Explaining Reinforcement Learning with Shapley Values: Theory and Algorithms

Daniel Beechey

Researcher, H Company

Mila · RL Sofa

September 4, 2026

Collaborators

art-ai

UNIVERSITY OF BATH

Thomas M. S. Smith

PhD Researcher

Bath Reinforcement Learning Lab

Department of Computer Science, University of Bath

Özgür Şimşek

Professor of Artificial Intelligence

Bath Reinforcement Learning Lab

Department of Computer Science, University of Bath

What is Reinforcement Learning?

Learn a policy $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})$ that maps each observation to a probability distribution over actions, **maximising the expected return** [14]:

$$\mathbb{E}[G_t] = \mathbb{E}\left[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} \right]$$

What Can Reinforcement Learning Do?

Atari [7]

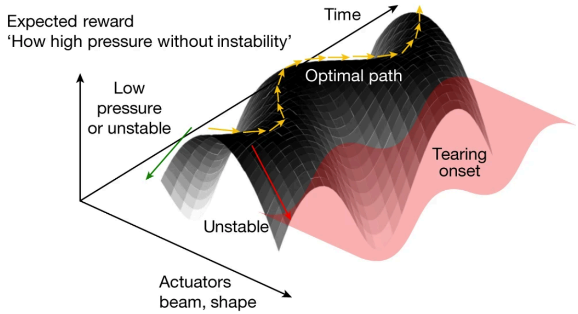
AlphaGo [9, 12, 21]

StarCraft II [17]

Gran Turismo [36]

Matrix Multiplication [32]

Stratospheric Balloons [18]



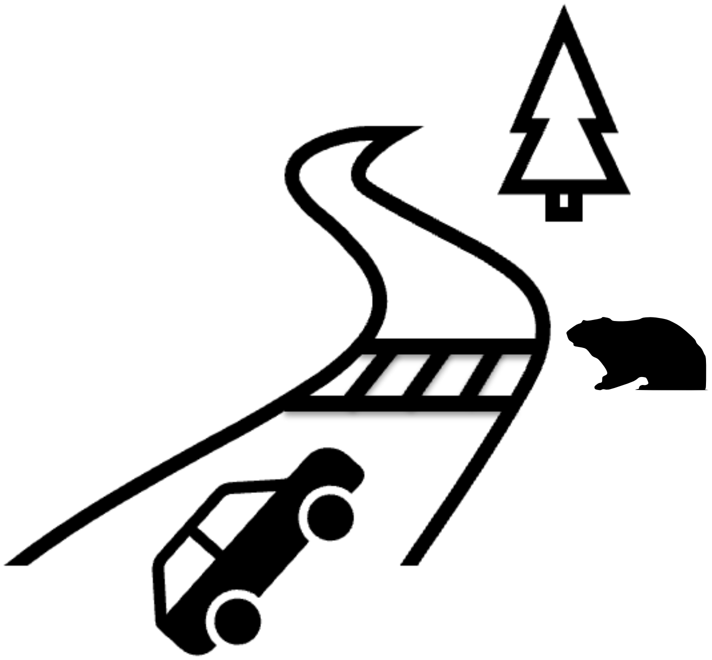
Nuclear Fusion Reactor Control [31, 39]

Reinforcement learning agents do not explain their actions.

Understanding Agent-Environment Interactions

Certain features of an agent's observations influence how they interact with their environment.

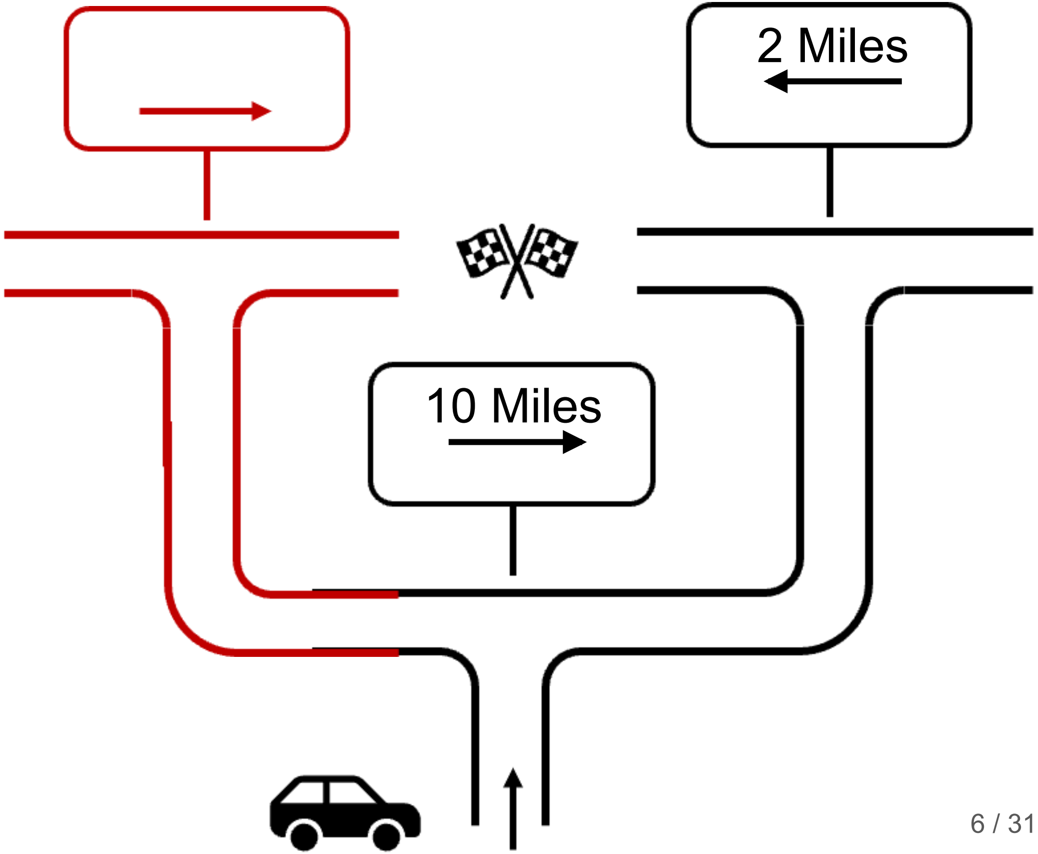
Contribution: A theoretical and computational framework for explaining agent-environment interactions using the influence of features.



Beechey, D., Smith, T.M. and Şimşek, Ö., 2023, July. Explaining reinforcement learning with Shapley values. In *International Conference on Machine Learning* (pp. 2003-2014). PMLR.

What Needs Explaining?

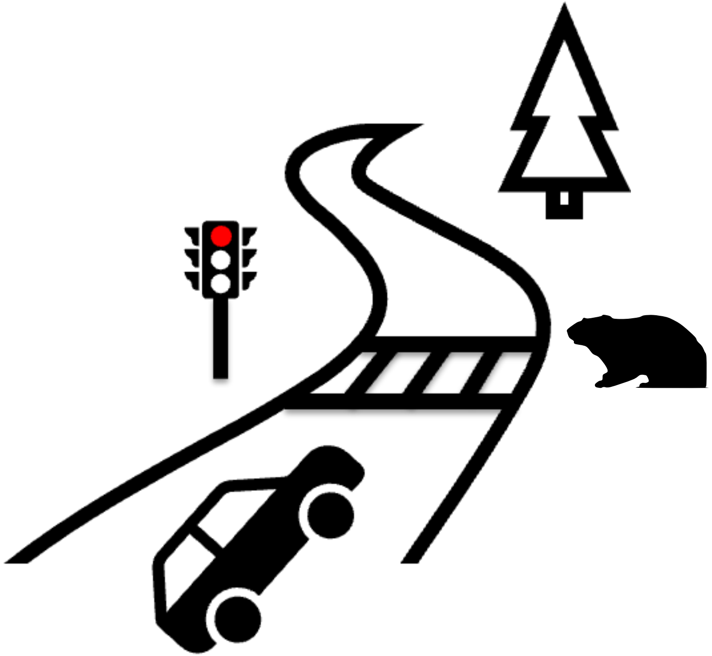
- Behaviour
- Outcomes
- Predictions



Computing Feature Influence

Compute the influence of features by observing the effect of their removal.

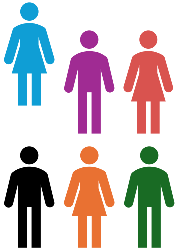
Features are interdependent; removing one feature does not properly capture its contribution.



A **cooperative game** is a set of players $\mathcal{F}$ and a characteristic function $v(\mathcal{C}) : 2^{|\mathcal{F}|} \to \mathbb{R}$.

How to assign the contribution $\phi_i(v)$ of player i to the outcome of the game $(\mathcal{F}, v)$? [1]

$$\phi_i(v) = \sum_{\mathcal{C} \subseteq \mathcal{F} \setminus \{i\}} \frac{|\mathcal{C}|!\,(|\mathcal{F}| - |\mathcal{C}| - 1)!}{|\mathcal{F}|!} \left(v(\mathcal{C} \cup \{i\}) - v(\mathcal{C})\right)$$



$$\phi_i(v) = \sum_{\mathcal{C} \subseteq \mathcal{F} \setminus \{i\}} \frac{|\mathcal{C}|!\,(|\mathcal{F}| - |\mathcal{C}| - 1)!}{|\mathcal{F}|!} \left(v(\mathcal{C} \cup \{i\}) - v(\mathcal{C})\right)$$

Shapley values $\phi_i(v)$ are the **unique solution** to the contribution assignment problem that satisfies four axioms of fair contribution. [1]

Efficiency: $v(\mathcal{F}) = v(\emptyset) + \sum_{i \in \mathcal{F}} \phi_i(v).$

Symmetry: $\phi_i(v) = \phi_j(v) \text{ if } v(\mathcal{C} \cup \{i\}) = v(\mathcal{C} \cup \{j\}) \;\; \forall \mathcal{C} \subseteq \mathcal{F} \setminus \{i, j\}.$

Nullity: $\phi_i(v) = 0 \text{ if } v(\mathcal{C} \cup \{i\}) = v(\mathcal{C}) \;\; \forall \mathcal{C} \subseteq \mathcal{F} \setminus \{i\}.$

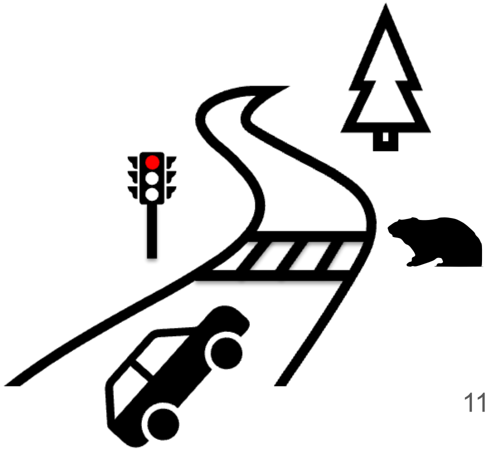
Linearity: $\phi_i(\alpha u + \beta v) = \alpha \phi_i(u) + \beta \phi_i(v).$

Shapley Values for Explaining Reinforcement Learning (SVERL)

Beechey, D., Smith, T. and Şimşek, Ö., 2025. A Theoretical Framework for Explaining Reinforcement Learning with Shapley Values. arXiv preprint arXiv:2505.07797.

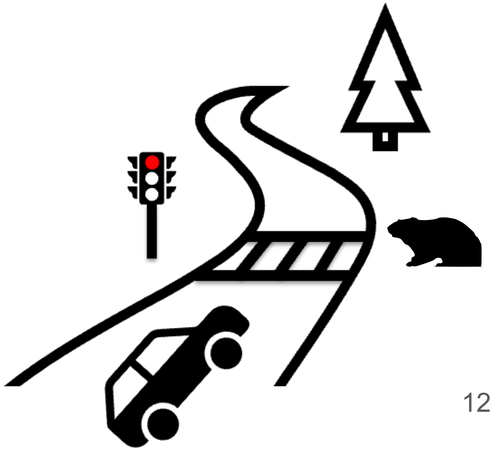
A collection of cooperative games played by the values of the features an agent observes, whose outcomes are different aspects of agent-environment interactions.

Explaining Behaviour. The contribution of feature values to the probability of selecting action a in state s.

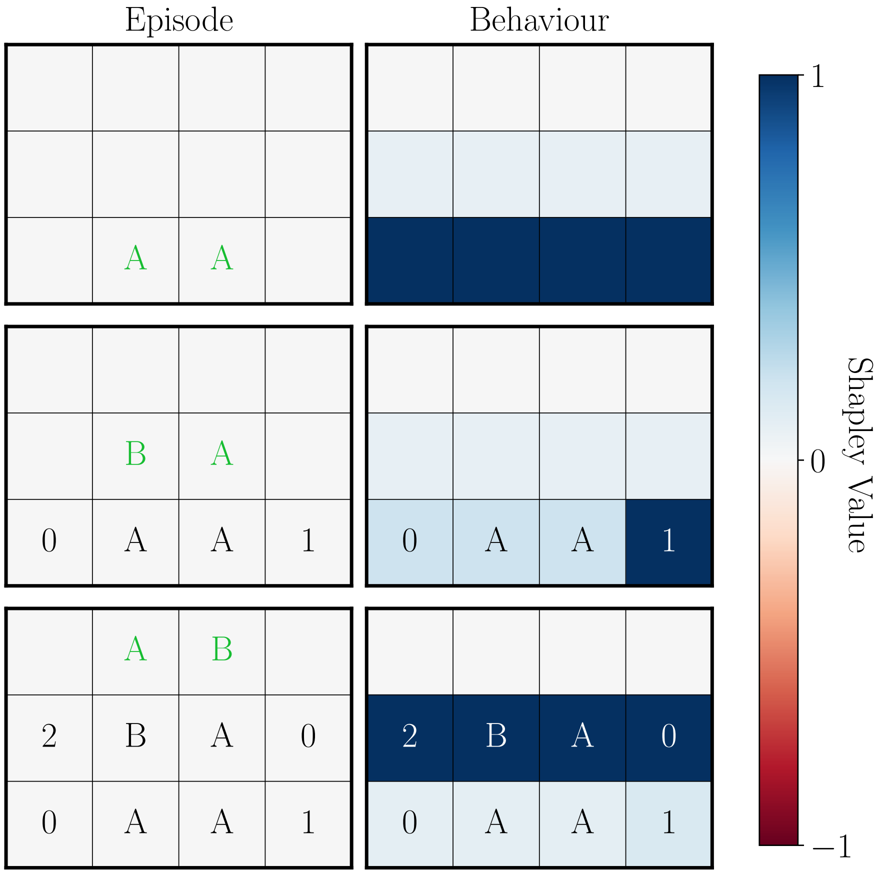


A cooperative game played by the values of the features at state s, whose outcome $\tilde{\pi}_s^a : 2^{|\mathcal{F}|} \rightarrow [0,1]$ is the probability of selecting action a at state s when only the values of features $\mathcal{C}$ are known.

$$\tilde{\pi}_s^a(\mathcal{C}) = \mathbb{E}\left[\pi(S,a) \mid S_{\mathcal{C}} = s_{\mathcal{C}}\right] = \sum_{s' \in \mathcal{S}^+} p^\pi(s' \mid s_{\mathcal{C}})\, \pi(s', a)$$



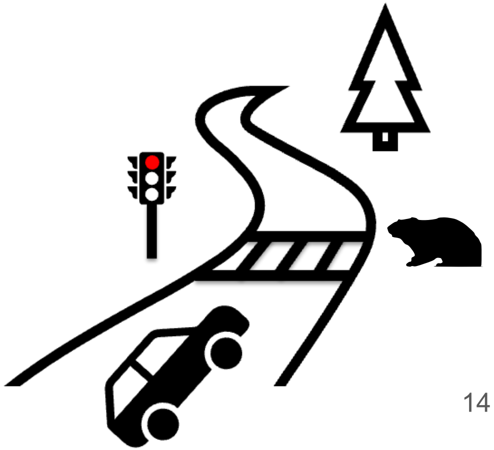
Explaining Behaviour in Mastermind



A collection of cooperative games played by the values of the features an agent observes, whose outcomes are different aspects of agent-environment interactions.

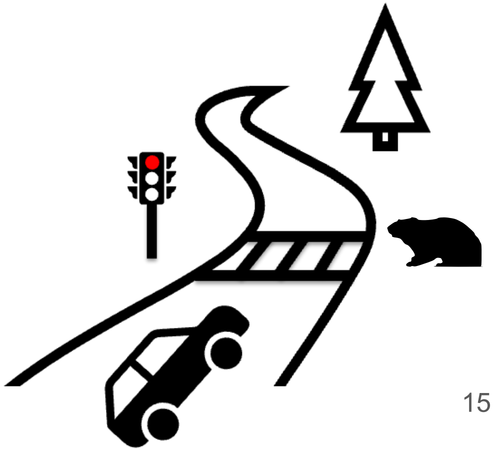
Explaining Behaviour. The contribution of feature values to the probability of selecting action a in state s.

Explaining Outcomes. The contribution of feature values to the expected return $v^{\pi}(s)$.

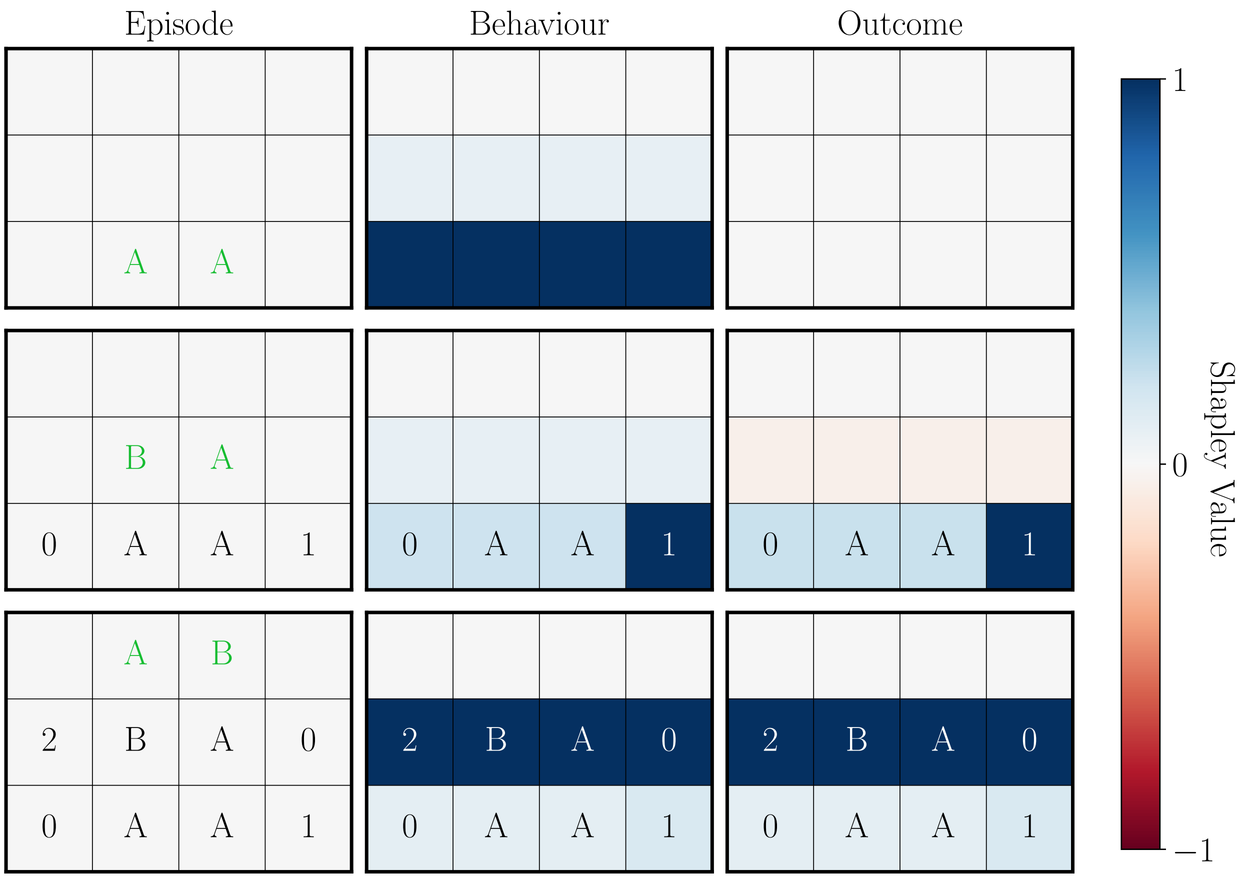


A cooperative game played by the values of the features at state s, whose outcome $\tilde{v}^{\pi}_{s} : 2^{|\mathcal{F}|} \to \mathbb{R}$ is the expected return from state s when policy π has access to only the values of features in $\mathcal{C}$.

$$\tilde{v}^{\pi}_{s}(\mathcal{C}) = \mathbb{E}_{\mu}\left[G_t \mid S_t = s\right], \quad \text{where} \quad \mu(s_t, a_t) = \begin{cases} \tilde{\pi}^{a_t}_{s_t}(\mathcal{C}) & \text{if } s_t = s, \\ \pi(s_t, a_t) & \text{otherwise.} \end{cases}$$



Explaining Outcomes in Mastermind

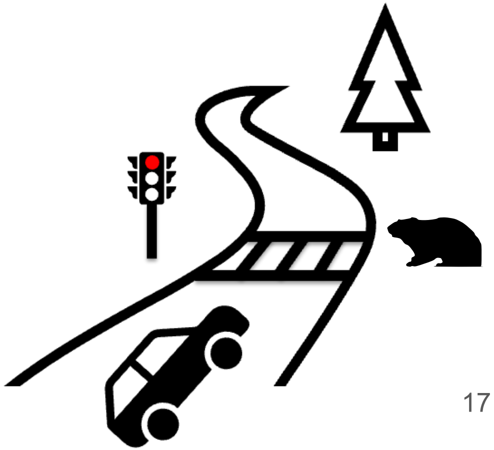


A collection of cooperative games played by the values of the features an agent observes, whose outcomes are different aspects of agent-environment interactions.

Explaining Behaviour. The contribution of feature values to the probability of selecting action a in state s.

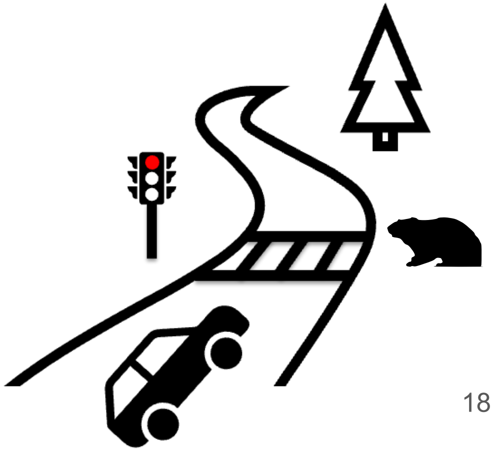
Explaining Outcomes. The contribution of feature values to the expected return $v^{\pi}(s)$.

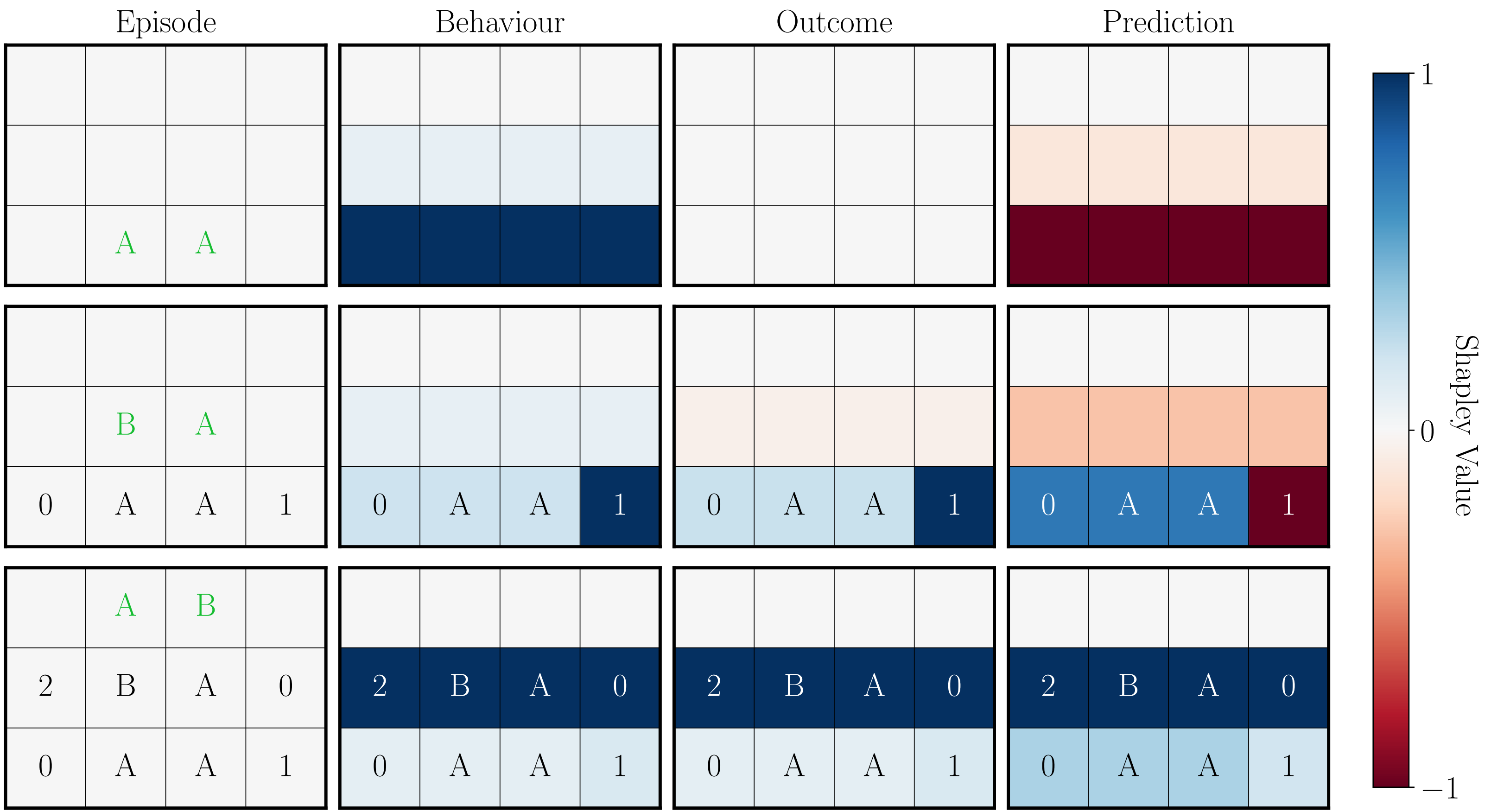
Explaining Predictions. The contribution of feature values to the predicted expected return $\hat{v}^{\pi}(s)$.



A cooperative game played by the values of the features at state s, whose outcome $\hat{v}^{\pi}_{s} : 2^{|\mathcal{F}|} \rightarrow \mathbb{R}$ is the predicted expected return using only the feature values $s_{\mathcal{C}}$.

$$\hat{v}^{\pi}_{s}(\mathcal{C}) = \mathbb{E}\left[\hat{v}^{\pi}(S) \mid S_{\mathcal{C}} = s_{\mathcal{C}}\right] = \sum_{s' \in \mathcal{S}^{+}} p^{\pi}(s' \mid s_{\mathcal{C}})\, \hat{v}^{\pi}(s')$$




Episode
Behaviour
Outcome
Prediction
A A
B A
0 A A 1
A B
2 B A 0
0 A A 1
1
0
−1
Shapley Value

Feature Importance Methods

Gradients [10] Perturbation [13, 20] Attention [15] LIME [8]

Shapley Values in Supervised Learning

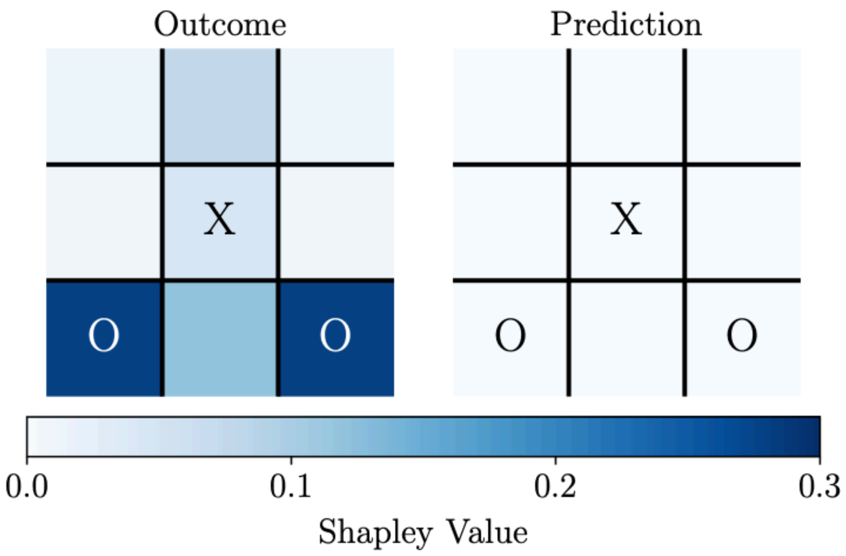
Feature attribution [4, 5, 6, 24, 25, 33, 38]

SHAP [11]

Shapley Values in Reinforcement Learning

SHAP applied to the value function [23, 29, 30]

SHAP applied to the policy [16, 19, 22, 26, 27, 28, 34, 35]



Approximating Shapley Explanations in Reinforcement Learning (FastSVERL)

Beechey, D. and Şimşek, Ö., 2025. Approximating Shapley explanations in reinforcement learning. In *Advances in Neural Information Processing Systems*.

Computing SVERL Exactly is Intractable

art-ai

UNIVERSITY OF BATH

Shapley values sum over every subset of features:

$$\phi_i(\tilde{\pi}_s^a) = \sum_{\mathcal{C} \subseteq \mathcal{F} \setminus \{i\}} \frac{|\mathcal{C}|!\,(|\mathcal{F}| - |\mathcal{C}| - 1)!}{|\mathcal{F}|!} \left(\tilde{\pi}_s^a(\mathcal{C} \cup \{i\}) - \tilde{\pi}_s^a(\mathcal{C})\right)$$

$2^{|\mathcal{F}|-1}$ **subsets**

Each characteristic value is an expectation over the state space:

$$\tilde{\pi}_s^a(\mathcal{C}) = \mathbb{E}\left[\pi(S, a) \mid S_\mathcal{C} = s_\mathcal{C}\right] = \sum_{s' \in \mathcal{S}^+} p^\pi(s' \mid s_\mathcal{C})\, \pi(s', a)$$

$|\mathcal{S}|$ **states**

$\mathcal{O}\left(2^{|\mathcal{F}|} \cdot |\mathcal{S}|\right)$ **per explanation**

The characteristic function is an expectation over the state space:

$$\tilde{\pi}_s^a(\mathcal{C}) = \mathbb{E}\left[\pi(S, a) \mid S_{\mathcal{C}} = s_{\mathcal{C}}\right] = \sum_{s' \in \mathcal{S}^+} p^\pi(s' \mid s_{\mathcal{C}})\, \pi(s', a)$$

Learn a model $\hat{\pi}(s, a \mid \mathcal{C}; \beta)$ that masks the features outside $\mathcal{C}$ [25]:

$$\mathcal{L}(\beta) = \mathop{\mathbb{E}}_{p^\pi(s)} \mathop{\mathbb{E}}_{\mathrm{Unif}(a)} \mathop{\mathbb{E}}_{p(\mathcal{C})} \left|\pi(s, a) - \hat{\pi}(s, a \mid \mathcal{C}; \beta)\right|^2$$

It **cannot** recover $\pi(s, a)$ from partial input, so it learns the average of $\pi(s', a)$ over the states sharing those values: the characteristic function.

Approximating the Shapley Values

$$\phi_i(\tilde{\pi}_s^a) = \sum_{\mathcal{C} \subseteq \mathcal{F} \setminus \{i\}} \frac{|\mathcal{C}|!\,(|\mathcal{F}| - |\mathcal{C}| - 1)!}{|\mathcal{F}|!} \left(\tilde{\pi}_s^a(\mathcal{C} \cup \{i\}) - \tilde{\pi}_s^a(\mathcal{C})\right)$$

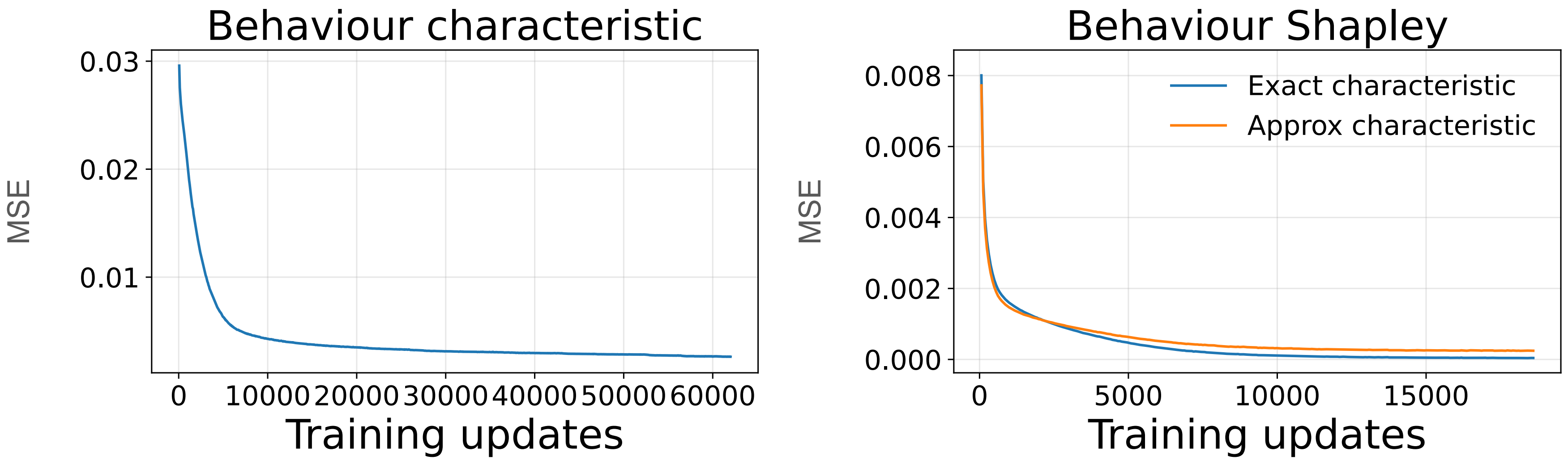
Shapley values also solve a weighted least-squares problem [2, 11].

Learn a model $\hat{\phi}(s, a; \theta)$ that predicts every feature's contribution at once [33]:

$$\mathcal{L}(\theta) = \mathop{\mathbb{E}}_{p^\pi(s)} \mathop{\mathbb{E}}_{\mathrm{Unif}(a)} \mathop{\mathbb{E}}_{p(\mathcal{C})} \left| \tilde{\pi}_s^a(\mathcal{C}) - \tilde{\pi}_s^a(\emptyset) - \sum_{i \in \mathcal{C}} \hat{\phi}^i(s, a; \theta) \right|^2$$

The cost is amortised: train once, then read off an explanation for any state. [33, 38]

Mastermind: 100,000+ states, 15 features, 27 actions.



Training the Shapley model on approximate characteristic values carries that error through.

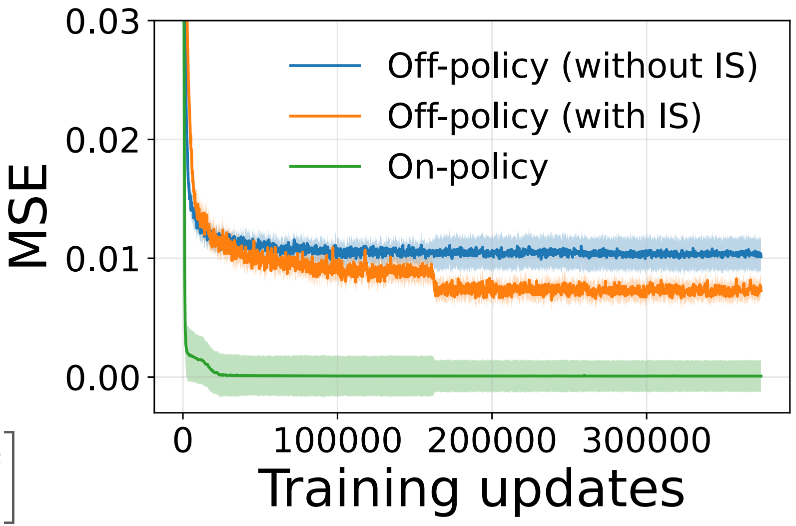
Take the behaviour characteristic loss:

$$\mathcal{L}(\beta) = \underset{p^\pi(s)}{\mathbb{E}} \underset{\mathrm{Unif}(a)}{\mathbb{E}} \underset{p(\mathcal{C})}{\mathbb{E}} \left|\pi(s,a) - \hat{\pi}(s,a \mid \mathcal{C};\beta)\right|^2$$

What if we cannot collect states from the policy we are explaining?

Draw states from the agent's own history, and reweight [3]:

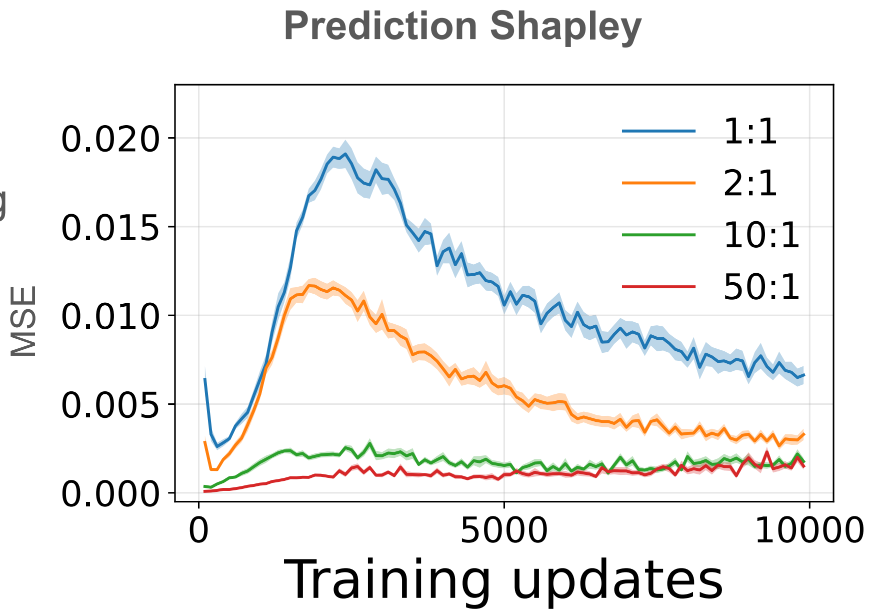
$$\mathcal{L}(\beta) \approx \underset{(s_t,a_t)\sim\mathcal{B}}{\mathbb{E}} \underset{p(\mathcal{C})}{\mathbb{E}} \left[\frac{\pi(s_t,a_t)}{\pi_t(s_t,a_t)} \cdot \left|\pi(s_t,a_t) - \hat{\pi}(s_t,a_t \mid \mathcal{C};\beta)\right|^2\right]$$



Explanations must track the agent's continuously changing policy.

Train the explanation models alongside it, using the same interaction data.

Error spikes when the policy shifts. More updates per policy update keeps them aligned.



Guess	Clue 1	Pos 1	Pos 2	Pos 3	Pos 4	Clue 2
6						
5		C	C	C	B	
4	2	B	C	C	C	2
3	2	C	B	C	C	2
2	0	C	C	C	C	3
1	0	A	A	C	A	1

3-letter alphabet. Over 2.8×10^{11} states, 36 features.

The recent guesses are enough to deduce the code is some combination of B, C, C, C.

The unused slots contribute nothing, recovering the nullity axiom.

Thank You for Listening!

Shapley Values for Explaining Reinforcement Learning (SVERL) [37, 40]

Explaining behaviour · Explaining outcomes · Explaining predictions

How to approximate SVERL in large-scale domains [41]

Parametric approximations of explanations

Learnt off-policy from the agent’s own history

Continually adapt to evolving agent behaviour

Interesting directions

A real-world application of SVERL · User studies

Website: djeb20.github.io

References

[1] Shapley, L.S. A value for n-person games. *Contributions to the Theory of Games* 2, 307-317 (1953).

[2] Charnes, A., Golany, B., Keane, M. et al. Extremal principle solutions of games in characteristic function form. *Econometrics of Planning and Efficiency* (1988).

[3] Precup, D., Sutton, R.S., Singh, S. Eligibility traces for off-policy policy evaluation. *ICML*, 759-766 (2000).

[4] Štrumbelj, E., Kononenko, I., Robnik Šikonja, M. Explaining instance classifications with interactions of subsets of feature values. *Data & Knowledge Engineering* 68(10), 886-904 (2009).

[5] Štrumbelj, E., Kononenko, I. An efficient explanation of individual classifications using game theory. *JMLR*, 11, 1-18 (2010).

[6] Štrumbelj, E., Kononenko, I. Explaining prediction models and individual predictions with feature contributions. *Knowl. Inf. Syst.* 41, 647-665 (2014).

[7] Mnih, V., Kavukcuoglu, K., Silver, D. et al. Human-level control through deep reinforcement learning. *Nature* 518, 529-533 (2015).

[8] Ribeiro, M.T., Singh, S., Guestrin, C. "Why should I trust you?" Explaining the predictions of any classifier. *KDD*, 1135-1144 (2016).

[9] Silver, D., Huang, A., Maddison, C. et al. Mastering the game of Go with deep neural networks and tree search. *Nature* 529, 484-489 (2016).

[10] Wang, Z., Schaul, T., Hessel, M. et al. Dueling network architectures for deep reinforcement learning. *ICML*, 1995-2003 (2016).

[11] Lundberg, S.M., Lee, S.-I. A unified approach to interpreting model predictions. *NeurIPS*, 30 (2017).

[12] Silver, D., Schrittwieser, J., Simonyan, K. et al. Mastering the game of Go without human knowledge. *Nature* 550, 354-359 (2017).

[13] Greydanus, S., Koul, A., Dodge, J. et al. Visualizing and understanding Atari agents. *ICML*, 1792-1801 (2018).

[14] Sutton, R.S., Barto, A.G. Reinforcement learning: An introduction. MIT Press (2018).

[15] Mott, A., Zoran, D., Chrzanowski, M. et al. Towards interpretable reinforcement learning using attention augmented agents. *NeurIPS* 32 (2019).

[16] Rizzo, S.G., Vantini, G., Chawla, S. Reinforcement learning with explainability for traffic signal control. *IEEE ITSC*, 3567-3572 (2019).

[17] Vinyals, O., Babuschkin, I., Czarnecki, W.M. et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575, 350-354 (2019).

[18] Bellemare, M.G., Candido, S., Castro, P.S. et al. Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature* 588, 77-82 (2020).

[19] Carbone, N.B. Explainable AI for path following with model trees. MSc thesis, NTNU (2020).

[20] Puri, N., Verma, S., Gupta, P. et al. Explain your move: understanding agent actions using feature attribution. *ICLR* (2020).

[21] Schrittwieser, J., Antonoglou, I., Hubert, T. et al. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature* 588, 604-609 (2020).

References

[22] Wang, Y., Mase, M., Egi, M. Attribution-based salience method towards interpretable reinforcement learning. *AAAI Spring Symposium* (2020).

[23] Zhang, K., Xu, P., Zhang, J. Explainable AI in deep reinforcement learning models: a SHAP method applied in power system emergency control. *IEEE EI2*, 711-716 (2020).

[24] Covert, I., Lundberg, S., Lee, S.-I. Explaining by removing: a unified framework for model explanation. *JMLR* 22(209), 1-90 (2021).

[25] Frye, C., de Mijolla, D., Begley, T. et al. Shapley explainability on the data manifold. *ICLR* (2021).

[26] He, L., Aouf, N., Song, B. Explainable deep reinforcement learning for UAV autonomous path planning. *Aerosp. Sci. Technol.* 118, 107052 (2021).

[27] Liessner, R., Dohmen, J., Wiering, M.A. Explainable reinforcement learning for longitudinal control. *ICAART*, 874-881 (2021).

[28] Løver, J., Gjærum, V.B., Lekkas, A.M. Explainable AI methods on a deep reinforcement learning agent for automatic docking. *IFAC-PapersOnLine* 54(16), 146-152 (2021).

[29] Schreiber, L., Ramos, G., Bazzan, A.L.C. Towards explainable deep reinforcement learning for traffic signal control. *LXAI Workshop at ICML* (2021).

[30] Zhang, K., Zhang, J., Xu, P.-D. et al. Explainable AI in deep reinforcement learning models for power system emergency control. *IEEE Trans. Comput. Soc. Syst.* 9(2), 419-427 (2021).

[31] Degrave, J., Felici, F., Buchli, J. et al. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature* 602, 414-419 (2022).

[32] Fawzi, A., Balog, M., Huang, A. et al. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature* 610, 47-53 (2022).

[33] Jethani, N., Sudarshan, M., Covert, I.C. et al. FastSHAP: Real-time Shapley value estimation. *ICLR* (2022).

[34] Remman, S.B., Strümke, I., Lekkas, A.M. Causal versus marginal Shapley values for robotic lever manipulation with deep reinforcement learning. *ACC*, 2683-2690 (2022).

[35] Theumer, P., Edenhofner, F., Zimmermann, R. et al. Explainable deep reinforcement learning for production control. *CPSL*, 809-818 (2022).

[36] Wurman, P.R., Barrett, S., Kawamoto, K. et al. Outracing champion Gran Turismo drivers with deep reinforcement learning. *Nature* 602, 223-228 (2022).

[37] Beechey, D., Smith, T.M.S., Şimşek, Ö. Explaining reinforcement learning with Shapley values. *ICML*, 2003-2014 (2023).

[38] Covert, I., Kim, C., Lee, S.-I. et al. Stochastic amortization: a unified approach to accelerate feature and data attribution. *arXiv*:2401.15866 (2024).

[39] Seo, J., Kim, S., Jalalvand, A. et al. Avoiding fusion plasma tearing instability with deep reinforcement learning. *Nature* 626, 746-751 (2024).

[40] Beechey, D., Smith, T.M.S., Şimşek, Ö. A theoretical framework for explaining reinforcement learning with Shapley values. *arXiv*:2505.07797 (2025).

[41] Beechey, D., Şimşek, Ö. Approximating Shapley explanations in reinforcement learning. *NeurIPS* (2025).