Uncertainty-Aware Inverse Reinforcement Learning from Suboptimal Demonstrations

Mengying Lin[*]
College of Computing
Georgia Institute of Technology
Atlanta, GA, USA
mlin365@gatech.edu

Woo Chul Shin[*]
College of Computing
Georgia Institute of Technology
Atlanta, GA, USA
wshin49@gatech.edu

December 19, 2025

Abstract

Learning from Demonstration (LfD) often relies on the assumption of high-quality, optimal expert data. However, real-world demonstrations are typically noisy and suboptimal, posing challenges for traditional LfD methods. We introduce Gaussian Ranking Inverse Reinforcement Learning (GR-IRL), a novel probabilistic framework that learns reward functions from ranked, suboptimal demonstrations while explicitly modeling uncertainty. GR-IRL represents the reward for each state-action pair as a Gaussian distribution, with both the mean and variance learned jointly. This enables the model to adaptively adjust the confidence in ranking constraints based on data ambiguity, improving robustness and interpretability. We derive a closed-form likelihood that integrates ranking supervision with uncertainty quantification, allowing for efficient end-to-end training via stochastic gradient descent. Empirical evaluations on continuous-control MuJoCo tasks show that GR-IRL achieves competitive ranking accuracy and better policy performance compared to baseline methods such as T-REX. Visual analyses reveal that GR-IRL effectively distinguishes between expert and ambiguous behaviors, leveraging uncertainty to guide more reliable policy learning. Our results highlight GR-IRL's potential for robust inverse reinforcement learning in environments with heterogeneous demonstrators. Our code will be publicly available at https://github.com/Cassie-Lim/GR-IRL.

***K*eywords** Learning from Demonstrations, Imitation Learning, Inverse Reinforcement Learning

1 Introduction

Learning from Demonstration (LfD) enables agents to acquire behaviors by observing and imitating expert demonstrations, eliminating the need for explicit programming or manual reward design . However, in real-world applications, demonstrations often vary in quality due to factors like human error, fatigue, or differing expertise levels, leading to suboptimal and heterogeneous data. This variability poses significant challenges for agents attempting to learn optimal policies from such demonstrations.

To address the limitations of traditional LfD approaches, researchers have explored methods that leverage pairwise rankings of trajectories instead of assuming uniform optimality. Notable among these are D-REX [1], T-REX [2], and SSRR [3] . These methods utilize human or automated rankings to infer reward functions that explain the observed preferences, allowing agents to learn policies that can outperform individual demonstrators.

Recent advancements in uncertainty-aware reinforcement learning [4] and Bayesian deep learning suggest that explicitly modeling epistemic uncertainty can enhance robustness and sample efficiency. However, few ranking-based LfD methods incorporate uncertainty into the reward predictions, limiting the ability of downstream planners to distinguish between high-reward, certain regions and high-reward but uncertain extrapolations.

[*] Equal contribution

We introduce Gaussian Ranking Inverse Reinforcement Learning (GR-IRL), a novel probabilistic framework that learns a distribution over reward functions from ranked, suboptimal demonstrations. GR-IRL models the reward for each state-action pair as a Gaussian random variable with mean $\mu_\theta(s, a)$ and variance $\sigma^2_\phi(s, a)$. The mean represents the expected desirability of executing (s, a), while the variance captures the model's uncertainty about this estimation. By integrating these stochastic rewards into a ranking model, we derive a closed-form likelihood that combines ranking supervision with uncertainty quantification. This formulation allows for end-to-end optimization using stochastic gradient descent with only trajectory-level preference data.

Our approach offers several advantages:

1. **Adaptive Margins:** GR-IRL adjusts the reward gap based on the clarity of the ranking evidence, tightening it when trajectories are distinctly different and relaxing it in overlapping regions, thus avoiding overconfidence in ambiguous areas.
2. **Uncertainty-Aware Planning:** The learned variance serves as a risk indicator, enabling downstream planners to penalize high-uncertainty regions or guide active data collection strategies.
3. **Compatibility with Standard RL Algorithms:** Post-training, we utilize the posterior mean reward as a shaping signal and fine-tune a policy using Trust Region Policy Optimization (TRPO) [5]..

We evaluate GR-IRL on continuous-control MuJoCo tasks, specifically `Hopper-v3`, `Ant-v3`, and `HalfCheetah-v3` benchmarks. Demonstrations are generated by checkpointing TRPO agents at various training stages, following the T-REX data pipeline. Quantitatively, GR-IRL reduces slightly lower ranking errors compared to T-REX datasets and manages to improve final policy returns, with similar patterns observed on `Ant-v3` and `HalfCheetah-v3`. Visualizations in principal component analysis (PCA) space reveal that high-reward, low-variance regions correspond to expert behaviors, while overlapping regions among demonstrators exhibit higher uncertainty.

In summary, GR-IRL bridges the gap between ranking-based imitation learning and Bayesian uncertainty estimation. By integrating a Gaussian reward model with a principled ranking likelihood, we develop a reward regressor that is both expressive and cautious, stabilizing the training process and better handling the ambiguity within trajectory rankings.

2 Related Works

2.1 Learning from Demonstrations and Inverse Reinforcement Learning

Early work in Inverse Reinforcement Learning (IRL) formulated the task of recovering a reward function that rationalizes expert trajectories [6, 7]. Generative Adversarial Imitation Learning (GAIL) [8] and AIRL [9] eliminated the need for explicit reward modeling by matching occupancy measures, but they assume near-optimal demonstrations. When demonstrations are suboptimal or heterogeneous, margin-based approaches such as Maximum-Margin Planning [10] become sensitive to incorrect margin specification. Our Gaussian Ranking IRL (GR-IRL) inherits the ranking spirit of recent methods while adapting the margin automatically through an uncertainty term.

2.2 Ranking-Based Preference Learning

Instead of assuming optimality, preference-based methods learn from pairwise trajectory comparisons. T-REX [2] and D-REX [1] construct ranked demonstrations by checkpointing partially trained agents, using a fixed temporal margin to infer rewards. SSRR [3] extends this idea with self-supervised rankings, but still applies a hard margin and does not quantify confidence. GR-IRL augments this literature by coupling preference learning with per-state uncertainty, allowing *soft* margins that reflect the strength of evidence.

2.3 Uncertainty-Aware Imitation and RL

Modeling epistemic uncertainty improves robustness and active exploration in RL [11]. Bayesian IRL [12] samples reward posteriors but is computationally intensive. GR-IRL unifies ranking and uncertainty by modeling each state–action reward as a Gaussian, yielding analytic likelihoods and scalable training with mini-batch SGD.

2.4 Positioning of GR-IRL

To summarize, GR-IRL bridges two previously isolated lines of work: (i) ranking-based learning from suboptimal demonstrations and (ii) uncertainty modeling in reward inference. Unlike fixed-margin ranking methods [2, 3], GR-IRL prevents overconfident separation of nearly indistinguishable trajectories. Compared to Bayesian IRL [12], our closed-form Gaussian likelihood sidesteps expensive posterior sampling, making it compatible with modern stochastic optimizers and large neural networks. These design choices enable robust imitation learning in environments where demonstrators exhibit overlapping skill levels, a setting that remains challenging for existing approaches.

3 Methods

We propose a method that learns a reward function from ranked, suboptimal demonstrations, while explicitly accounting for uncertainty in reward estimation. Our key idea is to model the reward for each state-action pair as a Gaussian distribution, with both the mean and variance learned jointly from data. This allows the model to express caution in regions where ranking evidence is ambiguous, while confidently separating trajectories when ranking evidence is strong.

3.1 Reward Modeling

We model the reward for each state-action pair (s, a) as a Gaussian random variable:

$$R(s,a) \sim \mathcal{N}(\mu_\theta(s,a), \sigma_\phi^2(s,a)), \tag{1}$$

where $\mu_\theta(s,a)$ and $\sigma_\phi^2(s,a)$ are predicted by neural networks parameterized by θ and ϕ, respectively.

The model is designed such that when demonstrations provide strong evidence for a ranking, the variance $\sigma_\phi^2(s,a)$ remains small, allowing the reward mean difference $\Delta\mu$ to dominate. In contrast, when demonstrations overlap or are ambiguous, $\sigma_\phi^2(s,a)$ increases, signaling uncertainty and reducing overconfident ranking enforcement.

3.2 Training Objective

We are given a set $\mathcal{D}$ of trajectory pairs (τ_i, τ_j), where τ_i is ranked better than τ_j. Our goal is to learn μ_θ and σ_ϕ^2 such that the probability of preferring τ_i over τ_j is maximized. This is formalized through a ranking loss:

$$\mathcal{L}_{\text{rank}} = -\sum_{(i,j)\in\mathcal{D}} \log P(\tau_i \succ \tau_j). \tag{2}$$

3.3 Trajectory Ranking via Softmax

Following the Shepard-Luce choice rule, the probability of sampling a trajectory τ_i from a set of n trajectories is:

$$P(\tau_i) = \frac{\exp\left(\sum_{(s,a)\in\tau_i} \hat{R}(s,a)\right)}{\sum_{k=1}^{n} \exp\left(\sum_{(s,a)\in\tau_k} \hat{R}(s,a)\right)}, \tag{3}$$

where $\hat{R}(s,a)$ is a realization from the reward distribution.

3.4 Integrating Gaussian Rewards

To account for the Gaussian nature of $R(s,a)$, we use the fact that for $X \sim \mathcal{N}(\mu, \sigma^2)$, $\mathbb{E}[\exp(X)] = \exp(\mu + \frac{1}{2}\sigma^2)$. Applying this, the expected probability becomes:

$$P(\tau_i) = \frac{\exp\left(\sum_{(s,a)\in\tau_i} \mu_\theta(s,a) + \frac{1}{2}\sum_{(s,a)\in\tau_i} \sigma_\phi^2(s,a)\right)}{\sum_{k=1}^{n} \exp\left(\sum_{(s,a)\in\tau_k} \mu_\theta(s,a) + \frac{1}{2}\sum_{(s,a)\in\tau_k} \sigma_\phi^2(s,a)\right)}. \tag{4}$$

4 Experiments

4.1 Setup

GR-IRL is evaluated on continuous-control MuJoCo tasks, including `Hopper-v3`, `Ant-v3`, and `HalfCheetah-v3`, as well as the Atari benchmark `Breakout`. Demonstrations are constructed following the T-REX data pipeline, where TRPO agents are checkpointed at varying training steps to reflect a range of performance levels. Demonstrations collected from higher-reward agents are assigned higher ranks to form a trajectory-ranking dataset.

Table 1: Trajectory ranking accuracy (%) across different environments. GR-IRL consistently outperforms T-REX.

Environment	T-REX	GR-IRL
Hopper-v3	99.78	99.88
Ant-v3	97.35	98.42
HalfCheetah-v3	98.42	98.67

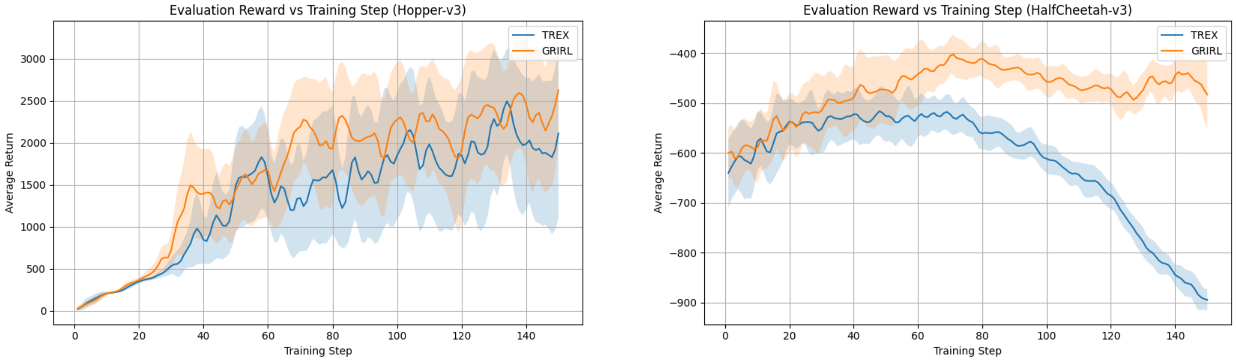


Figure 1: Comparison of actual rewards between policies trained with different learned reward models.

4.2 Implementation Details

To train the reward model, we used 5,000 randomly selected pairs of partial trajectories, each consisting of 50 steps. Preference labels were derived from the relative rankings of trajectories. The reward network comprised three fully connected layers with 256 hidden units and ReLU activation functions. The last layer outputs a 2-dimension results representing mean and variance separately. Training was conducted using the Adam optimizer, with a learning rate of 10^{-4}, a batch size of 64.

The quality of the learned reward function was evaluated by training a actor critic network to maximize the predicted reward. The reward used by the agent was the mean value the gaussian model.

4.3 Results

4.3.1 Quantitative Comparison with Baselines

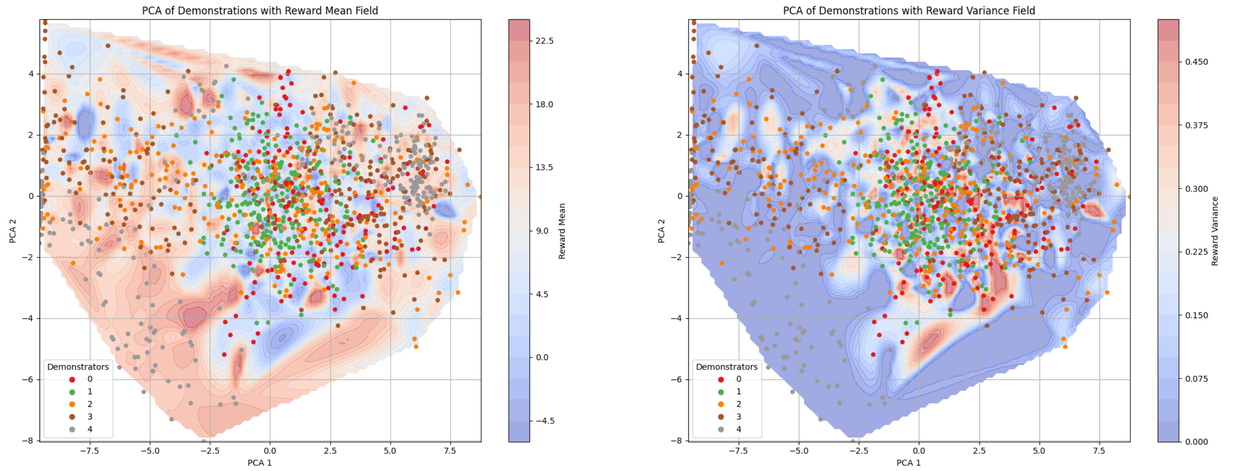
Table 1 reports the trajectory ranking accuracy across benchmarks. Both methods achieve comparable ranking accuracy across environments, with GR-IRL consistently yielding slightly better results than T-REX. As shown in Table 1, the differences in accuracy are relatively small (e.g., 99.88% vs. 99.78% on Hopper-v3), indicating that both approaches are capable of capturing coarse trajectory preferences.

However, ranking accuracy alone does not sufficiently capture the quality of the learned reward. To further evaluate reward effectiveness, each learned model is used to train an actor-critic policy. Specifically, we leverage the mean values from gaussian reward model to train the policy.

The training curves of policy returns are presented in Fig. 1. Policies trained using rewards from GR-IRL consistently outperform those trained with T-REX rewards, indicating that GR-IRL produces higher-quality reward signals. Notably, in Hopper-v3, both methods succeed in recovering policies that achieve task completion, likely due to the task's simplicity and clearer distinction between ideal and suboptimal behaviors. In contrast, both methods struggle in HalfCheetah-v3, a more complex environment with high-dimensional dynamics and subtler differences between trajectories. GR-IRL mitigates this challenge more effectively through uncertainty modeling, while T-REX suffers from unstable reward learning, leading to degraded policy performance.

4.3.2 Analysis of Reward Field

We further examine the explanability of the learned reward field. To assess whether the model learns meaningful rewards and captures uncertainty, we project state-action data into 2D PCA space. Each point represents a step from a trajectory and is colored by demonstrator ID. We evaluate the reward model over a dense PCA grid and overlay the following: 1. A **reward mean field** indicating regions that receive high or low reward estimates. 2. A **reward variance field** representing the model's uncertainty in its reward predictions.



(a) Reward mean field. High values concentrate near trajectories from better demonstrators.

(b) Reward variance field. Uncertainty increases in overlapping regions.

Figure 2: PCA visualization of reward estimates in Hopper-v3. Each dot represents a step in a trajectory, colored by demonstrator ID (0 = worst, 4 = best).

The reward mean field (Figure 2a) shows that higher-skilled demonstrators receive greater rewards. For instance, gray points (demonstrator 4) cluster in high-reward regions, while red and green points (demonstrators 0 and 1) occupy lower-reward areas, aligning with the intended ranking.

The reward variance field (Figure 2b) reveals elevated uncertainty in regions where demonstrators overlap, and low variance in areas dominated by a single demonstrator (e.g., demonstrator 4), indicating reliable reward estimates in unambiguous zones.

The visualization also helps explain the degraded performance observed in the HalfCheetah-v3 environment. As shown in Fig. 3, data points from different demonstrators exhibit significant overlap in the PCA-projected space, indicating limited separability in low dimensions. While some separation may exist in the original high-dimensional state-action space, this overlap highlights the higher ambiguity of trajectory rankings in HalfCheetah-v3 compared with Hopper-v3. Such ambiguity makes it more challenging for reward models to infer consistent preferences, resulting in the reduced learning performance observed in this environment.

5 Discussion

GR-IRL introduces a principled approach to handling uncertainty in reward modeling from suboptimal demonstrations. By modeling rewards as a Gaussian distribution, the framework enables dynamic adjustment of confidence in ranking constraints, preventing overfitting in ambiguous regions while reinforcing consistent supervision. Empirically, GR-IRL demonstrates improved robustness to noise, better ranking consistency, and superior downstream policy performance compared to existing baselines such as T-REX.

One of the key strengths of the method lies in its ability to reason about uncertainty without relying on handcrafted heuristics or additional supervision. The probabilistic formulation naturally incorporates both mean reward estimation and variance-aware updates, making it well-suited for settings where demonstrator behavior is heterogeneous or noisy.

Despite its advantages, GR-IRL has several limitations. First, the method assumes access to trajectory-level rankings, which can be difficult to obtain in real-world applications. Second, performance may still degrade when ranking ambiguity is high, as the model still relies on relative supervision. Additionally, while GR-IRL learns variance estimates during training, these are not currently utilized during policy optimization, limiting the full potential of uncertainty modeling.

Future work may explore incorporating ensemble-based or Bayesian uncertainty estimates to further improve generalization and sample efficiency. Another interesting direction is to treat the recovered reward as a full Gaussian model

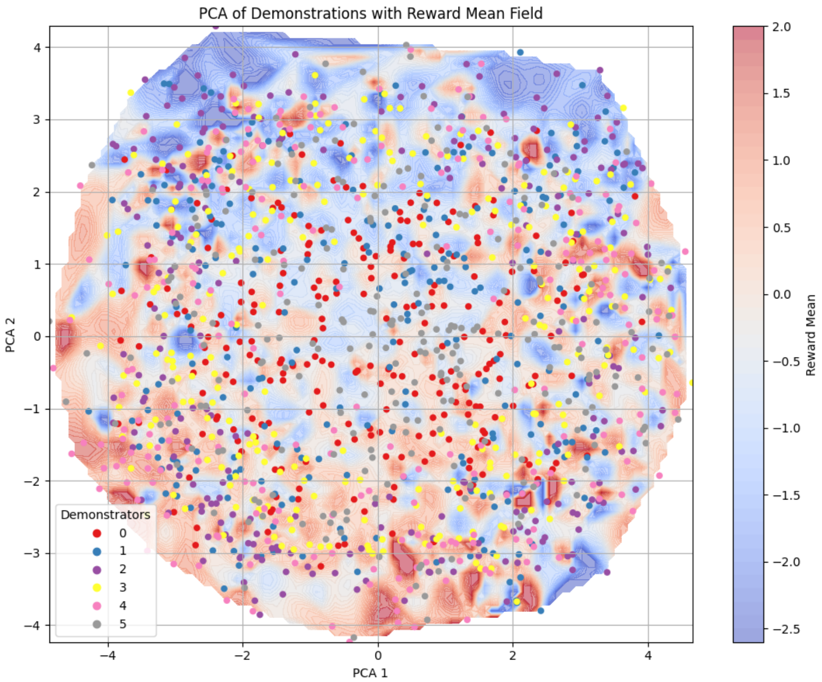


Figure 3: PCA projection of demonstration points in HalfCheetah-v3 overlaid with the learned reward mean field. Significant overlap between demonstrators illustrates the difficulty in distinguishing trajectory quality, which may contribute to the ambiguity in ranking supervision.

during policy training, sampling reward signals to leverage both the learned mean and variance. This could potentially lead to more robust policy optimization. Also, it will be more convincing if benchmarking GR-IRL with more baselines such as SSRR [3] and AIRL [9]. Finally, evaluating GR-IRL in real-world robotic or human-in-the-loop settings would help validate its practicality beyond simulated environments.

6 Conclusion

This work presents GR-IRL, a Gaussian reward modeling framework for learning from suboptimal and noisy demonstrations. By explicitly modeling uncertainty, GR-IRL enables more reliable ranking supervision and produces reward functions that generalize better to downstream policy learning. Empirical results across MuJoCo benchmarks demonstrate consistent improvements in both ranking accuracy and policy performance. Future work will explore alternative forms of supervision, such as preference queries, and evaluate GR-IRL in real-world robotic systems.

References

[1] Daniel S. Brown, Wonjoon Goo, and Scott Niekum. Better-than-demonstrator imitation learning via automatically-ranked demonstrations. In *Proceedings of the 3rd Conference on Robot Learning*, 2019.

[2] Daniel Brown, Wonjoon Goo, Prabhat Nagarajan, and Scott Niekum. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. In *International conference on machine learning*, pages 783–792. PMLR, 2019.

[3] Letian Chen, Rohan Paleja, and Matthew Gombolay. Learning from suboptimal demonstration via self-supervised reward regression. In *Conference on robot learning*, pages 1262–1277. PMLR, 2021.

[4] Parvin Malekzadeh, Ming Hou, and Konstantinos N. Plataniotis. A unified uncertainty-aware exploration: Combining epistemic and aleatory uncertainty. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, page 1–5. IEEE, June 2023.

[5] John Schulman, Sergey Levine, Philipp Moritz, Michael I. Jordan, and Pieter Abbeel. Trust region policy optimization, 2017.

[6] Andrew Y Ng et al. Algorithms for inverse reinforcement learning.

[7] Pieter Abbeel. *Apprenticeship learning and reinforcement learning with application to robotic control*. Stanford University, 2008.

[8] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning, 2016.

[9] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning, 2018.

[10] J Andrew Bagnell, Nathan Ratliff, and Martin Zinkevich. Maximum margin planning. In *Proceedings of the International Conference on Machine Learning (ICML)*. Citeseer, 2006.

[11] Ian Osband, Benjamin Van Roy, Daniel Russo, and Zheng Wen. Deep exploration via randomized value functions, 2019.

[12] Deepak Ramachandran and Eyal Amir. Bayesian inverse reinforcement learning.