HOW DOES UNCERTAINTY-AWARE SAMPLE-SELECTION HELP DECISION AGAINST ACTION NOISE?

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning from imperfect demonstrations has become a vital problem in imitation learning (IL). Since the assumption of the collected demonstrations are optimal cannot always hold in real-world tasks, many previous works considers learning from a mixture of optimal and sub-optimal demonstrations. On the other hand, video records can be hands-down demonstrations in practice. Leveraging such demonstrations requires labors to output action for each frame. However, action noise always occurs when the labors are not domain experts, or meet confusing state frames. Previous IL methods can be vulnerable to such demonstrations with state-dependent action noise. To tackle this problem, we propose a robust learning paradigm called USN, which bridges Uncertainty-aware Sample-selection with Negative learning. First, IL model feeds forward all demonstration data and estimates its predictive uncertainty. Then, we select large-loss samples in the light of the uncertainty measures. Next, we update the model parameters with additional negative learning on the selected samples. Empirical results on Box2D tasks and Atari games demonstrate that USN improves the performance of state-of-the-art IL methods by more than 10% under a large portion of action noise.

1 INTRODUCTION

Despite the great success of reinforcement learning (RL) (Sutton & Barto, 2018) over last few years, designing hand-crafted reward functions can be extremely difficult and even impossible in many real-world tasks (Ng et al., 1999; Amodei et al., 2016; Brown et al., 2019). Alternatively, imitation learning (IL) (Russell, 1998; Schaal, 1999; Abbeel & Ng, 2004; Argall et al., 2009; Hussein et al., 2017) aims to train an agent to mimic the demonstrations collected from an expert, without any access to hand-crafted reward signals. However, it is expensive and difficult to collect high-quality demonstrations in real-world tasks (Silver et al., 2013).

In practice, it is much cheaper to collect demonstrations from amateurs (Audiffren et al., 2015). Existing works (Tangkaratt et al., 2019; 2020; Zhang et al., 2021b) have studied imitation learning from a mixture of optimal and non-optimal demonstrations. Specifically, Tangkaratt et al. (2019) requires that all the actions for a demonstration are drawn from the same noisy distribution with sufficiently small variance. Following works (Tangkaratt et al., 2020) proposed robust imitation learning by optimizing a classification risk with a symmetric loss. The resulting algorithms RIL and RIL-CO still require more optimal demonstrations than non-optimal ones in the dataset.

On the other hand, in many practical activities like sport games, it is common for people to use camera to record the excellent behaviors of athletes in terms of sequences of pictures and videos. However, it usually be hard to get the exact action label for each picture. To leverage such data for imitation learning, we need to recruit labors to output action labels for pictures in the sequence. Limited by the quality of the labors, action noise always occur during the action-labeling procedure. An amateur labor may randomly pick an action for a picture that contains a state he never seen before. In this situation, the final demonstration will contain *state-independent action noise*. Besides, even an expert labor makes mistakes. This is especially true when the labor meets some similar and confusing states. In this situation, the labor will output noisy actions that are dependent on the confusing states, resulting a demonstration with *state-dependent action noise*.

Previous methods (Tangkaratt et al., 2019; 2020; Zhang et al., 2021b) focus on imitation learning from a mixture of optimal and non-optimal demonstrations or noisy demonstrations with small noise

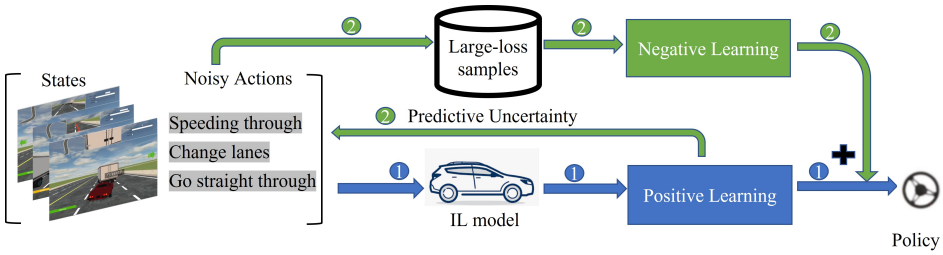


Figure 1: The full procedure of USN.

on the actions. They ignore the fact that both *state-independent action noise* and *state-independent action noise* widely exist in practice. Moreover, these methods are evaluated on low-dimensional environments, and are hard scale to high-dimensional environments. Therefore, existing methods usually fail in learning a good policy with action noise, especially in high-dimensional environments.

To tackle this challenge, we first conduct an investigation experiments to study the correlation between the dynamics of loss and predictive uncertainty metrics as the noise rate increase. Then, we propose a new method called uncertainty-aware sample-selection with soft negative learning (USN) based on the correlation. As shown in Figure 1, USN trains a policy with additional process of uncertainty-aware sample-selection for negative learning (USN). Specifically, in positive learning (marked in blue color), we train the IL model with any noise-tolerant loss function, and then estimate the predictive uncertainty measures during training. Then, we select large-loss samples using the estimated predictive uncertainty measure as the sampling threshold. This is motivated by recent works in the area of learning with noisy labels (Angluin & Laird, 1988; Smyth et al., 1994; Kalai & Servedio, 2005; Natarajan et al., 2013; Manwani & Sastry, 2013). Since deep networks learn easy patterns first (Arpit et al., 2017), they would first memorize training data of clean labels and then those of noisy labels with the assumption that clean labels are of the majority in a noisy class. Therefore, the large-loss samples can be regarded as noisy actions with high probability (Han et al., 2018; Yu et al., 2019; Wei et al., 2020; Yao et al., 2020) . Next, we employ negative learning (marked in green color) (Kim et al., 2019) on the automatically selected large-loss samples, along with positive learning on the full dataset.

USN is a meta-algorithm that is scalable to both offline imitation learning an online imitation learning. Compared with existing IL methods, our method has advantages in many aspects, including imitation performance, adaptivity and scalability . Our method can adaptively select large-loss samples for soft negative learning across different noise rates. Additionally, our method does not depend on any extra datasets, models or prior information about the noise model. That means our method avoids the extra efforts and drawbacks of estimating noise rates and transition matrix as previous noise-robust methods.

2 Background and Related Work

In this section, we firstly discuss existing offline and online imitation learning methods. Then, we discussed related works of learning with noisy labels.

2.1 Offline imitation learning

Behavioral cloning (BC) is probably the simplest offline imitation learning algorithm. For imitation learning in environments with discrete action space, the BC policy $\pi(a|s)$ is optimized by softmax cross-entropy loss.

Batch-constrained Q-learning (BCQ) (Fujimoto et al., 2019b) is the first continuous control algorithm capable of learning from arbitrary batch data, without exploration. BCQ aims to perform Q-learning while constraining the action space to eliminate actions that are unlikely to be selected by the behavioral policy π_b, and are therefore unlikely to be contained in the batch. At its core, BCQ uses a state-conditioned generative model $G : s \to a$ to model the distribution of data in the batch, $G \approx b$ akin to a behavioral cloning model. As it is easier to sample from $\pi_b(a|s)$ than model $\pi_b(a|s)$

exactly in a continuous action space, the policy is defined by sampling N actions a_i from $G(s)$ and selecting the highest valued action according to a Q-network.

Discrete BCQ (Fujimoto et al., 2019a) extends BCQ to discrete action spaces by computing the probabilities of every action $G \approx \pi_b(a|s)$, and instead utilizing some threshold to eliminate actions. To adaptively adjust this threshold, Fujimoto et al. (2019a) scale it by the maximum probability from the generative model over all actions, to only allow actions whose relative probability is above some threshold. Specifically, Fujimoto et al. (2019a) applies Double DQN (Van Hasselt et al., 2016) to select the max valued action with the current Q-network Q_θ, and evaluate with the target Q-network $Q_{\theta'}$. This results in an algorithm comparable to DQN (Mnih et al., 2015) where the policy is defined by a constrained argmax. The Q-network is trained by swapping the max operation with actions selected by the policy:

$$\mathcal{L}(\theta) = l_\kappa\Big(r + \gamma \max_{a|G(a|s)/\max \hat{a} G(\hat{a}|s) > \tau} Q_{\theta'}(s', a') - Q_\theta(s, a)\Big), \tag{1}$$

where l_κ defines the Huber loss (Watkins, 1989):

$$l_\kappa(\delta) = \begin{cases} 0.5\delta^2 & \text{if } \delta \leq \kappa \\ \kappa(|\delta| - 0.5\kappa) & \text{otherwise.} \end{cases} \tag{2}$$

With this threshold τ, Fujimoto et al. (2019a) maintain the original property of BCQ where setting $\tau = 0$ returns Q-learning and $\tau = 1$ returns an imitator of the actions contained in the batch. The generative model G, effectively a behavioral cloning network, is trained in a standard supervised learning fashion, with a cross-entropy loss. In this paper, we adopt the (Discrete) BCQ method for offline IL by training a intrinsic curiosity module (ICM) Pathak et al. (2017) for generating intrinsic rewards, instead of using the true reward data in the demonstration.

Uncertainty weighted actor-critic (UWAC) (Wu et al., 2021) is a recently proposed offline reinforcement learning algorithm that is able to detects OOD state-action pairs and down-weights their contribution in the training objectives accordingly. UWAC is closely related to our work, since it also uses the uncertainty estimation to improve the model robustness. Specifically, UWAC uses Monte Carlo (MC) dropout (Gal & Ghahramani, 2016) to estimate uncertainty of Q model. The model uncertainty is captured by the approximate predictive variance with respect to the estimated $\hat{Q}$ for T stochastic forward passes:

$$Var[Q(s, a)] \approx \sigma^2 + \frac{1}{T} \sum_{t=1}^{T} \hat{Q}_t(s, a)^\top \hat{Q}_t(s, a) - \mathbb{E}[\hat{Q}(s, a)]^\top \mathbb{E}[\hat{Q}(s, a)]. \tag{3}$$

Then, UWAC down-weighs the Bellman loss for the Q function by inverse the uncertainty of the Q-target $Q_{\theta'}(s', a')$:

$$\mathcal{L}(Q_\theta) = \mathbb{E}_{(s'|s,a)\sim\mathcal{D}} \mathbb{E}_{a'\sim\pi(\cdot|s')} \Big[\frac{\beta}{Var[Q_{\theta'}(s', a')]} Err(s, a, s', a')^2\Big], \tag{4}$$

where $Err(s, a, s', a') = Q_\theta(s, a) - (R(s, a)) + \gamma Q_{\theta'}(s', a')$. This directly reduces the effect that OOD backups has on the overall training process (Wu et al., 2021). We treat UWAC as a strong baseline in our paper, and we also show that our USN can scale to MC dropout for uncertainty estimation. The original UWAC was built on top of BEAR (Kumar et al., 2019). To enable a fair comparison, we implement UWAC on top of BCQ as follows:

$$\mathcal{L}(\theta) = l_\kappa\Big(r + \gamma \max_{a|G(a|s)/\max \hat{a} G(\hat{a}|s) > \tau} \frac{\beta}{Var[Q_{\theta'}(s', a')]} \big[Q_{\theta'}(s', a') - Q_\theta(s, a)\big]\Big). \tag{5}$$

2.2 Online imitation learning

Generative adversarial imitation learning (GAIL) (Ho & Ermon, 2016) is one of the state-of-the-art online IL methods. GAIL treats imitation learning as a distribution matching problem. Built of top of GAN (Goodfellow et al., 2014), GAIL and its many (robust) variants (Li et al., 2017; Peng et al., 2019; Tangkaratt et al., 2019; 2020; Wang et al., 2021) has achieved great success in imitation learning in low-dimensional space even with noisy demonstrations. However, GAIL fails to scale

to high-dimensional imitation learning tasks (Brown et al., 2019; Tucker et al., 2018). One simple solution to alleviate this issue is to initiate the actor of GAIL using behavior cloning.

Selective Adversarial Imitation Learning (SAIL) (Wang et al., 2021) was proposed to address imperfect demonstration issue, in which good demonstrations can be adaptively selected for training while bad demonstrations are abandoned. Specifically, a binary weight $w \in \{0, 1\}$ is assigned to each expert demonstration to indicate whether to select it for training. The weight is set to be determined by the reward function in Wasserstein GAIL (Xiao et al., 2019) (i.e. higher reward results in higher weight). The resulting algorithm - SAIL-hard is defined as follows:

$$\min_{\theta} \max_{\phi} \mathbb{E}_{(s,a)\sim\rho_E}[w(s,a)r_\phi(s,a) - Kw(s,a)] - \mathbb{E}_{(s,a)\sim\rho_\phi}[r_\phi(s,a)] + \lambda\Psi(r_\phi). \tag{6}$$

Besides hard binary weighting, they also propose a soft weighting scheme with the suggested optimal soft weight as $w^* = \frac{1}{1+e^{K-r_\phi(s,a)}}$. The corresponding algorithm - SAIL-soft is defined as follows:

$$\min_{\theta} \max_{\phi} \mathbb{E}_{(s,a)\sim\rho_E}[w(s,a)r_\phi(s,a) + f(w(s,a))] - \mathbb{E}_{(s,a)\sim\rho_\phi}[r_\phi(s,a)] + \lambda\Psi(r_\phi) \tag{7}$$

where $f(w) = w\log(w^{-1} - 1) - \log(1 - w) - Kw$. We regards GAIL, and its variants RIL, RIL-CO (Tangkaratt et al., 2020) and SAIL as baselines, for comparing their robustness against action noise.

2.3 Learning with noisy labels

Learning with noisy labels aims to learn a robust classifier f by exploiting training samples with only noisy labels $(x_i, \tilde{y}_i)_{i=1}^N$.

Generalized Cross Entropy (GCE) (Zhang & Sabuncu, 2018) is a generalization of Cross Entropy (CE) and Mean Absolute Error (MAE). To exploit the benefits of both the noise-robustness provided by MAE and the implicit weighting scheme of CCE, they proposed using the negative Box-Cox transformation (Box & Cox, 1964) as a loss function:

$$\mathcal{L}_q(f(x), e_j) = \frac{(1 - f_j(x)^q)}{q}, \tag{8}$$

where $q \in (0, 1]$. It is equivalent to CE for $\lim_{q\to 0} \mathcal{L}_q(f(x), e_j)$, and becomes MAE/unhinged loss when $q = 1$.

Label smoothing has been commonly used to improve the performance of deep learning models (Szegedy et al., 2016; Chorowski & Jaitly, 2016; Vaswani et al., 2017; Zoph et al., 2018; Real et al., 2019; Huang et al., 2019). It serves as a regularizer (Li et al., 2020; Pereyra et al., 2017; Müller et al., 2019; Yuan et al., 2020; Zhang et al., 2021a) that improves the generalization and model calibration (predictive uncertainty) by using smoothed labels $\tilde{y}$ that mix the original one-hot labels y with an uniform mixture over all possible labels for $\alpha \in [0, 1]$:

$$\tilde{y}_i = (1 - \alpha) \cdot y_i + \frac{\alpha}{K} \cdot \mathbf{1}, \tag{9}$$

where K is the number of classes. Recently, Lukasik et al. (2020) demonstrated label smoothing is also effective when learning with symmetric label noise.

3 How does action noise affect imitation learning?

In this section, we first introduce *state-independent action noise* and *state-dependent action noise*. Then, we briefly introduce widely used predictive uncertainty measures, i.e. calibration errors and MC dropout. Next, we study the correlation between the performance (i.e. average return), loss and predictive uncertainty of the imitation learning model, under multiple level of action noise. Finally, we introduce our uncertainty-aware robust learning paradigm using the observed correlation.

State-independent action noise: In traditional imitation learning, we consider the demonstrations $\mathcal{D} = \{s_i, a_i, s_{i+1}\}_{i=1}^N$ are collected from some expert demonstrator. In practice, we may recruit labors to give action labels for the recorded sequences of expert behaviors. The *the state-independent action noise* occurs when a amateur labor randomly pick an action for the states he never seen

before. The noise generation process is quite similar to the class-conditional label noise, where a corrupted label is randomly flipping from other classes (Symemetric noise) or its neighbor (Pairflip noise) (Natarajan et al., 2013; Han et al., 2018). Therefore, we build the noise modeling of *state-independent action noise* by borrowing the setting of class-conditional label noise. The resulting noisy demonstrations is noted as $\tilde{\mathcal{D}} = \{s_i, a_i, s_{i+1}\}_{i=1}^{N}$.

State-dependent action noise: The *state-dependent action noise* occurs when an expert labor meets confusing states. In this situation, we assume the noisy action is dependent on the state feature and is independent on the expert action: $P(\tilde{a}_i|a_i, s_i) = P(\tilde{a}_i|s_i)$. Figure 2 shows a graphical model for state-dependent action noise. We simulate the generation of *state-dependent action noise* following the instance-dependent label noise setting (Xia et al., 2020). The detailed generation process are summarized in the Appendix A.

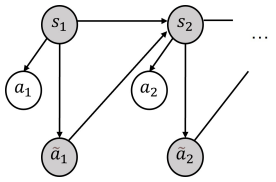


Figure 2: Graphical model describing state-dependent action noise.

3.1 Predictive uncertainty

Expected Calibration Error (ECE) (Naeini et al., 2015; Guo et al., 2017) is widely used to measure the predictive uncertainty (model calibration) of a deep network. To approximate the calibration error in expectation, ECE discretizes the probability interval into a fixed number of bins, and assigns each predicted probability to the bin that encompasses it. The calibration error is the difference between the fraction of predictions in the bin that are correct (accuracy) and the mean of the probabilities in the bin (confidence). Intuitively, the accuracy estimates $\mathbb{P}(Y = y|\hat{p} = p)$, and the average confidence is a setting of p. ECE computes a weighted average of this error across bins: $\mathrm{ECE} = \sum_{b=1}^{B} \frac{n_b}{N} |acc(b) - conf(b)|$, where n_b is the number of predictions in bin b, N is the total number of data points, and $acc(b)$ and $conf(b)$ are the accuracy and confidence of bin b, respectively. Recently, researchers have proposed some variants of ECE for uncertainty estimation, e.g. Adaptive Calibration Error (ACE) (Nixon et al., 2019), Thresholded Adaptive Calibration Error (TACE), Static Calibration Error (SCE) (Nixon et al., 2019), and Root Mean Square Calibration Error (RMSCE) (Hendrycks et al., 2018).

Monte Carlo (MC) dropout (Gal & Ghahramani, 2016) is another widely used method for uncertainty estimation. In reinforcement learning (Wu et al., 2021), the model uncertainty can be captured by the approximate predictive variance for T stochastic forward pass (see Eq. (3)). Instead of using the uncertainty for down-weighting the Bellman loss for Q function (Wu et al., 2021), we leverage uncertainty for robust imitation learning in a novel direction. Namely, we first study the correlation between loss and uncertainty; and then propose an robust paradigm called uncertainty-aware sample selection for negative learning (USN) based on the correlation.

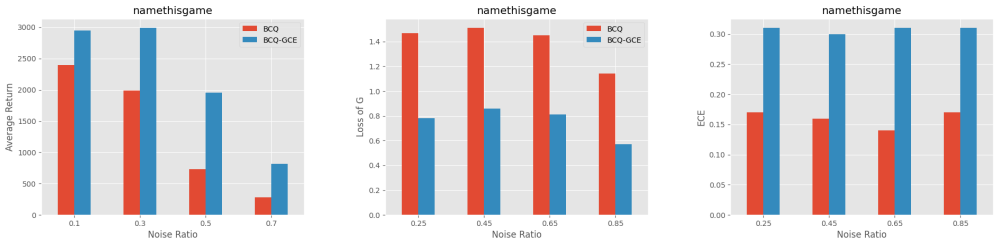
3.2 Correlation between loss and predictive uncertainty under action noise

To study the correlation between the model performance, loss and uncertainty, we perform an investigation experiments on an Atari game under diverse action noise. We first simulate *state-dependent action noise* following *instance-dependent label noise* (Xia et al., 2020; Cheng et al., 2020). We summarize the procedure for generating *state-dependent action noise* in Algorithm 4 of Appendix A. With the obtained noisy demonstrations, we study the performance of the original BCQ (Algorithm 1) and its robust variant BCQ-GCE. We train an ICM module (Pathak et al., 2017) for generating intrinsic rewards for the training of BCQ and its variants. BCQ starts by sampling a mini batch of transitions from the input demonstration data $\mathcal{D}$. Then, we use the generative model G to eliminate actions (step 5) for updating Q network (step 6). The generative model G is trained using a cross-entropy loss in step 7. BCQ-GCE improves BCQ by replacing the cross-entropy loss (step 7, Algorithm 1) with the generalized cross entropy (GCE) loss in Eq. (8).

We show the comparative results of BCQ and BCQ-GCE across diverse noise rates in the bar charts of Figure 3. We have two essential observations from the results: First, the original BCQ is fragile to the *state-dependent action noise*. The average return in Figure 3(a) shows that BCQ drops almost linearly with the increase of noise rates. The robustness of BCQ is improved when replacing CE with a noise-robust loss function, e.g. the GCE loss from the literature of learning with class-conditional

Algorithm 1 BCQ for offline imitation learning.

Input: Demonstration $\mathcal{D} = \{(s_i, a_i, s_{i+1})\}_{i=1}^{N}$, number of iterations T, target_update_rate, mini-batch size B, threshold τ.
Initialize Q-network Q_θ, generative model G with parameter ω and target network $Q_{\theta'}$ with $\theta' \leftarrow \theta$.
for $t = 1$ to T **do**
  Sample mini-batch M of B transitions (s, a, s') from $\mathcal{D}$.
  Train an ICM module to generate intrinsic reward: $r = \text{ICM}(s, a, s')$.
  $a' \leftarrow \arg\max_{a'|G(a'|s')/\max \hat{a} G(\hat{a}|s') > \tau} Q_\theta(s', a')$.
  $\theta \leftarrow \arg\min_\theta \sum_{(s,a,s')\in M} k_\kappa(r + \gamma Q_{\theta'}(s', a') - Q_\theta(s, a))$
  $\omega \leftarrow \arg\min_\omega -\sum_{(s,a)\in M} \log G(a|s)$. //Update G using cross-entropy loss.
  If t mod target_update_rate $= 0 : \theta' \leftarrow \theta$.
end for



(a) Average Return (b) Loss (c) Predictive Uncertainty

Figure 3: Bar chats showing the negative correlation between the generative model loss and predictive uncertainty (e.g. ECE) of BCQ and BCQ-GCE when coping with multi-ratio *state-dependent action noise* on the Atari game, NameThisGame.

label noise. Figure 3(a) shows that BCQ-GCE outperforms BCQ across all levels of action noise. Second, the loss of generative model G in both BCQ and BCQ-GCE goes up first and then drops as the noise rate increases. Interestingly, the calibration errors show a negative correlation with the dynamics of G's loss. The predictive uncertainty, e.g. the ECE drop slightly at first and then increase as the noise rate increases. These interesting observations motivates us to design a robust IL method that uses the negative correlation between G's loss and the predictive uncertainty.

4 METHOD

Based on the above observations, we propose a novel method called Uncertainty-aware sample-selection with Soft Negative learning (USN). We illustrate the full procedure of USN in Algorithm 2 . Given a mini-batch of demonstration data, we first employ a basic robust IL method, e.g. BCQ-GCE for positive learning (Eq. (10)) on the full-batch data:

$$\mathcal{L}_{pos} = \frac{(1 - G(a|s)^q)}{q}. \tag{10}$$

Performing BCQ-GCE alone can be robust to a sufficient amount of action noise; however, it is weak when noise rate is high. To address this challenge, we propose to employ soft negative learning on the 'large-loss' samples. Without using the gradient ascent technique used in previous work (Han et al., 2020), our method uses negative learning to reduce the negative effects of high-level action noise, and thus boost the imitation learning performance across different levels of action noise.

To achieve this goal, we estimate the predictive uncertainty measures (ECE, ACE, TACE, SCE, and RMSCE) of the positive learning model. The small-loss criterion is widely used to select samples that can be regarded as clean samples with high probability. Previous sample selection methods (Hafner et al., 2018; Han et al., 2020; Xia et al., 2021) usually assume that the noise rate is known, and

design the sample selection threshold using the noise rate. In contrast, we define the sample selection threshold τ_u using the estimated calibration errors (predictive uncertainty measures). Since the calibration errors adaptively changes during training, the sample selection threshold is automatically updated and is adaptive to different noise rates. We can maintain less negative samples with small noise rates and more negative learning when the noise rate is high, since the uncertainty measures decreases as the noise rate increases. In this way, our method can maximally reduces the negative effects of action noise using the adaptive threshold, without requiring any prior knowledge about the noise model.

Algorithm 2 Uncertainty-aware Sample-selection with Soft Negative learning (USN)

1: **Input:** A mini-batch M of B transitions (s, a, s') from $\mathcal{D}$, negative learning loss weight as λ_{neg}, model parameter ω.
2: Estimate predictive uncertainty, e.g. ECE, and regards it as the sample-selection threshold τ_u.
3: Sample the large-loss batch $\tilde{M}$ by selecting the large-loss samples with a length of: $\#neg = (1 - \tau_u) * B$.
4: Generate complementary actions for $\tilde{M}$: $\bar{a} = \text{Randomly select from}\{1, ..., |\mathcal{A}|\}\backslash\{a\}$, resulting a complementary batch $\bar{M}$ for negative learning.
5: $\omega \leftarrow \arg\min_\omega \sum_{(s,a)\in M} \mathcal{L}_{\text{pos}}(s, a) + \lambda_{neg} \sum_{(s,a)\in \bar{M}} \mathcal{L}_{\text{neg}}(s, a)$.
6: **Output:** w.

With the adaptive threshold, we select the remaining large-loss samples for **Soft Negative Learning.** Specifically, we employ the following negative log likelihood (NLL) loss for negative learning on the 'large-loss' samples with label smoothing:

$$\mathcal{L}_{neg} = \text{NLL}\Big(1 - G(a|s), (1 - \alpha) \cdot \bar{a} + \frac{\alpha}{|\mathcal{A}|} \cdot \mathbf{1}\Big), \tag{11}$$

where $\bar{a}$ is the complementary action of the 'large-loss' samples, and α is the smooth rate.

Algorithm 3 BCQ-USN for robust offline imitation learning against action noise.

1: **Input:** Demonstration $\mathcal{D} = \{(s_i, a_i, s_{i+1})\}_{i=1}^N$, number of iterations T, target_update_rate, mini-batch size B, threshold τ.
2: Initialize Q-network Q_θ, generative model G with parameter ω and target network $Q_{\theta'}$ with $\theta' \leftarrow \theta$.
3: **for** $t = 1$ to T **do**
4: Sample mini-batch M of B transitions (s, a, s') from $\mathcal{D}$.
5: Train an ICM module to generate intrinsic reward: $r = \text{ICM}(s, a, s')$.
6: $a' \leftarrow \arg\max_{a'|G(a'|s')/\max \hat{a} G(\hat{a}|s') > \tau} Q_\theta(s', a')$.
7: $\theta \leftarrow \arg\min_\theta \sum_{(s,a,s')\in M} k_\kappa(r + \gamma Q_{\theta'}(s', a') - Q_\theta(s, a))$
8: $\omega \leftarrow \text{USN}(M)$ // Update G using USN approach in Algorithm 2.
9: If t mod target_update_rate $= 0 : \theta' \leftarrow \theta$.
10: **end for**

Algorithm 3 presents an instance of our proposed robust IL method, BCQ-USN. BCQ-USN improves BCQ by updating the generative model G using our USN method (Algorithm 2) in step 7. USN improves the ability of generative model G for eliminating noisy actions for policy learning. Thus, BCQ-USN is more robust than BCQ and BCQ-GCE, achieving better imitation learning performance across diverse levels of action noise (Figure 3(a)).

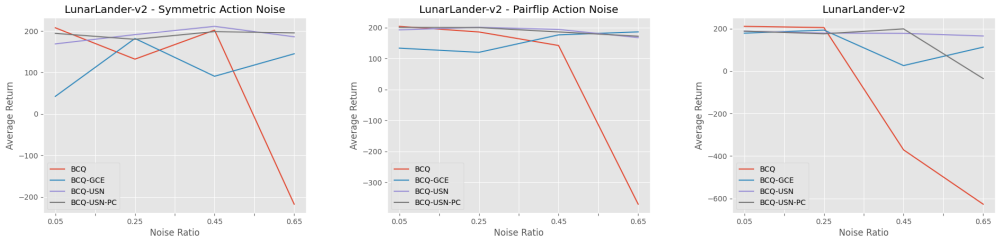
5 Experiments and Results

In this section, we evaluate our methods when learning with *state-independent action noise* and *state-dependent action noise*. We present empirical evaluation of BCQ-USN in a variety of settings. We start with description of our experimental setup, datasets and baseline algorithms.

Settings. We construct experiments on the widely used discrete control tasks and Atari games from OpenAI Gym environments. To evaluate the effectiveness of our methods, we pre-train expert agent

and then generate demonstrations with state-dependent noisy actions using the pre-trained agent. For each setting, we have two demonstrations, $\mathcal{D}$ for *normal demonstration* and $\tilde{\mathcal{D}}$ for demonstration with state-dependent noisy actions only, a.k.a. the *noisy demonstration*. Note that we use ground-truth rewards only to train the expert agent, and we discard the rewards afterward. We implement our methods using the Tianshou platform (Weng et al., 2021) for offline imitation learning, and train all of the algorithms with the default hyperparameters of BCQ. In **setting 1**, we use the control task from OpenAI Gym - LunarLander-v2. We pre-train a DQN policy as the expert agent, and generate demonstrations with 50K steps. In **setting 2**, we use widely used Atari games. We first pre-train a QRDQN policy as the expert agent. Then, we generate a 50K-step demonstration dataset. We set the negative learning loss weight $\lambda_{neg} = 0.01$ for discrete control task and $\lambda_{neg} = 1.0$ for Atari games. We compare BCQ-USN with two **Baseline algorithms:** (a) **BCQ** in Algorithm 1. and (b) **BCQ-GCE**, which is a robust BCQ by training generative model with GCE loss.

Robustness to *state-independent action noise*. In Figure 4(a)(b), we evaluate our method on two other types of state-independent action noise: symmetric noise and pairflip noise. These two types of noise have been widely studied in the literature of learning with noisy labels. We can see that the performance of BCQ drops when noise rate is larger than 0.45 in both Symmetric and Pairflip action noise. Our method BCQ-USN outperforms BCQ-GCE in both cases across diverse noise level.



(a) Symmetric noise (b) Pairflip noise (c) State-dependent noise

Figure 4: Average return of BCQ, BCQ-GCE and our BCQ-USN on LunarLander-v2 with Symmetric and Pairflip action noise.

Robustness to *state-dependent action noise*. Figure 4(c) shows the average return curves of BCQ, BCQ-GCE and our BCQ-USN on LunarLander-v2 across diverse levels of *state-dependent action noise*. For both types of action noise, BCQ-USN outperforms BCQ-GCE and BCQ across diverse noise levels. In Figure 7 of the Appendix, we show bar charts comparison of BCQ, BCQ-GCE and BCQ-USN for four Atari games. Our method, BCQ-USN consistently outperform BCQ-GCE and BCQ when the noise rate is 0.45. The performance improvements of BCQ-USN compared to other methods are usually significant. For this experiment, we compare our method with an related baseline - UWAC (Wu et al., 2021). In addition, we introduce a variants of our method called BCQ-USN-MC_dropout, which uses the MC dropout for uncertainty estimation in USN. The quantitative results are summarized in Table 1.

Table 1: Average return of BCQ, BCQ-GCE, BCQ-USN, BCQ-USN-MC_dropout, and UWAC when learning with *state-dependent action noise* with noise rate of 0.45 on four Atari games.

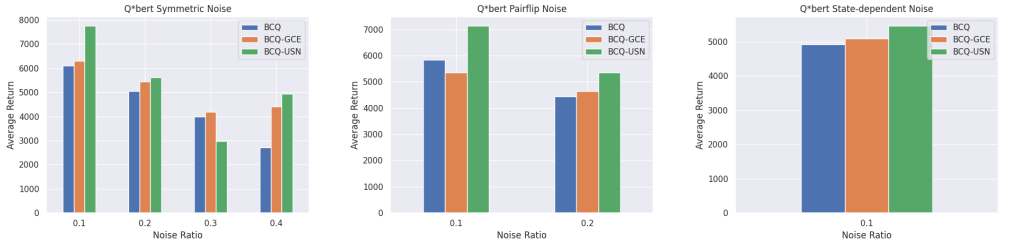
	Assault	Aline	AirRaid	NameThisGame
BCQ	646.4	445.6	1,894.5	2,580.4
BCQ-GCE	667.4	655.8	1,858.5	2,615.2
BCQ-USN (Ours)	678.3	**833.6**	2,170.0	**2,907.2**
BCQ-USN-MC_dropout (Ours)	**699.5**	539.6	**2,183.3**	1,723.6
UWAC (Wu et al., 2021)	695.3	510.6	**2,491.3**	2,317.3

The results show that UWAC only outperforms BCQ-GCE on the Assault and AirRaid games, while performs much worse than BCQ-GCE on two other games. Our variant BCQ-USN-MC_dropout outperforms UWAC and BCQ-GCE on two games. This demonstrates that our method USN can

use MC dropout as an uncertainty estimator, while overall it performs worse than the variant using calibration errors. Overall, BCQ-USN outperform BCQ-GCE by 14% across the four Atari games.

5.1 EXTENSION TO MORE GENERAL Q-LEARNING BASED IL METHODS.

To show the scalability of our method, we apply the USN method to a more general Q-learning based IL method without a generative model, e.g. conservative Q-learning (CQL) (Kumar et al., 2020). In Appendix C, we implement a robust CQL algorithm, CQL-GCE, by training the Q network with the original conservative loss and an additional GCE loss. Similar to BCQ-USN, we apply USN on the positive learning baseline - CQL-GCE. Specifically, we estimate calibration error metrics for the Q network, and then use the calibration error to determine the large-loss samples for negative learning. Empirically, we evaluate the resulting algorithm CQL-USN on the Q*bert Atari game with state-dependent action noise, symmetric action noise and pairflip action noise across different noise rates. Figure 5 shows the comparative results of CQL, CQL-GCE and CQL-USN on the Q*bert game. Impressively, CQL-USN usually outperform CQL and CQL-GCE by a large margin. Therefore, our USN method has good scalability to general Q-learning based IL methods, without requiring the generative model. More details about the algorithms and results are summarized in Appendix C.



(a) Symmetric Action Noise (b) Pairflip Action Noise (c) State-dependent Action Noise

Figure 5: Average return of CQL, CQL-GCE and CQL-USN on Q*bert game with state-depedent action noise, symmetric action noise and pairflip action noise.

5.2 EXTENSION TO ONLINE IMITATION LEARNING WITH NOISY ENVIRONMENT

For online imitation learning, we consider a more challenging setting that imitation learning from noisy demonstrations and noisy environment. Namely, both of the demonstration data and the collected data from the environments are full of action noise. As shown in Table 2, SOTA imitation learning method like GAIL total fails in this task. Our method GAIL-USN outperforms all of the baselines in the Atari game. RIL-CO also performs closely to the optimal performance GAIL*. However, it requires more computation cost due to the usage of co-training.

Table 2: Average return of GAIL, GAIL-USN, RIL, RIL-CO, SAIL-soft and SAIL-hard on the Atari game with 30% *state-dependent action noise* in both the demonstration and the environment.

	GAIL*	GAIL-USN (Ours)	GAIL	RIL	RIL-CO	SAIL-soft	SAIL-hard
Seaquest	1,553.5	**1,564.7**	159.4	776.3	1506.6	29.7	43.2

6 CONCLUSION

In this paper, we propose an novel paradigm for robust imitation learning with acion noise. Our method, uncertainty-aware sample selection for negative learning (USN), scales to both offline imitation learning and online imitation learning with *state-independent action noise* and *state-dependent action noise*. Moreover, our USN paradigm can use diverse predictive uncertainty for robust imitation learning, showing the nice adapativity and scalability of our method. In addition, the ablation studies in the Appendix also demonstrate that our method enables the uncertainty estimation to be multiple usage, and our method is not sensitive to the adaptation of the hyperparameters.

REFERENCES

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning*, pp. 1, 2004.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Dana Angluin and Philip Laird. Learning from noisy examples. *Machine Learning*, 2(4):343–370, 1988.

Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483, 2009.

Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International conference on machine learning*, pp. 233–242. PMLR, 2017.

Julien Audiffren, Michal Valko, Alessandro Lazaric, and Mohammad Ghavamzadeh. Maximum entropy semi-supervised inverse reinforcement learning. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

George EP Box and David R Cox. An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological)*, 26(2):211–243, 1964.

Daniel S Brown, Wonjoon Goo, Prabhat Nagarajan, and Scott Niekum. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. In *International Conference on Machine Learning*, 2019.

Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. Learning with instance-dependent label noise: A sample sieve approach. *arXiv preprint arXiv:2010.02347*, 2020.

Jan Chorowski and Navdeep Jaitly. Towards better decoding and language model integration in sequence to sequence models. *arXiv preprint arXiv:1612.02695*, 2016.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Scott Fujimoto, Edoardo Conti, Mohammad Ghavamzadeh, and Joelle Pineau. Benchmarking batch deep reinforcement learning algorithms. *arXiv preprint arXiv:1910.01708*, 2019a.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International Conference on Machine Learning*, pp. 2052–2062. PMLR, 2019b.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059. PMLR, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2017.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International Conference on Machine Learning*, 2018.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018.

Bo Han, Gang Niu, Xingrui Yu, Quanming Yao, Miao Xu, Ivor Tsang, and Masashi Sugiyama. Sigua: Forgetting may make learning with noisy labels more robust. In *International Conference on Machine Learning*, pp. 4006–4016. PMLR, 2020.

Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. In *Advances in Neural Information Processing Systems*, pp. 4565–4573, 2016.

Yanping Huang, Youlong Cheng, Ankur Bapna, Orhan Firat, Dehao Chen, Mia Chen, HyoukJoong Lee, Jiquan Ngiam, Quoc V Le, Yonghui Wu, et al. Gpipe: Efficient training of giant neural networks using pipeline parallelism. *Advances in neural information processing systems*, 32, 2019.

Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Computing Surveys (CSUR)*, 50(2):1–35, 2017.

Takashi Ishida, Gang Niu, Weihua Hu, and Masashi Sugiyama. Learning from complementary labels. *arXiv preprint arXiv:1705.07541*, 2017.

Adam Tauman Kalai and Rocco A Servedio. Boosting in the presence of noise. *Journal of Computer and System Sciences*, 71(3):266–290, 2005.

Youngdong Kim, Junho Yim, Juseung Yun, and Junmo Kim. Nlnl: Negative learning for noisy labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 101–110, 2019.

Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems*, 32, 2019.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *arXiv preprint arXiv:2006.04779*, 2020.

Weizhi Li, Gautam Dasarathy, and Visar Berisha. Regularization via structural label smoothing. In *International Conference on Artificial Intelligence and Statistics*, pp. 1453–1463. PMLR, 2020.

Yunzhu Li, Jiaming Song, and Stefano Ermon. Infogail: Interpretable imitation learning from visual demonstrations. *Advances in Neural Information Processing Systems*, 30, 2017.

Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *International Conference on Machine Learning*, pp. 6448–6458. PMLR, 2020.

Naresh Manwani and PS Sastry. Noise tolerance under risk minimization. *IEEE transactions on cybernetics*, 43(3):1146–1151, 2013.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

Andrew Y Ng, Daishi Harada, and Stuart Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pp. 278–287, 1999.

Jeremy Nixon, Michael W Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. Measuring calibration in deep learning. In *CVPR Workshops*, volume 2, 2019.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, pp. 2778–2787, 2017.

Xue Bin Peng, Angjoo Kanazawa, Sam Toyer, Pieter Abbeel, and Sergey Levine. Variational discriminator bottleneck: Improving imitation learning, inverse rl, and gans by constraining information flow. In *International Conference on Learning Representations*, 2019.

Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, 2017.

Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *Proceedings of the aaai conference on artificial intelligence*, volume 33, pp. 4780–4789, 2019.

Stuart Russell. Learning agents for uncertain environments. In *Proceedings of the eleventh annual conference on Computational learning theory*, pp. 101–103, 1998.

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

David Silver, J Andrew Bagnell, and Anthony Stentz. Learning autonomous driving styles and maneuvers from expert demonstration. In *Experimental Robotics*, pp. 371–386. Springer, 2013.

Padhraic Smyth, Usama Fayyad, Michael Burl, Pietro Perona, and Pierre Baldi. Inferring ground truth from subjective labelling of venus images. *Advances in neural information processing systems*, 7, 1994.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction (2nd Edition)*. MIT press, 2018.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

Michita Imai Takuma Seno. d3rlpy: An offline deep reinforcement library. In *NeurIPS 2021 Offline Reinforcement Learning Workshop*, December 2021.

Voot Tangkaratt, Bo Han, Mohammad Emtiyaz Khan, and Masashi Sugiyama. Vild: Variational imitation learning with diverse-quality demonstrations. *arXiv preprint arXiv:1909.06769*, 2019.

Voot Tangkaratt, Nontawat Charoenphakdee, and Masashi Sugiyama. Robust imitation learning from noisy demonstrations. *arXiv preprint arXiv:2010.10181*, 2020.

Aaron Tucker, Adam Gleave, and Stuart Russell. Inverse reinforcement learning for video games. *Workshop on Deep Reinforcement Learning at NeurIPS*, 2018.

Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 30, 2016.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Yunke Wang, Chang Xu, and Bo Du. Robust adversarial imitation learning via adaptively-selected demonstrations. In *IJCAI*, pp. 3155–3161, 2021.

Christopher John Cornish Hellaby Watkins. Learning from delayed rewards. 1989.

Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating noisy labels by agreement: A joint training method with co-regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13726–13735, 2020.

Jiayi Weng, Huayu Chen, Dong Yan, Kaichao You, Alexis Duburcq, Minghao Zhang, Hang Su, and Jun Zhu. Tianshou: A highly modularized deep reinforcement learning library. *arXiv preprint arXiv:2107.14171*, 2021.

Yue Wu, Shuangfei Zhai, Nitish Srivastava, Joshua Susskind, Jian Zhang, Ruslan Salakhutdinov, and Hanlin Goh. Uncertainty weighted actor-critic for offline reinforcement learning. *arXiv preprint arXiv:2105.08140*, 2021.

Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. *Advances in Neural Information Processing Systems*, 33:7597–7610, 2020.

Xiaobo Xia, Tongliang Liu, Bo Han, Mingming Gong, Jun Yu, Gang Niu, and Masashi Sugiyama. Sample selection with uncertainty of losses for learning with noisy labels. *arXiv preprint arXiv:2106.00445*, 2021.

Huang Xiao, Michael Herman, Joerg Wagner, Sebastian Ziesche, Jalal Etesami, and Thai Hong Linh. Wasserstein adversarial imitation learning. *arXiv preprint arXiv:1906.08113*, 2019.

Quanming Yao, Hansi Yang, Bo Han, Gang Niu, and James Tin-Yau Kwok. Searching to exploit memorization effect in learning with noisy labels. In *International Conference on Machine Learning*, pp. 10789–10798. PMLR, 2020.

Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pp. 7164–7173. PMLR, 2019.

Li Yuan, Francis EH Tay, Guilin Li, Tao Wang, and Jiashi Feng. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3903–3911, 2020.

Chang-Bin Zhang, Peng-Tao Jiang, Qibin Hou, Yunchao Wei, Qi Han, Zhen Li, and Ming-Ming Cheng. Delving deep into label smoothing. *IEEE Transactions on Image Processing*, 30:5984–5996, 2021a.

Songyuan Zhang, Zhangjie Cao, Dorsa Sadigh, and Yanan Sui. Confidence-aware imitation learning from demonstrations with varying optimality. *Advances in Neural Information Processing Systems*, 34, 2021b.

Zhilu Zhang and Mert R Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. In *32nd Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8697–8710, 2018.