithm 1 G-SCOPE(H T-length traj., $\epsilon, \delta, C_2 = 0$)

for $i = 1$ to D **do**
  $\hat{\Phi}_i \Leftarrow \emptyset$
  repeat
    $\Theta_i \Leftarrow \{(v, v(j), a) \in \Gamma^{|\hat{\Phi}_i|+1} \times A : j \in [D] \backslash \hat{\Phi}_i, |n(v, v(j), a)| > N(\epsilon, \delta)\}$
    For $N(\epsilon, \delta) = \frac{2\Gamma^2}{\epsilon^2} \ln\left(\frac{2\Gamma}{\delta_1}\right)$
    if $|\Theta| = 0$ **then**
      Break
    end if
    for $j = 1$ to D **do**
      $\text{diff}_j \Leftarrow \max_{(v,v(j),a)\in\Theta} \|\widehat{\Pr}(Y(i)|X(\hat{\Phi}_i \cup j) = (v, v(j)), a) - \widehat{\Pr}(Y(i)|X(\hat{\Phi}_i) = v, a)\|_1$
    end for
    $j^* \Leftarrow \arg\max_{j\in[D]} \text{diff}_j$
    if $\text{diff}_{j^*} > C_2 + 2\epsilon$ **then**
      $\hat{\Phi}_i \Leftarrow \hat{\Phi}_i \cup j^*$
    end if
  until $\text{diff}_{j^*} \leq C_2 + 2\epsilon$
end for
return $\{\hat{\Phi}_i\}_{i=1}^{D}$

G-SCOPE (Greedy Structure learning of faCtored MDPs for Off-Policy Evaluation) receives off-line batch data, two confidence parameters ϵ, δ and a minimum acceptable score C_2. The outputs $\hat{\Phi}_i$ are the estimated parents of each variable i. In the inner loop, the set Θ is defined as the set of all realization-action pairs which had been observed at least $N(\epsilon, \delta)$ times; These are the only pairs further considered. We then greedily add to $\hat{\Phi}_i$ the j'th variable which maximizes the L_1 difference between the old distribution depending only on $\hat{\Phi}_i$, and a distribution conditioned on the additional variable as well. Parents are no longer added when that difference is small, or when all possible realizations were not observed $N(\epsilon, \delta)$ times. The computational complexity of a naive implementation is $O(HT\Gamma D^2)$, since G-SCOPE sweeps the data for every input and output variable.

The idea beyond G-SCOPE is that having enough samples will result in an adequate estimate of the conditional prob-

abilities. Then, under regularity assumptions stated in Section 4, adding a non parent variable is unlikely. If parents have a higher effect than non-parents on the L_1 distance and non-parents have a weak effect, the $\arg\max$ procedure will most likely return only parents. When all prominent parents were found, or when there is not enough data for further inference, the algorithm stops. Once the set of assumed parents is available, we can build an estimated model and simulate *any* policy.

Notice that G-SCOPE algorithm does not necessarily find the actual parents. Instead, we settle on finding a subset of variables providing probably approximately correct transition probabilities. Hence, the number of considered parents scales with data available, a desired quality linking the model and sample complexity. Since we do not necessarily detect all parents, non-parents can have a non-zero influence on the target variable after all prominent parents have been detected. To avoid including these non-parents, the threshold to add a parent is C_2 plus some precision parameters. In practice, we use $C_2 = 0$ because including non-parents with an indirect influence on $\underline{Y}(i)$ may improve the quality of the model. However, in our analysis, we present Assumptions under which the true parents can be learned and explain C_2.

Finally, G-SCOPE can be modified to encode and construct the conditional probability distributions using decision trees. A different decision tree is constructed for each action and variable in the next state. Tree based models can produce more compact representations of the model than encoding the full conditional probability tables specified by $\hat{\Phi}_i$. While we analyze G-SCOPE as an algorithm that separates structure learning from estimating the conditional probability tables, for simplicity and clarity, in our experiments, we actually use a decision tree based algorithm. The modifications to the analysis for the tree based algorithm would add unnecessary complexity and distract from the key points of the analysis.

4. Analysis

By using a scalable but greedy approach to structure learning rather than a combinatorially exhaustive one, G-SCOPE can only learn arbitrarily well a subclass of models. In this section, we introduce three assumptions on the FMDP that describe this subclass, and then analyze the policy evaluation error for this subclass.

We divide Φ_i to non-overlapping "weak" (Φ_i^w) and "strong" (Φ_i^s) parents. We define these subsets formally later, but intuitively, parents in Φ_i^s have a large influence on $\underline{Y}(i)$ and are easy to detect while parents in Φ_i^w have a small influence that may be below the empirical noise threshold and hence not be detected. Our assumptions state

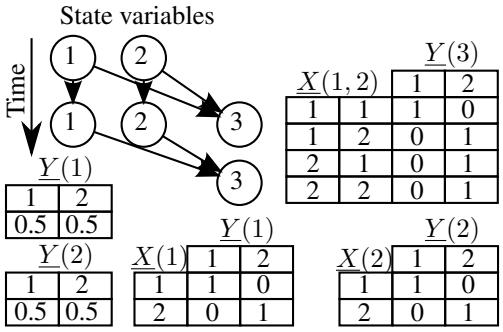

Figure 1. An FMDP that fails to satisfy Assumption 1. The factorization for a given action (not shown on the figure) is represented as a dynamic Bayesian network. States not relevant for the explanation are omitted. In the conditional transition probability tables, rows correspond to possible values of parent variables and columns to possible values of the variable. Cells at the intersection contain conditional probability values.

that (1) "strong" parents are sufficiently better than non-parents to be detected by G-SCOPE before non-parents; (2) conditionally on "strong" parents, non-parent have too little influence on $\underline{Y}(i)$ to be accepted by G-SCOPE and (3) conditioning on some "weak" parents does not increase the influence of other "weak" parents. The first two assumptions are used to bound the probability that G-SCOPE adds non parents in $\hat{\Phi}_i$ or does not add some strong parents, the last one to bound the error caused by the potential non-detection of weak parents.

Assumption 1. ***Strong parent superiority.*** *For every $i \in [D]$, there exists a "strong" subset of parents $\Phi_i^s \subseteq \Phi_i$ such that $\forall \Psi \subset \Phi_i$, $\Phi_i^s \backslash \Psi \neq \emptyset$, $j \in D \backslash \Phi_i$, $(v, v(j), a) \in \Gamma^{|\Psi \cup \{j\}|} \times A$, there exists $k \in \Phi_i^s \backslash \Psi$, such that $\forall (v', v'(k), a') \in \Gamma^{|\Psi \cup \{k\}|} \times A$: for some $C_1 \geq 0$,*

$$\begin{aligned} \| \Pr(\underline{Y}(i) | \underline{X}(\Psi \cup \{k\}) = (v', v'(k)), a') \\ - \Pr(\underline{Y}(i) | \underline{X}(\Psi) = v', a') \|_1 \geq \\ \| \Pr(\underline{Y}(i) | \underline{X}(\Psi \cup \{j\}) = (v, v(j), a) \\ - \Pr(\underline{Y}(i) | \underline{X}(\Psi) = v, a) \|_1 + C_1 \ . \end{aligned} \tag{5}$$

Assumption 1 ensures that, in terms of influence on the conditional distribution of the target, G-SCOPE finds at least one "strong" parent variable k more attractive than any non-parent variable j as long as $\Phi_i^s \backslash \hat{\Phi}_i \neq \emptyset$. This prevents extreme cases where due to large correlation between parents and non-parents factors, large numbers of non-parents could be added before finding the actual parents, thus considerably increasing the sample complexity. C_1 quantifies how much more information a true parent will provide than non-parents. The larger C_1 the less likely G-SCOPE will add a non-parent in $\hat{\Phi}_i$.

Figure 1 illustrates a subset of the state variables and corresponding conditional transition probability distributions

of an FMDP that, for the action implicitly considered, does not satisfy Assumption 1. In this setting, for $t \geq 3$ and considering $\Psi = \emptyset$, we have

$$\|\Pr(\underline{Y}(3)) - \Pr(\underline{Y}(3)|\underline{X}(3) = i)\|_1 = 2 \quad \forall i \in \{1,2\}$$
$$\|\Pr(\underline{Y}(3)) - \Pr(\underline{Y}(3)|\underline{X}(1) = j)\|_1 = 1 \quad \forall j \in \{1,2\}.$$

G-SCOPE would add $\underline{X}(3)$, a non-parent, before any true parent of $\underline{Y}(3)$ in the estimated parent set. Note that in this particular case it does not matter, as $\underline{X}(3)$ perfectly determines $\underline{Y}(3)$. However, adding noise in the transition probabilities would make $\underline{X}(3)$ less accurate than $\underline{X}(1)$ and $\underline{X}(2)$ together.

Assumption 2. ***Non-parent conditional weakness.*** *For every $i \in [D]$, Φ_i^s as in Assumption 1, $\forall \Psi : \Phi_i^s \subseteq \Psi \subseteq \Phi_i$, $j \in D \backslash \Phi_i$, $(v, v(j), a) \in \Gamma^{|\Psi \cup \{j\}|} \times A$: for some $C_2 \geq 0$,*

$$\begin{aligned}\|\Pr(\underline{Y}(i)|\underline{X}(\Psi \cup \{j\}) = (v, v(j), a) \\ - \Pr(\underline{Y}(i)|\underline{X}(\Psi) = v, a)\|_1 \leq C_2 \ .\end{aligned} \tag{6}$$

Assumption 2 ensures that, after G-SCOPE has detected all strong parents, non-parents have a low influence on the target variable and therefore G-SCOPE has a low probability to add them to $\hat{\Phi}_i$. If $\Phi_i^s = \Phi_i$, then $C_2 = 0$.

Assumption 3. ***Conditional diminishing returns.*** *There exists $C_3 \geq 0$ such that for every $i \in [D]$, Φ_i^s as in Assumptions 1 and 2, $\Psi : \Phi_i^s \subseteq \Psi \subseteq \Phi_i$, $j, k \in \Phi_i \setminus \Psi$, $(v, v(j), v(k), a) \in \Gamma^{|\Psi|+2} \times A$, if*

$$\begin{aligned}\|\Pr(\underline{Y}(i)|\underline{X}(\Psi \cup \{j\}) = (v, v(j)), a) \\ - \Pr(\underline{Y}(i)|\underline{X}(\Psi) = v, a)\|_1 \geq \\ \|\Pr(\underline{Y}(i)|\underline{X}(\Psi \cup \{k\}) = (v, v(k)), a) \\ - \Pr(\underline{Y}(i)|\underline{X}(\Psi) = v, a)\|_1,\end{aligned} \tag{7}$$

then:

$$\begin{aligned}\|\Pr(\underline{Y}(i)|\underline{X}(\Psi \cup \{j\}) = (v, v(j)), a) \\ - \Pr(\underline{Y}(i)|\underline{X}(\Psi) = v, a)\|_1 \geq \\ \|\Pr(\underline{Y}(i)|\underline{X}(\Psi \cup \{j,k\}) = (v, v(j), v(k), a) \\ - \Pr(\underline{Y}(i)|\underline{X}(\Psi \cup \{j\}) = (v, v(j)), a)\|_1 + C_3 \ .\end{aligned} \tag{8}$$

If conditioning on $\underline{X}(j)$ provides more knowledge on the output distribution than conditioning on another variable $\underline{X}(k)$, then it will also provide more knowledge than conditioning on $\underline{X}(k)$ given $\underline{X}(j)$. In simple words, Assumption 3 means that information inferred from variables is monotonic, so influential parents cannot go undetected. This assumption supports our greedy scheme, but there are trivial cases where it does not hold.

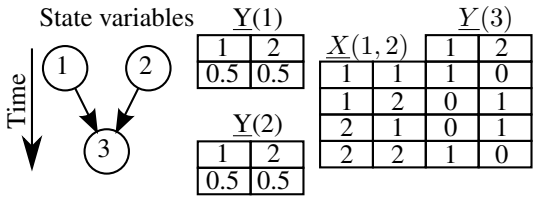

Figure 2. An FMDP that does not satisfy Assumption 3. See Figure 1 for an explanation of the representation.

Consider the substructure represented in Figure 2:

$$\underbrace{\|\Pr(\underline{Y}(3)|\underline{X}(1) = i) - \Pr(\underline{Y}(3))\|_1}_{=0} \not\geq$$
$$\underbrace{\|\Pr(\underline{Y}(3)|\underline{X}(1,2) = (i,j)) - \Pr(\underline{Y}(3)|\underline{X}(1) = i)\|_1}_{=1}.$$

Even though $\underline{X}(1,2)$ are together very informative about variable $\underline{Y}(3)$, any single one of them is not. In such a situation, useful variables cannot be detected by a greedy scheme. Assumption 3 prevents this problem.

These assumptions form the core hardness of the structure learning problem. From one side, there may be implicit dependencies between variables induced by the dynamics - making it hard to separate non-parents. From the other side, the conditional probabilities may belong to a family of XOR like function - initially hiding attractive true parents. Finally, while these assumptions are crucial for proper analysis, non-parent variables may have a beneficial effect on the actual evaluation error as they still contain information on the true parents values, and subsequently information on the output variable.

Theorem 1. *Suppose Assumptions 1, 2 and 3 hold, and let $\frac{C_1}{4} > \epsilon + \frac{C_2}{4}, \epsilon > 0, \delta_1 > 0$, and $m = \max_{i \in [D]} |\Phi_i|$. Then there exists*

$$H(\epsilon, \delta_1) = O\left(\frac{\Gamma^2}{\delta_1 \epsilon^2} \ln\left(\frac{\Gamma}{\delta_1}\right)\right)$$

such that if G-SCOPE is given H trajectories, with probability at least $1 - 2AD(m+2)(D+1-m)\Gamma^{m+1}\delta_1$, G-SCOPE returns an evaluation of π satisfying:

$$|\nu - \tilde{\nu}| \leq T^2(\delta^* + \epsilon^* D) \tag{9}$$

where

$$\epsilon^* = (4m+1)\epsilon + mC_2 + m^2 C_3, \quad \delta^* = A\Gamma^m \sum_{i=1}^{D} \psi_i \delta_1$$
$$\psi_i = \max_{(v,a) \in F_i} \frac{\sum_{t=1}^{T} \Pr(\underline{X}_t(\Phi_i) = v, a_t = a|\pi)}{\sum_{t=1}^{T} \Pr(\underline{X}_t(\Phi_i) = v, a_t = a|\pi_b)} \ . \tag{10}$$

The proof of Theorem 1 is divided in 4 parts, detailed in the supplementary material. First, we derive a simulation lemma for MDPs stating that for the target policy two MDPs with similar transition probability distributions have proximate value functions. We then consider the *number of samples* needed to estimate the transition probabilities of various realization-action pairs. Samples within a trajectory may not be independent so we derive a bound based on Azuma's inequality for martingales. Subsequently, we consider the *number of trajectories* needed to derive a model that evaluates the target policy accurately. If the behavior policy visits enough the parent realizations that the target policy is likely to visit, then the number of trajectories can be small. On the other hand, if the behavior never visits parent realizations that the target policy visits, then the number of trajectories may be infinite. This is captured by ψ_i. Finally, we bound the error due to greedy parent selection under Assumptions 1, 2 and 3.

The evaluation error bound depends on the horizon T, on the number of variables D, on the error bound ϵ^* on most transition probability values of the FMDP constructed by G-SCOPE and on the probability $T\delta^*$ that a trajectory will not visit a state with badly estimated probability values. The dependency of ϵ^* on m is the first advantage of the factorization. The constants C_1, C_2 and C_3, from Assumptions 1, 2 and 3, respectively, indicate the effect of the model "hardness" on the bound. When C_1 is large enough and $C_2 = C_3 = 0$ (the latter happens if all parents are strong parents), the true structure can be learned greedily and the error can be driven arbitrarily close to 0. In other cases, G-SCOPE may learn the wrong structure resulting in some approximation error.

Next, observe the probability that the bounds in Theorem 1 hold. The multiplicative term $A\Gamma^m$ is unavoidable since for each parents realization and action pair the estimation error on the transition probability must be bounded. The main advantage of this theorem is the lack of a Γ^D multiplicative term, which means the effective state space decreased exponentially. The factor $m+2$ is due to the number of iterations of G-SCOPE where a parent is added, and $D-m+1$ is due to bounds on non-parents that must be valid for all these iterations.

In δ^*, the ψ_i values characterize the mismatch between the behavior policy and the target policy. If the behavior policy visits all of the parent-action realizations that the target policy visits with sufficiently high probability, then the ψ_i parameters will be small. But if the target policy visits parent-action realizations that are never visited by the behavior policy, then the ψ_i values may be infinite. The ψ_i values are similar to importance sampling weights used by some model-free off-policy algorithms. However, unlike model-free approaches that depend on the differences in the state visitation distributions of the behavior policy and the target policy, the ψ_i values depend on the differences in the parent realization visitation distributions between the behavior policy and the target policy. This is more flexible because the ψ_i values can be small even when the behavior policy and the target policy visit different regions of the state-space.

5. Experiments

We compared G-SCOPE to other off-policy evaluation algorithms in the Taxi domain Dietterich (1998), randomly generated FMDPs, and the Space Invaders domain Bellemare et al. (2013). Since the domains compared in our experiments have different reward scales, we normalized the errors to compare $\frac{|\nu-\tilde{\nu}|}{|\nu|}$. The evaluation error always refers to the target policy's evaluation error, and all trajectory data is generated by the behavior policy. We compare G-SCOPE to the following algorithms:

- Model-Free Monte-Carlo (MFMC, Fonteneau et al. 2010): a model-free off-policy evaluation algorithm that constructs artificial trajectories by concatenating partial, behavior policy generated, transitions,
- Clipped Importance Sampling (CIS, Bottou et al. 2013): a model-free importance sampling algorithm that uses a heuristic approach to clip extremely large importance sampling ratios,
- Flat : a flat model-based approach that assumes no structure between any two state-action pairs and simply builds an empirical next state distribution for each state-action pair, and
- Known Structure (KS) : a model-based method that is given the true parents, but still needs to estimate the conditional probability tables from data generated by the behavior policy. KS should outperform G-SCOPE, because KS knows the structure. We introduce KS to differentiate the evaluation error due to insufficient samples from the evaluation error due to G-SCOPE selecting the wrong parent variables.

Our experimental results show that (1) model-based off-policy evaluation algorithms are more sample efficient than model-free methods, (2) exploiting structure can dramatically improve sample efficiency, and (3) G-SCOPE often provides a good evaluation of the target policy despite its greedy structure learning approach.

5.1. Taxi Domain

The objective in the Taxi domain Dietterich (1998) is for the agent to pickup a passenger from one location and to drop the passenger off at a destination. The state can be described by four variables. We selected the initial state according to a uniform random distribution and used a hori-

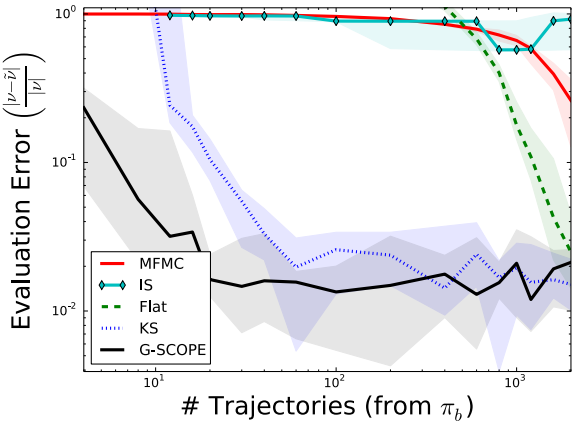

Figure 3. Taxi domain: Median evaluation error for the target policy (shaded region: $1^{\text{st}} - 3^{\text{rd}}$ quantiles) on log-scale varying the number of trajectories generated by the behavior policy. Without exploiting structure MFMC and Flat require many trajectories to achieve small evaluation error. Yet, KS and G-SCOPE achieve small evaluation error with just a few trajectories. Because G-SCOPE adapts the complexity of the model to the samples available, it achieve smaller estimation error than even KS for extremely few trajectories.

zon $T = 200$. The behavior policy selected actions uniform randomly, while the target policy was derived by solving the Taxi domain with the Rmax algorithm Brafman & Tennenholtz (2002). We discovered that the deterministic policy returned by Rmax was problematic for CIS, because the probability of almost all trajectories generated by the behavior policy were 0 with respect to the target policy. To resolve this problem, we modified the policy returned by Rmax to ensure that every action is selected in every state with probability at least $\varepsilon = 0.05$.

The Taxi domain is a useful benchmark because we know the true structure and the total number of states is only 500. Thus, we can compare G-SCOPE to KS and Flat.

Figure 3 presents the normalized evaluation error (on a log-scale) for MFMC, CIS, Flat, KS, and G-SCOPE over 2,000 trajectories generated by the behavior policy. Median and quantiles are estimated over 40 independent trials. For intermediate and large number of trajectories, G-SCOPE performs about the same as if the structure is given and achieves smaller error than the model-free algorithms (MFMC and CIS). Notice that MFMC, CIS, and Flat, which do not take advantage of the domains structure, require a large number of trajectories before they achieve low evaluation error. Interestingly, the Flat (model-based) approach appears to be more sample efficient than MFMC, which is in line with observations that model-based RL is more efficient than model-free RL Hester & Stone (2009); Jong & Stone (2007). KS and G-SCOPE, on the other hand, achieve low evaluation error after just a few trajectories and have similar performance,

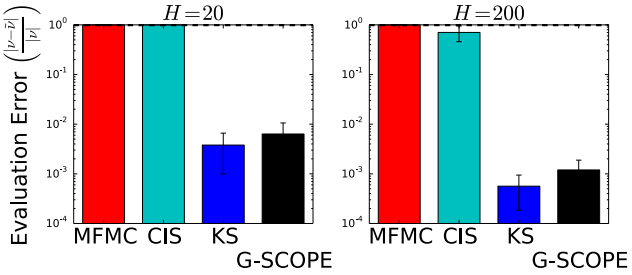

Figure 4. Random FMDP domain: Average evaluation error (± 1 std. deviation) on log-scale for MFMC, KS, and G-SCOPE (with $H = 20$ and 200 trajectories). G-SCOPE has slightly worse performance than Known Structure, but G-SCOPE achieves significantly lower evaluation error than MFMC.

except for very few trajectories where G-SCOPE can adapt the model complexity to the number of samples and therefore achieves a lower evaluation error than the algorithm knowing the structure. This provides one example where greedy structure learning is effective.

5.2. Randomly Generated Factored Domains

To test G-SCOPE in a higher dimensional problem, where we still know the true structure, we randomly generated FMDPs with $D = 20$ dimensional states. The domain of each variable was $\Gamma = \{1, 2\}$. For each state variable the number of parents was uniformly selected from 1 to 4 and the parents were also chosen randomly. Afterwards, the conditional probability tables were filled in uniformly and normalized to ensure they specified proper probability distributions. The FMDP was given a sparse reward function that returned 1 if and only if the last bit in the state-vector was 1 and returned 0 otherwise. We used a horizon $T = 200$. The behavior policy selected actions uniform randomly, while the target policy was derived by running SARSASutton & Barto (1998) with linear value function approximation on the FMDP for 5,000 episodes with a learning rate 0.1, discount factor 0.9, and epsilon-greedy parameter 0.05. After training SARSA, we extracted a stationary target policy. As in the Taxi domain, we modified the policy returned by SARSA to ensure that every action could be selected in every state with probability at least $\varepsilon = 0.05$.

For the randomly generated FMDPs, we could not construct a flat model because there are $2^{20} = 1,048,576$ states and the number of parameters in a flat model scales quadratically with the size of the state-space. However, we could still compare MFMC, CIS, KS, and G-SCOPE.

Figure 4 presents the normalized evaluation error (on a log-scale) for MFMC, CIS, KS, and G-SCOPE given $H = 20$ and $H = 200$ trajectories from the behavior policy. Average and standard deviations are estimated over 10 independent trials. MFMC fails because in this high-dimensional

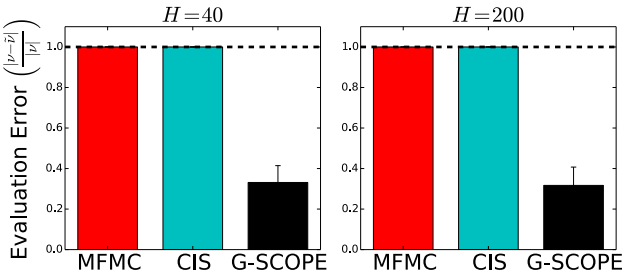

Figure 5. Space Invaders domain: Average evaluation error (± 1 std. deviation) for MFMC, CIS, and G-SCOPE (with $H = 40$ and 200 trajectories). G-SCOPE achieves significantly lower evaluation error than MFMC and CIS.

task there is not enough data to construct artificial trajectories for the target policy. CIS fairs only slightly better than MFMC, because it uses all of the trajectory data. Unfortunately, most of the trajectories generated by the behavior policy are not probable under the target policy and its evaluation of the target policy is pessimistic. G-SCOPE has slightly worse performance than KS, but G-SCOPE achieves significantly lower evaluation error than MFMC and CIS.

5.3. Space Invaders

In the Space Invaders (SI) domain using the Arcade Learning Environment Bellemare et al. (2013), not only do we not know the parent structure, we also cannot verify that the factored dynamics assumption even holds (2). Thus, SI presents a challenging benchmark for off-policy evaluation. We used the 1024-bit RAM as the state vector. We set the horizon $T = 1000$ so that the behavior policy would experience a diverse set of states.

As in the previous experiment, the behavior policy selected actions uniformly at random, while the target policy was derived by running SARSA Sutton & Barto (1998) with linear value function approximation on the FMDP with a learning rate 0.1, discount factor 0.9, and epsilon-greedy parameter 0.05. We only trained SARSA for 500 episodes, because of the time required to sample an episode. After training, we extracted a stationary target policy, which ensured all actions could be selected in all states with probability at least $\varepsilon = 0.05$.

Figure 5 shows the normalized evaluation error for MFMC, CIS, and G-SCOPE given $H = 40$ and $H = 200$ trajectories from the behavior policy. Averages and standard deviations are estimated over 5 independent trials. Again, the evaluation error of G-SCOPE is much smaller than MFMC and CIS. In fact, MFMC and CIS perform no better than a strategy that always predicts the target policy's value is 0. The poor performance of MFMC is due to the impossibility to construct artificial trajectories from samples in such a high dimensional space.

6. Discussion

We presented a finite sample analysis of G-SCOPE that shows how samples can be related to the evaluation error. When $m \ll D$, the sample complexity scales logarithmically with number of states, where $m = \arg\max_{i \in [D]} |\Phi_i|$.

Our experiments show that (1) model-based off-policy evaluation algorithms are more sample efficient than model-free methods, (2) exploiting structure can dramatically improve sample efficiency, and (3) G-SCOPE often provides a good evaluation of the target policy despite using a greedy structure learning approach. Thus, G-SCOPE provides a practical solution for evaluating new policies. Our empirical evaluation on large and small FMDPs shows our approach outperforms existing methods, which only exploit trajectories.

We analyzed G-SCOPE under three assumptions restricting the class of FMDPs that can be considered. These three assumptions imply that (1) including weak parent will not make any other weak parent (significantly) more informative than it was before, (2) strong parents are more relevant than non-parents, and (3) conditioned on the strong parents non-parents are non-informative. We believe that many real-world problems approximately satisfy these assumptions. If the problem under consideration does not satisfy them, then learning algorithms of combinatorial computational complexity in the number of state variables must be considered to correctly identify the true parents Chakraborty & Stone (2011).

To the best of our knowledge, this is the first model-based algorithm and analysis for off-policy evaluation in FMDPs. Moreover, G-SCOPE is a tractable algorithm for learning the structure of an FMDP even if no prior knowledge is given about the order in which variables should be considered. So, hopefully showing the effectiveness of structure learning for off-policy evaluation will encourage the adaptation of existing algorithms for learning the structure of FMDPs and more generally dynamic Bayesian networks for off-policy evaluation.

7. Acknowledgments

This Research was supported in part by the Israel Science Foundation (grant No. 920/12) and by the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013)/ ERC Grant Agreement n.306638.

References

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence*

Research, 47:253–279, 06 2013.

Bottou, L., Peters, J., Quiñonero Candela, J., Charles, D. X., Chickering, D. M., Portugaly, E., Ray, D., Simard, P., and Snelson, E. Counterfactual reasoning and learning systems: The example of computational advertising. *Journal of Machine Learning Research*, 14(1): 3207–3260, 2013.

Brafman, R. I. and Tennenholtz, M. R-MAX - a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3:213–231, 2002.

Chakraborty, D. and Stone, P. Structure learning in ergodic factored mdps without knowledge of the transition function's in-degree. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 737–744, 2011.

Degris, T., Sigaud, O., and Wuillemin, P.-H. Learning the structure of factored Markov decision processes in reinforcement learning problems. In *Proceedings of the 23rd international conference on Machine learning*, pp. 257–264. ACM, 2006.

Dietterich, T. G. The MAXQ method for hierarchical reinforcement learning. In *Proceedings of the 15th International Conference on Machine Learning*, pp. 118–126, 1998.

Diuk, C., Li, L., and Leffler, B. R. The adaptive k-meteorologists problem and its application to structure learning and feature selection in reinforcement learning. In *Proceedings of the 26th International Conference on Machine Learning*, pp. 249–256. ACM, 2009.

Fonteneau, R., Murphy, S., Wehenkel, L., and Ernst, D. Model-free monte carlo-like policy evaluation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2010), JMLR W&CP*, volume 9, pp. 217–224, 2010.

Friedman, N., Murphy, K., and Russell, S. Learning the structure of dynamic probabilistic networks. In *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*, pp. 139–147. Morgan Kaufmann Publishers Inc., 1998.

Guestrin, C., Patrascu, R., and Schuurmans, D. Algorithm-directed exploration for model-based reinforcement learning in factored mdps. In *Proceedings of the 19th International Conference on Machine Learning*, pp. 235–242, 2002.

Guestrin, C., Koller, D., Parr, R., and Venkataraman, S. Efficient solution algorithms for factored mdps. *J. Artif. Intell. Res.(JAIR)*, 19:399–468, 2003.

Hester, T. and Stone, P. Generalized model learning for reinforcement learning in factored domains. In *The Eighth International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, May 2009.

Jong, N. K. and Stone, P. Model-based function approximation in reinforcement learning. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, pp. 95:1–95:8, 2007.

Kearns, M. and Koller, D. Efficient reinforcement learning in factored mdps. In *IJCAI*, volume 16, pp. 740–747, 1999.

Kearns, M. and Singh, S. Near-optimal reinforcement learning in polynomial time. *Machine Learning*, 49(2): 209–232, 2002.

Li, L., Munos, R., and Szepesvari, C. On Minimax Optimal Offline Policy Evaluation. *ArXiv e-prints*, September 2014.

Precup, D. Eligibility traces for off-policy policy evaluation. *Computer Science Department Faculty Publication Series*, pp. 80, 2000.

Puterman, M. L. *Markov decision processes: discrete stochastic dynamic programming*, volume 414. John Wiley & Sons, 2009.

Richardson, M., Dominowska, E., and Ragno, R. Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th international conference on World Wide Web*, pp. 521–530. ACM, 2007.

Strehl, A. L., Diuk, C., and Littman, M. L. Efficient structure learning in factored-state MDPs. In *Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI-07)*, volume 7, pp. 645–650, 2007.

Sutton, R. and Barto, A. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

Thomas, P. S., Theocharous, G., and Ghavamzadeh, M. High confidence off-policy evaluation. In *Proceedings of the Twenty-Ninth Conference on Artificial Intelligence*, 2015.

Trabelsi, G., Leray, P., Ben Ayed, M., and Alimi, A. Dynamic MMHC: A local search algorithm for dynamic Bayesian network structure learning. In *Advances in Intelligent Data Analysis XII*, volume 8207 of *Lecture Notes in Computer Science*, pp. 392–403. Springer Berlin Heidelberg, 2013.