Deep PQR: Solving Inverse Reinforcement Learning using Anchor Actions

Presented by: Sinong Geng
Princeton University
Amazon

June, 2020 @ICML 2020

Authors

Sinong Geng
Princeton University

Houssam Nassif
Amazon

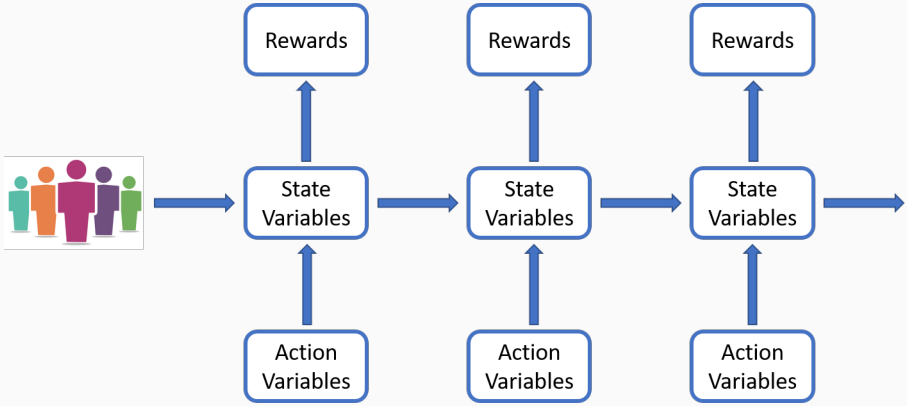
Carlos A. Manzanares
Amazon

A. Max Reppen
Princeton University

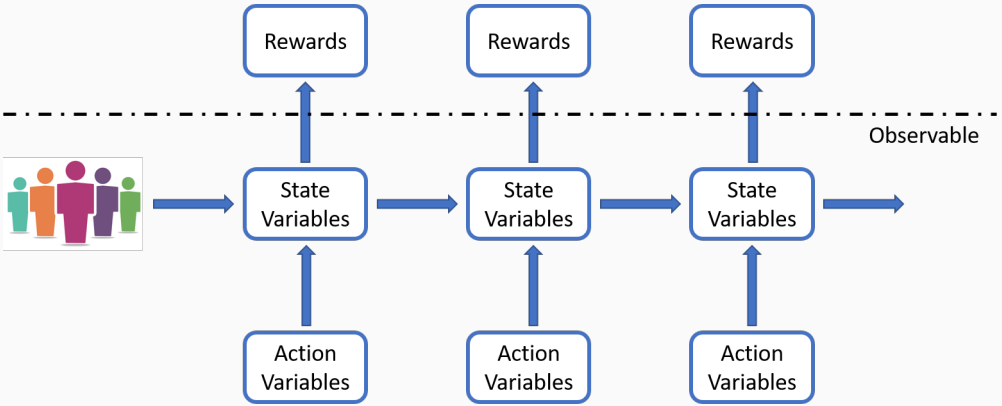
Ronnie Sircar
Princeton University

Supported by Amazon AWS Credits.
A. Max Reppen is supported by the Swiss National Science Foundation grant SNF 181815.

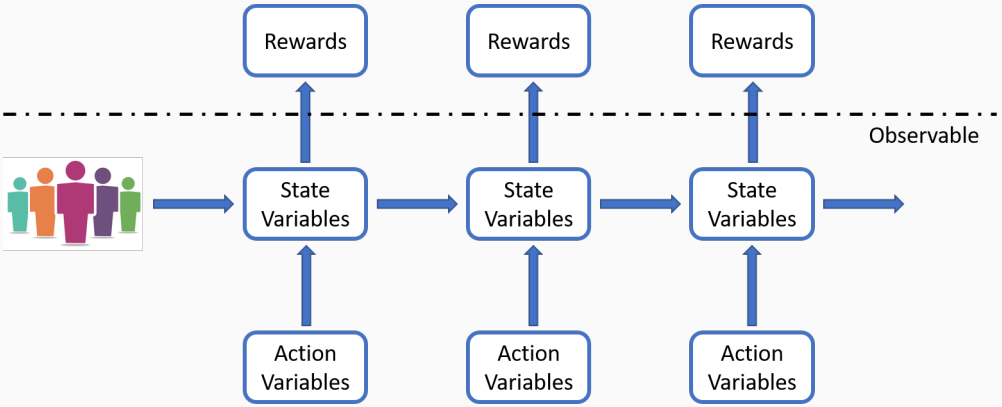
Inverse Optimal Control



Inverse Optimal Control



Inverse Optimal Control



→ **Goal**: Estimate reward functions from the (assumed optimal) demonstrations of agents.

Problem Formulation: Motivations

→ Economics: estimate the firm profit functions in industrial organization [Bajari et al., 2007].

→ Finance: estimate the risk tolerance of agents, suggesting how much compensation they require in exchange for asset volatility [Merton et al., 1973].

→ RL: replicate the decision-making behavior in novel environments [Fu et al., 2017].

Energy-based Modeling

- State Action Variables: $\mathbf{S}_t$ takes values in $\mathcal{S}$; $\mathbf{A}_t$ takes values in $\mathcal{A}$.
- Reward: We aim to estimate the reward $r(\mathbf{s}, \mathbf{a})$.
- Energy-based Stochastic Policy:

$$\pi(\mathbf{s}, \mathbf{a}) := \mathrm{P}(\mathbf{A}_t = \mathbf{a}_t \mid \mathbf{S}_t = \mathbf{s}) = \frac{\exp(-\mathcal{E}(\mathbf{s}, \mathbf{a}))}{\int_{\mathbf{a}' \in \mathcal{A}} \exp(-\mathcal{E}(\mathbf{s}, \mathbf{a}')d\mathbf{a}')}.$$

- Entropy-augmented reinforcement learning objective [Haarnoja et al., 2017]:

$$\begin{aligned} V(\mathbf{s}) &:= \max_{\pi} \sum_{t=0}^{\infty} \gamma^t \, \mathbb{E}[r(\mathbf{S}_t, \mathbf{A}_t) + \alpha \mathcal{H}(\pi(\mathbf{S}_t, \cdot)) \mid \mathbf{S}_0 = \mathbf{s}], \\ \mathcal{H}(\pi(\mathbf{s}, \cdot)) &:= -\int_{\mathcal{A}} \log(\pi(\mathbf{s}, \mathbf{a}))\pi(\mathbf{s}, \mathbf{a}) \, d\mathbf{a}. \end{aligned} \tag{1}$$

Energy-based Modeling

Lemma (Summary of Haarnoja et al. [2017])

When solving (1) *optimally with energy-based policies, the likelihood of* $\mathbb{X} = \{\mathbf{s}_0, \mathbf{a}_0, \mathbf{s}_1, \mathbf{a}_1, \cdots, \mathbf{s}_T, \mathbf{a}_T\}$ *is* $L(\mathbb{X}; r) = \prod_{t=0}^{T} \pi^*(\mathbf{s}_t, \mathbf{a}_t) \prod_{t=0}^{T-1} \mathsf{P}(\mathbf{s}_{t+1} \mid \mathbf{s}_t \mathbf{a}_t)$*, where the optimal policy function taken by agents follows*

$$\pi^*(\mathbf{s}, \mathbf{a}) = \frac{\exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a})\right)}{\int_{\mathbf{a}' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}')\right) d\mathbf{a}'}, \tag{2}$$

with

$$Q(\mathbf{s}, \mathbf{a}) := r(\mathbf{s}, \mathbf{a}) + \max_{\pi} \mathbb{E}\left\{ \sum_{t=1}^{\infty} \gamma^t \left[r(\mathbf{S}_t, \mathbf{A}_t) + \alpha \mathcal{H}(\pi(\mathbf{S}_t, \cdot)) \right] \mid \mathbf{s}, \mathbf{a} \right\}.$$

Agents are likely but not guaranteed to make better decisions.

PQR Method with 3 Steps

Target: estimate r from the dataset.

→ Given the dataset $\mathbb{X} = \{\mathbf{s}_0, \mathbf{a}_0, \mathbf{s}_1, \mathbf{a}_1, \cdots, \mathbf{s}_T, \mathbf{a}_T\}$ generated by solving (1) optimally, we estimate the reward function.

→ PQR Method: estimate the **P**olicy, the **Q** function, and the **R**eward function sequentially.

PQR Method with 3 Steps

Step 1: estimate policy function

→ $\hat{\pi}$ denotes the estimator.

→ Many existing methods can estimate the $\hat{\pi}$ [Geng et al., 2017, Kuang et al., 2017, Fu et al., 2017, Geng et al., 2018] .

→ We directly use AIRL in Fu et al. [2017]

PQR Method with 3 Steps

Identification Issue

→ For any ϕ, $Q'(\mathbf{s}, \mathbf{a}) := Q(\mathbf{s}, \mathbf{a}) + \phi(\mathbf{s})$ leads to the same behaviour of agents as $Q(\mathbf{s}, \mathbf{a})$:

$$\pi^*(\mathbf{s}, \mathbf{a}) = \frac{\exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a})\right)}{\int_{\mathbf{a}' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}')\right) d\mathbf{a}'} = \frac{\exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}) + \frac{1}{\alpha}\phi(\mathbf{s})\right)}{\int_{\mathbf{a}' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}') + \frac{1}{\alpha}\phi(\mathbf{s})\right) d\mathbf{a}'}.$$

PQR Method with 3 Steps

Identification Issue

→ For any ϕ, $Q'(\mathbf{s}, \mathbf{a}) := Q(\mathbf{s}, \mathbf{a}) + \phi(\mathbf{s})$ leads to the same behaviour of agents as $Q(\mathbf{s}, \mathbf{a})$:

$$\pi^*(\mathbf{s}, \mathbf{a}) = \frac{\exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a})\right)}{\int_{\mathbf{a}' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}')\right) d\mathbf{a}'} = \frac{\exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}) + \frac{1}{\alpha}\phi(\mathbf{s})\right)}{\int_{\mathbf{a}' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}') + \frac{1}{\alpha}\phi(\mathbf{s})\right) d\mathbf{a}'}.$$

→ **Anchor Action:** There exists a known anchor action $\mathbf{a}^A \in \mathcal{A}$ and a function $g : \mathcal{S} \longmapsto \mathbb{R}$, such that $r(\mathbf{s}, \mathbf{a}^A) = g(\mathbf{s})$.

PQR Method with 3 Steps

Identification Issue

→ For any ϕ, $Q'(\mathbf{s}, \mathbf{a}) := Q(\mathbf{s}, \mathbf{a}) + \phi(\mathbf{s})$ leads to the same behaviour of agents as $Q(\mathbf{s}, \mathbf{a})$:

$$\pi^*(\mathbf{s}, \mathbf{a}) = \frac{\exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a})\right)}{\int_{\mathbf{a}' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}')\right) d\mathbf{a}'} = \frac{\exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}) + \frac{1}{\alpha}\phi(\mathbf{s})\right)}{\int_{\mathbf{a}' \in \mathcal{A}} \exp\left(\frac{1}{\alpha} Q(\mathbf{s}, \mathbf{a}') + \frac{1}{\alpha}\phi(\mathbf{s})\right) d\mathbf{a}'}.$$

→ **Anchor Action:** There exists a known anchor action $\mathbf{a}^A \in \mathcal{A}$ and a function $g : \mathcal{S} \longmapsto \mathbb{R}$, such that $r(\mathbf{s}, \mathbf{a}^A) = g(\mathbf{s})$.

→ $g(\mathbf{s}) = 0$ indicates that there exists an anchor action providing no rewards. Any non-action (like not selling a good, not entering a market) will lead to zero rewards.

PQR Method with 3 Steps

Step 2: estimate Q-function

→ Estimate $Q(\mathbf{s}, \mathbf{a}^A)$

$$\hat{Q}(\mathbf{s}, \mathbf{a}^A) = \hat{\mathcal{T}}(\hat{Q}(\cdot, \mathbf{a}^A))(\mathbf{s}), \qquad (Q^A\text{-ESTIMATOR})$$

$$\hat{\mathcal{T}} f(\mathbf{s}) := g(\mathbf{s}) + \gamma \hat{\mathbb{E}}_{\mathbf{s}'} \left[-\alpha \log(\hat{\pi}(\mathbf{s}', \mathbf{a}^A)) + f(\mathbf{s}') \mid \mathbf{s}, \mathbf{a}^A \right]$$

PQR Method with 3 Steps

Step 2: estimate Q-function

→ Estimate $Q(\mathbf{s}, \mathbf{a}^A)$

$$\hat{Q}(\mathbf{s}, \mathbf{a}^A) = \hat{\mathcal{T}}(\hat{Q}(\cdot, \mathbf{a}^A))(\mathbf{s}), \qquad (Q^A\text{-Estimator})$$

$$\hat{\mathcal{T}} f(\mathbf{s}) := g(\mathbf{s}) + \gamma \hat{\mathbb{E}}_{\mathbf{s}'} \left[-\alpha \log(\hat{\pi}(\mathbf{s}', \mathbf{a}^A)) + f(\mathbf{s}') \mid \mathbf{s}, \mathbf{a}^A \right]$$

→ For other Q's:

$$\hat{Q}(\mathbf{s}, \mathbf{a}) := \alpha \log(\hat{\pi}(\mathbf{s}, \mathbf{a})) - \alpha \log(\hat{\pi}(\mathbf{s}, \mathbf{a}^A)) + \hat{Q}(\mathbf{s}, \mathbf{a}^A). \qquad (Q\text{-Estimator})$$

PQR Method with 3 Steps

Step 2: estimate Q-function

→ Estimate $Q(\mathbf{s}, \mathbf{a}^A)$

$$\hat{Q}(\mathbf{s}, \mathbf{a}^A) = \hat{\mathcal{T}}(\hat{Q}(\cdot, \mathbf{a}^A))(\mathbf{s}), \qquad (Q^A\text{-ESTIMATOR})$$

$$\hat{\mathcal{T}}f(\mathbf{s}) := g(\mathbf{s}) + \gamma \hat{\mathbb{E}}_{\mathbf{s}'}\left[-\alpha \log(\hat{\pi}(\mathbf{s}', \mathbf{a}^A)) + f(\mathbf{s}') \mid \mathbf{s}, \mathbf{a}^A\right]$$

→ For other Q's:

$$\hat{Q}(\mathbf{s}, \mathbf{a}) := \alpha \log(\hat{\pi}(\mathbf{s}, \mathbf{a})) - \alpha \log(\hat{\pi}(\mathbf{s}, \mathbf{a}^A)) + \hat{Q}(\mathbf{s}, \mathbf{a}^A). \qquad (Q\text{-ESTIMATOR})$$

→ We use fitted-Q-iteration method for Q in practice.

PQR Method with 3 Steps

Step 2: estimate Q-function

Algorithm 1 FQI-I

Input: Dataset: $\mathbb{X}$.
Input: γ, α, N, and $\hat{\pi}(\mathbf{s}, \mathbf{a})$.
Output: $\hat{Q}(\mathbf{s}, \mathbf{a})$.

Initialize:
    Initialize a deep function $h : \mathcal{S} \longmapsto \mathbb{R}$.
$y_t = 0$ for $t \in \{t \mid \mathbf{a}_t = \mathbf{a}^A\}$
for $k \in [N]$ **do**
    for $t \in \{t \mid \mathbf{a}_t = \mathbf{a}^A\}$ **do**
        $y_t \leftarrow g(\mathbf{s}_t) - \gamma\alpha \log(\hat{\pi}(\mathbf{s}_{t+1}, \mathbf{a}^A)) + \gamma h(\mathbf{s}_{t+1})$
    end for
    Update h using $\{y_t\}_{\{t \mid \mathbf{a}_t = \mathbf{a}^A\}}$ and $\{s_t\}_{\{t \mid \mathbf{a}_t = \mathbf{a}^A\}}$.
end for
$\hat{Q}(\mathbf{s}, \mathbf{a}) \leftarrow \alpha \log(\hat{\pi}(\mathbf{s}, \mathbf{a})) - \alpha \log(\hat{\pi}(\mathbf{s}, \mathbf{a}^A)) + h(\mathbf{s})$
return $\hat{Q}(\mathbf{s}, \mathbf{a})$

PQR Method with 3 Steps

Step 3: estimate reward

→ Estimate using $\hat{Q}$ and $\hat{\pi}$.

$$\hat{r}(\mathbf{s}, \mathbf{a}) := \hat{Q}(\mathbf{s}, \mathbf{a}) - \gamma \hat{\mathbb{E}}_{\mathbf{S}'} \Big[-\alpha \log(\hat{\pi}(\mathbf{s}', \mathbf{a}^A)) + \hat{Q}(\mathbf{s}', \mathbf{a}^A) \mid \mathbf{s}, \mathbf{a} \Big], \tag{R-Estimator}$$

→ The procedure is equivalent to solving a supervised learning problem.

PQR Method with 3 Steps

Step 3: estimate reward

Algorithm 2 Reward Estimation (RE)

Input: Dataset: $\mathbb{X} = \{\mathbf{s}_0, \mathbf{a}_0, \cdots, \mathbf{s}_T, \mathbf{a}_T\}$.
Input: $\hat{Q}(\mathbf{s}, \mathbf{a})$ and $\hat{\pi}(\mathbf{s}, \mathbf{a})$.
Output: $\hat{r}(\mathbf{s}, \mathbf{a})$.

1: **Initialize:**
   Initialize a deep function $h : \mathcal{S} \times \mathcal{A} \longmapsto \mathbb{R}$.
2: **for** $t \in [T]$ **do**
3: $y_t \leftarrow -\alpha \log(\hat{\pi}(\mathbf{s}_{t+1}, \mathbf{a}^A)) + \hat{Q}(\mathbf{s}_{t+1}, \mathbf{a}^A)$
4: **end for**
5: Train h using $\{y_t\}_{t=0}^{T-1}$ with $\{\mathbf{s}_t\}_{t=0}^{T-1}$ and $\{\mathbf{a}_t\}_{t=0}^{T-1}$.
6: **return** $\hat{r}(\mathbf{s}, \mathbf{a}) = \hat{Q}(\mathbf{s}, \mathbf{a}) - \gamma h(\mathbf{s}, \mathbf{a})$.

PQR Method with 3 Steps

Main Algorithm

Algorithm 3 Main Algorithm

Input: $\mathbb{X} = \{\mathbf{s}_0, \mathbf{a}_0, \cdots, \mathbf{s}_T, \mathbf{a}_T\}$.
Input: γ, α and N.
Output: $\hat{r}(\mathbf{s}, \mathbf{a})$.

1: $\hat{\pi}(\mathbf{s}, \mathbf{a}) \leftarrow$AIRL($\mathbb{X}$)
2: $\hat{Q}(\mathbf{s}, \mathbf{a}) \leftarrow$FQI-I($\mathbb{X}$, N, $\hat{\pi}(\mathbf{s}, \mathbf{a})$)
3: $\hat{r}(\mathbf{s}, \mathbf{a}) \leftarrow$RE($\mathbb{X}$, $\hat{\pi}(\mathbf{s}, \mathbf{a})$, $\hat{Q}(\mathbf{s}, \mathbf{a})$)
4: **return** $\hat{r}(\mathbf{s}, \mathbf{a})$.

PQR Method with 3 Steps

Intuitions

→ **Main Result**:

$$Q(\mathbf{s}, \mathbf{a}) = r(\mathbf{s}, \mathbf{a}) + \gamma \mathbb{E}\left[-\alpha \log(\pi^*(\mathbf{s}', \mathbf{a}^A)) + Q(\mathbf{s}', \mathbf{a}^A) \mid \mathbf{s}, \mathbf{a}\right].$$

PQR Method with 3 Steps

Intuitions

→ **Main Result**:

$$Q(\mathbf{s}, \mathbf{a}) = r(\mathbf{s}, \mathbf{a}) + \gamma \mathbb{E}\left[-\alpha \log(\pi^*(\mathbf{s}', \mathbf{a}^A)) + Q(\mathbf{s}', \mathbf{a}^A) \mid \mathbf{s}, \mathbf{a}\right].$$

→ Reward estimator:
$\hat{r}(\mathbf{s}, \mathbf{a}) := \hat{Q}(\mathbf{s}, \mathbf{a}) - \gamma \hat{\mathbb{E}}_{\mathbf{s}'}\left[-\alpha \log(\hat{\pi}(\mathbf{s}', \mathbf{a}^A)) + \hat{Q}(\mathbf{s}', \mathbf{a}^A) \mid \mathbf{s}, \mathbf{a}\right]$.

PQR Method with 3 Steps

Intuitions

→ **Main Result**:

$$Q(\mathbf{s}, \mathbf{a}) = r(\mathbf{s}, \mathbf{a}) + \gamma \mathbb{E}\left[-\alpha \log(\pi^*(\mathbf{s}', \mathbf{a}^A)) + Q(\mathbf{s}', \mathbf{a}^A) \mid \mathbf{s}, \mathbf{a}\right].$$

→ Reward estimator:
$\hat{r}(\mathbf{s}, \mathbf{a}) := \hat{Q}(\mathbf{s}, \mathbf{a}) - \gamma \hat{\mathbb{E}}_{\mathbf{s}'}\left[-\alpha \log(\hat{\pi}(\mathbf{s}', \mathbf{a}^A)) + \hat{Q}(\mathbf{s}', \mathbf{a}^A) \mid \mathbf{s}, \mathbf{a}\right]$.

→ $Q(\mathbf{s}, \mathbf{a}^A)$ estimator: (take $\mathbf{a} = \mathbf{a}^A$)
$\hat{Q}(\mathbf{s}, \mathbf{a}^A) := g(\mathbf{s}) + \gamma \hat{\mathbb{E}}_{\mathbf{s}'}\left[-\alpha \log(\hat{\pi}(\mathbf{s}', \mathbf{a}^A)) + \hat{Q}(\mathbf{s}', \mathbf{a}^A) \mid \mathbf{s}, \mathbf{a}^A\right]$.

PQR Method with 3 Steps

Intuitions

→ **Main Result**:

$$Q(\mathbf{s},\mathbf{a}) = r(\mathbf{s},\mathbf{a}) + \gamma \mathbb{E}\left[-\alpha \log(\pi^*(\mathbf{s}',\mathbf{a}^A)) + Q(\mathbf{s}',\mathbf{a}^A) \mid \mathbf{s},\mathbf{a}\right].$$

→ Reward estimator:
$\hat{r}(\mathbf{s},\mathbf{a}) := \hat{Q}(\mathbf{s},\mathbf{a}) - \gamma \hat{\mathbb{E}}_{\mathbf{s}'}\left[-\alpha \log(\hat{\pi}(\mathbf{s}',\mathbf{a}^A)) + \hat{Q}(\mathbf{s}',\mathbf{a}^A) \mid \mathbf{s},\mathbf{a}\right].$

→ $Q(\mathbf{s},\mathbf{a}^A)$ estimator: (take $\mathbf{a} = \mathbf{a}^A$)
$\hat{Q}(\mathbf{s},\mathbf{a}^A) := g(\mathbf{s}) + \gamma \hat{\mathbb{E}}_{\mathbf{s}'}\left[-\alpha \log(\hat{\pi}(\mathbf{s}',\mathbf{a}^A)) + \hat{Q}(\mathbf{s}',\mathbf{a}^A) \mid \mathbf{s},\mathbf{a}^A\right].$

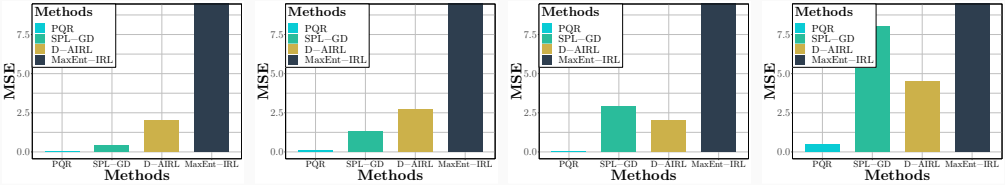
→ Policy is estimated by existing methods.

Theoretical Guarantees

→ When $\hat{\mathbb{E}} = \mathbb{E}$, the estimator uniquely recovers the true reward function.

→ When $\hat{\mathbb{E}} \neq \mathbb{E}$ and is estimated, the estimation error is nonasymptotically upper bounded.

Reward Estimation



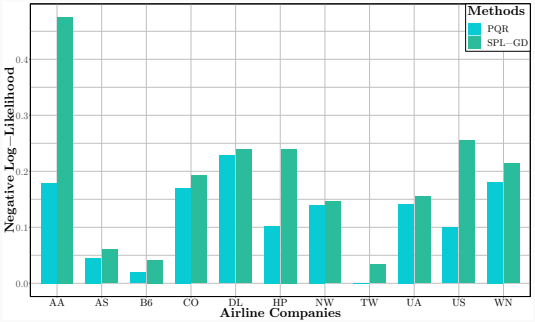
(a) $p = 5$ (b) $p = 10$ (c) $p = 20$ (d) $p = 40$

Figure: MSE (truncated at 9) for reward recovery with different state variable dimensions p

Sensitivity analysis, robustness analysis, hyperparameter selection are discussed.

Airline Company Market Entry Analysis

- → 11 Airline companies
- → 60 cities and $60 \times 59 \div 2 = 1770$ markets
- → Economic analysis is provided.



Thank you!

References I

P. Bajari, C. L. Benkard, and J. Levin. Estimating dynamic models of imperfect competition. *Econometrica*, 75(5): 1331–1370, 2007.

J. Fu, K. Luo, and S. Levine. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*, 2017.

S. Geng, Z. Kuang, and D. Page. An efficient pseudo-likelihood method for sparse binary pairwise markov network estimation. *arXiv preprint arXiv:1702.08320*, 2017.

References II

S. Geng, Z. Kuang, J. Liu, S. Wright, and D. Page. Stochastic learning for sparse discrete markov random fields with controlled gradient approximation error. In *Uncertainty in artificial intelligence: proceedings of the... conference. Conference on Uncertainty in Artificial Intelligence*, volume 2018, page 156. NIH Public Access, 2018.

T. Haarnoja, H. Tang, P. Abbeel, and S. Levine. Reinforcement learning with deep energy-based policies. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1352–1361. JMLR. org, 2017.

References III

Z. Kuang, S. Geng, and D. Page. A screening rule for l1-regularized ising model estimation. In *Advances in neural information processing systems*, pages 720–731, 2017.

R. C. Merton et al. An intertemporal capital asset pricing model. *Econometrica*, 41(5):867–887, 1973.