
Towards Inverse Reinforcement Learning for Limit Order Book Dynamics

Jacobo Roa-Vicens^{1,2} Cyrine Chtourou¹ Angelos Filos³ Francisco Rul-ian² Yarin Gal³ Ricardo Silva²

Abstract

Multi-agent learning is a promising method to simulate aggregate competitive behaviour in finance. Learning expert agents' reward functions through their external demonstrations is hence particularly relevant for subsequent design of realistic agent-based simulations. Inverse Reinforcement Learning (IRL) aims at acquiring such reward functions through inference, allowing to generalize the resulting policy to states not observed in the past. This paper investigates whether IRL can infer such rewards from agents within real financial stochastic environments: limit order books (LOB). We introduce a simple one-level LOB, where the interactions of a number of stochastic agents and an expert trading agent are modelled as a Markov decision process. We consider two cases for the expert's reward: either a simple linear function of state features; or a complex, more realistic non-linear function. Given the expert agent's demonstrations, we attempt to discover their strategy by modelling their latent reward function using linear and Gaussian process (GP) regressors from previous literature, and our own approach through Bayesian neural networks (BNN). While the three methods can learn the linear case, only the GP-based and our proposed BNN methods are able to discover the non-linear reward case. Our BNN IRL algorithm outperforms the other two approaches as the number of samples increases. These results illustrate that complex behaviours, induced by non-linear reward functions amid agent-based stochastic scenarios, can be deduced through inference, encouraging the use of inverse reinforcement learning for opponent-modelling in multi-agent systems.

¹JPMorgan Chase & Co., London, United Kingdom

²Department of Statistical Science, University College London, United Kingdom ³Department of Computer Science, University of Oxford, United Kingdom. Correspondence to: Jacobo Roa-Vicens <jacobo.roavicens@jpmorgan.com>.

Published as a workshop paper on AI in Finance: Applications and Infrastructure for Multi-Agent Learning at the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

1. Introduction

Limit order books play a central role in the formation of prices for financial securities in exchanges globally. These systems centralize limit orders of price and volume to buy or sell certain securities from large numbers of dealers and investors, matching bids and offers in a transparent process. The dynamics that emerge from this aggregate process (Preis et al., 2006; Cont et al., 2010) of competitive behaviour fall naturally within the scope of multi-agent learning, and understanding them is of paramount importance to liquidity providers (known as market makers) in order to achieve optimal execution of their operations. We focus on the problem of learning the latent reward function of an expert agent from a set of observations of their external actions in the LOB.

Reinforcement learning (RL) (Sutton & Barto, 2018) is a formal framework to study sequential decision-making, particularly relevant for modelling the behaviour of financial agents in environments like the LOB. In particular, RL allows to model their decision-making process as agents interacting with a dynamic environment through policies that seek to maximize their respective cumulative rewards. Inverse reinforcement learning (Russell, 1998) is therefore a powerful framework to analyze and model the actions of such agents, aiming at discovering their latent reward functions: the most "succinct, robust and transferable definition of a task" (Ng et al., 2000). Once learned, such reward functions can be generalized to unobserved regions of the state space, an important advantage over other learning methods.

Overview. The paper starts at Section 2 by introducing the foundations of IRL, relevant aspects of the maximum causal entropy model used, and useful properties of Gaussian processes and Bayesian neural networks applied to IRL. A formal description of the limit order book model follows. In Section 3, we formulate the one-level LOB as a finite Markov decision process (MDP) and express its dynamics in terms of a Poisson binomial distribution. Finally, we investigate the performance of two well-known IRL methods (maximum entropy IRL and GP-based IRL), and propose and test an additional IRL approach based on Bayesian neu-

Opinions expressed in this paper are those of the authors, and do not necessarily reflect the view of JP Morgan.

ral networks. Each IRL method is tested on two versions of the LOB environment, where the reward function of the expert agent may be either a simple linear function of state features, or a more complex and realistic non-linear reward function.

Related Work. Agent-based models of financial market microstructure are extensively used (Preis et al., 2006; Navarro & Larralde, 2017; Wang & Wellman, 2017). In most setups, mean-field assumptions (Lasry & Lions, 2007) are made to obtain closed form expressions for the dynamics of the complex, multi-agent environment of the exchanges. We make similar assumptions to obtain a tractable finite MDP model for the one-level limit order book.

Previous attempts have been made to model the evolution of the behaviour of large populations over discrete state spaces, combining MDPs with elements of game theory (Yang et al., 2017), using maximum causal entropy inverse reinforcement learning. Their focus had been around social media data. Recently, Hendricks et al. (2017) used IRL in financial market microstructure for modelling the behaviour of the different classes of agents involved in market exchanges (e.g. high-frequency algorithmic market makers, machine traders, human traders and other investors). We draw inspiration from them, and distinguish two types of agents: automatic agents that induce our environment’s dynamics, and active expert agents that trade in such environment. We focus on inferring the expert agents’ objectives from the point of view of an external observer.

Our simplified MDP model could be seen as a variant of the multi-agent Blotto environment (Borel, 1921; Tukey, 1949; Roberson, 2006; Balduzzi et al., 2019). Blotto is a resource distribution game consisting of two opponent armies having each a limited number of soldiers that need to be distributed across multiple areas or battlefields. Each area is won by the army that has the highest number of soldiers. The winner army is the one that has majority over the highest number of battlefields. This environment is often used to model electoral competition problems where parties have a limited budget and need to reach a maximum number of voters. In our environment, only two areas are used (best bid and ask), but the decisions are conditional to a state, hence the MDP could be seen as a contextual 2-area Blotto variant.

Contributions. We propose a multi-agent LOB model which provides the possibility of obtaining transition probabilities in closed form, enabling the use of model-based IRL, without giving up reasonable proximity to real world LOB settings. The reward functions we propose have a clear financial interpretation, and allow flexible and comparable testing of different IRL methods in a setup that can be scaled to higher dimension versions of the environment. We then provide results for the three IRL methods discussed: max-

imum entropy IRL, Gaussian process-based IRL, and our IRL approach based on Bayesian neural networks. Results show that BNNs are able to recover the target rewards, outperforming comparable methods both in IRL performance and in terms of computational efficiency.

2. Background

2.1. Inverse Reinforcement Learning

We base our IRL experiments on a Markov decision process consisting of a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma, P_0 \rangle$. \mathcal{S} represents the state space; \mathcal{A} the set of eligible actions; \mathcal{T} represents the model’s transition dynamics, where $\mathcal{T}(s', a, s) = p(s'|s, a)$ is the transition probability to state s' from state s through action a ; $r(s, a)$ is the unknown reward function we intend to recover; the discount factor γ takes values in $[0, 1]$; and P_0 represents the initial state distribution.

Reinforcement learning aims under its general forward formulation at finding an optimal policy π^* that maximizes the expected cumulative reward $\mathbb{E}[\sum_{t=0}^T \gamma^t r(s_t) | \pi^*]$, where the state-action pairs induced by policy π^* and transition dynamics \mathcal{T} are denoted in a sequence or trajectory $\mathbf{x} = \{(s_t, a_t)\}_{t=0}^T$.

On the other hand, inverse reinforcement learning aims at recovering through inference (see Figure 1) an unknown reward function $r(s, a)$, where we assume $\pi^*(a|s)$ to be an optimal policy from where a collection of expert demonstrations $\mathcal{D} = \{\mathbf{x}_n\}_{n=1}^N$ is drawn. However, one of the main difficulties of the IRL problem is its ill-posed nature, as there could be more than one optimal policy that explains a given set of demonstrations \mathcal{D} (Ng et al., 1999). This ambiguity is handled by the maximum entropy framework (Ziebart et al., 2008).

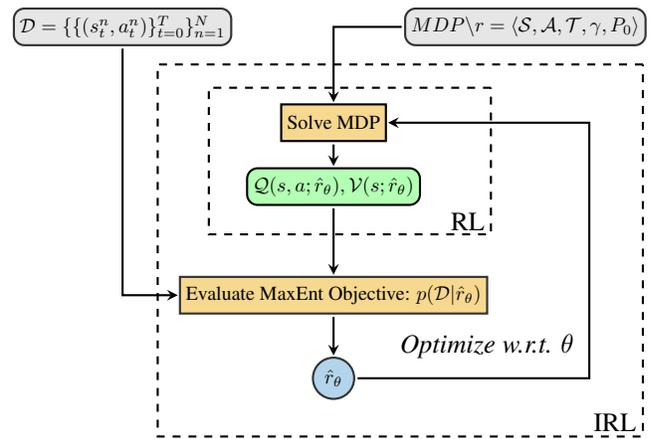


Figure 1. Flow diagram of maximum entropy-based IRL.

Maximum Causal Entropy Model. The principle of maximum causal entropy (Ziebart, 2010) assumes that the

distribution of trajectories generated by the optimal policy follows $\pi^*(a|s) \propto \exp\{Q'(s_t, a_t)\}$ (Ziebart et al., 2008; Haarnoja et al., 2017), where Q' is a soft Q-function that incorporates the treatment of sequentially-revealed information, regularizing the objective of forward reinforcement learning with respect to differential entropy $\mathcal{H}(\cdot)$ as described in Fu et al. (2017): $Q'(s_t, a_t) = r_t(s, a) + \mathbb{E}_{(s_{t+1}, \dots) \sim \pi} \left[\sum_{t'=t}^T \gamma^{t'} (r(s_{t'}, a_{t'}) + \mathcal{H}(\pi(\cdot|s_{t'}))) \right]$.

The inference problem for an unknown reward function $r(s, a)$ thus boils down to maximizing the likelihood $P(\mathcal{D}|r)$ as in Levine et al. (2011):

$$\exp \sum_i \sum_t \left[Q'_{s_i, t, a_i, t} - \log \sum_{a'} \exp(Q'_{s_i, t, a', i, t}) \right]$$

2.2. IRL methods considered

The three inverse reinforcement learning methods that we will test on our LOB model for both linear and exponential expert rewards are: maximum entropy IRL (MaxEnt), Gaussian processes-based IRL (GPIRL), and our implementation through Bayesian neural networks (BNN IRL). These methods are defined as follows:

Maximum Entropy IRL. Proposed by Ziebart (2010), this method builds on the maximum causal entropy objective function described above, combined with a linearity assumption on the structure of the reward as a function of the state features: $\mathbf{r}(s) = \mathbf{w}^T \phi(s)$, where $\phi(s) : \mathcal{S} \mapsto \mathbb{R}^m$ is the m -dimensional mapping from the state to the feature vector. The IRL problem then boils down to the inference of such \mathbf{w} .

Gaussian Process-based IRL. Maximum causal entropy provides a method to infer values of the reward function on specific points of the state space. This is enough for cases where the MDP is finite and where the observed demonstrations cover all the state space, which are not very common. In this context, it is important to learn a structure of rewards that mirrors as closely as possible the behaviour of the expert agent.

While many prior IRL methods assume linearity of the reward function, GP-based IRL (Levine et al., 2011), expands the function space of possible inferred rewards to non-linear reward structures. Under this model, the reward function r is modeled by a zero-mean GP prior with a kernel or covariance function k_θ . If $X_f \in \mathbb{R}^{n, m}$ is the feature matrix defining a finite number n of m -dimensional states, and f is the reward function evaluated at X_f : $f = r(X_f)$, then $f|X, \theta \sim \mathcal{N}(0, K_{f, f})$ where $[K_{f, f}]_{i, j} = k_\theta(x_i, x_j)$.

The GPIRL objective (Levine et al., 2011) is then maxi-

mized with respect to the finite rewards f and θ :

$$p(f, \theta, \mathcal{D}|X_f) = \left[\int_r p(\mathcal{D}|r) p(r|f, \theta, X_f) dr \right] p(f, \theta|X_f)$$

Inside the above integral, we can recognize the IRL objective $p(\mathcal{D}|r)$, the GP posterior $p(r|f, \theta, X_f)$ and the prior of f and θ . To mitigate the intractability of this integral, Levine et al. (2011) use the Deterministic Training Conditional (DTC) approximation, which reduces the GP posterior to its mean function.

Bayesian Neural Networks applied to IRL. A neural network (NN) is a superposition of multiple layers consisting of linear transformations and non-linear functions. Infinitely wide NNs whose weights are random variables have been proven to converge to a GP (Neal, 1995; Williams & Jacobs, 1997) and hence represent a universal function approximator (Cybenko, 1989). However, this convergence property is not applicable to finite NNs, which are the networks used in practice. Bayesian neural networks are equivalent networks in the finite case. BNNs have been the focus of multiple studies (Neal, 1995; MacKay, 1992; Gal & Ghahramani, 2016) and are known for their useful regularization properties. Since exact inference is computationally challenging, variational inference has instead been used to approximate these models (Hinton & Van Camp, 1993; Peterson, 1987; Graves, 2011; Gal & Ghahramani, 2016).

Given a dataset $\mathcal{Z} = \{(x_n, y_n)\}_{n=1}^N$, we wish to learn the mapping $\{(x_n \rightarrow y_n)\}_{n=1}^N$ in a robust way. A BNN can be characterized through a prior distribution $p(\mathbf{w})$ placed on its weights, and the likelihood $p(\mathcal{Z}|\mathbf{w})$. An approximate posterior distribution $q(\mathbf{w})$ can be fit through Bayesian variational inference, maximizing the evidence lower bound (ELBO):

$$\mathcal{L}_q = \mathbb{E}_q[\log p(\mathcal{Z}|\mathbf{w})] - \text{KL}[q(\mathbf{w})||p(\mathbf{w})]$$

We parametrize $q(\mathbf{w})$ with θ and use the mean-field variational inference approximation (Peterson, 1987), which assumes that the weights of each layer factorize to independent distributions. We choose a diagonal Gaussian prior distribution $p(\mathbf{w})$ and solve the optimization problem:

$$\max_{\theta} \mathcal{L}_{q\theta}$$

In the context of the IRL problem, we leverage the benefits of BNNs to generalize point estimates provided by maximum causal entropy to a reward function in a robust and efficient way. Unlike in GPIRL, where the full objective is maximized at each iteration, BNN IRL consists of two separate steps:

- **Inference step:** we first optimize the IRL objective $p(\mathcal{D}|\hat{r})$, obtaining a finite number of point estimates of the reward $\{\hat{r}(s_n) = \hat{r}_n\}_{n=1}^N$ over a finite number of states $\{s_n \in \mathcal{S}\}_{n=1}^N$
- **Learning step:** secondly, we use these point estimates to train a BNN that learns the mapping between state features $s \in \mathcal{S}$ and their respective inferred reward values $\hat{r}(s) \in \mathbb{R}$.

The number of point estimates used is the number of states existing in the expert’s demonstrations. This means that we do not use each state in the state space once, but as many times as it exists in the demonstrations \mathcal{D} . This is equivalent to tailoring the learning rate of the Bayesian neural network to match the state visitation counts.

2.3. Limit Order Book

Our experimental setup builds on limit order books (LOBs): here we introduce some basic definitions following the conventions of Gould et al. (2013) and Zhang et al. (2019). Two types of orders exist in an LOB: bids (buy orders) and asks (sell orders). A bid order for an asset is a quote to buy the underlying asset at or below a specified price $\mathbf{P}_b(t)$. Conversely, the ask order is a quote to sell the asset at or above a certain price $\mathbf{P}_a(t)$. For a certain price, the bid and ask orders have respective sizes (volumes) $\mathbf{V}_b(t)$ and $\mathbf{V}_a(t)$. Each level (l) of the LOB is represented by one pair $(p_b^{(l)}(t), v_b^{(l)}(t))$ or $(p_a^{(l)}(t), v_a^{(l)}(t))$, generally ranked in decreasing order of competitiveness.

3. Experiments

Experimental Setup. Our environment setup is a one-level LOB, resulting from the interaction of N (Markovian) trading agents (TA) and an expert agent (EA trader). Without loss of generality, the EA is regulated to have up to I_{\max} inventory (amount of securities in their portfolio). The EA is unaware of the strategies of each of the other trading agents, but adapts to them through trial and error, as in reinforcement learning frameworks (Sutton & Barto, 2018). The EA solves the following MDP $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma, P_0 \rangle$:

- **State Space \mathcal{S} .** The environment state at time-step t , \mathbf{s}_t , is a 3-dimensional vector:

$$\mathbf{s}_t = \left[v_b^{(1)}(t) \quad v_a^{(1)}(t) \quad i^{(\text{EA})}(t) \right]^T \in \mathcal{S} \quad (1)$$

where $v_b^{(1)}(t), v_a^{(1)}(t) \in \mathbb{R}_+$ are the volumes available in the one-level LOB through the best bid and ask levels, respectively, and $i^{(\text{EA})}(t) \in \{-I_{\max}, \dots, 0, \dots, +I_{\max}\}$ is the inventory held by the expert agent at time-step t .

- **Action Space \mathcal{A} .** The expert agent chooses to take an action \mathbf{a}_t at time-step t , by selecting volumes $v_b^{(\text{EA})}(t)$ and $v_a^{(\text{EA})}(t)$ to match the trading objectives defined by their reward with those available in the LOB through the other traders, at the best bid and ask, respectively:

$$\mathbf{a}_t = \left[v_b^{(\text{EA})}(t) \quad v_a^{(\text{EA})}(t) \right]^T \in \mathcal{A} \quad (2)$$

where $v_b^{(\text{EA})}(t) + v_a^{(\text{EA})}(t) \leq N$ (see *Transition Dynamics* below). The EA is here an active market participant, which actively sells at the best ask and buys at the best bid, while the trading agents on the other side of the LOB only place passive orders.

- **Transition Dynamics \mathcal{T} .** At each time-step t , the n -th trader, $n \in \{1, \dots, N\}$, places a single order, $o_t^{(n)}$ at either side of the LOB (best bid, or best ask). The traders follow stochastic policies:

$$\pi^{(n)}(\cdot | \mathbf{s}_t; \tau^{(n)}) = \text{Ber} \left[\frac{e^{\frac{v_b^{(1)}(t-1)}{\tau^{(n)}}}}{e^{\frac{v_b^{(1)}(t-1)}{\tau^{(n)}}} + e^{\frac{v_a^{(1)}(t-1)}{\tau^{(n)}}}} \right] \quad (3)$$

$$o_t^{(n)} \sim \pi^{(n)}(\cdot | \mathbf{s}_t; \tau^{(n)}) \quad (4)$$

where $o_t^{(n)} = 1$ corresponds to a trader placing an order at the best bid, and $o_t^{(n)} = 0$ at the best ask. By construction, each of the trading agents has to place exactly one order, a bid or an ask, at each time-step. $\text{Ber}(\cdot)$ denotes the Bernoulli distribution. The temperature parameters $\tau = (\tau_1, \dots, \tau_n)$ are independent and unknown to the expert agent.

Hence, the aggregate intermediate bid orders that were generated by the N trading agents, $v_b^{(\text{TA})}(t)$ are the sum of N independent Bernoulli variables, whose parameter is conditioned on the environment state \mathbf{s}_t and the idiosyncratic temperature parameters τ .

Therefore, the number of intermediate bids is distributed according to a Poisson binomial distribution (Shepp & Olkin, 1981), whose probability mass function can be expressed in closed form. Since each of the trading agents (but not the EA) should place exactly one order, the aggregate intermediate ask orders are $v_a^{(\text{TA})}(t) = N - v_b^{(\text{TA})}(t)$. The aggregate bids and asks of the trading agents form an intermediate state called the belief state $\mathbf{b}_t = (v_b^{(\text{TA})}(t), v_a^{(\text{TA})}(t))$.

Although the EA can only see the last available LOB snapshot contained in \mathbf{s}_t , their orders (\mathbf{a}_t) are executed against the intermediate state \mathbf{b}_t . This could be compared in real world LOBs to a slippage effect. Based on the number of bids and asks finally matched by the

EA in the LOB, their inventory is updated as a net long or short position. Considering the above dynamics of the LOB, the transition matrix \mathcal{T} , can be calculated exactly. This allows for an exact solution of the MDP. An illustration of the dynamics is provided in Figure 2.

- **Reward Function r .** The selection of the reward function is crucial, since it induces the behaviour of the agent (Berger, 2013). The reward function can be either a function of state s_t and action a_t ; or equivalently, following the dynamics \mathcal{T} , of the next state s_{t+1} , (Sutton & Barto, 2018). We follow the latter convention, and use the following two reward functions for the expert agent to test each IRL method considered:

1. **Linear reward**, equal to the EA’s hit count:

$$r(s) = N - v_b^{(1)} - v_a^{(1)} \quad (5)$$

This reward is equivalent to the hit count of the EA because there are always exactly N orders in the intermediate state b_t , and therefore the next state s_t resulting from the execution of the EA’s active orders a_t against the TA orders b_t , can be effectively seen as the remaining orders, i.e. the number of orders out of a maximum of N that the EA was not able to match.

This reward function does not incentivize directly to sell inventory. However, since the episode is terminated when maximum inventory I_{\max} is exceeded, the market maker is implicitly motivated not to violate this constraint, since the simulation will then be terminated and the cumulative reward will be reduced.

2. **Exponential reward** is a model widely used in economic theory to characterize levels of risk aversion (Pratt, 1964). We may use it to define a non-linear reward function that depends on both the inventory of the EA and their hit count:

$$r(s) = 1 - e^{-\alpha*(N-v_b^{(1)}-v_a^{(1)}-\beta*|i^{(\text{EA})}|)} \quad (6)$$

where $\alpha, \beta \in \mathbb{R}_{\geq 0}$ are chosen based on the level of risk aversion of the agent.

Both reward structures are illustrated in Figures 3- 4.

Results and Discussion. Our experiments are performed on synthetic data, where $N = 3$ is the number of trading agents with temperatures $\tau = (0.1, 0.5, 1.0)$, and $I_{\max} = 5$. We also consider an episodic setup, where the simulation is terminated either when the maximum inventory is violated, or $T = 5$ time-steps are completed. The initial state is uniformly sampled from the non-terminal states in \mathcal{S} .

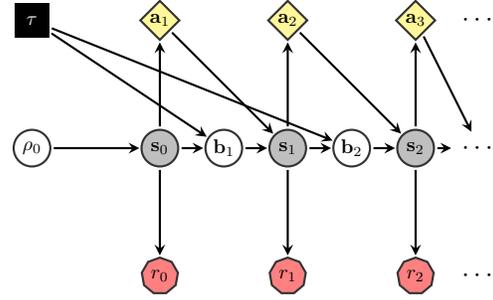


Figure 2. Model of our Markov decision process for a one-level LOB environment. The expert agent takes actions a_t (diamond yellow nodes), conditioned on the observed states s_t (shaded circle nodes), receiving reward r_t (polygon red nodes). The trading agents, given the deterministic temperature parameters τ (black square node), place orders which comprise the latent belief state b_t . The next state s_{t+1} is a function of the expert agent’s action and the latent belief states, which the expert agent should learn to infer in order to optimize cumulative returns.

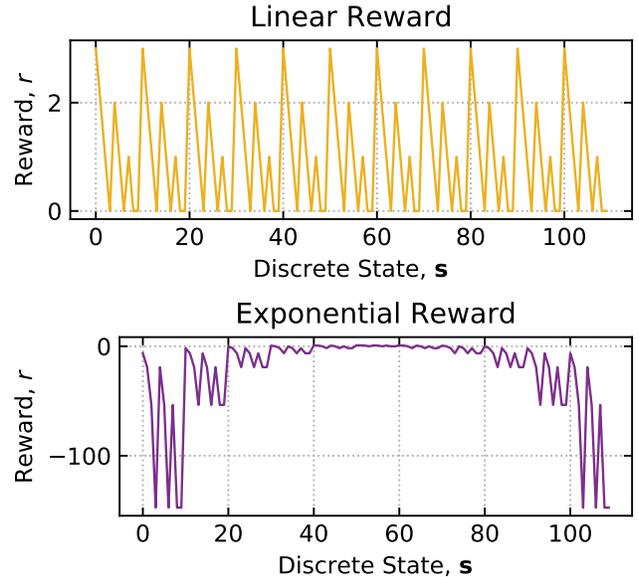


Figure 3. Reward functions (i.e. linear and exponential) with respect to the discrete (enumerated) states, e.g. $s = 0$ corresponds to an inventory of $-I_{\max} = -5$ and to bid and ask volumes of 0; while $s = 1$ corresponds to an inventory of -5, a bid volume of 0 and and ask volume of 1, etc.

- **Performance metric.** Following previous IRL literature (Jin et al., 2017; Wulfmeier et al., 2015) we evaluate the performance of each method through their respective Expected Value Differences (EVD). EVD quantifies the difference between:

1. the expected cumulative reward earned by following the optimal policy π^* implied by the true rewards;

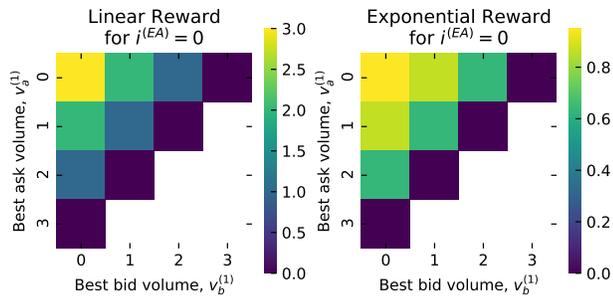


Figure 4. Reward functions (linear and exponential) with respect to the state features: volume of best bid and ask and expert agent’s inventory.

- the expected cumulative reward earned by following the policy $\hat{\pi}$ implied by the rewards inferred through IRL.

$$\text{EVD} = \mathbb{E} \left[\sum_{t=0}^T \gamma^t r(s_t) | \pi^* \right] - \mathbb{E} \left[\sum_{t=0}^T \gamma^t r(s_t) | \hat{\pi} \right] \quad (7)$$

Since the expert’s observed behaviour could have been generated by different reward functions, we compare the EVD yielded by inferred rewards per method, rather than directly comparing each inferred reward against the ground truth reward.

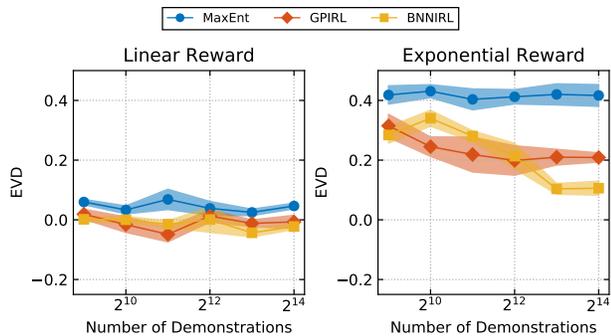


Figure 5. EVD for both the linear and the exponential reward functions as inferred through MaxEnt, GP and BNN IRL algorithms for increasing numbers of demonstrations. EVDs are all evaluated on 100,000 trajectories each. The standard errors are plotted for 10 independent evaluations.

We run two versions of our experiments, where the expert agent has either a linear or an exponential reward function. Each IRL method is run for 512, 1024, 2048, 4096, 8192 and 16384 demonstrations. The results obtained are presented in Figure 5: as expected, all three IRL methods tested (MaxEnt IRL, GPIRL, BNN-IRL), learn fairly well linear reward functions. However, for an agent with an exponential reward, GPIRL and BNN-IRL are able to discover the latent function significantly better, with BNN outperforming as the number of demonstrations increases. In addition to

improved EVD, our BNN-IRL experiments provide a significant improvement in computational time as compared to GPIRL, hence enabling potentially more efficient scalability of IRL on LOBs to state spaces of higher dimensions.

4. Conclusions

In this paper we attempt an application of IRL to a stochastic, non-stationary financial environment (the LOB) whose competing agents may follow complex reward functions. While as expected all the methods considered are able to recover linear latent reward functions, only GP-based IRL (Levine et al., 2011) and our implementation through BNNs are able to recover more realistic non-linear expert rewards, thus mitigating most of the challenges imposed by this stochastic multi-agent environment. Moreover, our BNN method outperforms GPIRL for larger numbers of demonstrations, and is less computationally intensive. This may enable future work to study LOBs of higher dimensions, and on increasingly realistic number and complexity of agents involved.

References

- Balduzzi, D., Garnelo, M., Bachrach, Y., Czarnecki, W. M., Perolat, J., Jaderberg, M., and Graepel, T. Open-ended learning in symmetric zero-sum games. *arXiv preprint arXiv:1901.08106*, 2019.
- Berger, J. O. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 2013.
- Borel, E. La théorie du jeu et les équations intégrales à noyau symétrique. *Comptes rendus de l’Académie des Sciences*, 173(1304-1308):58, 1921.
- Cont, R., Stoikov, S., and Talreja, R. A stochastic model for order book dynamics. *Operations Research*, 58(3): 549–563, 2010.
- Cybenko, G. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.
- Fu, J., Luo, K., and Levine, S. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*, 2017.
- Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 1050–1059. PMLR, 20–22 Jun 2016.
- Gould, M. D., Porter, M. A., Williams, S., McDonald, M., Fenn, D. J., and Howison, S. D. Limit order books. *Quantitative Finance*, 13(11):1709–1742, 2013.

- Graves, A. Practical variational inference for neural networks. In *Advances in neural information processing systems*, pp. 2348–2356, 2011.
- Haarnoja, T., Tang, H., Abbeel, P., and Levine, S. Reinforcement learning with deep energy-based policies. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1352–1361. JMLR. org, 2017.
- Hendricks, D., Cobb, A., Everett, R., Downing, J., and Roberts, S. J. Inferring agent objectives at different scales of a complex adaptive system. *arXiv preprint arXiv:1712.01137*, 2017.
- Hinton, G. and Van Camp, D. Keeping neural networks simple by minimizing the description length of the weights. In *in Proc. of the 6th Ann. ACM Conf. on Computational Learning Theory*. Citeseer, 1993.
- Jin, M., Damianou, A. C., Abbeel, P., and Spanos, C. J. Inverse reinforcement learning via deep gaussian process. In *UAI*. AUAI Press, 2017.
- Lasry, J.-M. and Lions, P.-L. Mean field games. *Japanese journal of mathematics*, 2(1):229–260, 2007.
- Levine, S., Popovic, Z., and Koltun, V. Nonlinear inverse reinforcement learning with gaussian processes. In *Advances in Neural Information Processing Systems*, pp. 19–27, 2011.
- MacKay, D. J. A practical bayesian framework for backpropagation networks. *Neural computation*, 4(3):448–472, 1992.
- Navarro, R. M. and Larralde, H. A detailed heterogeneous agent model for a single asset financial market with trading via an order book. *PloS one*, 12(2):e0170766, 2017.
- Neal, R. M. *Bayesian learning for neural networks*, volume 118. Springer Science & Business Media, 1995.
- Ng, A. Y., Harada, D., and Russell, S. J. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning, ICML '99*, pp. 278–287, 1999. ISBN 1-55860-612-2.
- Ng, A. Y., Russell, S. J., et al. Algorithms for inverse reinforcement learning. In *ICML*, pp. 663–670, 2000.
- Peterson, C. A mean field theory learning algorithm for neural networks. *Complex systems*, 1:995–1019, 1987.
- Pratt, J. W. Risk aversion in the small and in the large. *Econometrica*, 32(1/2):122–136, 1964. ISSN 00129682, 14680262.
- Preis, T., Golke, S., Paul, W., and Schneider, J. J. Multi-agent-based order book model of financial markets. *Europhysics Letters (EPL)*, 75(3):510–516, aug 2006. doi: 10.1209/epl/i2006-10139-0.
- Roberson, B. The colonel blotto game. *Economic Theory*, 29(1):1–24, 2006.
- Russell, S. Learning agents for uncertain environments (extended abstract). In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pp. 101–103. ACM Press, 1998.
- Shepp, L. A. and Olkin, I. Entropy of the sum of independent bernoulli random variables and of the multinomial distribution. In *Contributions to probability*, pp. 201–206. Elsevier, 1981.
- Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.
- Tukey, J. W. A problem of strategy. *Econometrica*, 17(1): 73, 1949.
- Wang, X. and Wellman, M. P. Spoofing the limit order book: An agent-based model. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, pp. 651–659. International Foundation for Autonomous Agents and Multiagent Systems, 2017.
- Williams, L. R. and Jacobs, D. W. Stochastic completion fields: A neural model of illusory contour shape and salience. *Neural computation*, 9(4):837–858, 1997.
- Wulfmeier, M., Ondruska, P., and Posner, I. Maximum entropy deep inverse reinforcement learning. *arXiv preprint arXiv:1507.04888*, 2015.
- Yang, J., Ye, X., Trivedi, R., Xu, H., and Zha, H. Deep mean field games for learning optimal behavior policy of large populations. *CoRR*, abs/1711.03156, 2017.
- Zhang, Z., Zohren, S., and Roberts, S. Deeplob: Deep convolutional neural networks for limit order books. *IEEE Transactions on Signal Processing*, 2019.
- Ziebart, B. D. *Modeling Purposeful Adaptive Behavior with the Principle of Maximum Causal Entropy*. PhD thesis, Pittsburgh, PA, USA, 2010. AAI3438449.
- Ziebart, B. D., Maas, A. L., Bagnell, J. A., and Dey, A. K. Maximum entropy inverse reinforcement learning. In *AAAI*, volume 8, pp. 1433–1438. Chicago, IL, USA, 2008.

Disclaimer

Opinions and estimates constitute our judgement as of the date of this Material, are for informational purposes only and are subject to change without notice. This Material

is not the product of J.P. Morgan's Research Department and therefore, has not been prepared in accordance with legal requirements to promote the independence of research, including but not limited to, the prohibition on the dealing ahead of the dissemination of investment research. This Material is not intended as research, a recommendation, advice, offer or solicitation for the purchase or sale of any financial product or service, or to be used in any way for evaluating the merits of participating in any transaction. It is not a research report and is not intended as such. Past performance is not indicative of future results. Please consult your own advisors regarding legal, tax, accounting or any other aspects including suitability implications for your particular circumstances. J.P. Morgan disclaims any responsibility or liability whatsoever for the quality, accuracy or completeness of the information herein, and for any reliance on, or use of this material in any way.

Important disclosures at: www.jpmorgan.com/disclosures