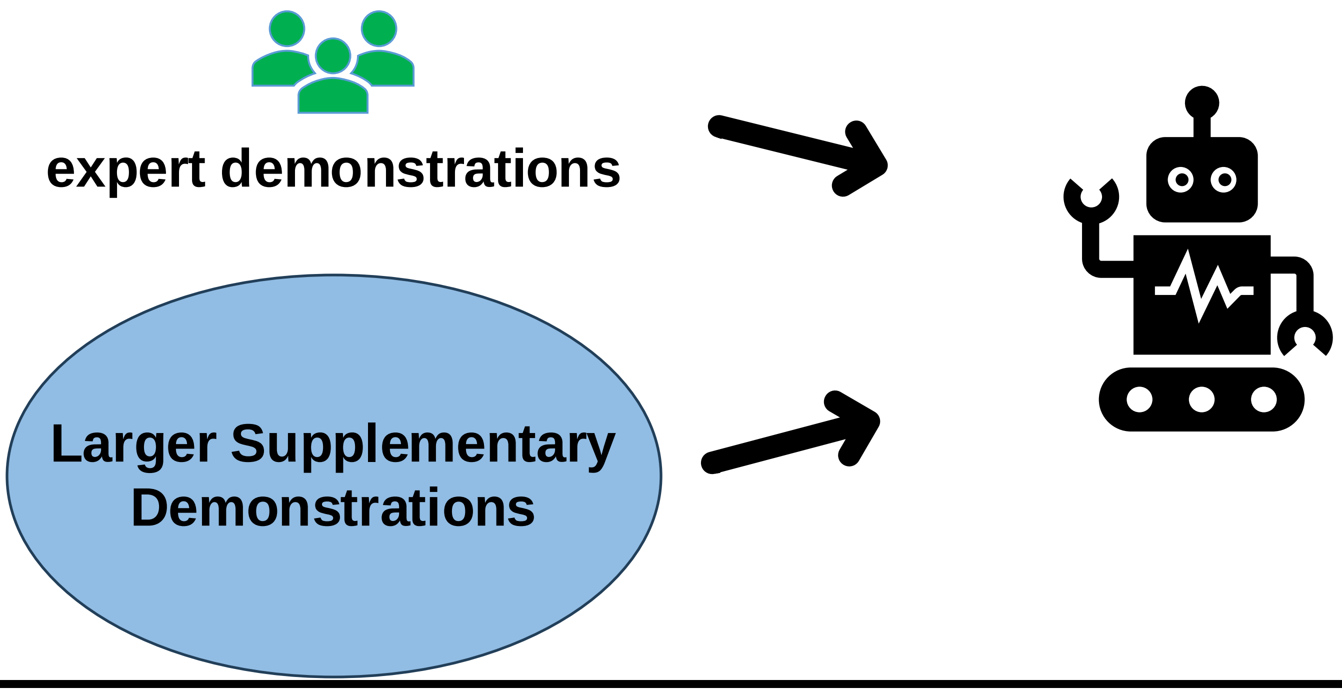
School of
Computing and
Information Systems

SPRINQL: Sub-optimal Demonstrations driven Offline Imitation Learning

Huy Hoang, Tien Mai, Pradeep Varakantham
Singapore Management Univerisity

Offline Imitation Learning
With Supplementary Demonstrations

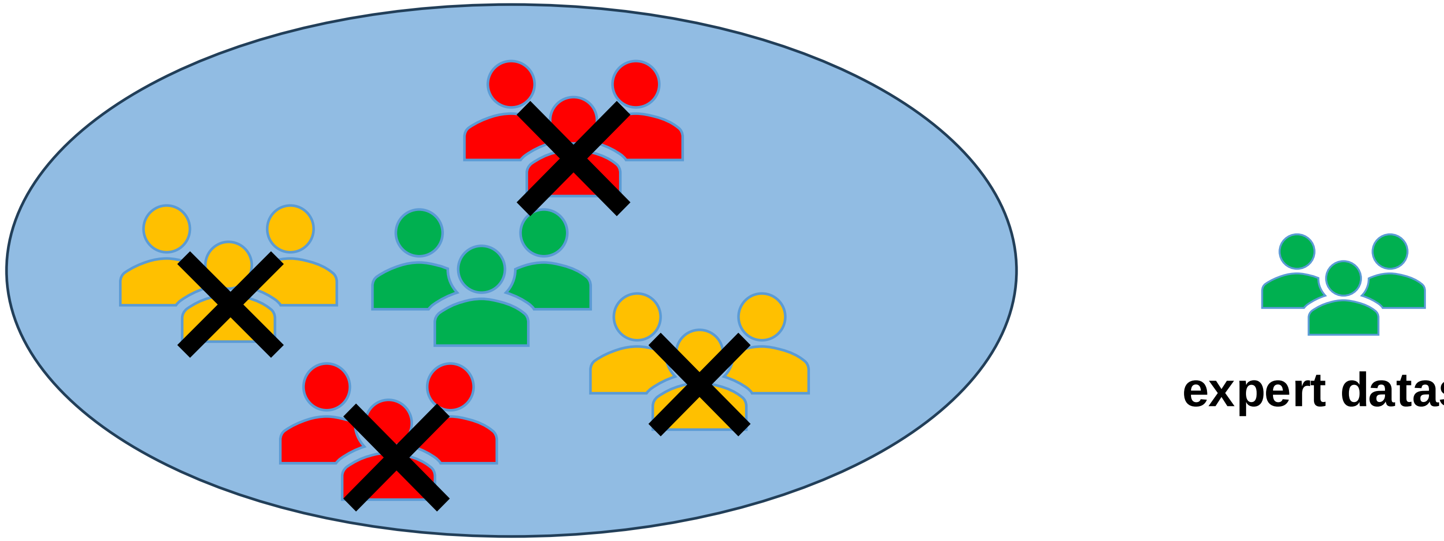
Offline Imitation Learning with Supplementary demonstrations



- ✓Leverage expert demonstrations and additional data.
- ✓Working completely Offline.
- ✓Reduce the number of expert demonstrations.
- ✓Enhance generalization.

Motivation - Existing methods

Existing methods



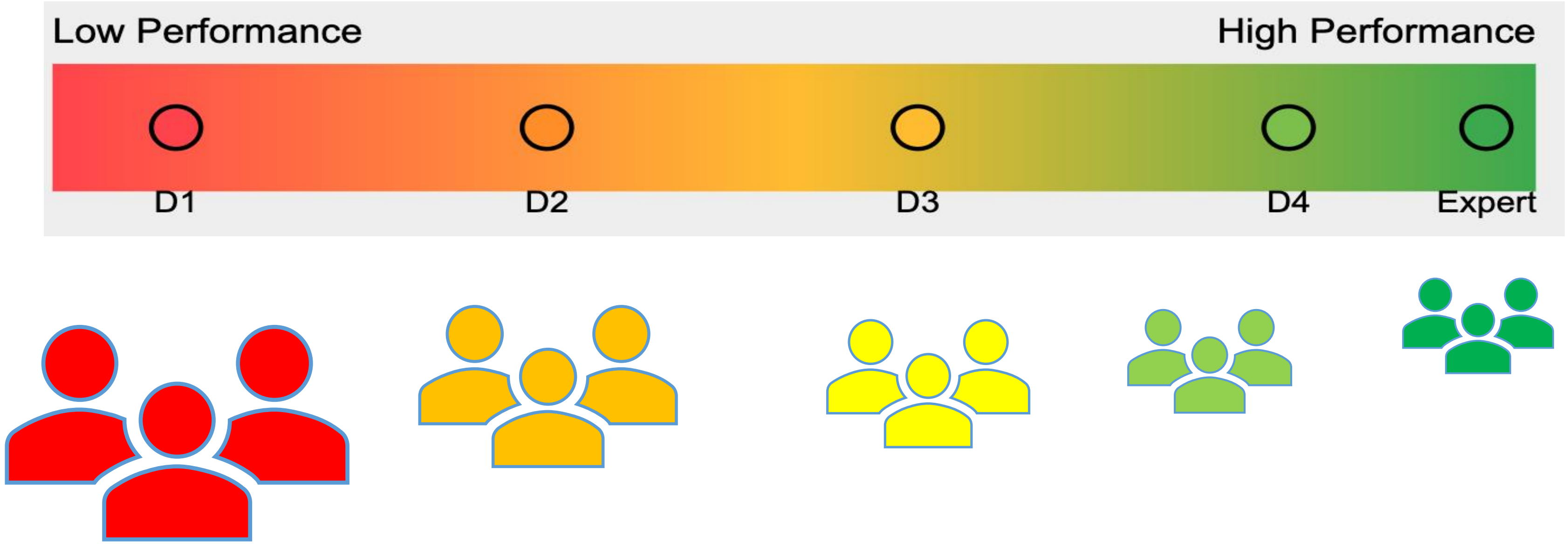
Unlabeled mixed dataset

expert dataset

Remove all non-expert demonstrations

SPRINQL Idea

SPRINQL



✓ **Learning from all demonstrations**

✓ **Reduce the number of dataset**

Sub-optimal Demonstration driven Offline Imitation Learning

Imitation Learning with multiple expertise levels

- Given several sets of different expertise levels $\mathcal{D}^1 > \mathcal{D}^2 > \cdots > \mathcal{D}^N$, we have:

$$\mathbb{E}_{\rho^1}[r^*(s,a)] > \mathbb{E}_{\rho^2}[r^*(s,a)] > \cdots > \mathbb{E}_{\rho^N}[r^*(s,a)],$$

where ρ^k is the occupancy measures of expertise level k policy, and $r^*(.,.)$ is the ground-truth rewards.

- The the expert level dataset significant smaller than others

$$\|\mathcal{D}^1\| \ll \|\mathcal{D}^i\|$$

SPRINQL – MaxEnt IRL for distribution matching

We formulate the Max Entropy Inversed RL [1] for multiple levels:

$$\max_{r} \min_{\pi} \sum_{i \in [N]} w_i \mathbb{E}_{\rho^i}[r(s,a)] - \mathbb{E}_{\rho_\pi}[r(s,a)] + \mathbb{E}_{\rho_\pi}[\log \pi(s,a)]$$

where $w_i \geq 0$ is the weight of expertise level i:

$$w_1 > w_2 > \ldots > w_N$$

$$\sum_{i \in [N]} w_i = 1$$

SPRINQL – MaxEnt IRL for distribution matching

To simplify, the objective can be rewritten as:

$$\mathbb{E}_{\rho^U}[r(s,a)] - \mathbb{E}_{\rho_\pi}[r(s,a)] - \mathbb{E}_{\rho_\pi}[\log \pi(s,a)],$$

Where $\rho^U = \sum_{i \in [N]} w_i \rho^i$

However, the dataset of expert-level is sufficient small, leading to inaccurate $\mathbb{E}_{\rho^1}[r(s,a)]$ estimation.

SPRINQL – reward regularization with reference reward

We define a reference reward function $\bar{r}$ that:

$$\bar{r}(s,a) > \bar{r}(s',a'), \forall(s,a) \in \mathcal{D}^1 \text{ and } (s',a') \notin \mathcal{D}^1 \text{ and}$$

$$\bar{r}(s,a) > \bar{r}(s',a'), \forall(s,a) \in \mathcal{D}^2 \text{ and } \forall(s',a') \notin \mathcal{D}^2 \cup \mathcal{D}^1 \text{ and so on}$$

Combine with the MaxEnt IRL objective:

$$\max_r \min_\pi \left\{ \underbrace{\mathbb{E}_{\rho^U}[r(s,a)] - \mathbb{E}_{\rho_\pi}[r(s,a)] + \mathbb{E}_{\rho_\pi}[\log \pi(s,a)]}_{\text{Occupancy matching}} - \underbrace{\alpha \mathbb{E}_{\rho^U}[(r(s,a) - \bar{r}(s,a))^2]}_{\text{Reward regularizer}} \right\}$$

SPRINQL – Inverse Soft-Q with reward regularization

We transform the objective into Q-space (IQ-learn [2]):

$$\max_{Q} \min_{\pi} \Big\{ \mathcal{H}(Q,\pi) \stackrel{def}{=} \mathbb{E}_{\rho^U}[\mathcal{T}^\pi[Q](s,a))] - \mathbb{E}_{\rho_\pi}[\mathcal{T}^\pi[Q](s,a))] + \mathbb{E}_{\rho_\pi}[\log \pi(s,a)] - \alpha \mathbb{E}_{\rho^U}[(\mathcal{T}^\pi[Q](s,a)) - \overline{r}(s,a))^2] \Big\}$$

Where $r(s,a)$ is replaced by $\mathcal{T}^\pi[Q](s,a)$

$$\mathcal{T}^\pi[Q](s,a) = Q(s,a) - \gamma \mathbb{E}_{s'}[V^\pi(s')],\ V^\pi(s) = \mathbb{E}_{a\sim\pi(a|s)}[Q(s,a) - \log \pi(a|s)]$$

However, this new objective do not have a unique saddle point as IQ-learn.

SPRINQL – final objective

We arrive at a final objective that retains desirable properties from the original IQ-Learn (proofs provided in the paper).

$$\widehat{\mathcal{H}}(Q,\pi) \stackrel{def}{=} \sum_{i\in[N]} w_i \mathbb{E}_{\rho^i}[\mathcal{T}^\pi[Q](s,a))] - (\mathbb{E}_{\rho_\pi}[\mathcal{T}^\pi[Q](s,a))] - \mathbb{E}_{\rho_\pi}[\log \pi(s,a)])$$

$$- \alpha \mathbb{E}_{\rho^U}\Big[(Q(s,a) - \overline{r}(s,a))^2 + (\mathbb{E}_{s'} V^\pi(s'))^2 + 2\mathrm{ReLU}(\overline{r}(s,a) - Q(s,a))\mathbb{E}_{s'} V^\pi(s')\Big]$$

SPRINQL – Estimate reference reward function

We automatically learn the reference rewards $\bar{r}$:

$$\min_{\bar{r}}\{\mathcal{L}(\bar{r}) = \sum_{i\in[N]} \sum_{(s,a),(s',a')\in\mathcal{D}^i} (\bar{r}(s,a) - \bar{r}(s',a'))^2 - \sum_{h,k\in[N],h>k,\tau_i\in\mathcal{D}^h,\tau_j\in\mathcal{D}^k} \ln P(\tau_i \prec \tau_j)\}$$

Where $P(\tau_i \prec \tau_j) = \frac{\exp(R(\tau_j))}{\exp(R(\tau_i))+\exp(R(\tau_j))}$ is Bradley-Terry model of preferences.

SPRINQL – Preference-based Weight Learning

In the occupancy matching term, we assign a weight parameter to each expertise level, which should reflect the quality of that level:

$$w_i = \frac{\mathbb{E}_{(s,a)\sim D^i}[\bar{r}(s,a)]}{\sum_{j\in[N]} \mathbb{E}_{(s,a)\sim D^j}[\bar{r}(s,a)]}$$

SPRINQL – Conservative soft-Q learning

CQL [3] is added into the objective to overcome the out-of-distribution actions problem:

$$\widehat{\mathcal{H}}^C(Q,\pi) = -\beta \sum_{s\sim\mathcal{D},\ a\sim\mu(a|s)} [Q(s,a)] + \widehat{\mathcal{H}}(Q,\pi)$$

Experiments

Experiments - Baselines

We compare our method against several baselines:

- **BC, IQ [2] (-E, -O, -both)**: Offline imitation learning variants using only expert data (-E), only sub-optimal data (-O), and both expert and sub-optimal data (-both).
- **W-BC**: Weighted Behavioral Cloning, which applies preference-based weights to the datasets.

We compare with state-of-the-art offline imitation learning methods that leverage supplementary demonstrations:

- **TRAIL** [4]
- **DemoDICE** [5]
- **DWBC** [6]

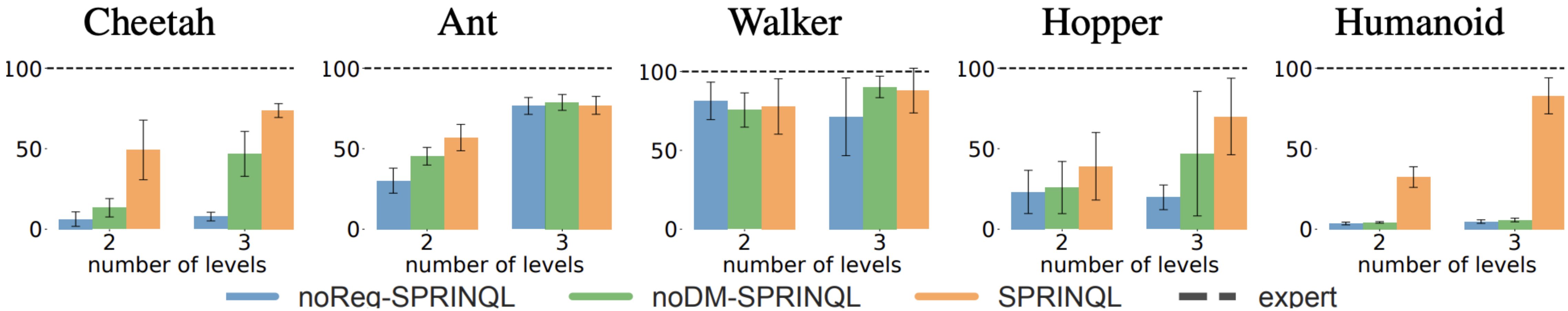
Experiments - Main Comparison Results

We compare SPRINQL with other algorithms in 3 level dataset scenario In Mujoco [7] and Panda-gym [8] environment:

	Mujoco			Panda-gym			
	Cheetah	Ant	Humanoid	Push	PnP	Slide	Avg
BC-E	-3.2±0.9	6.4±19.1	1.3±0.2	8.2±3.8	3.7±2.7	0.0±0.0	2.7
BC-O	14.2±2.9	35.2±20.1	10.6±6.3	8.8±4.5	3.9±2.7	0.1±0.3	12.1
BC-both	13.2±3.6	47.0±5.9	9.0±3.5	9.0±4.3	4.4±3.0	0.1±0.4	13.8
W-BC	12.9±2.8	47.3±6.4	19.6±19.0	8.8±4.3	3.7±2.8	0.0±0.0	15.4
TRAIL	-4.1±0.3	-4.7±1.9	2.6±0.6	11.7±4.0	7.8±3.7	1.7±1.8	3.9
IQ-E	-3.4±0.6	-3.4±1.3	2.4±0.6	26.3±10.9	18.1±12.5	0.1±0.4	6.7
IQ-both	-6.1±1.4	-58.2±0.0	0.8±0.0	8.3±3.9	3.8±3.3	0.0±0.2	-8.6
SQIL-E	-5.0±0.7	-33.8±7.4	0.9±0.1	9.6±3.3	3.2±2.9	0.1±0.3	-4.2
SQIL-both	-5.6±0.5	-58.0±0.4	0.8±0.0	8.2±3.8	3.3±2.3	0.1±0.3	-12.6
DemoDICE	0.4±2.0	31.7±8.9	2.6±0.8	8.1±3.7	4.3±2.4	0.1±0.5	7.9
DWBC	-0.2±2.5	10.4±5.0	3.7±0.3	36.9±7.4	25.0±6.3	11.6±4.4	14.6
SPRINQL (ours)	**73.6±4.3**	**77.0±5.6**	**82.9±11.2**	**72.0±5.3**	**63.2±6.4**	**37.7±6.6**	**67.7**

Importance of Distribution Matching and Reward Regularizer

In this experiment, we test the importance of two term of our objective:



Other experiment concerns

Moreover, in our paper, we conduct a comprehensive set of experiments to address the following questions:

- ***(Q3)*** *What happens if we augment (or reduce) the expert data while maintaining the sub-optimal datasets?*
- ***(Q4)*** *What happens if we augment (or reduce) the sub-optimal data while maintaining the expert dataset?*
- ***(Q5)*** *How does the conservative term help in our approach?*
- ***(Q6)*** *How does increasing N (the number of expertise levels) affect the performance of SPRINQL?*
- ***(Q7)*** *Does the preference-based weight learning approach provide good values for the weights?*
- ***(Q8)*** *How does SPRINQL perform in recovering the ground-truth reward function?*

Conclusion

- SPRINQL is offline imitation learning for ranked datasets.
- SPRINQL have several favorable properties, contributing to its well-behaved, stable, and scalable nature.
- SPRINQL can utilize all expertise datasets instead of remove sub-optimals.

Limitation:

- lack of theoretical investigation on how the sizes of the expert and non-expert datasets affect the performance of Q-learning.
- lacks a theoretical exploration of how the reward regularizer term enhances the distribution matching term when expert samples are low.

References

[1] Ziebart, Brian D., et al. "Maximum entropy inverse reinforcement learning." *Aaai*. Vol. 8. 2008.

[2] Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, and Stefano Ermon. Iq-learn: Inverse soft-q learning for imitation. Advances in Neural Information Processing Systems, 34:4028–4039, 2021.

[3] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. Advances in Neural Information Processing Systems, 33:1179– 1191, 2020.

[4] Mengjiao Yang, Sergey Levine, and Ofir Nachum. Trail: Near-optimal imitation learning with suboptimal data. In International Conference on Learning Representations, 2021.

[5] Geon-Hyeong Kim, Seokin Seo, Jongmin Lee, Wonseok Jeon, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. Demodice: Offline imitation learning with supplementary imperfect demonstrations. In International Conference on Learning Representations, 2021.

[6] Haoran Xu, Xianyuan Zhan, Honglei Yin, and Huiling Qin. Discriminator-weighted offline imitation learning from suboptimal demonstrations. In Proceedings of the 39th International Conference on Machine Learning, pages 24725–24742, 2022.

[7] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In 2012 IEEE/RSJ international conference on intelligent robots and systems, pages 5026–5033. IEEE, 2012.

[8] Quentin Gallouédec, Nicolas Cazin, Emmanuel Dellandréa, and Liming Chen. pandagym: Open-source goal-conditioned environments for robotic learning. arXiv preprint arXiv:2106.13687, 2021.