Imitation Learning from Imperfect Demonstration

Yueh-Hua Wu [1 2] **Nontawat Charoenphakdee** [3 2] **Han Bao** [3 2] **Voot Tangkaratt** [2] **Masashi Sugiyama** [2 3]

Abstract

Imitation learning (IL) aims to learn an optimal policy from demonstrations. However, such demonstrations are often imperfect since collecting optimal ones is costly. To effectively learn from imperfect demonstrations, we propose a novel approach that utilizes *confidence scores*, which describe the quality of demonstrations. More specifically, we propose two confidence-based IL methods, namely two-step importance weighting IL (2IWIL) and generative adversarial IL with imperfect demonstration and confidence (IC-GAIL). We show that confidence scores given only to a small portion of sub-optimal demonstrations significantly improve the performance of IL both theoretically and empirically.

1. Introduction

Imitation learning (IL) has become of great interest because obtaining demonstrations is usually easier than designing reward. Reward is a signal to instruct agents to complete the desired tasks. However, ill-designed reward functions usually lead to unexpected behaviors (Everitt & Hutter, 2016; Dewey, 2014; Amodei et al., 2016). There are two main approaches that can be used to solve IL: behavioral cloning (BC) (Schaal, 1999), which adopts supervised learning approaches to learn an action predictor that is trained directly from demonstration data; and apprenticeship learning (AL), which attempts to find a policy that is better than the expert policy for a class of cost functions (Abbeel & Ng, 2004). Even though BC can be trained with supervised learning approaches directly, it has been shown that BC cannot imitate the expert policy without a large amount of demonstration data for not considering the transition of environments (Ross et al., 2011). In contrast, AL approaches learn from interacting with environments and optimize objectives such as maximum entropy (Ziebart et al., 2008).

A state-of-the-art approach *generative adversarial imitation learning* (GAIL) is proposed by Ho & Ermon (2016). The method learns an optimal policy by performing occupancy measure matching (Syed et al., 2008). An advantage of the matching method is that it is robust to demonstrations generated from a stochastic policy. Based on the concept proposed in GAIL, variants have been developed recently for different problem settings (Li et al., 2017; Kostrikov et al., 2019).

Despite that GAIL is able to learn an optimal policy from optimal demonstrations, to apply IL approaches to solve real-world tasks, the difficulty in obtaining such demonstration data should be taken into consideration. However, demonstrations from an optimal policy (either deterministic or stochastic) are usually assumed to be available in the above mentioned works, which can be barely fulfilled by the fact that most of the accessible demonstrations are imperfect or even from different policies. For instance, to train an agent to play basketball with game-play videos of the National Basketball Association, we should be aware that there are 14.3 turnovers per game[1], not to mention other kinds of mistakes that may not be recorded. The reason why optimal demonstrations are hard to obtain can be attributed to the limited attention and the presence of distractions, which make humans hard to follow optimal policies all the time. As a result, some parts of the demonstrations may be optimal and the others are not.

To mitigate the above problem, we propose to use confidence scores, which indicate the probability that a given trajectory is optimal. An intuitive example to collect the confidence scores is crowdsourcing, where data can be hard-labeled as being optimal or non-optimal by multiple labelers. We can extract confidence from the proportion of labelers giving a particular label, which corresponds to the probability that the instance has this label. Another example is when it is difficult to judge whether the demonstration is optimal. A labeler may rate a score digitized from $0, 0.1, \ldots, 1$ and we can use it as confidence. Since the attained confidence may not be perfect, to show the practicality of our methods, in our experiments, we use estimated confidence instead

[1]National Taiwan University, Taiwan [2]RIKEN Center for Advanced Intelligence Project, Japan [3]The University of Tokyo, Japan. Correspondence to: Yueh-Hua Wu <d06922005@csie.ntu.edu.tw>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

[1]https://www.basketball-reference.com/leagues/NBA_stats.html

of the ground truth value. In addition, experiments with different levels of noise are conducted to further justify their robustness to noisy labelers.

To further reduce the additional cost to learn an optimal policy, we consider a more realistic setting that the given demonstrations are partially equipped with confidence. As a result, the goal of this work is to utilize imperfect demonstrations where some are equipped with confidence while some are not (we refer to demonstrations without confidence as "unlabeled demonstrations").

In this work, we consider the setting where the given imperfect demonstrations are a mixture of optimal and non-optimal demonstrations. The setting is common when the demonstrations are collected via crowdsourcing (Serban et al., 2017; Hu et al., 2018; Shah et al., 2018) and learning from different sources such as videos (Tokmakov et al., 2017; Pathak et al., 2017; Supancic III & Ramanan, 2017; Yeung et al., 2017; Liu et al., 2018), where demonstrations can be generated from different policies.

We propose two methods, *two-step importance weighting imitation learning* (2IWIL) and *generative adversarial imitation learning with imperfect demonstration and confidence* (IC-GAIL), based on the idea of reweighting but from different perspectives. To utilize both confidence and unlabeled data, for 2IWIL, it predicts confidence scores for unlabeled data by optimizing the proposed objective based on empirical risk minimization (ERM) (Vapnik, 1998), which has flexibility for different loss functions, models, and optimizers; on the other hand, instead of directly reweighting to the optimal distribution and perform GAIL with reweighting, IC-GAIL reweights to the *non-optimal* distribution and match the optimal occupancy measure based on our mixture distribution setting. Since the derived objective of IC-GAIL depends on the proportion of the optimal demonstration in the demonstration mixture, we empirically show that IC-GAIL converges slower than 2IWIL but achieves better performance, which forms a trade-off between the two methods. We show that the proposed methods are both theoretically and practically sound.

2. Related work

In this section, we provide a brief survey about making use of non-optimal demonstrations and semi-supervised classification with confidence data.

2.1. Learning from non-optimal demonstrations

Learning from non-optimal demonstrations is nothing new in IL and reinforcement learning (RL) literature, but previous works utilized different information to learn a better policy. *Distance minimization inverse RL* (DM-IRL) (Burchfiel et al., 2016) utilized a feature function of states and assumed that the true reward function is linear in the features. The feedback from human is an estimate of accumulated reward, which is harder to be given than confidence because multiple reward functions may correspond to the same optimal policy.

Semi-supervised IRL (SSIRL) (Valko et al., 2012) extends the IRL method proposed by Abbeel & Ng (2004), where the reward function can be learned by matching the *feature expectations* of the optimal demonstrations. The difference from Abbeel & Ng (2004) is that in SSIRL, optimal and sub-optimal trajectories from other performers are given. Transductive SVM (Schölkopf et al., 1999) was used in place of vanilla SVM in Abbeel & Ng (2004) to recognize optimal trajectories in the sub-optimal ones. In our setting, the confidence scores are given instead of the optimal demonstrations. DM-IRL and SSIRL are not suitable for high-dimensional problems due to its dependence on the linearity of reward functions and good feature engineering.

2.2. Semi-supervised classification with confidence data

In our 2IWIL method, we train a probabilistic classifier with confidence and unlabeled data by optimizing the proposed ERM objective. There are similar settings such as *semi-supervised classification* (Chapelle et al., 2006), where few hard-labeled data $y \in \{0, 1\}$ and some unlabeled data are given.

Zhou et al. (2014) proposed to use hard-labeled instances to estimate confidence scores for unlabeled samples using Gaussian mixture models and principal component analysis. Similarly, for an input instance x, Wang et al. (2013) obtained an upper bound of confidence $\Pr(y = 1|x)$ with hard-labeled instances and a kernel density estimator, then treated the upper bound as an estimate of probabilistic class labels.

Another related scheme was considered in El-Zahhar & El-Gayar (2010) where they considered soft labels $z \in [0, 1]$ as fuzzy inputs and proposed a classification approach based on k-nearest neighbors. This method is difficult to scale to high-dimensional tasks, and lacks theoretical guarantees. Ishida et al. (2018) proposed another scheme that trains a classifier only from positive data equipped with confidence. Our proposed method, 2IWIL, also considers training a classifier with confidence scores of given demonstrations. Nevertheless, 2IWIL can train a classifier from fewer confidence data, with the aid of a large number of unlabeled data.

3. Background

In this section, we provide backgrounds of RL and GAIL.

3.1. Reinforcement Learning

We consider the standard Markov Decision Process (MDP) (Sutton & Barto, 1998). MDP is represented by a tuple $\langle S, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$, where S is the state space, $\mathcal{A}$ is the action space, $\mathcal{P}(s_{t+1}|s_t, a_t)$ is the transition density of state s_{t+1} at time step $t+1$ given action a_t made under state s_t at time step t, $\mathcal{R}(s, a)$ is the reward function, and $\gamma \in (0, 1)$ is the discount factor.

A stochastic policy $\pi(a|s)$ is a density of action a given state s. The performance of π is evaluated in the γ-discounted infinite horizon setting and its expectation can be represented with respect to the trajectories generated by π:

$$\mathbb{E}_\pi[\mathcal{R}(s, a)] = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}(s_t, a_t)\right], \tag{1}$$

where the expectation on the right-hand side is taken over the densities $p_0(s_0)$, $\mathcal{P}(s_{t+1}|s_t, a_t)$, and $\pi(a_t|s_t)$ for all time steps t. Reinforcement learning algorithms (Sutton & Barto, 1998) aim to maximize Eq. (1) with respect to π.

To characterize the distribution of state-action pairs generated by an arbitrary policy π, the occupancy measure is defined as follows.

Definition 3.1 (Puterman (1994))**.** *Define occupancy measure $\rho_\pi : S \times \mathcal{A} \to \mathbb{R}$,*

$$\rho_\pi(s, a) = \pi(a|s) \sum_{t=0}^{\infty} \gamma^t \Pr(s_t = s|\pi), \tag{2}$$

where $\Pr(s_t = s|\pi)$ is the probability density of state s at time step t following policy π.

The occupancy measure of π, $\rho_\pi(s, a)$, can be interpreted as an unnormalized density of state-action pairs.

The occupancy measure plays an important role in IL literature because of the following one-to-one correspondence with the policy.

Theorem 3.2. *(Theorem 2 of Syed et al. (2008)) Suppose ρ is the occupancy measure for $\pi_\rho(a|s) \triangleq \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')}$. Then π_ρ is the only policy whose occupancy measure is ρ.*

In this work, we also define the *normalized occupancy measure* $p(s, a)$,

$$\begin{aligned} p(s, a) \triangleq & \frac{\rho(s, a)}{\sum_{s,a} \rho(s, a)} \\ = & \frac{\rho(s, a)}{\sum_{s,a} \pi(a|s) \sum_{t=0}^{\infty} \gamma^t \Pr(s_t = s|\pi)} \\ = & \frac{\rho(s, a)}{\sum_{t=0}^{\infty} \gamma^t} = (1 - \gamma)\rho(s, a). \end{aligned}$$

The normalized occupancy measure can be interpreted as a probability density of state-action pairs that an agent experiences in the environment with policy π.

3.2. Generative adversarial imitation learning (GAIL)

The problem setting of IL is that given trajectories $\{(s_i, a_i)\}_{i=1}^n$ generated by an expert π_E, we are interested in optimizing the agent policy π_θ to recover the expert policy π_E with $\{(s_i, a_i)\}_{i=1}^n$ and the MDP tuple without reward function $\mathcal{R}$.

GAIL (Ho & Ermon, 2016) is a state-of-the-art IL method that performs occupancy measure matching to learn a parameterized policy. Occupancy measure matching aims to minimize the objective $d(\rho_{\pi_\mathrm{E}}, \rho_{\pi_\theta})$, where d is a distance function. The key idea behind GAIL is that it uses generative adversarial training to estimate the distance and minimize it alternatively. To be precise, the distance is the Jensen-Shannon divergence (JSD), which is estimated by solving a binary classification problem. This leads to the following min-max optimization problem:

$$\min_\theta \max_w \mathop{\mathbb{E}}_{s,a \sim p_\theta} [\log D_w(s, a)] + \mathop{\mathbb{E}}_{s,a \sim p_\mathrm{opt}} [\log(1 - D_w(s, a))], \tag{3}$$

where p_θ and p_opt are the corresponding normalized occupancy measures for π_θ and π_opt respectively. D_w is called a discriminator and it can be shown that if the discriminator has infinite capacity, the global optimum of Eq. (3) corresponds to the JSD up to a constant (Goodfellow et al., 2014). To update the agent policy π_θ, GAIL treats the loss $-\log(D_w(s, a))$ as a reward signal and the agent can be updated with RL methods such as trust region policy optimization (TRPO) (Schulman et al., 2015). A weakness of GAIL is that if the given demonstrations are non-optimal then the learned policy will be non-optimal as well.

4. Imitation learning with confidence and unlabeled data

In this section, we present two approaches to learning from imperfect demonstrations with confidence and unlabeled data. The first approach is *2IWIL*, which aims to learn a probabilistic classifier to predict confidence scores of unlabeled demonstration data and then performs standard GAIL with reweighted distribution. The second approach is *ICGAIL*, which forgoes learning a classifier and learns an optimal policy by performing occupancy measure matching with unlabeled demonstration data. Details of derivation and proofs in this section can be found in Appendix.

4.1. Problem setting

Firstly, we formalize the problem setting considered in this paper. For conciseness, in what follows we use x in place of (s, a). Consider the case where given imperfect demonstrations are sampled from an optimal policy π_opt and non-optimal policies $\Pi = \{\pi_i\}_{i=1}^n$. Denote that the corresponding normalized occupancy measure of π_opt and Π are p_opt

and $\{p_i\}_{i=1}^n$, respectively. The normalized occupancy measure $p(x)$ of a state-action pair x is therefore the weighted sum of $p_{\rm opt}$ and $\{p_i\}_{i=1}^n$,

$$\begin{aligned} p(x) =& \alpha p_{\rm opt}(x) + \sum_{i=1}^{n} \nu_i p_i(x) \\ =& \alpha p_{\rm opt}(x) + (1-\alpha) p_{\rm non}(x), \end{aligned}$$

where $\alpha + \sum_{i=1}^n \nu_i = 1$ and $p_{\rm non}(x) = \frac{1}{(1-\alpha)} \sum_{i=1}^n \nu_i p_i(x)$. We may further follow traditional classification notation by defining $p_{\rm opt}(x) \triangleq p(x|y=+1)$ and $p_{\rm non}(x) \triangleq p(x|y=-1)$, where $y=+1$ indicates that x is drawn from the occupancy measure of the optimal policy and $y=-1$ indicates the non-optimal policies. Here, $\alpha = \Pr(y=+1)$ is the class-prior probability of the optimal policy. We further assume that an oracle labels state-action pairs in the demonstration data with *confidence scores* $r(x) \triangleq p(y=+1|x)$. Based on this, the normalized occupancy measure of the optimal policy can be expressed by the Bayes' rule as

$$p(x|y=+1) = \frac{r(x)p(x)}{\alpha}. \tag{4}$$

We assume that labeling state-action pairs by the oracle can be costly and only some pairs are labeled with confidence. More precisely, we obtain demonstration datasets as follows,

$$\begin{aligned} \mathcal{D}_{\rm c} &\triangleq \{(x_{{\rm c},i}, r_i)\}_{i=1}^{n_{\rm c}} \overset{\text{i.i.d.}}{\sim} q(x,r), \\ \mathcal{D}_{\rm u} &\triangleq \{x_{{\rm u},i}\}_{i=1}^{n_{\rm u}} \overset{\text{i.i.d.}}{\sim} p(x), \end{aligned}$$

where $q(x,r) = p(x)p_{\rm r}(r|x)$ and $p_{\rm r}(r_i|x) = \delta(r_i - r(x))$ is a delta distribution. Our goal is to consider the case where $\mathcal{D}_c$ is scarce and we want to learn the optimal policy $\pi_{\rm opt}$ with $\mathcal{D}_c$ and $\mathcal{D}_u$ jointly.

4.2. Two-step importance weighting imitation learning

We first propose an approach based on the importance sampling scheme. By Eq. (4), the GAIL objective in Eq. (3) can be rewritten as follows:

$$\begin{aligned} \min_{\theta} \max_{w} & \mathbb{E}_{x\sim p_\theta} [\log D_w(x)] \\ & + \mathbb{E}_{x,r\sim q} \left[\frac{r}{\alpha} \log(1 - D_w(x)) \right]. \end{aligned} \tag{5}$$

In practice, we may use the mean of confidence scores to estimate the class prior α. Although we can reweight the confidence data $\mathcal{D}_{\rm c}$ to match the optimal distribution, we have a limited number of confidence data and it is difficult to perform accurate sample estimation. To make full use of unlabeled data, the key idea is to identify confidence scores of the given unlabeled data $\mathcal{D}_{\rm u}$ and reweight both confidence data and unlabeled data. To achieve this, we train a probabilistic classifier from confidence data and unlabeled data, where we call this learning problem *semi-conf (SC) classification*.

Let us first consider a standard binary classification problem to classify samples into $p_{\rm opt}$ ($y=+1$) and $p_{\rm non}$ ($y=-1$). Let $g\colon \mathbb{R}^d \to \mathbb{R}$ be a prediction function and $\ell\colon \mathbb{R} \to \mathbb{R}_+$ be a loss function. The optimal classifier can be learned by minimizing the following risk:

$$\begin{aligned} R_{{\rm PN},\ell}(g) = & \alpha \mathbb{E}_{x\sim p_{\rm opt}} [\ell(g(x))] \\ & + (1-\alpha) \mathbb{E}_{x\sim p_{\rm non}} [\ell(-g(x))], \end{aligned} \tag{6}$$

where PN stands for "positive-negative". However, as we only have samples from the mixture distribution p instead of samples separately drawn from $p_{\rm opt}$ and $p_{\rm non}$, it is not straightforward to conduct sample estimation of the risk in Eq. (6). To overcome this issue, we express the risk in an alternative way that can be estimated only from $\mathcal{D}_{\rm c}$ and $\mathcal{D}_{\rm u}$ in the following theorem.

Theorem 4.1. *The classification risk* (6) *can be equivalently expressed as*

$$\begin{aligned} R_{{\rm SC},\ell}(g) = & \mathbb{E}_{x,r\sim q}[r(\ell(g(x)) - \ell(-g(x))) \\ & + (1-\beta)\ell(-g(x))] + \mathbb{E}_{x\sim p}[\beta \ell(-g(x))], \end{aligned} \tag{7}$$

where $\beta \in [0,1]$ is an arbitrary weight.

Thus, we can obtain a probabilistic classifier by minimizing Eq. (7), which can be estimated only with $\mathcal{D}_{\rm c}$ and $\mathcal{D}_{\rm u}$. Once we obtain the prediction function g, we can use it to give confidence scores for $\mathcal{D}_{\rm u}$.

To make the prediction function g estimate confidence accurately, the loss function ℓ in Eq. (7) should come from a class of *strictly proper composite loss* (Buja et al., 2005; Reid & Williamson, 2010). Many losses such as the squared loss, logistic loss, and exponential loss are proper composite. For example, if we obtain $g^*_{\log}$ that minimizes a logistic loss $\ell_{\log}(z) = (\log(1+\exp(-z))$, we can obtain confidence scores by passing prediction outputs to a sigmoid function $\widehat{p}(y=1|x) = [1+\exp(-g^*_{\log}(x))]^{-1}$ (Reid & Williamson, 2010). On the other hand, the hinge loss cannot be applied since it is not a proper composite loss and cannot estimate confidence reliably (Bartlett & Tewari, 2007; Reid & Williamson, 2010). Therefore, we can obtain a probabilistic classifier from the prediction function g that learned from a strictly proper composite loss. After obtaining a probabilistic classifier, we optimize the importance weighted objective in Eq. (5), where both $\mathcal{D}_{\rm c}$ and $\mathcal{D}_{\rm u}$ are used to estimate the second expectation. We summarize this training procedure in Algorithm 1.

Next, we discuss the choice of the combination coefficient β. Since we have access to the empirical unbiased estimator $\widehat{R}_{{\rm SC},\ell}(g)$ from Eq. (7), it is natural to find the minimum

Algorithm 1 2IWIL

Input: Expert trajectories and confidence $\mathcal{D}_{\mathrm{c}} = \{(x_{\mathrm{c},i}, r_i)\}_{i=1}^{n_{\mathrm{c}}}$, $\mathcal{D}_{\mathrm{u}} = \{x_{\mathrm{u},i}\}_{i=1}^{n_{\mathrm{u}}}$
Estimate the class prior by $\widehat{\alpha} = \frac{1}{n_{\mathrm{c}}} \sum_{i=1}^{n_{\mathrm{c}}} r_i$
Train a probabilistic classifier by minimizing Eq. (7) with $\beta = \frac{n_u}{n_u + n_c}$
Predict confidence scores $\{\widehat{r}_{\mathrm{u},i}\}_{i=1}^{n_{\mathrm{u}}}$ for $\{x_{\mathrm{u},i}\}_{i=1}^{n_{\mathrm{u}}}$
for $i = 0, 1, 2, \ldots$ **do**
  Sample trajectories $\{x_i\}_{i=1}^{n_a} \sim \pi_\theta$
  Update the discriminator parameters by maximizing Eq. (5)
  Update π_θ with reward $-\log D_w(x)$ using TRPO
end for

variance estimator among them. The following theorem gives the optimal β in terms of the estimator variance.

Proposition 4.2 (variance minimality)**.** *Let σ_{cov} denote the covariance between $n_{\mathrm{c}}^{-1} \sum_{i=1}^{n_{\mathrm{c}}} r_i \{\ell(g(x_{\mathrm{c},i})) - \ell(-g(x_{\mathrm{c},i}))\}$ and $n_{\mathrm{c}}^{-1} \sum_{i=1}^{n_{\mathrm{c}}} \ell(-g(x_{\mathrm{c},i}))$. For a fixed g, the estimator $\widehat{R}_{\mathrm{SC},\ell}(g)$ has the minimum variance when $\beta = \mathrm{clip}_{[0,1]}(\frac{n_{\mathrm{u}}}{n_{\mathrm{c}}+n_{\mathrm{u}}} + \frac{\sigma_{\mathrm{cov}}}{\mathrm{Var}(\ell(-g(x)))} \frac{n_{\mathrm{c}} n_{\mathrm{u}}}{n_{\mathrm{c}}+n_{\mathrm{u}}})$.*[2]

Thus, β lies in $(0, 1)$ when the covariance σ_{cov} is not so large. If $\beta \neq 0$, it means that the unlabeled data $\mathcal{D}_{\mathrm{u}}$ does help the classifier by reducing empirical variance when Eq. (7) is adopted. However, computing the β that minimizes empirical variance is computationally inefficient since it involves computing σ_{cov} and $\mathrm{Var}(l(-g(x)))$. In practice, we use $\beta = \frac{n_{\mathrm{u}}}{n_{\mathrm{c}}+n_{\mathrm{u}}}$ for all experiments by assuming that the covariance is small enough.

In our preliminary experiments, we sometimes observed that the empirical estimate $\widehat{R}_{\mathrm{SC},\ell}$ of Eq. (7) became negative and led to overfitting. We can mitigate this phenomenon by employing a simple yet highly effective technique from Kiryo et al. (2017), which is proposed to solve a similar overfitting problem (see Appendix for implementation details).

4.2.1. Theoretical Analysis

Below, we show that the estimation error of Eq. (7) can be bounded. This means that its minimizer is asymptotically equivalent to the minimizer of the standard classification risk $R_{\mathrm{PN},\ell}$, which provides a consistent estimator of $p(y = +1|x)$. We provide the estimation error bound with Rademacher complexity (Bartlett & Mendelson, 2002). Denote $\mathfrak{R}_n(\mathcal{G})$ be the Rademacher complexity of the function class $\mathcal{G}$ with the sample size n.

Theorem 4.3. *Let $\mathcal{G}$ be the hypothesis class we use. Assume that the loss function ℓ is ρ_ℓ-Lipschitz continuous, and that there exists a constant $C_\ell > 0$ such that $\sup_{x \in \mathcal{X}, y \in \{\pm 1\}} |\ell(yg(x))| \leq C_\ell$ for any $g \in \mathcal{G}$. Let $\widehat{g} \triangleq \arg\min_{g \in \mathcal{G}} \widehat{R}_{\mathrm{SC},\ell}(g)$ and $g^* \triangleq \arg\min_{g \in \mathcal{G}} R_{\mathrm{SC},\ell}(g)$. For $\delta \in (0, 1)$, with probability at least $1 - \delta$ over repeated sampling of data for training $\widehat{g}$,*

$$R_{\mathrm{SC},\ell}(\widehat{g}) - R_{\mathrm{SC},\ell}(g^*) \leq 16\rho_\ell((3-\beta)\mathfrak{R}_{n_{\mathrm{c}}}(\mathcal{G}) + \beta\mathfrak{R}_{n_{\mathrm{u}}}(\mathcal{G})) + 4C_\ell \sqrt{\frac{\log(8/\delta)}{2}} \left((3-\beta) n_{\mathrm{c}}^{-\frac{1}{2}} + \beta n_{\mathrm{u}}^{-\frac{1}{2}}\right).$$

Thus, we may safely obtain a probabilistic classifier by minimizing $\widehat{R}_{\mathrm{SC},\ell}$, which gives a consistent estimator.

4.3. IC-GAIL

Since 2IWIL is a two-step approach by first gathering more confidence data and then conducting importance sampling, the error may accumulate over two steps and degrade the performance. Therefore, we propose IC-GAIL that can be trained in an end-to-end fashion and perform occupancy measure matching with the optimal normalized occupancy measure p_{opt} directly.

Recall that $p = \alpha p_{\mathrm{opt}} + (1-\alpha) p_{\mathrm{non}}$. Our key idea here is to minimize the divergence between p and p', where $p' = \alpha p_\theta + (1-\alpha) p_{\mathrm{non}}$. Intuitively, the divergence between p_θ and p_{opt} is minimized if that between p and p' is minimized. For Jensen-Shannon divergence, this intuition can be justified in the following theorem.

Theorem 4.4. *Denote that*

$$V(\pi_\theta, D_w) = \mathbb{E}_{x \sim p}[\log(1 - D_w(x))] + \mathbb{E}_{x \sim p'}[\log D_w(x)],$$

and that $C(\pi_\theta) = \max_w V(\pi_\theta, D_w)$. Then, $V(\pi_\theta, D_w)$ is maximized when $D_w = \frac{p'}{p+p'} (\triangleq D_{w^*})$, and its maximum value is $C(\pi_\theta) = -\log 4 + 2\mathrm{JSD}(p \| p')$. Thus, $C(\pi_\theta)$ is minimized if and only if $p_\theta = p_{\mathrm{opt}}$ almost everywhere.

Theorem 4.4 implies that the optimal policy can be found by solving the following objective,

$$\min_\theta \max_w \mathbb{E}_{x \sim p}[\log(1 - D_w(x))] + \mathbb{E}_{x \sim p'}[\log D_w(x)]. \tag{8}$$

The expectation in the first term can be approximated from $\mathcal{D}_{\mathrm{u}}$, while the expectation in the second term is the weighted sum of the expectation over p_θ and p_{non}. Data $\mathcal{D}_{\mathrm{a}} = \{x_{\mathrm{a},i}\}_i^{n_{\mathrm{a}}}$ sampled from p_θ can be obtained by executing the current policy π_θ. However, we cannot directly obtain samples from p_{non} since it is unknown. To overcome this issue, we establish the following theorem.

Theorem 4.5. *$V(\pi_\theta, D_w)$ can be transformed to $\widetilde{V}(\pi_\theta, D_w)$, which is defined as follows:*

$$\widetilde{V}(\pi_\theta, D_w) = \mathbb{E}_{x \sim p}[\log(1 - D_w(x))] + \alpha \mathbb{E}_{x \sim p_\theta}[\log D_w(x)] + \mathbb{E}_{x,r \sim q}[(1-r) \log D_w(x)]. \tag{9}$$

[2] $\mathrm{clip}_{[l,u]}(v) \triangleq \max\{l, \min\{v, u\}\}$.

We can approximate Eq. (9) given finite samples $\mathcal{D}_{\mathrm{c}}$, $\mathcal{D}_{\mathrm{u}}$, and $\mathcal{D}_{\mathrm{a}}$. In practice, we perform alternative gradient descent with respect to θ and w to solve this optimization problem. Below, we show that the estimation error of $\widetilde{V}$ can be bounded for a fixed agent policy π_θ.

4.3.1. Theoretical analysis

In this subsection, we show that the estimation error of Eq. (9) can be bounded, given a fixed agent policy π_θ. Let $\widehat{V}(\pi_\theta, D_w)$ be the empirical estimate of Eq. (9).

Theorem 4.6. *Let $\mathcal{W}$ be a parameter space for training the discriminator and $D_{\mathcal{W}} \triangleq \{D_w \mid w \in \mathcal{W}\}$ be its hypothesis space. Assume that there exist a constant $C_L > 0$ such that $|\log D_w(x)| \leq C_L$ and $|\log(1 - D_w(x))| \leq C_L$ for any $x \in \mathcal{X}$ and $w \in \mathcal{W}$. Assume that both $\log D_w(x)$ and $\log(1 - D_w(x))$ for any $w \in \mathcal{W}$ have Lipschitz norms no more than $\rho_L > 0$. For a fixed agent policy π_θ, let $\{x_{\mathrm{a},i}\}_{i=1}^{n_a}$ be a sample generated from π_θ, $D_{\widehat{w}} \triangleq \arg\max_{w \in \mathcal{W}} \widehat{V}(\pi_\theta, D_w)$, and $D_{w^*} \triangleq \arg\max_{w \in \mathcal{W}} V(\pi_\theta, D_w)$. Then, for $\delta \in (0, 1)$, the following holds with probability at least $1 - \delta$:*

$$
\begin{aligned}
&V(\pi_\theta, D_{w^*}) - V(\pi_\theta, D_{\widehat{w}}) \\
&\quad \leq 16\rho_L(\mathfrak{R}_{n_{\mathrm{u}}}(D_{\mathcal{W}}) + \alpha\mathfrak{R}_{n_{\mathrm{a}}}(D_{\mathcal{W}}) + \mathfrak{R}_{n_{\mathrm{c}}}(D_{\mathcal{W}})) \\
&\qquad + 4C_L\sqrt{\frac{\log(6/\delta)}{2}}\left(n_{\mathrm{u}}^{-\frac{1}{2}} + \alpha n_{\mathrm{a}}^{-\frac{1}{2}} + n_{\mathrm{c}}^{-\frac{1}{2}}\right).
\end{aligned}
$$

Theorem 4.6 guarantees that the estimation of Eq. (9) provides a consistent maximizer with respect to the original objective in Eq. (8) at each step of the discriminator training.

4.3.2. Practical implementation of IC-GAIL

Even though Eq. (9) is theoretically supported, when the class prior α is low, the influence of the agent become marginal in the discriminator training. This issue can be mitigated by thresholding α in Eq. (9) as follows:

$$
\begin{aligned}
\min_\theta \max_w &\mathbb{E}_{x\sim p}[\log(1 - D_w(x))] + \lambda\mathbb{E}_{x\sim p_\theta}[\log D_w(x)] \\
&+ (1 - \lambda)\mathbb{E}_{x,r\sim q}\left[\frac{(1 - r)}{(1 - \alpha)}\log D_w(x)\right], \quad (10)
\end{aligned}
$$

where $\lambda = \max\{\tau, \alpha\}$ and $\tau \in (0, 1]$. The training procedure of IC-GAIL is summarized in Algorithm 2. Note that Eq. (10) returns to Eq. (3) and learns an sub-optimal policy when $\tau = 1$.

4.4. Discussion

To understand the difference between 2IWIL and IC-GAIL, we discuss it from three different perspectives: unlabeled data, confidence data, and the class prior.

Role of unlabeled data: It should be noted that unlabeled data plays different roles in the two methods. In 2IWIL, we show that unlabeled data reduces the variance of the empirical risk estimator as shown in Proposition 4.2.

Algorithm 2 IC-GAIL

Input: Expert trajectories, confidence, and weight threshold $\{x_{\mathrm{u},i}\}_{i=1}^{n_{\mathrm{u}}}$, $\{(x_{\mathrm{c},i}, r_i)\}_{i=1}^{n_{\mathrm{c}}}$, τ
Estimate the class prior by $\widehat{\alpha} = \frac{1}{n_{\mathrm{c}}}\sum_{i=1}^{n_{\mathrm{c}}} r_i$
$\lambda = \max\{\tau, \widehat{\alpha}\}$
for $i = 0, 1, 2, \ldots$ **do**
  Sample trajectories $\{x_i\}_{i=1}^{n_a} \sim \pi_\theta$
  Update the discriminator parameters by maximizing Eq. (10)
  Update π_θ with reward $-\log D_w(x)$ using TRPO
end for

On the other hand, in addition to making more accurate estimation, the usefulness of unlabeled data in IC-GAIL is similar to guided exploration (Kang et al., 2018). We may analogize confidence information in the imperfect demonstration setting to reward functions since both of them allow agents to learn an optimal policy in IL and RL, respectively. Likewise, fewer confidence data can be analogous to sparse reward functions. Even though a small number of confidence data and sparse reward functions do not make objective such as Eqs. (5) and (1) biased, they cause practical issues such as a deficiency in information for exploration. To mitigate the problem, we imitate from sub-optimal demonstrations and use confidence information to refine the learned policy, which is similar to Kang et al. (2018) in the sense that they imitate a sub-optimal policy to guide RL algorithms in the sparse reward setting.

Role of confidence data: Confidence data is utilized to train a classifier and to reweight p_{opt} in 2IWIL, which causes the two-step training scheme and therefore the error is accumulated in the prediction phase and the occupancy measure matching phase. Differently, IC-GAIL instead compensates the p_{non} portion in the given imperfect demonstrations by mimicking the composition of p. The advantage of IC-GAIL over 2IWIL is that it avoids the prediction error by employing an end-to-end training scheme.

Influence of the class-prior α: The class prior in 2IWIL as shown in Eq. (5) serves as a normalizing constant so that the weight $\frac{r(x)}{\alpha}$ for reweighting p to p_{opt} has unit mean. Consequently, the class prior α does not affect the convergence of the agent policy. On the other hand, the term with respect to the agent p_θ is directly scaled by α in Eq. (9) of IC-GAIL. To comprehend the influence, we may expand the reward function from the discriminator $-\log D^*_w(x) = -\log\left(\left(\frac{\alpha}{(1-\alpha)}p_\theta + p_{\mathrm{non}}\right) / \left(\frac{\alpha}{(1-\alpha)}(p_{\mathrm{opt}} + p_\theta) + 2p_{\mathrm{non}}\right)\right)$ and it shows that the agent term is scaled by $\frac{\alpha}{(1-\alpha)}$, which makes the reward function prone to be a constant when α is small. Therefore the agent learns slower than in 2IWIL,

Table 1. Comparison between proposed methods (IC-GAIL and 2IWIL) and baselines.

METHOD	INPUT	OBJECTIVE
IC-GAIL	$\mathcal{D}_{\mathrm{u}} \cup \mathcal{D}_{\mathrm{c}}$	EQ. (9)
2IWIL	$\mathcal{D}_{\mathrm{u}} \cup \mathcal{D}_{\mathrm{c}}$	EQ. (7)
GAIL (U+C)	$\mathcal{D}_{\mathrm{u}} \cup \mathcal{D}_{\mathrm{c}}^{x}$	EQ. (3)
GAIL (C)	$\mathcal{D}_{\mathrm{c}}^{x}$	EQ. (3)
GAIL (REWEIGHT)	$\mathcal{D}_{\mathrm{c}}$	EQ. (5)

where the reward function is $-\log\left(p_\theta/(p_\theta + p_{\mathrm{opt}})\right)$.

5. Experiments

In this section, we aim to answer the following questions with experiments. (1) *Do 2IWIL and IC-GAIL methods allow agents to learn near-optimal policies when limited confidence information is given?* (2) *Are the methods robust enough when the given confidence is less accurate?* and (3) *Do more unlabeled data results in better performance in terms of average return?* The discussions are given in Sec. 5.1, 5.2, and 5.3 respectively.

Setup To collect demonstration data, we train an optimal policy (π_{opt}) using TRPO (Schulman et al., 2015) and select two intermediate policies (π_1 and π_2). The three policies are used to generate the same number of state-action pairs. In real-world tasks, the confidence should be given by human labelers. We simulate such labelers by using a probabilistic classifier $p^\star(y=+1|x)$ pre-trained with demonstration data and randomly choose 20% of demonstration data to label confidence scores $r(x) = p^\star(y=+1|x)$.

We compare the proposed methods against three baselines. Denote that $\mathcal{D}_{\mathrm{c}}^{x} \triangleq \{x_{\mathrm{c},i}\}_{i=1}^{n_{\mathrm{c}}}$, $\mathcal{D}_{\mathrm{c}}^{r} \triangleq \{r_i\}_{i=1}^{n_{\mathrm{c}}}$, and $\mathcal{D}_{\mathrm{u}}^{x} \triangleq \mathcal{D}_{\mathrm{u}}$. GAIL (U+C) takes all the pairs as input without considering confidence. To show if reweighting using Eq. (5) makes difference, GAIL (C) and GAIL (Reweight) use the same state-action pairs $\mathcal{D}_{\mathrm{c}}^{x}$ but GAIL (Reweight) additionally utilizes reweighting with confidence information $\mathcal{D}_{\mathrm{c}}^{r}$. The baselines and the proposed methods are summarized in Table 1.

To assess our methods, we conduct experiments on Mujoco (Todorov et al., 2012). Each experiment is performed with five random seeds. The hyper-parameter τ of IC-GAIL is set to 0.7 for all tasks. To show the performance with respect to the optimal policy that we try to imitate, the accumulative reward is normalized with that of the optimal policy and a uniform random policy so that 1.0 indicates the optimal policy and 0.0 the random one. Due to space limit, we defer implementation details, the performance of the optimal and the random policies, the specification of each task, and the uncropped figures of Ant-v2 to Appendix.

5.1. Performance comparison

The average return against training epochs in Fig. 1 shows that the proposed IC-GAIL and 2IWIL outperform other baselines by a large margin. Due to the mentioned experiment setup, the class prior of the optimal demonstration distribution is around 33%. To interpret the experiment results, we would like to emphasize that our experiments are under incomplete optimality setting such that confidence itself is not enough to learn the optimal policy as indicated by the GAIL (Reweight) baseline. Since the difficulty of each task varies, we use different number of $n_c + n_u$ for different tasks. Our contribution is that in addition to the confidence, our methods are able to utilize the demonstration mixture (sub-optimal demonstration) and learn near-optimal policies.

We can observe that IC-GAIL converges slower than 2IWIL. As discussed in Section 4.4, it can be attributed to that the term with respect to the agent in Eq. (10) is scaled by 0.7 as specified by τ, which decreases the influence of the agent policy in updating discriminator. The faster convergence of 2IWIL can be an advantage over IC-GAIL when interactions with environments are expensive. Even though the objective of IC-GAIL becomes biased by not using the class prior α, it still converges to near-optimal policies in four tasks.

In Walker2d-v2, the improvement in performance of our methods is not as significant as in other tasks. We conjecture that it is caused by the insufficiency of confidence information. This can be verified by observing that the GAIL (Reweight) baseline in Walker2d-v2 gradually converges to 0.2 whereas in other tasks it achieves the performance of at least 0.4. In HalfCheetah-v2, we observe that the discriminator is stuck in a local maximum in the middle of learning, which influences all methods significantly.

The baseline GAIL (Reweight) surpasses GAIL (C) in all tasks, which shows that reweighting enables the agent to learn policies that obtain higher average return. However, since the number of confidence instances is small, the information is not enough to derive the optimal policies. GAIL (U+C) is the standard GAIL without considering confidence information. Although the baseline uses the same number of demonstrations $n_c + n_u$ as our proposed methods, the performance difference is significant due to the use of confidence.

5.2. Robustness to Gaussian noise in confidence

In practice, the oracle that gives confidence scores is basically human labelers and they may not be able to accurately label confidence all the time. To investigate robustness of our approaches against noise in the confidence scores, we further conduct an experiment on Ant-v2 where the Gaussian noise is added to confidence scores as follows:

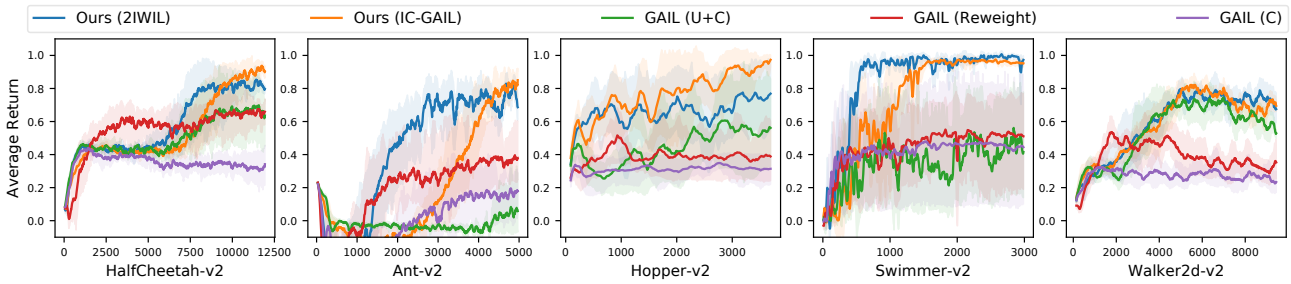

Figure 1. Learning curves of our 2IWIL and IC-GAIL versus baselines given imperfect demonstrations. The x-axis is the number of training epochs and the shaded area indicates standard error.

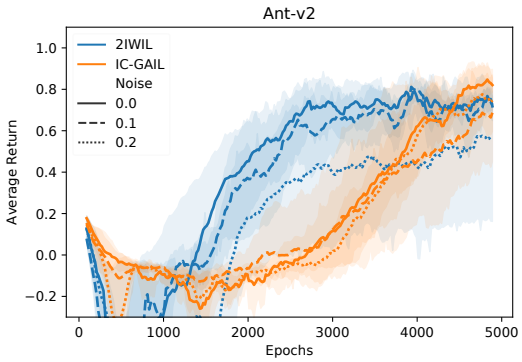

Figure 2. Learning curves of proposed methods with different standard deviations of Gaussian noise added to confidence. The numbers in the legend indicate the standard deviation of the Gaussian noise.

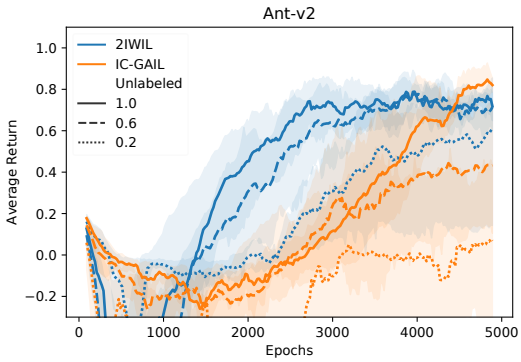

Figure 3. Learning curves of the proposed methods with different number of unlabeled data. The numbers in the legend suggest the proportion of unlabeled data used as demonstrations. 1.0 is the same as the data used in Fig. 1.

$r(x) = p^\star(y = 1|x) + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Fig. 2 shows the performance of our methods in this noisy confidence scenario. It reveals that both methods are quite robust to noisy confidence, which suggests that the proposed methods are robust enough to human labelers, who may not always correctly assign confidence scores.

5.3. Influence of unlabeled data

In this experiment, we would like to evaluate the performance of both 2IWIL and IC-GAIL with different numbers of unlabeled data to verify whether unlabeled data is useful. As we can see in Fig. 3, the performance of both methods grows as the number of unlabeled data increases, which confirms our motivation that using unlabeled data can improve the performance of imitation learning when confidence data is scarce. As discussed in Sec. 4.4, the different roles of unlabeled data in the two proposed methods result in dissimilar learning curves with respect to unlabeled data.

6. Conclusion

In this work, we proposed two general approaches IC-GAIL and 2IWIL, which allow the agent to utilize both confidence and unlabeled data in imitation learning. The setting considered in this paper is usually the case in real-world scenarios because collecting optimal demonstrations is normally costly. In 2IWIL, we utilized unlabeled data to derive a risk estimator and obtained the minimum variance with respect to the combination coefficient β. 2IWIL predicts confidence scores for unlabeled data and matches the optimal occupancy measure based on the GAIL objective with importance sampling. For IC-GAIL, we showed that the agent learns an optimal policy by matching a mixture of normalized occupancy measures p' with the normalized occupancy measure of the given demonstrations p.

Practically, we conducted extensive experiments to show that our methods outperform baselines by a large margin, to confirm that our methods are robust to noise, and to verify that unlabeled data has a positive correlation with the performance. The proposed approaches are general and can be easily extended to other IL and IRL methods (Li et al., 2017; Fu et al., 2018; Kostrikov et al., 2019).

For future work, we may extend it to a variety of applications such as discrete sequence generation because the confidence in our work can be treated as a property indicator. For instance, to generate soluble chemicals, we may not have enough soluble chemicals, whereas the Crippen function (Crippen & Snow, 1990) can be used to evaluate the solubility as the confidence in this work easily.

Acknowledgement

We thank Zhenghang Cui for helpful discussion. MS was supported by KAKENHI 17H00757, NC was supported by MEXT scholarship, and HB was supported by JST, ACT-I, Grant Number JPMJPR18UI, Japan.

References

Abbeel, P. and Ng, A. Y. Apprenticeship learning via inverse reinforcement learning. In *ICML*, pp. 1–8, 2004.

Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., and Mané, D. Concrete problems in AI safety. *CoRR*, abs/1606.06565, 2016.

Bartlett, P. L. and Mendelson, S. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Bartlett, P. L. and Tewari, A. Sparseness vs estimating conditional probabilities: Some asymptotic results. *Journal of Machine Learning Research*, 8:775–790, 2007.

Buja, A., Stuetzle, W., and Shen, Y. Loss functions for binary class probability estimation and classification: Structure and applications. Technical report, 2005.

Burchfiel, B., Tomasi, C., and Parr, R. Distance minimization for reward learning from scored trajectories. In *AAAI*, pp. 3330–3336, 2016.

Chapelle, O., Scholkopf, B., and Zien, A. *Semi-Supervised Learning*. MIT press, 2006.

Crippen, G. M. and Snow, M. E. A 1.8 å resolution potential function for protein folding. *Biopolymers: Original Research on Biomolecules*, 29(10-11):1479–1489, 1990.

Dewey, D. Reinforcement learning and the reward engineering principle. In *AAAI Spring Symposium Series*, 2014.

El-Zahhar, M. M. and El-Gayar, N. F. A semi-supervised learning approach for soft labeled data. In *International Conference on Intelligent Systems Design and Applications*, pp. 1136–1141, 2010.

Everitt, T. and Hutter, M. Avoiding wireheading with value reinforcement learning. pp. 12–22, 2016.

Fu, J., Luo, K., and Levine, S. Learning robust rewards with adversarial inverse reinforcement learning. In *ICLR*, 2018.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *NeurIPS*, pp. 2672–2680, 2014.

Ho, J. and Ermon, S. Generative adversarial imitation learning. In *NeurIPS*, pp. 4565–4573, 2016.

Hu, Z., Liang, Y., Zhang, J., Li, Z., and Liu, Y. Inference aided reinforcement learning for incentive mechanism design in crowdsourcing. In *NeurIPS*, pp. 5508–5518, 2018.

Ishida, T., Niu, G., and Sugiyama, M. Binary classification from positive-confidence data. In *NeurIPS*, pp. 5919–5930, 2018.

Kang, B., Jie, Z., and Feng, J. Policy optimization with demonstrations. In *ICML*, pp. 2474–2483, 2018.

Kiryo, R., Niu, G., du Plessis, M. C., and Sugiyama, M. Positive-unlabeled learning with non-negative risk estimator. In *NeurIPS*, pp. 1675–1685, 2017.

Kostrikov, I., Agrawal, K. K., Levine, S., and Tompson, J. Addressing sample inefficiency and reward bias in inverse reinforcement learning. In *ICLR*, 2019.

Ledoux, M. and Talagrand, M. *Probability in Banach Spaces: Isoperimetry and Processes*. Springer, 1991.

Li, Y., Song, J., and Ermon, S. InfoGAIL: Interpretable imitation learning from visual demonstrations. In *NeurIPS*, pp. 3812–3822, 2017.

Liu, Y., Gupta, A., Abbeel, P., and Levine, S. Imitation from observation: Learning to imitate behaviors from raw video via context translation. In *ICRA*, pp. 1118–1125, 2018.

McDiarmid, C. On the method of bounded differences. *Surveys in Combinatorics*, 141(1):148–188, 1989.

Pathak, D., Girshick, R. B., Dollár, P., Darrell, T., and Hariharan, B. Learning features by watching objects move. In *CVPR*, 2017.

Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994. ISBN 0471619779.

Reid, M. D. and Williamson, R. C. Composite binary losses. *Journal of Machine Learning Research*, 11:2387–2422, 2010.

Ross, S., Gordon, G., and Bagnell, D. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635, 2011.

Schaal, S. Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences*, 3(6):233–242, 1999.

Schölkopf, B., Burges, C. J. C., and Smola, A. J. *Advances in Kernel Methods: Support Vector Learning*. MIT press, 1999.

Schulman, J., Levine, S., Abbeel, P., Jordan, M., and Moritz, P. Trust region policy optimization. In *ICML*, pp. 1889–1897, 2015.

Serban, I. V., Sankar, C., Germain, M., Zhang, S., Lin, Z., Subramanian, S., Kim, T., Pieper, M., Chandar, S., Ke, N. R., et al. A deep reinforcement learning chatbot. *CoRR*, abs/1709.02349, 2017.

Shah, P., Hakkani-Tur, D., Liu, B., and Tur, G. Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 41–51, 2018.

Supancic III, J. S. and Ramanan, D. Tracking as online decision-making: Learning a policy from streaming videos with reinforcement learning. In *ICCV*, pp. 322–331, 2017.

Sutton, R. S. and Barto, A. G. *Introduction to Reinforcement Learning*, volume 135. MIT press, 1998.

Syed, U., Bowling, M., and Schapire, R. E. Apprenticeship learning using linear programming. In *ICML*, pp. 1032–1039, 2008.

Todorov, E., Erez, T., and Tassa, Y. Mujoco: A physics engine for model-based control. In *IROS*, pp. 5026–5033, 2012.

Tokmakov, P., Alahari, K., and Schmid, C. Learning motion patterns in videos. In *CVPR*, pp. 531–539, 2017.

Valko, M., Ghavamzadeh, M., and Lazaric, A. Semi-supervised apprenticeship learning. In *EWRL*, pp. 131–142, 2012.

Vapnik, V. *Statistical Learning Theory*, volume 3. Wiley, New York, 1998.

Wang, W., Wang, Y., Chen, F., and Sowmya, A. A weakly supervised approach for object detection based on soft-label boosting. In *IEEE Workshop on Applications of Computer Vision*, pp. 331–338, 2013.

Yeung, S., Ramanathan, V., Russakovsky, O., Shen, L., Mori, G., and Fei-Fei, L. Learning to learn from noisy web videos. In *CVPR*, pp. 7455–7463, 2017.

Zhou, D., Quost, B., and Frémont, V. Soft label based semi-supervised boosting for classification and object recognition. In *International Conference on Control Automation Robotics & Vision*, pp. 1062–1067, 2014.

Ziebart, B. D., Maas, A. L., Bagnell, J. A., and Dey, A. K. Maximum entropy inverse reinforcement learning. In *AAAI*, pp. 1433–1438, 2008.