Iterative Value-Aware Modeling Learning

Amir-massoud Farahmand

Presenters: Dami Choi and Chris Zhang

Questions for Professor Animesh

- What is the point of MBRL?
  - Sample cost for MBRL is regarding learning the model or using the model to learn a policy, or both. (assuming sample from model is free)
- If sample efficiency is regarding *after* having a model
  - VAML can't be reused. Then point is not so much to reuse it for different tasks
  - Is the point that we can get more than single sample estimates during q learning?

Model Based Reinforcement Learning

Model Free RL: Learn a value function or policy by directly interacting with environment

Model based RL: Use interactions with environment to learn a model of the environment

Advantages

- Learning model is more sample efficient than policy (sometimes)
- Model can be reused to learn other policies

Potential difficulty: A little bit wrong in the model can be a lot wrong in the policy (which is what matters ultimately)

Motivation

Why prior methods might be failing

- Conventional MBRL learns a model by minimizing probabilistic loss,
  - Then uses the model for planning
  - E.g. Garbage picking robot in art museum. Overkill maybe?
- Solving the unsupervised problem (model learning) in a vacuum ignores the decision problems we eventually need to solve

Let’s do decision aware model learning (DAML)!

Contributions

- A decision-aware method for model based RL
  - Take into account how value based planner would use a model
- An easier optimization problem than prior work
  - Reuses some computation, tradeoff with robustness
- Theoretical analysis
  - What are the effects of errors on the final resulting policy?

Background

MDP

$$(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$$

State space

Action space

Reward distribution

Transition probability kernel

Discount factor

Background

MDP

$$(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$$

State space

Action space

Reward distribution

Transition probability kernel

Discount factor

$$\mathcal{D}_n = \{(X_i, A_i, R_i, X_i')\}_{i=1}^n$$

$$Z_i = (X_i, A_i) \sim \nu(\mathcal{X} \times \mathcal{A})$$

$$R_i \sim \mathcal{R}^*(\cdot | X_i, A_i)$$

$$X_i' \sim \mathcal{P}^*(\cdot | X_i, A_i)$$

Background

MDP

$$(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$$

State space

Action space

Reward distribution

Transition probability kernel

Discount factor

Goal: Find $\hat{\mathcal{P}}$ close to $\mathcal{P}^*$

Background

MDP

$$(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$$

State space

Action space

Reward distribution

Transition probability kernel

Discount factor

Estimation

True

Goal: Find $\hat{\mathcal{P}}$ close to $\mathcal{P}^*$

Background: Value Iteration

One way to find an optimal policy given a model

$$Q_{k+1}(x, a) \leftarrow r(x, a) + \gamma \int \mathcal{P}^*(\mathrm{d}x' | x, a) \max_{a'} Q_k(x', a')$$

Background: Value Iteration

One way to find an optimal policy given a model

$$Q_{k+1}(x,a) \leftarrow r(x,a) + \gamma \int \mathcal{P}^*(\mathrm{d}x' | x,a) \max_{a'} Q_k(x',a')$$

$$\Updownarrow$$

$$Q_{k+1} \leftarrow T^*_{\mathcal{P}^*} Q_k \triangleq r + \gamma \mathcal{P}^* V_k$$

Background: Value Iteration

One way to find an optimal policy given a model

$$Q_{k+1}(x,a) \leftarrow r(x,a) + \gamma \int \mathcal{P}^*(\mathrm{d}x'|x,a) \max_{a'} Q_k(x',a')$$

$$\Updownarrow$$

Bellman optimality operator

$$Q_{k+1} \leftarrow \underline{T^*_{\mathcal{P}^*} Q_k} \triangleq r + \gamma \mathcal{P}^* V_k$$

Background: Value Iteration

One way to find an optimal policy given a model

$$Q_{k+1}(x,a) \leftarrow r(x,a) + \gamma \int \mathcal{P}^*(\mathrm{d}x'|x,a) \max_{a'} Q_k(x',a')$$

$$\Updownarrow$$

Bellman optimality operator

$$Q_{k+1} \leftarrow \underline{T^*_{\mathcal{P}^*} Q_k} \triangleq r + \gamma \mathcal{P}^* V_k$$

Value-Aware Model Learning (Farahmand et al 2017)

Goal: Find a model such that the resulting policy is good

Consider: How to derive a policy using Value Iteration

$$T^* : Q \mapsto r + \gamma \mathcal{P}^* \max_a Q$$

What we want

$$\hat{T}^* : Q \mapsto r + \gamma \hat{\mathcal{P}} \max_a Q$$

What we have

Goal: Find a P such that $T^* Q = \hat{T}^* Q$

Value-Aware Model Learning

$$T^* : Q \mapsto r + \gamma \mathcal{P}^* \max_a Q$$

$$\hat{T}^* : Q \mapsto r + \gamma \hat{\mathcal{P}} \max_a Q$$

Goal: Find a P such that $$T^* Q = \hat{T}^* Q$$

To do that: Minimize $$\mathbb{E}[\mathcal{P}^* V - \hat{\mathcal{P}} V] = \mathbb{E}[(\mathcal{P}^* - \hat{\mathcal{P}})V]$$

In expectation over data,
How different is value under dynamics of my model,
compared to true model

Value-Aware Model Learning (Farahmand et al 2017)

$$\mathbb{E}[\mathcal{P}^* V - \hat{\mathcal{P}} V] = \mathbb{E}[(\mathcal{P}^* - \hat{\mathcal{P}})V]$$

Is there a problem?

We don't have this!

$$c_{2,\nu}^2(\hat{\mathcal{P}}, \mathcal{P}^*; V) = \int \mathrm{d}\nu(x,a) \left| \int \left[\mathcal{P}^*(\mathrm{d}x'|x,a) - \hat{\mathcal{P}}(\mathrm{d}x'|x,a) \right] V(x') \right|^2$$

Value-Aware Model Learning (Farahmand et al 2017)

Is there a problem?

We don't have this!

$$c_{2,\nu}^2(\hat{\mathcal{P}}, \mathcal{P}^*; V) = \int \mathrm{d}\nu(x,a) \left| \int \left[\mathcal{P}^*(\mathrm{d}x'|x,a) - \hat{\mathcal{P}}(\mathrm{d}x'|x,a) \right] V(x') \right|^2$$

$$c_{2,\nu}^2(\hat{\mathcal{P}}, \mathcal{P}^*) = \int \mathrm{d}\nu(x,a) \sup_{V \in \mathcal{F}} \left| \int \left[\mathcal{P}^*(\mathrm{d}x'|x,a) - \hat{\mathcal{P}}(\mathrm{d}x'|x,a) \right] V(x') \right|^2 .$$

Idea: Be robust and consider worse case

VAML algorithm

$$\hat{Q}_0 \leftarrow r$$

$$\hat{Q}_1 \leftarrow r + \gamma \hat{\mathcal{P}}^{(1)} \hat{V}_0$$

$$\hat{Q}_2 \leftarrow r + \gamma \hat{\mathcal{P}}^{(2)} V_1$$

When we have this

Collect data using Q

Solve robust problem $\hat{\mathcal{P}}^{(k)} \leftarrow \arg\min_{\mathcal{P} \in \mathcal{M}} \sup_{V \in \mathcal{F}} \mathbb{E}[(\mathcal{P}^* - \mathcal{P})V]$

Why do this

Iter VAML algorithm

$$\hat{Q}_0 \leftarrow r$$

$$\hat{Q}_1 \leftarrow r + \gamma \hat{\mathcal{P}}^{(1)} \hat{V}_0$$

$$\hat{Q}_2 \leftarrow r + \gamma \hat{\mathcal{P}}^{(2)} V_1$$

When we have this

$$\hat{\mathcal{P}}^{(k)} \leftarrow \arg\min_{\mathcal{P} \in \mathcal{M}} \sup_{V \in \mathcal{F}} \mathbb{E}[(\mathcal{P}^* - \mathcal{P})V]$$

$$\hat{\mathcal{P}}^{(k)} \leftarrow \arg\min_{\mathcal{P} \in \mathcal{M}} \mathbb{E}[(\mathcal{P}^* - \mathcal{P})V_{k-1}]$$

Iterative VAML - Estimates needed

Ideal

$$\mathbb{E}[\mathcal{P}^* V - \hat{\mathcal{P}} V] = \mathbb{E}[(\mathcal{P}^* - \hat{\mathcal{P}}) V]$$

$$\hat{\mathcal{P}}^{(k)} \leftarrow \underset{\mathcal{P} \in \mathcal{M}}{\operatorname{argmin}} \left\| (\mathcal{P} - \mathcal{P}^*) \hat{V}_k \right\|_2^2 = \int \left| (\mathcal{P} - \mathcal{P}^*)(\mathrm{d}x'|z) \max_{a'} \hat{Q}_k(x', a') \right|^2 \mathrm{d}\nu(z),$$

Things we want but dont have

Estimate

$$\hat{\mathcal{P}}^{(k+1)} \leftarrow \underset{\mathcal{P} \in \mathcal{M}}{\operatorname{argmin}} \frac{1}{n} \sum_{(X_i, A_i, X_i') \in \mathcal{D}_n} \left| \hat{V}_k(X_i') - \int \mathcal{P}(\mathrm{d}x'|X_i, A_i) \hat{V}_k(x') \right|^2 .$$

Monte Carlo Estimate of population

Single-sample estimate of P*

Iterative VAML - Estimates needed

Ideal

$$\hat{Q}_{k+1} \leftarrow T^*_{\hat{\mathcal{P}}^{(k)}} \hat{Q}_k$$

Approximate Value Iteration (Fitted Value or Q-Iteration)

$$\hat{Q}_{k+1} \leftarrow \operatorname*{argmin}_{Q \in \mathcal{F}^{|\mathcal{A}|}} \frac{1}{n} \sum_{(X_i, A_i, R_i) \in \mathcal{D}_n} \left| Q(X_i, A_i) - \left(R_i + \gamma \int \hat{\mathcal{P}}^{(k+1)}(\mathrm{d}x' | X_i, A_i) \hat{V}_k(x') \right) \right|^2 .$$

$$\int \hat{\mathcal{P}}^{(k+1)}(\mathrm{d}x' | X_i, A_i) \hat{V}_k(x') \approx \frac{1}{m} \sum_{j=1}^{m} \hat{V}_k(X'_{i,j}),$$

Algorithm 1 Model-based Reinforcement Learning Algorithm with Iterative VAML

// MDP $(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$
// K: Number of iterations
// $\mathcal{M}$: Space of transition probability kernels
// $\mathcal{F}^{|\mathcal{A}|}$: Space of action-value functions
// $\mathcal{G}$: Space of reward functions
Initialize a policy π_0 and a value function $\hat{V}_0$.
for $k = 0$ to $K - 1$ **do**
  Generate training set $\mathcal{D}_n^{(k)} = \{(X_i, A_i, R_i, X_i')\}_{i=1}^n$ by interacting with the true environment (potentially using π_k), i.e., $(X_i, A_i) \sim \nu_k$ with $X_i' \sim \mathcal{P}^*(\cdot|X_i, A_i)$ and $R_i \sim \mathcal{R}^*(\cdot|X_i, A_i)$.
  $\hat{\mathcal{P}}^{(k+1)} \leftarrow \mathrm{argmin}_{\mathcal{P} \in \mathcal{M}} \left\| \hat{V}_k(X_i') - \int \mathcal{P}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
  $\hat{r} \leftarrow \mathrm{argmin}_{r \in \mathcal{G}} \mathrm{Loss}_{\mathcal{R}}(r; \cup_{i=0}^k \mathcal{D}_n^{(i)})$
  $\hat{Q}_{k+1} \leftarrow \mathrm{argmin}_{Q \in \mathcal{F}^{|\mathcal{A}|}} \left\| Q(X_i, A_i) - \left(\hat{r}(X_i, A_i) + \gamma \int \hat{\mathcal{P}}^{(k+1)}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right) \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
  $\pi_{k+1} \leftarrow \hat{\pi}(\cdot; \hat{Q}_{k+1})$.
end for
return π_K

Algorithm 1 Model-based Reinforcement … VAML

$$\hat{Q}_0 \leftarrow r$$
$$\hat{Q}_1 \leftarrow r + \gamma \hat{\mathcal{P}}^{(1)} \hat{V}_0$$
$$\hat{Q}_2 \leftarrow r + \gamma \hat{\mathcal{P}}^{(2)} V_1$$

// MDP $(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$
// K: Number of iterations
// $\mathcal{M}$: Space of transition probability kernels
// $\mathcal{F}^{|\mathcal{A}|}$: Space of action-value functions
// $\mathcal{G}$: Space of reward functions
Initialize a policy π_0 and a value function $\hat{V}_0$
for $k = 0$ to $K - 1$ **do**
  Generate training set $\mathcal{D}_n^{(k)} = \{(X_i, A_i, R_i, X_i')\}_{i=1}^n$ by interacting with the true environment (potentially using π_k), i.e., $(X_i, A_i) \sim \nu_k$ with $X_i' \sim \mathcal{P}^*(\cdot|X_i, A_i)$ and $R_i \sim \mathcal{R}^*(\cdot|X_i, A_i)$.
  $\hat{\mathcal{P}}^{(k+1)} \leftarrow \text{argmin}_{\mathcal{P} \in \mathcal{M}} \left\| \hat{V}_k(X_i') - \int \mathcal{P}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
  $\hat{r} \leftarrow \text{argmin}_{r \in \mathcal{G}} \text{Loss}_{\mathcal{R}}(r; \cup_{i=0}^k \mathcal{D}_n^{(i)})$
  $\hat{Q}_{k+1} \leftarrow \text{argmin}_{Q \in \mathcal{F}^{|\mathcal{A}|}} \left\| Q(X_i, A_i) - \left(\hat{r}(X_i, A_i) + \gamma \int \hat{\mathcal{P}}^{(k+1)}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right) \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
  $\pi_{k+1} \leftarrow \hat{\pi}(\cdot; \hat{Q}_{k+1})$.
end for
return π_K

Algorithm 1 Model-based Reinforcement Learning Algorithm with Iterative VAML

// MDP $(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$
// K: Number of iterations
// $\mathcal{M}$: Space of transition probability kernels
// $\mathcal{F}^{|\mathcal{A}|}$: Space of action-value functions
// $\mathcal{G}$: Space of reward functions
Initialize a policy π_0 and a value function $\hat{V}_0$.
for $k = 0$ to $K - 1$ **do**
  Generate training set $\mathcal{D}_n^{(k)} = \{(X_i, A_i, R_i, X_i')\}_{i=1}^n$ by interacting with the true environment (potentially using π_k), i.e., $(X_i, A_i) \sim \nu_k$ with $X_i' \sim \mathcal{P}^*(\cdot|X_i, A_i)$ and $R_i \sim \mathcal{R}^*(\cdot|X_i, A_i)$.
  $\hat{\mathcal{P}}^{(k+1)} \leftarrow \mathrm{argmin}_{\mathcal{P} \in \mathcal{M}} \left\| \hat{V}_k(X_i') - \int \mathcal{P}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
  $\hat{r} \leftarrow \mathrm{argmin}_{r \in \mathcal{G}} \mathrm{Loss}_{\mathcal{R}}(r; \cup_{i=0}^k \mathcal{D}_n^{(i)})$
  $\hat{Q}_{k+1} \leftarrow \mathrm{argmin}_{Q \in \mathcal{F}^{|\mathcal{A}|}} \left\| Q(X_i, A_i) - \left(\hat{r}(X_i, A_i) + \gamma \int \hat{\mathcal{P}}^{(k+1)}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right) \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
  $\pi_{k+1} \leftarrow \hat{\pi}(\cdot; \hat{Q}_{k+1})$.
end for
return π_K

Algorithm 1 Model-based Reinforcement Learning Algorithm with Iterative VAML

// MDP $(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$
// K: Number of iterations
// $\mathcal{M}$: Space of transition probability kernels
// $\mathcal{F}^{|\mathcal{A}|}$: Space of action-value functions
// $\mathcal{G}$: Space of reward functions
Initialize a policy π_0 and a value function $\hat{V}_0$.
for $k = 0$ to $K - 1$ **do**
Generate training set $\mathcal{D}_n^{(k)} = \{(X_i, A_i, R_i, X_i')\}_{i=1}^n$ by interacting with the true environment (potentially using π_k), i.e., $(X_i, A_i) \sim \nu_k$ with $X_i' \sim \mathcal{P}^*(\cdot|X_i, A_i)$ and $R_i \sim \mathcal{R}^*(\cdot|X_i, A_i)$.
$\boxed{\hat{\mathcal{P}}^{(k+1)} \leftarrow \mathrm{argmin}_{\mathcal{P} \in \mathcal{M}} \left\| \hat{V}_k(X_i') - \int \mathcal{P}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}.}$ $\longleftarrow$ $\boxed{\hat{\mathcal{P}} V_k \approx \mathcal{P}^* V_k}$
$\hat{r} \leftarrow \mathrm{argmin}_{r \in \mathcal{G}} \mathrm{Loss}_{\mathcal{R}}(r; \cup_{i=0}^k \mathcal{D}_n^{(i)})$
$\hat{Q}_{k+1} \leftarrow \mathrm{argmin}_{Q \in \mathcal{F}^{|\mathcal{A}|}} \left\| Q(X_i, A_i) - \left(\hat{r}(X_i, A_i) + \gamma \int \hat{\mathcal{P}}^{(k+1)}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right) \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}.$
$\pi_{k+1} \leftarrow \hat{\pi}(\cdot; \hat{Q}_{k+1})$.
end for
return π_K

Algorithm 1 Model-based Reinforcement Learning Algorithm with Iterative VAML

// MDP $(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$
// K: Number of iterations
// $\mathcal{M}$: Space of transition probability kernels
// $\mathcal{F}^{|\mathcal{A}|}$: Space of action-value functions
// $\mathcal{G}$: Space of reward functions
Initialize a policy π_0 and a value function $\hat{V}_0$.
for $k = 0$ to $K - 1$ **do**
    Generate training set $\mathcal{D}_n^{(k)} = \{(X_i, A_i, R_i, X_i')\}_{i=1}^n$ by interacting with the true environment (potentially using π_k), i.e., $(X_i, A_i) \sim \nu_k$ with $X_i' \sim \mathcal{P}^*(\cdot|X_i, A_i)$ and $R_i \sim \mathcal{R}^*($

$$\hat{Q}_{k+1} \leftarrow T^*_{\hat{\mathcal{P}}^{(k)}} \hat{Q}_k$$

    $\hat{\mathcal{P}}^{(k+1)} \leftarrow \mathrm{argmin}_{\mathcal{P} \in \mathcal{M}} \left\| \hat{V}_k(X_i') - \int \mathcal{P}(\mathrm{d}x'|X_i, A_i) \hat{V}_k(x') \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$
    $\hat{r} \leftarrow \mathrm{argmin}_{r \in \mathcal{G}} \mathrm{Loss}_{\mathcal{R}}(r; \cup_{i=0}^k \mathcal{D}_n^{(i)})$
    $\hat{Q}_{k+1} \leftarrow \mathrm{argmin}_{Q \in \mathcal{F}^{|\mathcal{A}|}} \left\| Q(X_i, A_i) - \left(\hat{r}(X_i, A_i) + \gamma \int \hat{\mathcal{P}}^{(k+1)}(\mathrm{d}x'|X_i, A_i) \hat{V}_k(x') \right) \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
    $\pi_{k+1} \leftarrow \hat{\pi}(\cdot; \hat{Q}_{k+1})$.
end for
return π_K

Algorithm 1 Model-based Reinforcement Learning Algorithm with Iterative VAML

// MDP $(\mathcal{X}, \mathcal{A}, \mathcal{R}^*, \mathcal{P}^*, \gamma)$
// K: Number of iterations
// $\mathcal{M}$: Space of transition probability kernels
// $\mathcal{F}^{|\mathcal{A}|}$: Space of action-value functions
// $\mathcal{G}$: Space of reward functions
Initialize a policy π_0 and a value function $\hat{V}_0$.
for $k = 0$ to $K - 1$ **do**
  Generate training set $\mathcal{D}_n^{(k)} = \{(X_i, A_i, R_i, X_i')\}_{i=1}^n$ by interacting with the true environment (potentially using π_k), i.e., $(X_i, A_i) \sim \nu_k$ with $X_i' \sim \mathcal{P}^*(\cdot|X_i, A_i)$ and $R_i \sim \mathcal{R}^*(\cdot|X_i, A_i)$.
  $\hat{\mathcal{P}}^{(k+1)} \leftarrow \text{argmin}_{\mathcal{P} \in \mathcal{M}} \left\| \hat{V}_k(X_i') - \int \mathcal{P}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
  $\hat{r} \leftarrow \text{argmin}_{r \in \mathcal{G}} \text{Loss}_{\mathcal{R}}(r; \cup_{i=0}^k \mathcal{D}_n^{(i)})$
  $\hat{Q}_{k+1} \leftarrow \text{argmin}_{Q \in \mathcal{F}^{|\mathcal{A}|}} \left\| Q(X_i, A_i) - \left(\hat{r}(X_i, A_i) + \gamma \int \hat{\mathcal{P}}^{(k+1)}(\mathrm{d}x'|X_i, A_i)\hat{V}_k(x') \right) \right\|^2_{\cup_{i=0}^k \mathcal{D}_n^{(i)}}$.
  $\boxed{\pi_{k+1} \leftarrow \hat{\pi}(\cdot; \hat{Q}_{k+1}).}$
end for
return π_K

Greedy policy wrt Q:

$$\hat{\pi}(x; Q) = \text{argmax}_{a \in \mathcal{A}} Q(x, a)$$

Iterative VAML

Summary

- VAML brings value function into model learning
- VAML uses worst case value function for robustness
    - Requires solving minimax
- IterVAML uses intermediate value functions from AVI
    - Intuition: Why always use worst case when intermediate results are available for use and approaches the true value function

Theoretical Results

Main question we want to answer:

How do various errors affect the quality of the outcome policy?

Approach:

1. What's the error in **one iteration** of model learning?
2. How do errors **propagate** throughout iterations and affect the **final policy**?
3. Putting the two together will give us the answer!

What’s the error in one iteration of model learning?

Formally, we want to provide a bound on the error $\|(\hat{\mathcal{P}}^{(k+1)} - \mathcal{P}^*)V_k\|_2$

Insight 1:

Express in terms of the best possible error given our model class + a constant

- i.e. control the **excess error**

What are the sources of excess error?

Insight 2:

Error is introduced whenever we do **empirical estimates** of our loss function

Main intuitive idea**:**

Control excess error by providing **probabilistic** bounds on how far away **empirical things** might get from their expected value

What's the error in one iteration of model learning?

Compare and contrast **true** loss function vs. **empirical** loss function

True (what we want)

$$l(z;\mathcal{P}) \triangleq |(\mathcal{P}_z - \mathcal{P}^*_z)V|^2$$

$$L(\mathcal{P}) = \mathbb{E}\left[l(Z;\mathcal{P})\right] = \|(\mathcal{P}_z - \mathcal{P}^*_z)V\|^2_{2,\nu}$$

$$L_n(\mathcal{P}) = \frac{1}{n}\sum_{i=1}^{n} l(Z_i;\mathcal{P}),$$

Depends on P* which we don't have

Empirical (what we have)

Single-sample estimate of P*

$$\hat{l}(z, x'; \mathcal{P}) = |\mathcal{P}_z V - V(x')|^2$$

$$\hat{L}(\mathcal{P}) = \mathbb{E}[\hat{l}(Z, X'; \mathcal{P})]$$

$$\hat{L}_n(\mathcal{P}) = \frac{1}{n}\sum_{i=1}^{n} \hat{l}(Z_i, X'_i; \mathcal{P}).$$

Additional Monte carlo estimate

What's the error in one iteration of model learning?

Types of loss functions and the sources of error

	Real — Depends on P* which we don't have	Estimate — Single-sample estimate of P*
Pointwise	$l(z;\mathcal{P}) \triangleq \lvert(\mathcal{P}_z - \mathcal{P}_z^*)V\rvert^2$	$\hat{l}(z,x';\mathcal{P}) = \lvert\mathcal{P}_z V - V(x')\rvert^2$
Population	$L(\mathcal{P}) = \mathbb{E}\left[l(Z;\mathcal{P})\right]$	
Empirical Monte carlo estimate	$L_n(\mathcal{P}) = \frac{1}{n}\sum_{i=1}^{n} l(Z_i;\mathcal{P}),$	$\hat{L}_n(\mathcal{P}) = \frac{1}{n}\sum_{i=1}^{n} \hat{l}(Z_i,X_i';\mathcal{P}).$

What’s the error in one iteration of model learning?

Reminder: We wanted to control **excess error**, which is how much worse is our solution vs the best possible

$$\tilde{\mathcal{P}} \leftarrow \operatorname*{argmin}_{\mathcal{P}\in\mathcal{M}} L(\mathcal{P}).$$

Best possible

$$\hat{\mathcal{P}} \leftarrow \operatorname*{argmin}_{\mathcal{P}\in\mathcal{M}} \hat{L}_n(\mathcal{P})$$

IterVAML’s solution

Formally, excess error is controlled if we can show

$$L(\hat{\mathcal{P}}) \leq L(\tilde{\mathcal{P}}) + C(1/\delta)$$ **with probability at least** $1-\delta$

What's the error in one iteration of model learning?

To show $L(\hat{\mathcal{P}}) - L(\tilde{\mathcal{P}}) \leq C(1/\delta)$ **with probability at least** $1-\delta$

Relate population real $L(\hat{\mathcal{P}}) - L(\tilde{\mathcal{P}})$ and empirical real $L_n(\hat{\mathcal{P}}) - L_n(\tilde{\mathcal{P}})$

Relate empirical real $L_n(\mathcal{P})$ and empirical estimate $\hat{L}_n(\mathcal{P})$

Population real
The thing we actually care about

Empirical real
Intermediate step

Empirical estimate
The only thing we can actually compute

Goal: Express **population real** in terms of **empirical real** in terms of **empirical estimate** in terms of <u>**constants**</u>

What's the error in one iteration of model learning?

Relate population real $L(\hat{\mathcal{P}}) - L(\tilde{\mathcal{P}})$ and empirical real $L_n(\hat{\mathcal{P}}) - L_n(\tilde{\mathcal{P}})$

Let's define a space of functions that maps (s,a) to excess error

$$\mathcal{G} = \{\, z \mapsto l_{\mathcal{P}}(z) - l_{\tilde{\mathcal{P}}}(z) \,:\, \mathcal{P} \in \mathcal{M} \,\}.$$

Bartlett et al. [2005] showed with probability at least $1 - \delta_1$

$$L(\mathcal{P}) - L(\tilde{\mathcal{P}}) \leq 2\left(L_n(\mathcal{P}) - L_n(\tilde{\mathcal{P}})\right) + \frac{2c_1}{B} r^*(\mathcal{G}) + \frac{(11 \times 8V_{\max}^2 + 2c_2 B)\ln(\frac{1}{\delta_1})}{n}$$

What’s the error in one iteration of model learning?

How did we do this?

$$\mathcal{G} = \{ z \mapsto l_{\mathcal{P}}(z) - l_{\tilde{\mathcal{P}}}(z) \; : \; \mathcal{P} \in \mathcal{M} \}.$$

$$L(\mathcal{P}) - L(\tilde{\mathcal{P}}) \leq 2\left(L_n(\mathcal{P}) - L_n(\tilde{\mathcal{P}})\right) + \frac{2c_1}{B} r^*(\mathcal{G}) + \frac{(11 \times 8V_{\max}^2 + 2c_2 B)\ln(\frac{1}{\delta_1})}{n}$$

Required assumption:

A certain model space complexity → bounded local Rademacher complexity of G

Required assumption:
Value function is bounded

Intuition: **Empirical real** loss probably lies close to **population real** loss if make assumptions on

1. Complexity of the model function space
2. Boundedness of value function

What's the error in one iteration of model learning?

Relate empirical real $L_n(\mathcal{P})$ and empirical estimate $\hat{L}_n(\mathcal{P})$

$$\hat{l}(z, x'; \mathcal{P}) = |\mathcal{P}_z V - V(x')|^2 = |\mathcal{P}_z V - \mathcal{P}_z^* V + \mathcal{P}_z^* V - V(x')|^2$$

$$= \underbrace{|(\mathcal{P}_z - \mathcal{P}_z^*)|^2}_{= l_{\mathcal{P}}(z)} + |\mathcal{P}_z^* V - V(x')|^2 + 2\left[(\mathcal{P}_z - \mathcal{P}_z^*)V\right]\left[\mathcal{P}_z^* V - V(x')\right]$$

$$L_n(\mathcal{P}) = \hat{L}_n(\mathcal{P}) - \underbrace{\frac{1}{n}\sum_{i=1}^{n} |\mathcal{P}_{Z_i}^* V - V(X_i')|^2}_{\triangleq e_\sigma} + \underbrace{2 \times \frac{1}{n}\sum_{i=1}^{n} \left[(\mathcal{P}_{Z_i}^* - \mathcal{P}_{Z_i})V\right]\left[\mathcal{P}_{Z_i}^* V - V(X_i')\right]}_{\triangleq e_I(\mathcal{P})}.$$

What’s the error in one iteration of model learning?

$$L_n(\mathcal{P}) = \hat{L}_n(\mathcal{P}) - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\left|\mathcal{P}^*_{Z_i}V - V(X'_i)\right|^2}_{\triangleq e_\sigma} + 2\times \underbrace{\frac{1}{n}\sum_{i=1}^{n}\left[(\mathcal{P}^*_{Z_i} - \mathcal{P}_{Z_i})V\right]\left[\mathcal{P}^*_{Z_i}V - V(X'_i)\right]}_{\triangleq e_I(\mathcal{P})}.$$

Step 1: Rearrange equation so green is on the left and red is on the right
Step 2: Bound red in terms of constants

$$\begin{aligned}
J(\hat{\mathcal{P}}, \tilde{\mathcal{P}}) &\le c(\alpha, V_{\max}, R) \left\| (\hat{\mathcal{P}}_{Z_i} - \tilde{\mathcal{P}}_{Z_i})V \right\|_n^{1-\alpha} t_n \\
&\le c(\alpha, V_{\max}, R) t_n \times \\
&\quad \left[2\mathbb{E}\left[\left| (\hat{\mathcal{P}} - \tilde{\mathcal{P}})V \right|^2 \right] + \frac{c_1}{4V_{\max}^2} r^*(\mathcal{F}) + \frac{(44V_{\max}^2 + c_2 \times 4V_{\max}^2)\ln(1/\delta_3)}{n} \right]^{\frac{1-\alpha}{2}} \\
&\le c(\alpha, V_{\max}, R) t_n \times \\
&\quad \left[2\left(L(\hat{\mathcal{P}}) - L(\tilde{\mathcal{P}}) \right) + \frac{c_1}{4V_{\max}^2} r^*(\mathcal{F}) + \frac{(88V_{\max}^2 + c_2 \times 4V_{\max}^2)\ln(1/\delta_3)}{n} \right]^{\frac{1-\alpha}{2}},
\end{aligned}$$

What’s the error in one iteration of model learning?

$$L_n(\mathcal{P}) = \hat{L}_n(\mathcal{P}) - \underbrace{\frac{1}{n}\sum_{i=1}^{n}\left|\mathcal{P}^*_{Z_i}V - V(X'_i)\right|^2}_{\triangleq e_\sigma} + 2 \times \underbrace{\frac{1}{n}\sum_{i=1}^{n}\left[(\mathcal{P}^*_{Z_i} - \mathcal{P}_{Z_i})V\right]\left[\mathcal{P}^*_{Z_i}V - V(X'_i)\right]}_{\triangleq e_I(\mathcal{P})}.$$

Step 1: Rearrange equation so green is on the left and red is on the right
Step 2: Bound red in terms of constants

Empirical real also probably lies close to **empirical estimates,** which can subsequently be expressed as constants by making the same assumptions on:

1. Complexity of the model space
2. Boundedness of value function

What's the error in one iteration of model learning?

With a lot of work, we can put everything together

A1. New i.i.d. Data
A2. Model capacity
A3. Bounded Value

Theorem 1. *Suppose that Assumptions A1, A2, and A3 hold. Consider $\hat{\mathcal{P}}$ obtained by solving (11). There exists a finite $c(\alpha) > 0$, depending only on α, such that for any $\delta > 0$, with probability at least $1 - \delta$, we have*

$$\left\|(\hat{\mathcal{P}}_z - \mathcal{P}_z^*)V\right\|_{2,\nu}^2 \leq \inf_{\mathcal{P} \in \mathcal{M}} \|(\mathcal{P}_z - \mathcal{P}_z^*)V\|_{2,\nu}^2 + \frac{c(\alpha) V_{max}^2 R^{\frac{2\alpha}{1+\alpha}} \sqrt{\log(1/\delta)}}{n^{\frac{1}{1+\alpha}}}.$$

$$L(\hat{\mathcal{P}}) \leq L(\tilde{\mathcal{P}}) + C(1/\delta)$$

What's the error in one iteration of model learning?

With a lot of work, we can put everything together

A1. New i.i.d. Data
A2. Model capacity
A3. Bounded Value

Theorem 1. *Suppose that Assumptions A1, A2, and A3 hold. Consider $\hat{\mathcal{P}}$ obtained by solving (11). There exists a finite $c(\alpha) > 0$, depending only on α, such that for any $\delta > 0$, with probability at least $1 - \delta$, we have*

$$\left\|(\hat{\mathcal{P}}_z - \mathcal{P}_z^*)V\right\|_{2,\nu}^2 \leq \inf_{\mathcal{P} \in \mathcal{M}} \|(\mathcal{P}_z - \mathcal{P}_z^*)V\|_{2,\nu}^2 + \frac{c(\alpha) V_{max}^2 R^{\frac{2\alpha}{1+\alpha}} \sqrt{\log(1/\delta)}}{n^{\frac{1}{1+\alpha}}}.$$

Model Learning Error

Approximation Error

Estimation error

How do errors propagate and affect the final policy?

How do errors propagate and affect the final policy?

Let's consider a sequence of $\hat{Q}_0, \hat{Q}_1, \ldots, \hat{Q}_K$,

and the final resulting policy $\pi_K(x) = argmax_{a \in \mathcal{A}} \hat{Q}(x, a)$.

How do errors propagate and affect the final policy?

Let's consider a sequence of $\hat{Q}_0, \hat{Q}_1, \ldots, \hat{Q}_K$,

and the final resulting policy $\pi_K(x) = argmax_{a \in \mathcal{A}} \hat{Q}(x, a)$.

We are interested in $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

How do errors propagate and affect the final policy?

Let's consider a sequence of $\hat{Q}_0, \hat{Q}_1, \ldots, \hat{Q}_K$,

and the final resulting policy $\pi_K(x) = argmax_{a \in \mathcal{A}} \hat{Q}(x, a)$.

We are interested in $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

Optimal value

Value of $\pi_K(x)$

How do errors propagate and affect the final policy?

Let's consider a sequence of $\hat{Q}_0, \hat{Q}_1, \ldots, \hat{Q}_K$,

and the final resulting policy $\pi_K(x) = argmax_{a \in \mathcal{A}} \hat{Q}(x, a)$.

We are interested in $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

Optimal value

Value of $\pi_K(x)$

Sources of error?

How do errors propagate and affect the final policy?

Let's consider a sequence of $\hat{Q}_0, \hat{Q}_1, \ldots, \hat{Q}_K$,

and the final resulting policy $\pi_K(x) = argmax_{a \in \mathcal{A}} \hat{Q}(x, a)$.

We are interested in $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

Modeling error

$$e_k = \left(\mathcal{P}^* - \hat{\mathcal{P}}^{(k+1)}\right) \max_{a'} \hat{Q}_k(\cdot, a'), \qquad k = 0, 1, \ldots, K-1$$

We just upper bounded this in Theorem 1!

How do errors propagate and affect the final policy?

Let's consider a sequence of $\hat{Q}_0, \hat{Q}_1, \ldots, \hat{Q}_K$,

and the final resulting policy $\pi_K(x) = argmax_{a \in \mathcal{A}} \hat{Q}(x, a)$.

We are interested in $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

Modeling error

$$e_k = \left(\mathcal{P}^* - \hat{\mathcal{P}}^{(k+1)}\right) \max_{a'} \hat{Q}_k(\cdot, a'), \qquad k = 0, 1, \ldots, K-1$$

Regression error

$$\varepsilon_k = T^*_{\hat{\mathcal{P}}^{(k+1)}} \hat{Q}_k - \hat{Q}_{k+1}, \qquad k = 0, 1, \ldots, K-1$$

How do errors propagate and affect the final policy?

We want to represent $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

Modeling error

$$e_k = \left(\mathcal{P}^* - \hat{\mathcal{P}}^{(k+1)}\right) \max_{a'} \hat{Q}_k(\cdot, a'), \qquad k = 0, 1, \ldots, K-1$$

We just upper bounded this in Theorem 1!

How do errors propagate and affect the final policy?

We want to represent $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

Modeling error

$$e_k = \left(\mathcal{P}^* - \hat{\mathcal{P}}^{(k+1)}\right) \max_{a'} \hat{Q}_k(\cdot, a'), \qquad k = 0, 1, \ldots, K-1$$

Regression error

$$\varepsilon_k = T^*_{\hat{\mathcal{P}}^{(k+1)}} \hat{Q}_k - \hat{Q}_{k+1}, \qquad k = 0, 1, \ldots, K-1$$

We want $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

We want $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

How good we **actually** are at the end

We want $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

How good we **actually** are at the end

How good we **think** we are at the end

We want $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

   How good we **actually** are at the end

   How good we **think** we are at the end

2. Represent $Q^* - \hat{Q}_K$ in terms of $Q^* - \hat{Q}_{k'}$

   and ε_k and e_k (error propagation)

   How good we **think** we are at each step

We want $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

   How good we **actually** are at the end

   How good we **think** we are at the end

2. Represent $Q^* - \hat{Q}_K$ in terms of $Q^* - \hat{Q}_k$,

   and ε_k and e_k (error propagation)

   How good we **think** we are at each step

3. Take expectation of $Q^* - Q^{\pi_K}$ to get $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

$$\left(Q^* - Q^{\pi_K}\right) \leq \gamma \left(\mathbf{I} - \gamma \mathcal{P}^{\pi_K}\right)^{-1} \left(\mathcal{P}^{\pi^*} - \mathcal{P}^{\pi_K}\right) \left(Q^* - \hat{Q}_K\right)$$

2. Represent $Q^* - \hat{Q}_K$ in terms of $Q^* - \hat{Q}_k$, ε_k and e_k

We first upper and lower bound $Q^* - \hat{Q}_k$

2. Represent $Q^* - \hat{Q}_K$ in terms of $Q^* - \hat{Q}_k$, ε_k and e_k

We first upper and lower bound $Q^* - \hat{Q}_k$

$$Q^* - \hat{Q}_{k+1} \leq \gamma \mathcal{P}^{\pi^*} (Q^* - \hat{Q}_k) + \Delta_k$$

and

$$Q^* - \hat{Q}_{k+1} \geq \gamma \mathcal{P}^{\pi_k} \left(Q^* - \hat{Q}_k \right) + \Delta_k$$

Where $\Delta_k = \varepsilon_k + \gamma e_k$

2. Represent $Q^* - \hat{Q}_K$ in terms of $Q^* - \hat{Q}_k$, ε_k and e_k

We first upper and lower bound $Q^* - \hat{Q}_k$

$$Q^* - \hat{Q}_{k+1} \leq \gamma \mathcal{P}^{\pi^*} (Q^* - \hat{Q}_k) + \Delta_k$$

and

$$Q^* - \hat{Q}_{k+1} \geq \gamma \mathcal{P}^{\pi_k} \left(Q^* - \hat{Q}_k\right) + \Delta_k$$

Where $\Delta_k = \varepsilon_k + \gamma e_k$

Q: How do we get from $Q^* - \hat{Q}_{k+1}$ to $Q^* - \hat{Q}_K$?

2. Represent $Q^* - \hat{Q}_K$ in terms of $Q^* - \hat{Q}_k$, ε_k and e_k

By **induction,**

$$Q^* - \hat{Q}_{k+1} \leq \gamma \mathcal{P}^{\pi^*}(Q^* - \hat{Q}_k) + \Delta_k$$

$$\vdots$$

$$Q^* - \hat{Q}_K \leq \sum_{k=0}^{K-1} \gamma^{K-k-1} (\mathcal{P}^{\pi^*})^{K-k-1} \Delta_k + \gamma^K (\mathcal{P}^{\pi^*})^K (Q^* - \hat{Q}_0)$$

where $\Delta_k = \varepsilon_k + \gamma e_k$

2. Represent $Q^* - \hat{Q}_K$ in terms of $Q^* - \hat{Q}_k$, ε_k and e_k

By **induction,**

$$Q^* - \hat{Q}_{k+1} \leq \gamma \mathcal{P}^{\pi^*}(Q^* - \hat{Q}_k) + \Delta_k$$

$$\vdots$$

$$Q^* - \hat{Q}_K \leq \sum_{k=0}^{K-1} \gamma^{K-k-1} (\mathcal{P}^{\pi^*})^{K-k-1} \Delta_k + \gamma^K (\mathcal{P}^{\pi^*})^K (Q^* - \hat{Q}_0)$$

where $\Delta_k = \varepsilon_k + \gamma e_k$

2. Represent $Q^* - \hat{Q}_K$ in terms of $Q^* - \hat{Q}_k$, ε_k and e_k

Similarly, by **induction**,

$$Q^* - \hat{Q}_{k+1} \geq \gamma \mathcal{P}^{\pi_k} \left(Q^* - \hat{Q}_k \right) + \Delta_k$$

$$\vdots$$

$$Q^* - \hat{Q}_K \geq \sum_{k=0}^{K-1} \gamma^{K-1-k} \left(\mathcal{P}^{\pi_{K-1}} \cdots \mathcal{P}^{\pi_{k+1}} \right) \Delta_k + \gamma^K \left(\mathcal{P}^{\pi_{K-1}} \cdots \mathcal{P}^{\pi_0} \right) \left(Q^* - \hat{Q}_0 \right)$$

where $\Delta_k = \varepsilon_k + \gamma e_k$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

$$\left(Q^* - Q^{\pi_K}\right) \leq \gamma \left(\mathbf{I} - \gamma \mathcal{P}^{\pi_K}\right)^{-1} \left(\mathcal{P}^{\pi^*} - \mathcal{P}^{\pi_K}\right) \left(Q^* - \hat{Q}_K\right)$$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

$$\left(Q^* - Q^{\pi_K}\right) \leq \gamma \left(\mathbf{I} - \gamma \mathcal{P}^{\pi_K}\right)^{-1} \left(\mathcal{P}^{\pi^*} - \mathcal{P}^{\pi_K}\right) \left(Q^* - \hat{Q}_K\right)$$

$$Q^* - \hat{Q}_K \leq \sum_{k=0}^{K-1} \gamma^{K-k-1} (\mathcal{P}^{\pi^*})^{K-k-1} \Delta_k + \gamma^K (\mathcal{P}^{\pi^*})^K (Q^* - \hat{Q}_0)$$

$$Q^* - \hat{Q}_K \geq \sum_{k=0}^{K-1} \gamma^{K-1-k} \left(\mathcal{P}^{\pi_{K-1}} \cdots \mathcal{P}^{\pi_{k+1}}\right) \Delta_k + \gamma^K \left(\mathcal{P}^{\pi_{K-1}} \cdots \mathcal{P}^{\pi_0}\right) \left(Q^* - \hat{Q}_0\right)$$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

$$Q^* - Q^{\pi_K} \leq \gamma \left(\mathbf{I} - \gamma \mathcal{P}^{\pi_K}\right)^{-1} \left[\sum_{k=0}^{K-1} \gamma^{K-k-1} \left(\left(\mathcal{P}^{\pi^*}\right)^{K-k} + \left(\mathcal{P}^{\pi_K} \cdots \mathcal{P}^{\pi_{k+1}}\right) \right) |\Delta_k| + \right.$$

$$\left. \gamma^K \left(\left(\mathcal{P}^{\pi^*}\right)^{K+1} + \left(\mathcal{P}^{\pi_K} \cdots \mathcal{P}^{\pi_0}\right) \right) \left| Q^* - \hat{Q}_0 \right| \right]$$

1. Represent $Q^* - Q^{\pi_K}$ in terms of $Q^* - \hat{Q}_K$

$$Q^* - Q^{\pi_K} \leq \gamma \left(\mathbf{I} - \gamma \mathcal{P}^{\pi_K}\right)^{-1} \left[\sum_{k=0}^{K-1} \gamma^{K-k-1} \left((\mathcal{P}^{\pi^*})^{K-k} + (\mathcal{P}^{\pi_K} \cdots \mathcal{P}^{\pi_{k+1}}) \right) |\Delta_k| + \right.$$

$$\left. \gamma^K \left((\mathcal{P}^{\pi^*})^{K+1} + (\mathcal{P}^{\pi_K} \cdots \mathcal{P}^{\pi_0}) \right) \left| Q^* - \hat{Q}_0 \right| \right]$$

To simplify notation:

$$Q^* - Q^{\pi_K} \leq \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k A_k |\Delta_k| + \alpha_K A_K |Q^* - \hat{Q}_0| \right]$$

3. Take expectation of $Q^* - Q^{\pi_K}$ to get $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

$$Q^* - Q^{\pi_K} \leq \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k A_k |\Delta_k| + \alpha_K A_K |Q^* - \hat{Q}_0| \right]$$

$$\Downarrow$$

$$\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k \rho A_k |\Delta_k| + \alpha_K \rho A_K |Q^* - \hat{Q}_0| \right]$$

3. Take expectation of $Q^* - Q^{\pi_K}$ to get $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

$$\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k \rho A_k |\Delta_k| + \alpha_K \rho A_K |Q^* - \hat{Q}_0| \right]$$

3. Take expectation of $Q^* - Q^{\pi_K}$ to get $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

$$\|Q^* - Q^{\pi_K}\|_{1,\rho} \le \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k \rho A_k |\Delta_k| + \alpha_K \rho A_K |Q^* - \hat{Q}_0| \right]$$

3. Take expectation of $Q^* - Q^{\pi_K}$ to get $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

$$\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k \rho A_k |\Delta_k| + \alpha_K \rho A_K |Q^* - \hat{Q}_0| \right]$$

1. Upper-bound $|Q^* - \hat{Q}_0|$ using the fact that $|Q^* - \hat{Q}_0| \leq 2V_{\max}$

3. Take expectation of $Q^* - Q^{\pi_K}$ to get $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

$$\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k \rho A_k |\Delta_k| + \alpha_K \rho A_K |Q^* - \hat{Q}_0| \right]$$

1. Upper-bound $|Q^* - \hat{Q}_0|$ using the fact that $|Q^* - \hat{Q}_0| \leq 2V_{\max}$
2. Allow expectation of Δ_k w.r.t. data generating distribution ν

3. Take expectation of $Q^* - Q^{\pi_K}$ to get $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

$$\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k \rho A_k |\Delta_k| + \alpha_K \rho A_K |Q^* - \hat{Q}_0| \right]$$

1. Upper-bound $|Q^* - \hat{Q}_0|$ using the fact that $|Q^* - \hat{Q}_0| \leq 2V_{\max}$

2. Allow expectation of Δ_k w.r.t. data generating distribution ν

Recall: $\Delta_k = \varepsilon_k + \gamma e_k$

$$\varepsilon_k = T^*_{\hat{\mathcal{P}}(k+1)} \hat{Q}_k - \hat{Q}_{k+1}, \qquad k = 0, 1, \ldots, K-1$$

$$e_k = \left(\mathcal{P}^* - \hat{\mathcal{P}}^{(k+1)} \right) \max_{a'} \hat{Q}_k(\cdot, a'), \qquad k = 0, 1, \ldots, K-1$$

3. Take expectation of $Q^* - Q^{\pi_K}$ to get $\|Q^* - Q^{\pi_K}\|_{1,\rho}$

$$\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \lambda_K \left[\sum_{k=0}^{K-1} \alpha_k \rho A_k |\Delta_k| + \alpha_K \rho A_K |Q^* - \hat{Q}_0| \right]$$

1. Upper-bound $\left|Q^* - \hat{Q}_0\right|$ using the fact that $\left|Q^* - \hat{Q}_0\right| \leq 2V_{\max}$

2. Allow expectation of Δ_k w.r.t. data generating distribution ν

$$\Downarrow$$

$$\boxed{\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \frac{2\gamma}{(1-\gamma)^2} \left[\bar{C}(\rho, \nu) \max_{0 \leq k \leq K-1} \left(\|\varepsilon_k\|_{2,\nu} + \gamma \|e_k\|_{2,\nu} \right) + 2\gamma^K R_{max} \right]}$$

How do errors propagate and affect the final policy?

Theorem 2. *Consider a sequence of action-value function* $(\hat{Q}_k)_{k=0}^K$*, and their corresponding* $(\hat{V}_k)_{k=0}^K$*, each of which is defined as* $\hat{V}_k(x) = \max_a \hat{Q}_k(x, a)$*. Suppose that the MDP is such that the expected rewards are* R_{max}*-bounded, and* $\hat{Q}_0$ *is initialized such that it is* $V_{max} \leq \frac{R_{max}}{1-\gamma}$*-bounded. Let* $\varepsilon_k = T^*_{\hat{\mathcal{P}}^{(k+1)}} \hat{Q}_k - \hat{Q}_{k+1}$ *(regression error) and* $e_k = (\mathcal{P}^* - \hat{\mathcal{P}}^{(k+1)})\hat{V}_k$ *(modelling error) for* $k = 0, 1, \ldots, K-1$*. Let* π_K *be the greedy policy w.r.t.* $\hat{Q}_K$*, i.e.,* $\pi_K(x) = \text{argmax}_{a \in \mathcal{A}} \hat{Q}(x, a)$ *for all* $x \in \mathcal{X}$*. Consider probability distributions* $\rho, \nu \in \bar{\mathcal{M}}(\mathcal{X} \times \mathcal{A})$*. We have*

$$\boxed{\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \frac{2\gamma}{(1-\gamma)^2} \left[\bar{C}(\rho, \nu) \max_{0 \leq k \leq K-1} \left(\|\varepsilon_k\|_{2,\nu} + \gamma \|e_k\|_{2,\nu} \right) + 2\gamma^K R_{max} \right]}$$

How do errors propagate and affect the final policy?

1. We want Q* - Q^pi_K
2. Get a bound for Q* - Qhat K in terms of delta
    a. Start off with Q*- Qhat k+1
    b. Express Qhatk+1 with delta error
    c. Get upper and lower bound using P* and Ppi_k respectively
3. Relate Q* - Q^pi_k to Q* - Qhat K by adding and subtracting term
4. Add rho
5. Remove Q* - Q0 by Vmax
6. Substitute in c(rho, nu) to get expectation in data

How do errors propagate and affect the final policy?

Theorem 2. *Consider a sequence of action-value function* $(\hat{Q}_k)_{k=0}^{K}$*, and their corresponding* $(\hat{V}_k)_{k=0}^{K}$*, each of which is defined as* $\hat{V}_k(x) = \max_a \hat{Q}_k(x,a)$*. Suppose that the MDP is such that the expected rewards are* R_{max}*-bounded, and* $\hat{Q}_0$ *is initialized such that it is* $V_{max} \leq \frac{R_{max}}{1-\gamma}$*-bounded. Let* $\varepsilon_k = T^*_{\hat{\mathcal{P}}^{(k+1)}}\hat{Q}_k - \hat{Q}_{k+1}$ *(regression error) and* $e_k = (\mathcal{P}^* - \hat{\mathcal{P}}^{(k+1)})\hat{V}_k$ *(modelling error) for* $k = 0, 1, \ldots, K-1$*. Let* π_K *be the greedy policy w.r.t.* $\hat{Q}_K$*, i.e.,* $\pi_K(x) = \text{argmax}_{a \in \mathcal{A}} \hat{Q}(x,a)$ *for all* $x \in \mathcal{X}$*. Consider probability distributions* $\rho, \nu \in \bar{\mathcal{M}}(\mathcal{X} \times \mathcal{A})$*. We have*

$$\boxed{\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \frac{2\gamma}{(1-\gamma)^2}\left[\bar{C}(\rho,\nu) \max_{0 \leq k \leq K-1}\left(\|\varepsilon_k\|_{2,\nu} + \gamma\|e_k\|_{2,\nu}\right) + 2\gamma^K R_{max}\right]}$$

Use Theorem 1 to bring everything together

How do errors propagate and affect the final policy?

From Theorem 1:

$$\|e_k\|_{2,\nu}^2 = \left\|(\hat{\mathcal{P}}_z^{(k+1)} - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2 \leq \inf_{\mathcal{P}\in\mathcal{M}} \left\|(\mathcal{P}_z - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2 + \frac{c(\alpha)V_{\max}^2 R^{\frac{2\alpha}{1+\alpha}}\sqrt{\log(K/\delta)}}{n^{\frac{1}{1+\alpha}}}$$

with probability at least $1 - \delta/K$.

How do errors propagate and affect the final policy?

From Theorem 1:

$$\|e_k\|_{2,\nu}^2 = \left\|(\hat{\mathcal{P}}_z^{(k+1)} - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2 \leq \underline{\inf_{\mathcal{P}\in\mathcal{M}} \left\|(\mathcal{P}_z - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2} + \frac{c(\alpha)V_{\max}^2 R^{\frac{2\alpha}{1+\alpha}}\sqrt{\log(K/\delta)}}{n^{\frac{1}{1+\alpha}}}$$

with probability at least $1 - \delta/K$.

Since $\hat{V}_k$ is random, we upper bound the model approximation error

How do errors propagate and affect the final policy?

From Theorem 1:

$$\|e_k\|_{2,\nu}^2 = \left\|(\hat{\mathcal{P}}_z^{(k+1)} - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2 \leq \underline{\inf_{\mathcal{P}\in\mathcal{M}} \left\|(\mathcal{P}_z - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2} + \frac{c(\alpha)V_{\max}^2 R^{\frac{2\alpha}{1+\alpha}}\sqrt{\log(K/\delta)}}{n^{\frac{1}{1+\alpha}}}$$

with probability at least $1 - \delta/K$.

Since $\hat{V}_k$ is random, we upper bound the model approximation error

$$\sup_{V\in\mathcal{F}^+} \inf_{\mathcal{P}\in\mathcal{M}} \|(\mathcal{P}_z - \mathcal{P}_z^*)V\|_{2,\nu}$$

How do errors propagate and affect the final policy?

From Theorem 1:

$$\|e_k\|_{2,\nu}^2 = \left\|(\hat{\mathcal{P}}_z^{(k+1)} - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2 \leq \inf_{\mathcal{P}\in\mathcal{M}} \left\|(\mathcal{P}_z - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2 + \frac{c(\alpha)V_{\max}^2 R^{\frac{2\alpha}{1+\alpha}}\sqrt{\log(K/\delta)}}{n^{\frac{1}{1+\alpha}}}$$

with probability at least $1 - \delta/K$.

Since $\hat{V}_k$ is random, we upper bound the model approximation error

$$\sup_{V\in\mathcal{F}^+} \inf_{\mathcal{P}\in\mathcal{M}} \|(\mathcal{P}_z - \mathcal{P}_z^*)V\|_{2,\nu}$$

Apply union bound over all k such that all k = 0, …, K-1 satisfy with probability $1 - \delta$

How do errors propagate and affect the final policy?

From Theorem 1:

$$\|e_k\|_{2,\nu}^2 = \left\|(\hat{\mathcal{P}}_z^{(k+1)} - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2 \leq \inf_{\mathcal{P}\in\mathcal{M}} \left\|(\mathcal{P}_z - \mathcal{P}_z^*)\hat{V}_k\right\|_{2,\nu}^2 + \frac{c(\alpha)V_{\max}^2 R^{\frac{2\alpha}{1+\alpha}}\sqrt{\log(K/\delta)}}{n^{\frac{1}{1+\alpha}}}$$

with probability at least $1-\delta/K$.

Since $\hat{V}_k$ is random, we upper bound the model approximation error

$$\sup_{V\in\mathcal{F}^+} \inf_{\mathcal{P}\in\mathcal{M}} \|(\mathcal{P}_z - \mathcal{P}_z^*)V\|_{2,\nu}$$

Apply union bound over all k such that all k = 0, …, K-1 satisfy with probability $1-\delta$

$$\|e_k\|_{2,\nu}^2 \leq \sup_{V\in\mathcal{F}^+} \inf_{\mathcal{P}\in\mathcal{M}} \|(\mathcal{P}_z - \mathcal{P}_z^*)V\|_{2,\nu}^2 + \frac{c(\alpha)V_{\max}^2 R^{\frac{2\alpha}{1+\alpha}}\sqrt{\log(K/\delta)}}{n^{\frac{1}{1+\alpha}}}$$

How do errors propagate and affect the final policy?

Applying Theorem 2 with $\|e_k\|^2_{2,\nu}$ gives:

Theorem 3. *Consider the IterVAML procedure in which at the k-th iteration the model* $\hat{\mathcal{P}}^{(k+1)}$ *is obtained by solving* (11) *and* $\hat{Q}_{k+1}$ *is obtained by solving* (12). *Let* $\varepsilon_k = T^*_{\hat{\mathcal{P}}^{(k+1)}} \hat{Q}_k - \hat{Q}_{k+1}$ *be the regression error. Suppose that Assumptions A1, A2, and A4 hold. Consider the greedy policy* π_K *w.r.t.* $\hat{Q}_K$. *For any* $\rho \in \bar{\mathcal{M}}(\mathcal{X} \times \mathcal{A})$, *there exists a finite* $c(\alpha) > 0$, *depending only on* α, *such that for any* $\delta > 0$, *with probability at least* $1 - \delta$, *we have*

$$\|Q^* - Q^{\pi_K}\|_{1,\rho} \leq \frac{2\gamma}{(1-\gamma)^2} \left[\bar{C}(\rho, \nu) \left(\max_{0 \leq k \leq K-1} \|\varepsilon_k\|_{2,\nu} + \gamma e_{model}(n) \right) + 2\gamma^K R_{max} \right]$$

where

$$e_{model}(n) = \sup_{V \in \mathcal{F}^+} \inf_{\mathcal{P} \in \mathcal{M}} \|(\mathcal{P}_z - \mathcal{P}^*_z) V\|_{2,\nu} + \frac{c(\alpha) V_{max} R^{\frac{\alpha}{1+\alpha}} \sqrt[4]{\log(K/\delta)}}{n^{\frac{1}{2(1+\alpha)}}},$$

and $\mathcal{F}^+ = \left\{ \max_a Q(\cdot, a) \, : \, Q \in \mathcal{F}^{|\mathcal{A}|} \right\}$.

Limitations

- Lack of experiments (see Lambert et al., 2020)
  - Bounds might be vacuous empirically
- Value aware model learning is less transferable
- Requires assumptions on model space complexity

Contributions (recap)

- A decision-aware method for model based RL
  - Take into account how value based planner would use a model
- An easier optimization problem than prior work
  - Reuses some computation, tradeoff with robustness
- Theoretical analysis
  - What are the effects of errors on the final resulting policy?

Questions to Consider

- How does IterVAML save computation from VAML?
- Name 2 important assumptions needed error analysis
- What proof technique is used to get $Q^* - \hat{Q}_{k+1}$ to $Q^* - \hat{Q}_K$

Contributions (Recap)

Approximately one bullet for each of the following (the paper on 1 slide)

- Model based reinforcement suffers from objective mismatch
-
- What is the key limitation of prior work
- What is the key insight(s) (try to do in 1-3) of the proposed work
- What did they demonstrate by this insight? (tighter theoretical bounds, state of the art performance on X, etc)

Contributions (recap)

Analysis provided probabilistic guarantees on error in final resulting policy due to modeling and regression error propagation

>=1 slide

What conclusions are drawn from the results?

Are the stated conclusions fully supported by the results and references? If so, why? (Recap the relevant supporting evidences from the given results + refs)

Critique / Limitations / Open Issues

1 or more slides: What are the key limitations of the proposed approach / ideas? (e.g. does it require strong assumptions that are unlikely to be practical? Computationally expensive? Require a lot of data? Find only local optima?)

- If follow up work has addressed some of these limitations, include pointers to that. But don’t limit your discussion only to the problems / limitations that have already been addressed.