Towards Uniformly Superhuman Autonomy via Subdominance Minimization

Brian D. Ziebart [1] **Sanjiban Choudhury** [2] **Xinyan (Shane) Yan** [2] **Paul Vernaza** [2]

Abstract

Prevalent imitation learning methods seek to produce behavior that matches or exceeds *average* human performance. This often prevents achieving expert-level or superhuman performance when identifying the better demonstrations to imitate is difficult. We instead assume demonstrations are of varying quality and seek to induce behavior that is unambiguously better (i.e., Pareto dominant or minimally subdominant) than *all* human demonstrations. Our **minimum subdominance inverse optimal control** training objective is primarily defined by high quality demonstrations; lower quality demonstrations, which are more easily dominated, are effectively ignored instead of degrading imitation. With increasing probability, our approach produces superhuman behavior incurring lower cost than demonstrations on the demonstrator's unknown cost function—even if that cost function differs for each demonstration. We apply our approach on a computer cursor pointing task, producing behavior that is 78% superhuman, while minimizing demonstration suboptimality provides 50% superhuman behavior—and only 72% even after selective data cleaning.

1. Introduction

Learning from human demonstrations is a desirable alternative to hand-crafting an autonomous system's policy or specifying its cost function (Osa et al., 2018). Inverse reinforcement learning (Ng & Russell, 2000; Abbeel & Ng, 2004) seeks to learn cost functions that reflect human preferences and induce human-like behaviors across different decision processes. However, human demonstrations of sequential control (even from experts) often vary in quality due in part to visuomotor system imprecisions (Wolpert et al., 1995) and/or bounded rationality (Simon, 1997). Low quality demonstrations (Figure 1) pose significant challenges to existing cost function learning approaches that seek to minimize the suboptimality of demonstrated behavior (Ratliff et al., 2006; Ziebart et al., 2008). The set of training demonstrations may be carefully cleansed of harmful outliers, but this can become more art than science, with undesirable sensitivities to laborious or haphazard cleaning processes.

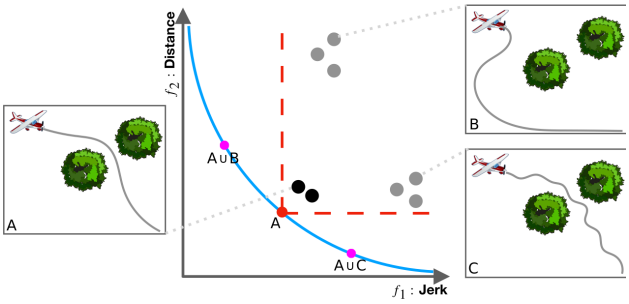

Figure 1. Combining high quality demonstrations (black points A) with low quality demonstrations (gray points B or C) to train existing methods shifts the high quality learned behavior (red point, learned from A only) to lower quality (magenta points, A ∪ B and A ∪ C) on the Pareto frontier (blue curve). Our approach instead seeks to Pareto dominate (red dashed lines) all demonstrations.

We instead seek **uniformly superhuman behavior** that is unambiguously better than all demonstrations (i.e., **Pareto dominant** or smaller in all cost features). Achieving this by a large margin enables better generalization to the broader population distribution of human behaviors. Since low quality demonstrations are typically easy to Pareto dominate, their influence on learned behavior is minimal. Unfortunately, strict improvement over all demonstrated behaviors is often impossible, so we relax our objective to **minimization of subdominance**. This hinge-loss surrogate of the Pareto dominance measures the largest (or sum of) difference(s) in costs preventing induced behavior from Pareto dominating a demonstration. This is reminiscent of structured support vector machines (Tsochantaridis et al., 2004; Taskar et al., 2003): higher quality demonstrations that are not sufficiently dominated by the learned behavior serve as *support vectors* for the learned cost function.

Our learned behavior's **superhuman percentile** (i.e., percentage of demonstrations that it Pareto dominates) on unseen demonstrations is bounded in expectation by the rela-

[1]Computer Science, University of Illinois Chicago [2]Aurora Innovation. Correspondence to: B. Ziebart <bziebart@uic.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

tive frequency of non-support vectors (Vapnik & Chapelle, 2000) and by the relationship between the sample mean and sample deviation of demonstration subdominances. Additionally, unlike previous margin-based approaches for imitation learning (Ratliff et al., 2006), subdominance minimization is Fisher consistent, meaning that under ideal learning conditions it learns to produce behaviors that are maximally superhuman for the given demonstrations.

We evaluate our approach on a computer cursor pointing task that illustrates the varying qualities of demonstrated behaviors. By minimizing subdominance, our approach produces behavior that is 78% superhuman, while maximum entropy inverse reinforcement learning (Ziebart et al., 2008), which minimizes demonstration costs relative to the softmin distribution of trajectories, is only 50% superhuman—and only achieves 72% even after selective data cleaning.

2. Problem Formulation & Related Work

2.1. Imitation Learning Task

Imitation learning is often framed using a Markov Decision Process, $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R})$, from which a demonstrator produces trajectories, $\tilde{\xi} = (\tilde{s}_1, \tilde{a}_1, \ldots, \tilde{a}_{T-1}, \tilde{s}_T)$, of states, $\tilde{s}_t \in \mathcal{S}$, and actions, $\tilde{a}_t \in \mathcal{A}$, according to the demonstrator's policy, π, and the state transition dynamics, $\mathcal{T}$. Often, trajectories are obtained from related decision processes (e.g., differing initial/goal states). The imitation learning task is to estimate a policy, $\hat{\pi} : \mathcal{S} \to \mathcal{A}$, (or stochastic policy, $\hat{\pi} : \mathcal{S} \to \Delta_{\mathcal{A}}$) producing behavior that is similar, in some sense, to the demonstrated trajectories, $\{\tilde{\xi}\}$, even when applied to states or entire decision processes that have not previously been demonstrated. Though a reward or cost function $\mathcal{R}$ may motivate the demonstrator's behavior, it is unknown to the imitation learner. Instead, state features, $\mathbf{f} : \mathcal{S} \to \mathbb{R}^K_{\geq 0}$ (or state-action features, $\mathbf{f} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^K_{\geq 0}$), and other characteristics of the decision process, are often assumed to be available for the imitation learner to appropriately generalize to new settings.

2.2. Behavioral Cloning

Behavioral cloning (Pomerleau, 1989) frames policy estimation as a direct supervised learning problem in which the action $\tilde{a}_t$ should be predicted given the state $\tilde{s}_t$. Newell (1994) describes this as "treat[ing] the mind as one big monster response function." The advantages are in the simplicity of the approach: reasoning about state-transition dynamics $\mathcal{T}$ or reward/cost function $\mathcal{R}$ is completely avoided, as is interacting with or simulating the decision process. However, policies learned in this manner may be much less compact than the underlying structure of the decision process from which they were produced, making learning much less data efficient. Additionally, estimating actions independently allows errors to compound over time, potentially leading to a shift between the distribution of states encountered by the demonstrator and those encountered by the imitator (Ross et al., 2011). Covariate shift correction methods may adequately address this when demonstrations have sufficient coverage of the state space (Spencer et al., 2021), but soliciting additional demonstrations may be required when the shift is substantial (Ross et al., 2011). Lastly, behavioral cloning generally limits the imitation learner, at best, to the plurality action of multiple noisy demonstrators rather than extrapolating beyond them to provide better performance.

2.3. Feature Matching & Suboptimality

Cost function learning methods—known as inverse reinforcement learning (IRL) (Ng & Russell, 2000) or inverse optimal control (IOC) (Kalman, 1963)—seek to rationalize demonstrations by making them (near) optimal solutions of the decision process. **Feature matching** (Abbeel & Ng, 2004) is a foundational idea for accomplishing this. It guarantees the estimated policy $\hat{\pi}$ has expected cost under the demonstrator's unknown fixed cost function weights $\tilde{\mathbf{w}} \in \mathbb{R}^K$ equal to the average of the demonstration policies $\{\tilde{\pi}_i\}$ if the expected feature counts match:

$$\mathbb{E}_{\xi\sim\pi}\left[f_k(\xi)\right] = \frac{1}{N}\sum_{i=1}^{N} f_k(\tilde{\xi}_i), \forall k \tag{1}$$

$$\implies \mathbb{E}_{\xi\sim\hat{\pi}}\left[\text{cost}_{\tilde{\mathbf{w}}}(\xi)\right] = \frac{1}{N}\sum_{i=1}^{N} \text{cost}_{\tilde{\mathbf{w}}}(\tilde{\xi}_i),$$

where $f_k(\xi) = \sum_{s_t\in\xi} f_k(s_t)$. This feature-matching constraint (1) can be enforced using a potential term measuring the **suboptimality** of the demonstrations $\tilde{\xi}$ relative to the induced behavior ξ,

$$\text{subopt}_{\hat{\mathbf{w}}}(\tilde{\xi}, \xi) \triangleq \sum_{k=1}^{K} \hat{\mathbf{w}}_k \big(f_k(\tilde{\xi}) - f_k(\xi)\big), \tag{2}$$

where $\hat{\mathbf{w}}^{\mathrm{T}}\mathbf{f}(\xi)$ is interpreted as the cost of the trajectory parameterzied by learned cost weights $\hat{\mathbf{w}}$.

Existing methods minimize the expected demonstration suboptimality in various ways—for example, by augmenting it with a structured loss (Ratliff et al., 2006) or using probabilistic induced behavior (Baker et al., 2007; Neu & Szepesvári, 2007; Ramachandran & Amir, 2007; Ziebart et al., 2008; Babes et al., 2011; Finn et al., 2016; Bobu et al., 2020). Many recent methods do not assume a fixed feature representation and instead minimize an integral probability metric (Sun et al., 2019; Swamy et al., 2021) or use a generative-adversarial discriminator (Ho & Ermon, 2016).

Maximum entropy IRL (Ziebart et al., 2008) might seem to appropriately facilitate superhuman imitation. The cost function it learns from noisy demonstrations can be optimized, via optimal control or reinforcement learning, to a

degree that none of the demonstrations achieve. However, MaxEnt IRL's Boltzmann distribution for demonstration noise makes the method particularly sensitive to low quality outlier demonstrations that often do not reflect the cost feature trade-offs of high quality demonstrations.

Noise models for specific human biases have been developed (Evans et al., 2016; Majumdar et al., 2017; Reddy et al., 2018; Kwon et al., 2020; Zhi-Xuan et al., 2020) primarily using simple controlled cognitive science experiments. Unfortunately, choosing appropriate noise models automatically for more complex tasks is impossible in general (Armstrong & Mindermann, 2018) without strong assumptions, and often difficult even when imposing unrealistic ones (Shah et al., 2019). We argue that generatively modeling demonstrations is inherently more difficult than imitation, and propose a more discriminative imitation learning approach that avoids the challenges of human bias modeling.

Closer to our approach, Syed & Schapire (2008) leverages known signs of each feature's contribution to the cost function to outperform the demonstrator by considering the worst possible $\tilde{\mathbf{w}}$:

$$\max_{\pi} \min_{\hat{\mathbf{w}}:||\hat{\mathbf{w}}||_1=1, \hat{\mathbf{w}}\succeq\mathbf{0}} \mathbb{E}_{\xi\sim\pi}[\text{subopt}_{\hat{\mathbf{w}}}(\tilde{\xi}, \xi)]. \quad (3)$$

Unfortunately, when demonstration quality widely varies, neither matching nor outperforming demonstration averages guarantees good performance relative to high quality demonstrations. As illustrated by Figure 1, sacrificing low suboptimality on high quality demonstrations to lower the average suboptimality is preferred by these methods.

2.4. Ranking & Confidence Outperformance Methods

Manually ranked sets of demonstrations (Ibarz et al., 2018; Brown et al., 2019; Novoseller et al., 2020; Zhang et al., 2021; Myers et al., 2021; Bıyık et al., 2022) or demonstration significance weights (Wu et al., 2019) can enable the imitator to match or outperform the highest quality demonstrations. However, providing this information, like data cleaning, is an annotation burden we seek to avoid, despite active learning methods that have been designed to reduce this burden (Sadigh et al., 2017; Bıyık & Sadigh, 2018).

Extensions that automatically learn to rank or provide significance weights assume that demonstration-based policy estimates have better rank than more random policies (Brown et al., 2020), that demonstrations follow specific noise models or optimality prevalences (Tangkaratt et al., 2020; 2021; Wang et al., 2021a), or relationships between weights and the advantage function (Wang et al., 2021b). Chen et al. (2020) also uses noise-augmented demonstrations, but learns how rewards degrade as a function of the noise. Unfortunately, available demonstrations can violate these assumptions (e.g., the majority of demonstrations being extremely low quality), producing negative end results.

3. Subdominance Minimization

We seek to learn cost weights that induce **uniformly superhuman behavior** in deterministic decision processes.[1] Pareto dominance (§3.1) and subdominance (§3.2) with respect to demonstrations are key measures for achieving this. We employ a margin-based formulation for our learning task (§3.3) that provides generalization bounds (§3.4) and Fisher consistency (§3.5).

3.1. Pareto Dominance & Superhuman Behavior

A bolder aim than matching (1) or outperforming (3) *average* human performance is outperforming *all demonstrations*—ideally from the population distribution. **Pareto dominance** of behavior ξ over behavior $\tilde{\xi}$, $\mathbf{f}(\xi) \preceq \mathbf{f}(\tilde{\xi})$, is a concept from multi-objective optimization that helps formalize this aim. It requires demonstration $\tilde{\xi}$ to have larger feature counts, guaranteeing the imitator, ξ, no worse cost than the demonstrator's cost, as described in Theorem 1.

Theorem 1. *If ξ Pareto dominates $\tilde{\xi}_i$, then it has an expected cost no worse than the demonstrator under the demonstrator's unknown cost weights $\tilde{\mathbf{w}}^{(i)} \in \mathbb{R}^K_{\geq 0}$:*

$$\mathbf{f}(\xi) \preceq \mathbf{f}(\tilde{\xi}_i) \iff f_k(\xi) \leq f_k(\tilde{\xi}_i), \forall k$$
$$\implies \text{cost}_{\tilde{\mathbf{w}}^{(i)}}(\xi) \leq \text{cost}_{\tilde{\mathbf{w}}^{(i)}}(\tilde{\xi}_i). \quad (4)$$

This guarantee holds even if each demonstration has distinct cost function weights $\tilde{\mathbf{w}}^{(i)}$, which is more realistic than assuming static weights (or human biases, §2.3) for all demonstrations (e.g., due to unmodeled side information). We refer to autonomous behavior that is unambiguously better than human demonstrations as **superhuman**.

Definition 2 (Superhuman percentile)**.** *An autonomous system with behavior ξ is $\gamma-$***superhuman*** *with percentile γ for features $\mathbf{f}$ if: $P(\mathbf{f}(\xi) \preceq \mathbf{f}(\tilde{\xi}_i)) \geq \gamma$.*

We argue that the ideal goal of imitation learning—and artificial intelligence more broadly—is to produce **uniformly superhuman behavior** (i.e., 1-superhuman) *on the population distribution of human behaviors*.

3.2. Subdominance: Variants & Properties

Directly seeking uniformly superhuman behavior (or maximization of γ), poses some technical challenges. First, achieving it on training demonstrations may not generalize to the population distribution of human behaviors. We address this weakness by seeking to outperform each demon-

[1] We consider deterministic dynamics in our derivations and analyses, and generalize to stochastic dynamics, albeit with more complicated notation, in Appendix B.

stration in each feature f_k by a margin β_k. Second, uniformly superhuman behavior may be impossible to achieve when no single behavior Pareto dominates all demonstrations. To deal with this potential impossibility, we define (using $[x]_+ \triangleq \max(x, 0)$ to denote the hinge function) two generalized feature-based notions of **subdominance**:

$$\text{subdom}^k_{\alpha_k,\beta_k}(\xi, \tilde{\xi}) \triangleq \left[\alpha_k \left(f_k(\xi) - f_k(\tilde{\xi})\right) + \beta_k\right]_+; \quad (5)$$

$$\text{relsubdom}^k_{\alpha_k,\beta_k}(\xi, \tilde{\xi}) \triangleq \left[\alpha_k \left(\frac{f_k(\xi)}{f_k(\tilde{\xi})} - 1\right) + \beta_k\right]_+. \quad (6)$$

These measure how far behavior ξ is from Pareto dominating demonstration $\tilde{\xi}$ absolutely (5) or relatively (6) by a margin $\beta_k \geq 0$ with weight $\alpha_k \geq 0$ for cost dimension k. To incorporate multiple feature dimensions, we introduce max-based (7) and sum-based (8) aggregations (Figure 2) of the individual (relative) subdominances over the $k \in \{1, \ldots, K\}$ features:[2]

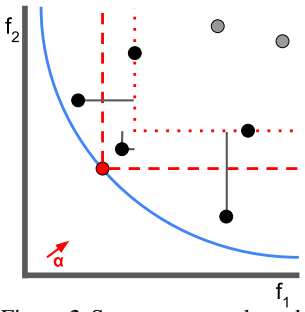

Figure 2. Sum-aggregated subdominance (black lines) measures how far demonstrations (black points) are from margin boundaries (dotted red lines) defined by behavior ξ (red point) and margin parameters α and β.

$$[\text{rel}]\text{subdom}_{\alpha,\beta}(\xi, \tilde{\xi}) \triangleq \max_k [\text{rel}]\text{subdom}^k_{\alpha_k,\beta_k}(\xi, \tilde{\xi}); \quad (7)$$

$$[\text{rel}]\text{subdom}^{\Sigma}_{\alpha,\beta}(\xi, \tilde{\xi}) \triangleq \sum_{k=1}^{K} [\text{rel}]\text{subdom}^k_{\alpha_k,\beta_k}(\xi, \tilde{\xi}). \quad (8)$$

The sum-aggregated subdominance relates to suboptimality as an α-limited worst-case suboptimality (Theorem 3) and as a complementary loss that together define an α-weighted L^1 distance (Theorem 4).

Theorem 3. *The worst case suboptimality with weights $\tilde{\mathbf{w}}$ bounded by α (i.e., $0 \leq w_k \leq \alpha_k \; \forall k$) is the subdominance:*

$$\max_{\mathbf{0} \preceq \tilde{\mathbf{w}} \preceq \alpha} \text{subopt}_{\tilde{\mathbf{w}}_i}(\xi, \tilde{\xi}) = \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\xi, \tilde{\xi}). \quad (9)$$

Theorem 4. *The α-weighted L^1-norm of feature differences, $L^1_\alpha \triangleq \sum_{k=1}^{K} \alpha_k \left|f_k(\xi) - f_k(\tilde{\xi})\right|$, equals the demonstration suboptimality plus twice the subdominance:*

$$L^1_\alpha(\xi, \tilde{\xi}) = \text{subopt}_\alpha(\tilde{\xi}, \xi) + 2\,\text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\xi, \tilde{\xi}).$$

Theorem 4 provides additional justification for the subdominance as an imitation learning loss function. Since some suboptimality is inherent for demonstrations of varying quality, seeking to minimize suboptimality is unnecessary. When it is removed from a natural measure of the difference between trajectories (L^1_α distance), the subdominance that our approach seeks to minimize is what remains. We analyze this decomposition in our experiments (§4.4) to understand what is learned by different methods.

3.3. Margin-Based Formulation and Optimization

Degenerate solutions are a well-known problem for inverse reinforcement learning (Ng & Russell, 2000). The $\mathbf{0}$ weight vector, for example, minimizes suboptimality, $\text{subopt}_{\mathbf{0}}(\tilde{\xi}, \xi) = 0$. However, using a margin threshold $\beta > 0$ in our defined subdominance avoids this degeneracy:[3] $\text{subdom}_{\mathbf{0},\beta}(\tilde{\xi}, \xi) = \beta$. This can equivalently be viewed as leveraging margin-based learning ideas from structured support vector machines (Tsochantaridis et al., 2004) and maximum margin structured prediction (Taskar et al., 2003) by employing a hinge loss surrogate of the Pareto dominance: the subdominance.

Definition 5 (MinSub IOC)**.** *Minimally subdominant inverse optimal control minimizes the subdominance of the minimum cost trajectory, $\xi^*(\mathbf{w})$, induced by learned weights $\mathbf{w}$, with respect to the set of demonstration trajectories $\{\tilde{\xi}_i\}$ using hinge slopes α:*

$$\min_{\mathbf{w} \succeq \mathbf{0}} \min_{\alpha \succeq \mathbf{0}} \frac{1}{N} \sum_{i=1}^{N} [\text{rel}]\text{subdom}^{[\Sigma]}_{\alpha,\mathbf{1}}\left(\xi^*(\mathbf{w}), \tilde{\xi}_i\right) + \frac{\lambda}{2}||\alpha||^2.$$

We fix the margin amounts β in this training objective to one in this paper, and learn the hinge loss slopes α from data. These α values provide the relative sensitivity for failing to sufficiently outperform the demonstrations on the different cost features, i.e., a larger α_k implies greater sensitivity to that feature, and are chosen to minimize subdominance since it upper bounds the generalization error of this approach (§3.4). Regularizing α provides a max margin solution when all demonstrations can be Pareto dominated.

Algorithm 1 Update $\mathbf{w}$ and α from demonstration(s) $\tilde{\xi}$

Obtain optimal behavior $\xi^*(\mathbf{w})$ for weights $\mathbf{w}$ **(Step 1)**
Find support vectors: $\tilde{\Xi}_{\text{SV}_k}(\mathbf{w}, \alpha_k)$ given ξ^* **(Step 2)**
for k, $\tilde{\xi} \in \tilde{\Xi}_{\text{SV}_k}$ **do**
  Update α_k: $\alpha_k \leftarrow \alpha_k e^{\eta_t(f_k(\tilde{\xi}) - f_k(\xi^*) - \lambda\alpha_k)}$ **(Step 3)**
  Update w_k: $w_k \leftarrow w_k e^{\eta_t \alpha_k}$ **(Step 4)**
end for

Algorithm 1 describes the four main steps for updating our model parameters from a mini-batch of demonstrations

[2] We use $[\text{rel}]\text{subdom}^{[\Sigma]}_{\alpha,\beta}(\xi, \tilde{\xi})$ to denote relative or absolute and max- or sum-aggregated subdominance. We refer to $[\text{rel}]\text{subdom}^{[\Sigma]}_{\alpha,\beta}(\tilde{\xi}, \xi)$ as the "reverse" subdominance.

[3] Maximum margin planning (Ratliff et al., 2006) employs a similar margin-based approach for the suboptimality, but is susceptible to the degeneracy $\mathbf{w}$=$\mathbf{0}$ when demonstrations are sufficiently noisy (Fisher inconsistency), which our approach avoids (§3.5).

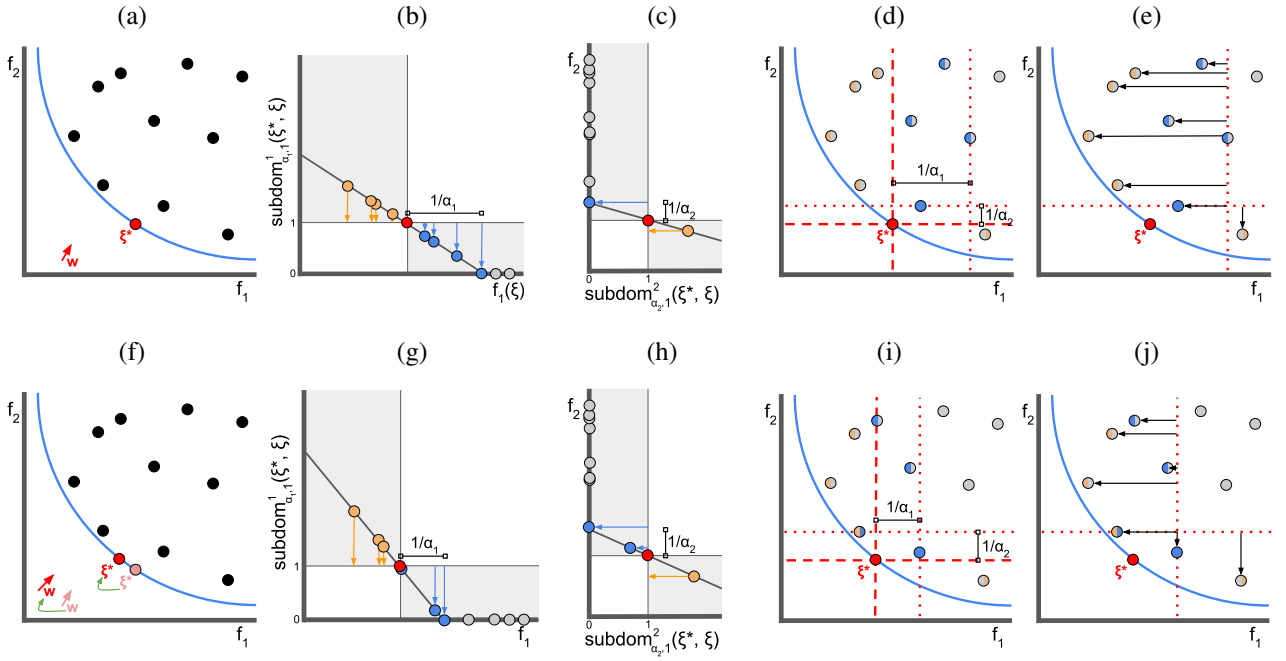

Figure 3. The process for updating model parameters α and $\mathbf{w}$. The optimal behavior ξ^* (red dot) for an initial weight vector $\mathbf{w}$ is obtained (a). The optimal slopes α_1 (b) and α_2 (c) are chosen from all possible slopes (gray regions) to minimize the subdominance, which is achieved by including just enough dominated support vectors (blue) to offset the gradients from nondominated support vectors (orange). These define the margin boundaries (dotted red lines) in (d). Since the number of demonstrations with positive subdominance from feature f_1 (weighted by α_1) is much larger than that of feature f_2 (e), the weight for feature f_1 is increased and a new optimal behavior is obtained (f). Optimizing α_1 (g) and α_2 (h) for this new optimal behavior provides new margin boundaries in (i) and weight gradients (j).

and Figure 3 provides geometric interpretations. Parameter updates are repeatedly applied until convergence.

Step 1: Optimal behaviors. Existing optimal control techniques (or reinforcement learning algorithms, particularly when using a simulator) are used to obtain optimal behavior ξ^* for cost weights $\mathbf{w}$ in the first step of Algorithm 1 (Figures 3a and 3f). Function classes for the policy (with different parameterizations) that avoid solving optimal control problems could instead be employed, but we defer this to future work.

Step 2: Support vectors. The support vector demonstrations of feature k are defined by the optimal trajectory from Step 1 and α_k for the relative subdominance (10) as the set of demonstrations residing exactly on or below the margin boundary (i.e., where the subdominance becomes zero):

$$\tilde{\Xi}_{\mathrm{SV}_k}(\xi^*, \alpha_k) = \left\{ \tilde{\xi} : f_k(\tilde{\xi}) \leq \frac{\alpha_k}{\alpha_k - 1} f_k(\xi^*) \right\}. \quad (10)$$

For max-based subdominance, each demonstration may only belong to one set of support vectors.

Step 3: Hinge slope α updates. The slope parameters α defining the subdominance margins are chosen to minimize the subdominance. As shown in Figures 3b, 3c, 3g, and 3h, this is achieved when the gradients from dominated support vectors match or minimally exceed the gradients from nondominated support vectors:

$$0 \in \partial_{\alpha_k} \sum_{i=1}^{N} \mathrm{subdom}^{[\Sigma]}_{\alpha_k,1}\left(\xi^*(\mathbf{w}), \tilde{\xi}_i\right)\Big|_{\alpha_k=\alpha_k^*} \quad (11)$$

$$\iff \alpha_k^* = \operatorname*{argmin}_{\alpha_k} \sum_{\tilde{\xi}_i \in \tilde{\Xi}_{\mathrm{SV}_k}(\mathbf{w},\alpha_k)} \left(f_k(\xi^*) - f_k(\tilde{\xi}_i)\right) \geq 0.$$

We optimize each α_k using stochastic exponentiated gradient descent: $\alpha_k \leftarrow \alpha_k e^{\eta_t (f_k(\tilde{\xi}) - f_k(\xi^*) - \lambda \alpha_k)}$ using an appropriately decaying learning rate η_t, as shown in Algorithm 1. In other words, we increase α_k for feature k when the optimal behavior outperforms a support vector demonstration, and decrease α_k when it is outperformed by one.

Step 4: Cost weight $\mathbf{w}$ updates. We employ a similar exponentiated subgradient update for cost weights $\mathbf{w}$:

$$\mathbf{w} \leftarrow \mathbf{w} \odot \exp\left(-\eta_t \partial_{\mathbf{w}} \mathrm{subdom}^{k}_{\alpha_k,1}(\xi^*(\mathbf{w}), \tilde{\xi})\right), \quad (12)$$

for the k in which $\tilde{\xi}$ is a support vector, and where $\odot$ denotes element-wise multiplication. We first note that $\mathbf{0}$ is a subgradient for all examples $\tilde{\xi}_i$ with zero subdominance, and

the objective is smooth for examples with positive subdominance, assuming the underlying optimal control problem is smooth. Thus, we can focus our attention strictly on calculating: $\nabla_{\mathbf{w}} \text{subdom}^{k}_{\alpha_k,1}(\xi^*(\mathbf{w}), \tilde{\xi})$ for examples $\tilde{\xi}$ with positive subdominance for k using the chain rule. We introduce an intermediary variable, $f^*_k(\mathbf{w}) = f_k(\xi^*(\mathbf{w}))$, to facilitate this (rather than differentiating with respect to ξ^*):

$$\nabla_{\mathbf{w}} \text{subdom}^{k}_{\alpha_k,1}(f^*_k(\mathbf{w}), \tilde{\xi}) = \frac{\partial}{\partial f^*_k} \text{subdom}^{k}_{\alpha_k,1}(f^*_k(\mathbf{w}), \tilde{\xi})\ \nabla_{\mathbf{w}} f^*_k(\mathbf{w}).$$

The first partial derivative is simply α_k. The remaining portion, $\nabla_{\mathbf{w}} f^*_k(\mathbf{w})$, can either be: (1) computed analytically (Amos et al., 2018), when possible; (2) approximated using a set of finite differences:

$$\frac{\partial}{\partial w_j} f^*_k(\mathbf{w}) \approx \frac{f_k(\xi^*(\mathbf{w} + \epsilon \mathbf{e}_j)) - f_k(\xi^*(\mathbf{w}))}{\epsilon}, \quad (13)$$

where $\mathbf{e}_k$ is the k$^{\text{th}}$ standard unit vector; or (3) approximated using pseudo-gradient optimization (Poljak & Tsypkin, 1973) that only updates the weights corresponding to the features that incur subdominance loss, i.e., $w_k \leftarrow w_k e^{\eta_t \alpha_k}$, as shown in Algorithm 1. The latter two approaches have the benefit of remaining applicable when the underlying optimal control problem is discrete or continuous but not smooth, and therefore not differentiable.

3.4. Generalization Bounds

How well does the learned cost function from this approach generalize to new demonstration samples? Similarly to support vector machines (Vapnik & Chapelle, 2000), the frequency of non-support vectors in the training set bounds the average generalized loss.

Theorem 6. *A MinSub IOC policy with $\{\tilde{\Xi}_{SV_k}(\mathbf{w}, \alpha_k)\}$ support vectors trained on a set of N IID examples to minimize absolute or relative subdominance, [rel]subdom$_{\alpha,\mathbf{1}}(\xi^*(\mathbf{w}), \tilde{\xi}_i)$ is on average (over training samples) γ-**superhuman** on the population distribution with: $\gamma \geq 1 - \frac{1}{N}\left|\left|\bigcup_{k=1}^{K} \tilde{\Xi}_{\text{SV}_k}(\mathbf{w}, \alpha_k)\right|\right|$.*

Minimizing the sum-aggregated subdominance, [rel]subdom$^{\Sigma}_{\alpha,\mathbf{1}}(\xi^*(\mathbf{w}), \tilde{\xi}_i)$, instead provides per-feature guarantees:

$$\mathbb{E}\Big[f_k(\xi^*(\mathbf{w})) \leq f_k(\tilde{\xi})\Big] \geq 1 - \frac{1}{N}\left|\left|\tilde{\Xi}_{\text{SV}_k}(\mathbf{w}, \alpha_k)\right|\right|.$$

The superhuman generalization is also bounded using the sample mean and sample standard deviation of the subdominance of training demonstrations.

Theorem 7. *A MinSub IOC policy with N IID subdominance samples $\{$[rel]subdom$^{[\Sigma]}_{\alpha,\mathbf{1}}(\xi^*(\mathbf{w}), \tilde{\xi}_i)\}_{i=1:N}$ with sample mean $\tilde{\mu}$ and sample standard deviation $\tilde{\sigma}$, is γ-**superhuman** on the population distribution with: $\gamma \geq 1 - \frac{1}{N} - \frac{(N^2-1)}{N^2}\frac{\tilde{\sigma}^2}{(1-\tilde{\mu})^2}$ when $\tilde{\mu} < 1$.*

Thus, when margins can be chosen that make this sample mean and standard deviation small, the rightmost expression approaches zero and the bound on γ tightens towards $\frac{N-1}{N}$. Tighter bounds may also be realized by incorporating physiological limitations of human demonstrators (e.g., reaction times).

3.5. Fisher Consistency

Fisher consistency guarantees that under ideal learning conditions (i.e., learning over the class of all measurable functions using the population distribution), the supervised learner produces the Bayes optimal decision. Unfortunately, margin-based methods for structured prediction generally inherit the Fisher inconsistency of multiclass support vector machines: if no majority label exists in the population distribution conditioned on a particular input (i.e., $\max_y P(y|\mathbf{x}) < 0.5$), the Crammer-Singer SVM (Crammer & Singer, 2001) can fail to learn to predict a plurality label (i.e., $\text{argmax}_y P(y|\mathbf{x})$) (Liu, 2007).

Does minimizing the margin-augmented subdominance suffer from similar Fisher inconsistency as these previous margin-based methods? Permitting learning over all measurable functions is too flexible for the Pareto dominance objective; all the demonstrations can be made Pareto dominatable by any planner behavior. Instead, we consider strictly increasing functions that preserve the Pareto dominance of the original feature space.

Theorem 8. *Letting a new set of feature mappings $\{\phi_k\}$ that are strictly increasing functions of the original features $\{f_k\}$ be learned from the population distribution, the minimization of absolute or relative subdom$_{\alpha,\mathbf{1}}(\xi^*(\mathbf{w}), \tilde{\xi}_i)$ maximizes the the γ frequency of superhuman behavior:*

$$\xi^*(\mathbf{w}) \in \underset{\xi}{\text{argmax}} \max_{\gamma} \gamma\text{-}\mathbf{superhuman}(\xi),$$

or, for the sum-aggregated subdominance, $\xi^*(\mathbf{w}) \in \text{argmax}_{\xi} \sum_{k=1}^{K} P(f_k(\xi) \leq f_k(\tilde{\xi}))$; with the optimal trajectory redefined in terms of $\{\phi_k\}$ and $\{f_k\}$ as: $\xi^*(\mathbf{w}) = \text{argmin}_{\xi} \sum_{k=1}^{K} w_k \phi_k(f_k(\xi))$.

4. Cursor Pointing Inverse Optimal Control

To substantiate our approach's premises and demonstrate its benefits, we consider a simple, but ubiquitous control task: navigating the computer cursor to a target position (i.e., pointing). Though performed virtually on a computer screen, this task—and the variability in human demonstration quality—is representative of many physical pointing tasks (Fitts, 1954).

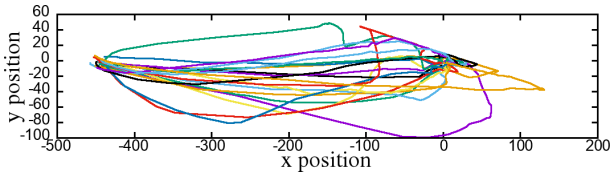

Figure 4. Twenty cursor trajectories for pointing at a target centered at $(0, 0)$ with a 10 pixel radius starting from near $(-450, 0)$.

4.1. Cursor Pointing Tasks & Dataset

In contrast with previous work on learning from demonstrations of varying quality, which have been evaluated almost exclusively on computer-generated demonstrations, we focus our experiments on human-generated demonstrations. We analyze pointing task data gathered from 20 non-motor impaired individuals each performing 300 pointing tasks. Each pointing task requires navigating the computer cursor to a circle of radius 10, 20, or 40 pixels located in a randomized position of the graphical user interface at least 200 pixels (Euclidean distance) away from the starting point. Cursor positions are recorded at a rate of 100Hz.

Following previous work (Ziebart et al., 2012), the state of the cursor is defined using x_t as the cursor position at timestep t along the axis between the starting position and the target position, and using y_t to define the cursor position along the orthogonal axis. The center of the target circle is defined as the origin $(0, 0)$ of the coordinate system. We clean the raw demonstration trajectories in two ways: (1) by removing repeated positions at the beginning ($x_1 = x_2 \wedge y_1 = y_2$) and end ($x_T = x_{T-1} \wedge y_T = y_{T-1}$) of the trajectory; and (2) by removing single timestep "jitters" defined at timestep t by:

$$\left|\left|\begin{bmatrix} x_t \\ y_t \end{bmatrix} - \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix}\right|\right|_2 \Bigg/ \left|\left|\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} - \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix}\right|\right|_2 > 5.0.$$

Figure 4 shows demonstrated trajectories in this coordinate frame. The positions of the demonstrated trajectories over time in this frame are plotted in Figure 5. Some pointing motions appear to consist of a coarse initial movement followed by a more precise corrective movement. Others appear to be smoother single motions. Outliers in position and time are both common. We randomly split the dataset into a training set of 200 tasks and a testing set of 100 tasks.

4.2. Inverse Optimal Control Formulation & Methods

Maximum entropy inverse reinforcement learning has been previously applied using a linear-quadratic regulation (LQR) formulation for this pointing task (Ziebart et al., 2012). The state at timestep t is defined as the position (x_t, y_t), velocity $(\dot{x}_t = x_t - x_{t-1}, \dot{y}_t = y_t - y_{t-1})$, and acceleration $(\ddot{x}_t = \dot{x}_t - \dot{x}_{t-1}, \ddot{y}_t = \dot{y}_t - \dot{y}_{t-1})$: $\mathbf{s}_t = [x_t\ y_t\ \dot{x}_t\ \dot{y}_t\ \ddot{x}_t\ \ddot{y}_t]^T$. The cost function is linear in the outer product of the state

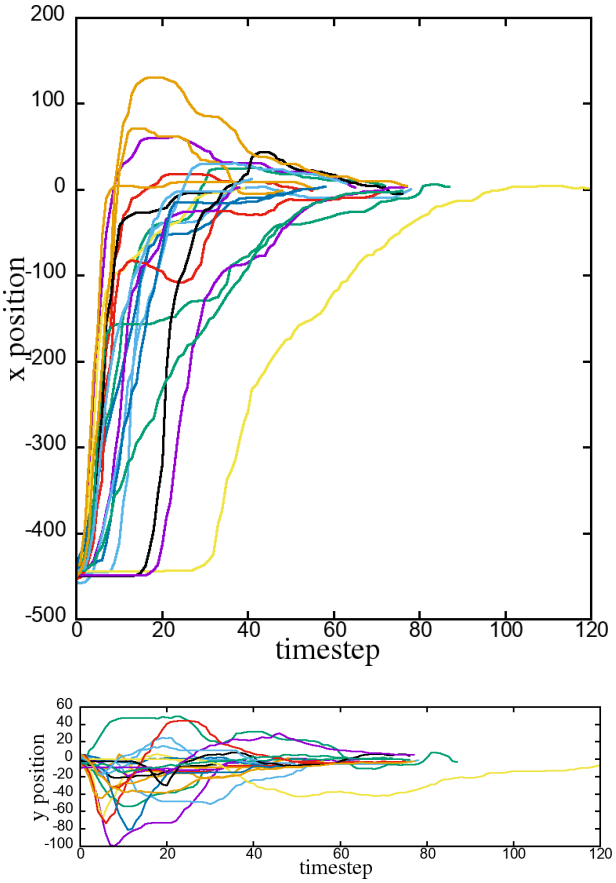

Figure 5. Cursor positions over time in the target-aligned (top) and target-orthogonal (bottom) dimensions.

and can be written using a weight matrix $\mathbf{W}$ and the Frobenius inner product as: $\text{cost}_{\mathbf{W}}(\mathbf{s}_{1:T}) = \left\langle \mathbf{W}, \sum_t \mathbf{s}_t \mathbf{s}_t^T \right\rangle_F$. We consider a subset of the features employed in that previous work: $\left\{\sum_t x_t^2, \sum_t \dot{x}_t^2, \sum_t \ddot{x}_t^2, \sum_t y_t^2, \sum_t \dot{y}_t^2, \sum_t \ddot{y}_t^2\right\}$, and denote the corresponding cost function weights as: $\{w_{x,x}, w_{\dot{x},\dot{x}}, w_{\ddot{x},\ddot{x}}, w_{y,y}, w_{\dot{y},\dot{y}}, w_{\ddot{y},\ddot{y}}\}$. We note that demonstrations cleaned (§4.1) using either of our procedures (or both) Pareto dominate the original demonstrations. We expect this relationship to frequently hold for observation noise and model misspecification in imitation learning tasks.

We compare against MaxEnt IRL apprenticeship learning for LQR tasks (Ziebart et al., 2012), which employs the optimal trajectory for the cost function learned via MaxEnt IRL. Rather than learning confidences or significance weights (§2.4), we employ additional data cleaning via demonstration selection in §4.5 using an auxiliary performance metrics: the task completion time. To evaluate MinSub IOC, we employ sum-aggregated (8), relative subdominance (6) minimization in Algorithm 1, so that short and long distance pointing tasks more equally contribute to the training objective.

4.3. Learned Cost Function Weights

The cost weights learned by each approach are shown in Figure 6. We first focus on the orthogonal y dimension of the trajectories. MaxEnt IRL seeks to fit to the demonstration data by maximizing its likelihood, and therefore learns appropriate weights for $w_{y,y}$, $w_{\dot{y},\dot{y}}$, and $w_{\ddot{y},\ddot{y}}$ for the demonstrated variance of those features. In contrast, all demonstrations are trivially dominated with $\epsilon > 0$ values for $w_{y,y}$, so MinSub IOC does not optimize further. Similarly, MaxEnt IRL learns a large value for $w_{\ddot{x},\ddot{x}}$. However, since dominating demonstrations in terms of squared velocity often implies dominance in terms of squared acceleration, MinSub IOC focuses on the trade-off between $w_{x,x}$ and $w_{\dot{x},\dot{x}}$ with minimal weight learned for $w_{\ddot{x},\ddot{x}}$. This is illustrated in Figure 7 by the learned optimal trajectory and margin boundaries for a single task.

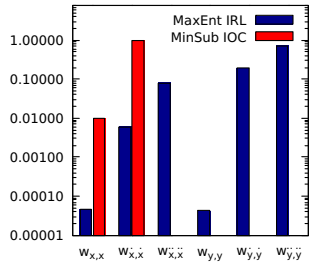

Figure 6. Learned cost weights (log scale, normalized) for MaxEnt IRL and MinSub IOC.

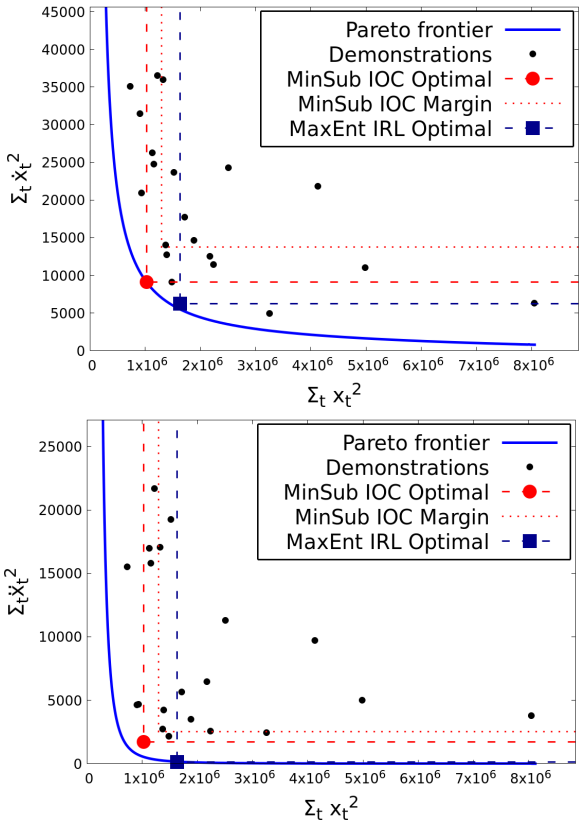

Figure 7. Demonstrations and Pareto frontier in the space of features for the pairs: $\sum_t \dot{x}_t^2$ (y axis) and $\sum_t x_t^2$ (x axis) in the top plot; $\sum_t \ddot{x}_t^2$ (y axis) and $\sum_t x_t^2$ (x axis) in the bottom plot. The pointing task corresponds to Figures 4 and 5.

4.4. Demonstration Loss Analysis

Motivated by the decomposition of the L^1 distance (Theorem 4), we compare the average subdominance, demonstration suboptimality, and L^1 distance of MinSub IOC and MaxEnt IRL on test data in Figure 8. The differences in MinSub IOC and MaxEnt IRL training objectives are apparent from this analysis: the MinSub IOC policy generally has much lower subdominance, while the MaxEnt IRL policy provides lower suboptimality for both learned MinSub IOC weights. Loss analysis using the MaxEnt IRL weights illustrates additional weaknesses of MaxEnt IRL. As shown, the learned weights emphasize suboptimality and deemphasize subdominance, reducing alignment with higher quality demonstrations. Additionally, though trained so that expected features match the mean demonstration features, the mode of the MaxEnt IRL distribution differs more greatly from demonstrations than the MinSub IOC policy on all three loss measures.

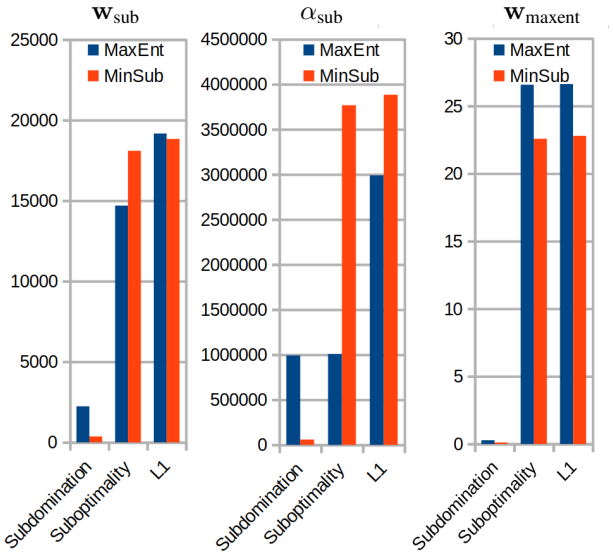

Figure 8. The subdominance ($\beta = \mathbf{0}$), suboptimality, and L^1 feature distance weighted using the MinSub cost weights $\mathbf{w}_{\text{sub}}$ (left) and α_{sub} values (center), and MaxEnt cost weights $\mathbf{w}_{\text{maxent}}$ (right).

The lower subdominance of MinSub IOC corresponds to more frequent Pareto dominance, with MinSub IOC achieving 78% superhuman behavior and MaxEnt IRL only achieving 50% superhuman behavior. In other words, for 28% more test demonstrations, no possible weights $\tilde{\mathbf{w}}^{(i)}$ exist that make the demonstration lower cost than the produced behavior from MinSub IOC compared to MaxEnt IRL.

4.5. Data Cleaning Impact on Superhuman Percentile

We next analyze the impact of additional data cleaning, in the form of training demonstration selection, on the superhuman percentile of learned behavior. We remove various

percentages of the training trajectories with the longest durations (task completion time) for each task. Though we argue that completion time is an adequate (or better) surrogate of learned data selection criteria (§2.4) for this task, such demonstration quality signals are not generally available for more complex tasks in which completion time may be one of many competing cost features.

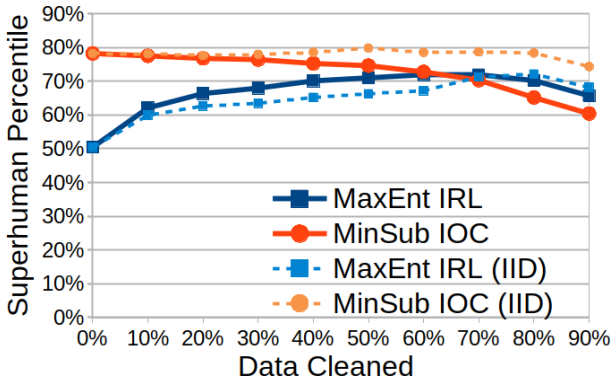

Figure 9. Pareto dominance over withheld test data when demonstrations with longer durations are removed from training data only (solid lines) and from both training and testing data (dashed lines).

Figure 9 shows that this data cleaning improves MaxEnt IRL as it learns from a pool of higher quality demonstrations with fewer large $\sum_t x_t^2$ terms. The sharpest increase is when the initial lowest quality demonstrated are removed, reflecting the strong sensitivity of the MaxEnt IRL approach to outliers. In contrast, data cleaning slowly degrades MinSub IOC as it increasingly removes support vectors that are representative of testing data. At the 70% level of cleaning and above, MaxEnt IRL's performance exceeds MinSub IOC. However, when the testing data is similarly cleaned, providing an IID learning problem, MinSub IOC's performance advantage over MaxEnt IRL remains for all levels of cleaning. Furthermore, for all levels of cleaning, MaxEnt IRL is unable to achieve MinSub IOC's performance on the uncleaned demonstration dataset.

4.6. Sensitivity to Sample Noise

We lastly investigate robustness to sample noise by measuring how learning from single pointing tasks and from a large training set differ. We learn a cost function for each of the 100 testing set tasks and evaluate the suboptimality of the resulting behavior compared to the cost function learned from the entire training set (200 tasks). A scatterplot with the suboptimality for each method is shown in Figure 10.

The correlation in suboptimality across methods is moderate ($r = 0.5$) for the entire set, but weak ($r = 0.3$) when the rightmost outlier is removed. This weak or moderate correlation is due to the sensitivity of each method to different types of noise: suboptimality for MaxEnt IRL and

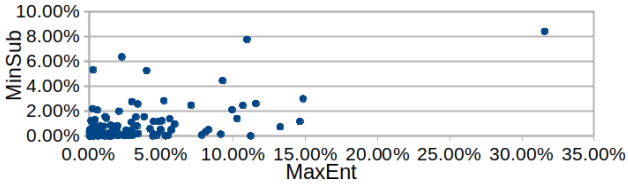

Figure 10. The relative suboptimality of the behavior learned from a single task relative to the behavior produced from 200 tasks: $\text{subopt}_{\mathbf{w}_{\text{full}}}(\xi_{\text{single}}, \xi_{\text{full}}) / \text{cost}_{\mathbf{w}_{\text{full}}}(\xi_{\text{full}})$.

subdominance for MinSub IOC. On average, MinSub IOC is 1.03% suboptimal and MaxEnt IRL is 3.68% suboptimal. This indicates that MaxEnt IRL is over three times more sensitive to sample noise than MinSub IOC.

5. Discussion & Future Work

The variability of human demonstration quality often poses significant challenges that prevent existing imitation learning methods from producing behaviors that reliably match or exceed expert human performance. We argue that outperforming all demonstrations—or minimizing the degree to which this is not achieved—is a better objective for guiding imitation than objectives based on averages over demonstrations. Our approach is more discriminative and avoids the daunting task of understanding human biases—either to construct generative noise models or to automatically rank demonstrations. Our margin-based approach provides useful generalization guarantees for outperforming human behavior under relaxations of classical inverse reinforcement learning assumptions that more realistically allow cost function weights to vary for each demonstration.

We investigate a smooth optimal control problem with deterministic dynamics using a linearly-parameterized cost function in this paper. Extensions to discrete or non-smooth continuous control tasks, stochastic dynamics, and/or nonlinear cost feature functions are important areas for future investigation. Additionally, we assume the dynamics of the decision process are known, or at least can be simulated, and that cost features are provided to the imitation learner. Appropriately incorporating dynamics estimation and cost feature representation learning into our imitation learning approach while maintaining useful generalization guarantees is an important topic of future work. We plan to explore these future directions on more complex decision processes, including Atari (Bellemare et al., 2013) and OpenAI Gym (Brockman et al., 2016) testbeds.

Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant Nos. 1652530, 1838770, and 1939743.

References

Abbeel, P. and Ng, A. Y. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning*, pp. 1–8, 2004.

Amos, B., Rodriguez, I., Sacks, J., Boots, B., and Kolter, Z. Differentiable MPC for end-to-end planning and control. In *Advances in Neural Information Processing Systems*, pp. 8289–8300, 2018.

Armstrong, S. and Mindermann, S. Occam's razor is insufficient to infer the preferences of irrational agents. *Advances in Neural Information Processing Systems*, pp. 5603–5614, 2018.

Babes, M., Marivate, V. N., Subramanian, K., and Littman, M. L. Apprenticeship learning about multiple intentions. In *International Conference on Machine Learning*, pp. 897–904, 2011.

Baker, C. L., Tenenbaum, J. B., and Saxe, R. R. Goal inference as inverse planning. In *Annual Meeting of the Cognitive Science Society*, pp. 779–784, 2007.

Bellemare, M. G., Naddaf, Y., Veness, J., and Bowling, M. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

Bıyık, E. and Sadigh, D. Batch active preference-based learning of reward functions. In *Conference on Robot Learning*, volume 87, pp. 519–528. PMLR, 2018.

Bıyık, E., Losey, D. P., Palan, M., Landolfi, N. C., Shevchuk, G., and Sadigh, D. Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences. *The International Journal of Robotics Research*, 41(1):45–67, 2022.

Bobu, A., Scobee, D. R., Fisac, J. F., Sastry, S. S., and Dragan, A. D. Less is more: Rethinking probabilistic models of human behavior. In *Proceedings ACM/IEEE International Conference on Human-Robot Interaction*, pp. 429–437, 2020.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. OpenAI gym. *arXiv preprint arXiv:1606.01540*, 2016.

Brown, D., Goo, W., Nagarajan, P., and Niekum, S. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. In *International Conference on Machine Learning*, pp. 783–792. PMLR, 2019.

Brown, D. S., Goo, W., and Niekum, S. Better-than-demonstrator imitation learning via automatically-ranked demonstrations. In *Conference on Robot Learning*, pp. 330–359. PMLR, 2020.

Chen, L., Paleja, R., and Gombolay, M. Learning from suboptimal demonstration via self-supervised reward regression. *arXiv preprint arXiv:2010.11723*, 2020.

Crammer, K. and Singer, Y. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2(Dec):265–292, 2001.

Evans, O., Stuhlmüller, A., and Goodman, N. D. Learning the preferences of ignorant, inconsistent agents. In *AAAI*, pp. 323–329, 2016.

Finn, C., Levine, S., and Abbeel, P. Guided cost learning: Deep inverse optimal control via policy optimization. In *International Conference on Machine Learning*, pp. 49–58. PMLR, 2016.

Fitts, P. M. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47(6):381, 1954.

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

Ibarz, B., Leike, J., Pohlen, T., Irving, G., Legg, S., and Amodei, D. Reward learning from human preferences and demonstrations in Atari. *Advances in Neural Information Processing Systems*, 31, 2018.

Kabán, A. Non-parametric detection of meaningless distances in high dimensional data. *Statistics and Computing*, 22(2):375–385, 2012.

Kalman, R. E. When is a linear control system optimal? In *Joint Automatic Control Conference*, pp. 1–15, 1963.

Kwon, M., Biyik, E., Talati, A., Bhasin, K., Losey, D. P., and Sadigh, D. When humans aren't optimal: Robots that collaborate with risk-aware humans. In *2020 15th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pp. 43–52. IEEE, 2020.

Liu, Y. Fisher consistency of multicategory support vector machines. In *Artificial Intelligence and Statistics*, pp. 291–298. PMLR, 2007.

Majumdar, A., Singh, S., Mandlekar, A., and Pavone, M. Risk-sensitive inverse reinforcement learning via coherent risk models. In *Robotics: Science and Systems*, 2017.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of Machine Learning*. MIT press, 2018.

Myers, V., Biyik, E., Anari, N., and Sadigh, D. Learning multimodal rewards from rankings. In *Conference on Robot Learning*, pp. 342–352. PMLR, 2021.

Neu, G. and Szepesvári, C. Apprenticeship learning using inverse reinforcement learing and gradient methods. In *Uncertainty in Artificial Intelligence*, pp. 295–302, 2007.

Newell, A. *Unified Theories of Cognition*. Harvard University Press, 1994.

Ng, A. Y. and Russell, S. J. Algorithms for inverse reinforcement learning. In *International Conference on Machine Learning*, pp. 663–670, 2000.

Novoseller, E., Wei, Y., Sui, Y., Yue, Y., and Burdick, J. Dueling posterior sampling for preference-based reinforcement learning. In *Conference on Uncertainty in Artificial Intelligence*, pp. 1029–1038. PMLR, 2020.

Osa, T., Pajarinen, J., Neumann, G., Bagnell, J. A., Abbeel, P., Peters, J., et al. An algorithmic perspective on imitation learning. *Foundations and Trends® in Robotics*, 7 (1-2):1–179, 2018.

Poljak, B. and Tsypkin, Y. Z. Pseudogradient adaptation and training algorithms. *Automation and Remote Control*, 34:45–67, 1973.

Pomerleau, D. A. ALVINN: An autonomous land vehicle in a neural network. In *Advances in Neural Information Processing Systems*, pp. 305–313, 1989.

Ramachandran, D. and Amir, E. Bayesian inverse reinforcement learning. In *International Joint Conference on Artificial Intelligence*, volume 7, pp. 2586–2591, 2007.

Ratliff, N. D., Bagnell, J. A., and Zinkevich, M. A. Maximum margin planning. In *International Conference on Machine Learning*, pp. 729–736, 2006.

Reddy, S., Dragan, A., and Levine, S. Where do you think you're going?: Inferring beliefs about dynamics from behavior. *Advances in Neural Information Processing Systems*, 31:1454–1465, 2018.

Ross, S., Gordon, G., and Bagnell, D. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 627–635, 2011.

Sadigh, D., Dragan, A. D., Sastry, S., and Seshia, S. A. Active preference-based learning of reward functions. In *Robotics: Science and Systems*, 2017.

Saw, J. G., Yang, M. C., and Mo, T. C. Chebyshev inequality with estimated mean and variance. *The American Statistician*, 38(2):130–132, 1984.

Shah, R., Gundotra, N., Abbeel, P., and Dragan, A. On the feasibility of learning, rather than assuming, human biases for reward inference. In *International Conference on Machine Learning*, pp. 5670–5679. PMLR, 2019.

Simon, H. A. *Models of bounded rationality: Empirically grounded economic reason*, volume 3. MIT press, 1997.

Spencer, J., Choudhury, S., Venkatraman, A., Ziebart, B., and Bagnell, J. A. Feedback in imitation learning: The three regimes of covariate shift. *arXiv preprint arXiv:2102.02872*, 2021.

Sun, W., Vemula, A., Boots, B., and Bagnell, D. Provably efficient imitation learning from observation alone. In *International Conference on Machine Learning*, pp. 6036–6045, 2019.

Swamy, G., Choudhury, S., Bagnell, J. A., and Wu, S. Of moments and matching: A game-theoretic framework for closing the imitation gap. In *International Conference on Machine Learning*, pp. 10022–10032, 2021.

Syed, U. and Schapire, R. E. A game-theoretic approach to apprenticeship learning. In *Advances in Neural Information Processing Systems*, pp. 1449–1456, 2008.

Tangkaratt, V., Han, B., Khan, M. E., and Sugiyama, M. Variational imitation learning with diverse-quality demonstrations. In *International Conference on Machine Learning*, pp. 9407–9417. PMLR, 2020.

Tangkaratt, V., Charoenphakdee, N., and Sugiyama, M. Robust imitation learning from noisy demonstrations. In *International Conference on Artificial Intelligence and Statistics*, 2021.

Taskar, B., Guestrin, C., and Koller, D. Max-margin Markov networks. *Advances in Neural Information Processing Systems*, 16:25–32, 2003.

Tsochantaridis, I., Hofmann, T., Joachims, T., and Altun, Y. Support vector machine learning for interdependent and structured output spaces. In *International Conference on Machine Learning*, pp. 104–111, 2004.

Vapnik, V. and Chapelle, O. Bounds on error expectation for support vector machines. *Neural computation*, 12(9): 2013–2036, 2000.

Wang, Y., Xu, C., and Du, B. Robust adversarial imitation learning via adaptively-selected demonstrations. In *International Joint Conference on Artificial Intelligence*, pp. 3155–3161, 2021a.

Wang, Y., Xu, C., Du, B., and Lee, H. Learning to weight imperfect demonstrations. In *International Conference on Machine Learning*, pp. 10961–10970, 2021b.

Wolpert, D. M., Ghahramani, Z., and Jordan, M. I. An internal model for sensorimotor integration. *Science*, 269 (5232):1880–1882, 1995.

Wu, Y.-H., Charoenphakdee, N., Bao, H., Tangkaratt, V., and Sugiyama, M. Imitation learning from imperfect demonstration. In *International Conference on Machine Learning*, pp. 6818–6827. PMLR, 2019.

Zhang, S., Cao, Z., Sadigh, D., and Sui, Y. Confidence-aware imitation learning from demonstrations with varying optimality. In *Advances in Neural Information Processing Systems*, pp. 12340–12350, 2021.

Zhi-Xuan, T., Mann, J., Silver, T., Tenenbaum, J., and Mansinghka, V. Online Bayesian goal inference for boundedly rational planning agents. *Advances in Neural Information Processing Systems*, pp. 19238–19250, 2020.

Ziebart, B., Dey, A., and Bagnell, J. A. Probabilistic pointing target prediction via inverse optimal control. In *ACM International Conference on Intelligent User Interfaces*, pp. 1–10, 2012.

Ziebart, B. D., Maas, A., Bagnell, J. A., and Dey, A. K. Maximum entropy inverse reinforcement learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, pp. 1433–1438, 2008.

A. Proofs

A.1. Domination and Subdominance Properties

Proof of Theorem 1. The first relationship,

$$\mathbf{f}(\xi) \preceq \mathbf{f}(\tilde{\xi}_i) \iff f_k(\xi) \leq f_k(\tilde{\xi}_i), \forall k,$$

is the definition of Pareto dominance. Under the assumption that $f_k(\xi) \geq 0, \forall k$, multiplication by all positive weights and addition maintains the inequality:

$$\begin{aligned} &f_k(\xi) \leq f_k(\tilde{\xi}_i), \forall k \implies w_k f_k(\xi) \leq w_k f_k(\tilde{\xi}_i)\ \forall k, w_k \geq 0, \\ &\implies \sum_{k=1}^{K} w_k f_k(\xi) \leq \sum_{k=1}^{K} w_k f_k(\tilde{\xi}_i)\ \forall w_k \geq 0, \\ &\implies \text{cost}_{\mathbf{w}}(\xi) \leq \text{cost}_{\mathbf{w}}(\tilde{\xi}_i)\ \forall \mathbf{w} \succeq \mathbf{0}. \end{aligned}$$

□

Proof of Theorem 3.

$$\begin{aligned} &\max_{\mathbf{0} \preceq \mathbf{w} \preceq \alpha} \text{subopt}_{\mathbf{w}}(\xi, \tilde{\xi}) \\ &\overset{(a)}{=} \max_{\{w_k \in \{0, \alpha_k\}\}} \sum_{k=1}^{K} w_k \left(f_k(\xi) - f_k(\tilde{\xi}) \right) \\ &\overset{(b)}{=} \sum_{k=1}^{K} \max_{w_k \in \{0, \alpha_k\}} w_k \left(f_k(\xi) - f_k(\tilde{\xi}) \right) \\ &\overset{(c)}{=} \sum_{k=1}^{K} \left[\alpha_k \left(f_k(\xi) - f_k(\tilde{\xi}) \right) \right]_+ \\ &\overset{(d)}{=} \text{subdom}^{\Sigma}_{\alpha,0}(\xi, \tilde{\xi}) \end{aligned}$$

We expand the definition of suboptimality in (a), and since the function is linear in each w_k only its extreme values, $\{0, \alpha_k\}$ need be considered. After algebraic rearrangement (b) and employing the definition of the hinge function (c), we arrive at the definition of the subdominance (d). □

Lemma 9. *The L^1_α feature distance decomposes into the forward subdominance and the reverse subdominance:*

$$L^1_\alpha(\xi, \tilde{\xi}) = \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\xi, \tilde{\xi}) + \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\tilde{\xi}, \xi).$$

Proof. This follows from the basic hinge function identity: $|x - y| = [x - y]_+ + [y - x]_+$.

$$\begin{aligned} &\forall k, |f_k(\xi) - f_k(\tilde{\xi})| \qquad (14) \\ &\quad = [f_k(\xi) - f_k(\tilde{\xi})]_+ + [f_k(\tilde{\xi}) - f_k(\xi)]_+ \\ &\implies \sum_k \alpha_k |f_k(\xi) - f_k(\tilde{\xi})| \qquad (15) \\ &\quad = \sum_k \alpha_k \left([f_k(\xi) - f_k(\tilde{\xi})]_+ + [f_k(\tilde{\xi}) - f_k(\xi)]_+ \right) \end{aligned}$$

□

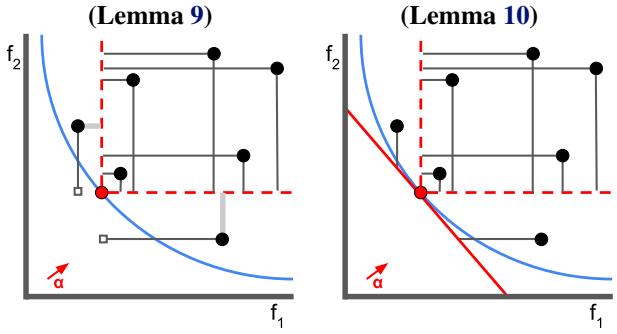

Figure 11. **Lemma 9:** The forward (thick gray lines) and reverse (black lines) subdominances additively form the L^1 loss: $L^1_\alpha(\xi, \tilde{\xi}) = \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\xi, \tilde{\xi}) + \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\tilde{\xi}, \xi)$. **Lemma 10:** Suboptimality (black lines) and reverse subdominance differ only for examples with positive forward subdominance, with resulting losses equivalent to "clipping" the losses of (a) at the optimal tangent plane (solid red line): $\text{subopt}_\alpha(\tilde{\xi}, \xi) = \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\tilde{\xi}, \xi) - \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\xi, \tilde{\xi})$.

Lemma 10. *The suboptimality is equal to the reverse subdominance minus the forward subdominance:*

$$\text{subopt}_\alpha(\tilde{\xi}, \xi) = \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\tilde{\xi}, \xi) - \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\xi, \tilde{\xi}).$$

Proof. This follows from the basic hinge function identity: $x - y = [x - y]_+ - [y - x]_+$

$$\begin{aligned} &\forall k, f_k(\tilde{\xi}) - f_k(\xi) \qquad (16) \\ &\quad = [f_k(\tilde{\xi}) - f_k(\xi)]_+ - [f_k(\xi) - f_k(\tilde{\xi})]_+ \\ &\implies \sum_k \alpha_k f_k(\tilde{\xi}) - f_k(\xi)| \qquad (17) \\ &\quad = \sum_k \alpha_k \left([f_k(\tilde{\xi}) - f_k(\xi)]_+ - [f_k(\xi) - f_k(\tilde{\xi})]_+ \right) \end{aligned}$$

□

Proof of Theorem 4. The result follows by applying Lemmas 9 and 10:

$$\begin{aligned} L^1_\alpha(\xi, \tilde{\xi}) &= \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\xi, \tilde{\xi}) + \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\tilde{\xi}, \xi) \\ &= \text{subopt}_\alpha(\tilde{\xi}, \xi) + 2\ \text{subdom}^{\Sigma}_{\alpha,\mathbf{0}}(\tilde{\xi}, \xi). \end{aligned}$$

□

Corollary 11. *A corresponding relative decomposition of the L^1_α feature distances exists when $f_k(\tilde{\xi}) > 0\ \forall k$,*

$$\textit{rel}L^1_\alpha(\xi, \tilde{\xi}) = \textit{relsubopt}_\alpha(\tilde{\xi}, \xi) + 2\ \textit{relsubdom}_\alpha(\tilde{\xi}, \xi),$$

in which the relative L^1 distance and relative suboptimality

are defined as:

$$relL^1_\alpha(\xi, \tilde{\xi}) = \sum_k \alpha_k \frac{|f_k(\xi) - f_k(\tilde{\xi})|}{f_k(\tilde{\xi})} \text{ and} \tag{18}$$

$$relsubopt_\alpha(\tilde{\xi}, \xi) = \sum_k \alpha_k \frac{f_k(\xi) - f_k(\tilde{\xi})}{f_k(\tilde{\xi})}. \tag{19}$$

Proof. Lemma 9 and Lemma 10 are easily extended to the relative case by dividing both sides of (14) and (16) by $f_k(\tilde{\xi})$, which then carries through for both lemmas and the corollary when the reverse relative subdominance is also relative to the demonstrated trajectory features. □

A.2. Generalization Bounds

Proof of Theorem 6. The leave-one-out cross validation error (i.e., failing to Pareto dominate or achieve superhuman performance) on n demonstrations is an unbiased estimate of the loss with $(n-1)$ demonstrations, which upper bounds the loss with n demonstrations (and is an *almost unbiased* estimate) under *IID* assumptions (Vapnik & Chapelle, 2000; Mohri et al., 2018).

Consider the model trained using all n demonstrations. By definition, non-support vectors are Pareto dominated by the induced behavior in this model, incurring no error. If one of these non-support vectors is removed from the training set, the learned model does not change. Thus, each non-support vector contributes no error to the overall leave-one-out cross validation error. Each support vector may, in the worst case, incur an error when removed from the training set during leave-one-out cross validation. Together, these provide the bound. □

Proof of Theorem 7. Letting $\tilde{\mu}$ and $\tilde{\sigma}$ represent the sample mean and sample variance of the subdominance measurements of N IID samples $\{X_1, X_2, \ldots, X_N\}$ where $X_i = \text{[rel]subdom}^{[\Sigma]}_{\alpha,\mathbf{1}}(\xi^*(\mathbf{w}), \tilde{\xi}_i)$ drawn from an unknown distribution (formally, with zero probability that all samples are equal to zero), we start from Kabán (2012)'s simplified bound for a new sample based on a finite sample Chebyshev's inequality (Saw et al., 1984) for $\epsilon > 0$:

$$\begin{aligned}
&P(|X - \tilde{\mu}| \geq \epsilon\tilde{\mu}) \leq \frac{N^2-1}{N^2}\frac{1}{\epsilon^2}\frac{\tilde{\sigma}^2}{\tilde{\mu}^2} + \frac{1}{N} \\
\overset{(a)}{\Longrightarrow}\ &P(X - \tilde{\mu} \geq \epsilon\tilde{\mu}) \leq \frac{N^2-1}{N^2}\frac{1}{\epsilon^2}\frac{\tilde{\sigma}^2}{\tilde{\mu}^2} + \frac{1}{N} \\
\overset{(b)}{\Longrightarrow}\ &P(X - \tilde{\mu} \geq 1 - \tilde{\mu}) \leq \frac{N^2-1}{N^2}\frac{\tilde{\sigma}^2}{(1-\tilde{\mu})^2} + \frac{1}{N} \\
\overset{(c)}{\Longrightarrow}\ &P(X \geq 1) \leq \frac{N^2-1}{N^2}\frac{\tilde{\sigma}^2}{(1-\tilde{\mu})^2} + \frac{1}{N},
\end{aligned}$$

where: (a) follows from taking just one of the two-sided bounds; (b) is obtained by substituting $\epsilon = \frac{1}{\tilde{\mu}} - 1 = \frac{1-\tilde{\mu}}{\tilde{\mu}} > 0$; and (c) results from adding the sample mean to both sides. Then, $P(X \geq 1) \geq P(\bigcup_k f_k(\xi) \geq f_k(\tilde{\xi}))$, thus providing the overall bound. □

A.3. Fisher Consistency

Definition 12. *A learner is* **Fisher consistent** *if it learns to make Bayes optimal decisions when trained from any population distribution using a fully expressive function class (e.g., all measurable functions).*

Remark 13. *Maximum margin planning (Ratliff et al., 2006) inherits the Fisher inconsistency of the Crammer & Singer (2001) multiclass support vector machine (Liu, 2007) and is therefore also not Fisher consistent.*

This inconsistency arises when demonstrations are sufficiently noisy—for example, when no majority action exists under the distribution of demonstrations. More concretely, maximum margin planning (MMP) is Fisher inconsistent for the Markov decision process in Figure 12 when the demonstration distribution is $P(a_1) = 0.4$, $P(a_2) = 0.3$, and $P(a_3) = 0.3$, in which case $cost(s_1) = cost(s_2) = cost(s_3) = 0$ minimizes the MMP training objective, but does not induces the Bayes optimal policy (under 0-1 loss).

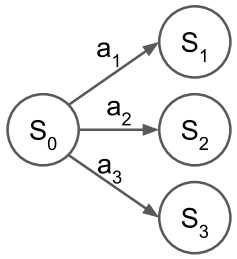

Figure 12. A simple one timestep Markov decision process with three actions.

Proof of Theorem 8. We first note that optimization of ϕ completely subsumes the optimization of α, leaving:

$$\min_{\mathbf{w} \geq \mathbf{0}} \sup_{\phi : \nabla\phi > \mathbf{0}} \sum_{i=1}^{N} \underbrace{\max_k \left[\phi_k(f_k(\xi^*(\mathbf{w}))) - \phi_k(f_k(\tilde{\xi}_i)) + 1\right]_+}_{\text{subdom}_{\phi,\mathbf{1}}(\xi^*(\mathbf{w}), \tilde{\xi}_i)}.$$

We assume,[4] having been selected from a numerical optimization procedure or noisily produced from a human demonstrator, that $f_k(\xi^*(\mathbf{w})) \neq f_k(\tilde{\xi}_i)$. The optimal $\{\phi_k\}$ for some $\xi(\mathbf{w})$ has two cases. If behavior $\tilde{\xi}_i$ is not dominated in k (i.e., $f_k(\tilde{\xi}_i) < f_k(\xi^*(\mathbf{w}))$), $\text{subdom}^k_{\phi_k,1}(\xi^*(\mathbf{w}), \tilde{\xi}_i)$ is minimized to 1 by choosing $\sup_{\phi_k} \phi_k(f_k(\tilde{\xi}_i)) < \phi_k(f_k(\xi^*(\mathbf{w})))$.

[4] Alternatively, the Hamming loss (or similar loss function) can be used for the subdominance margin rather than a fixed value.

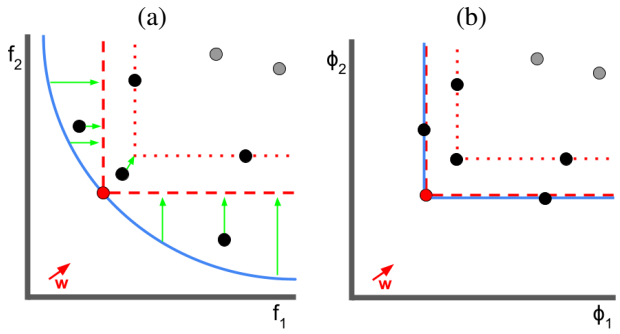

Figure 13. Feature mapping ϕ transforms the original feature space (a) based on the green arrows so that all Pareto-dominated demonstrations have zero subdominance and all non-dominated demonstrations have subdominance of one (b).

For the remaining behaviors, $f_k(\tilde{\xi}_i) > f_k(\xi^*(\mathbf{w}))$, ϕ_k can be made arbitrarily large at $f_k(\tilde{\xi})$, providing a subdominance of 0 for those examples. We have ignored the requirement that ϕ_k be strictly increasing in f_k across demonstrations, but this added constraint does not change the solution. Given this, the training objective for choosing the optimal behavior weights $\mathbf{w}$ is equivalent to the frequency of non-Pareto-dominated demonstrations in the training set:

$$\min_{\mathbf{w}\geq\mathbf{0}} \frac{1}{N}\sum_{i=1}^{N} I\left[\mathbf{f}(\xi^*(\mathbf{w})) \not\preceq \mathbf{f}(\tilde{\xi}_i)\right],$$

using I to denote a binary-valued indicator function. As $N \to \infty$, the training set converges to the population distribution and the objective is equivalent to maximizing the probability of Pareto dominance on the population distribution. □

B. Formulation for Stochastic Dynamics

For tasks with stochastic dynamics, we must reason about demonstrated policies, $\tilde{\pi} \in \Pi_{\text{dp}}$. Conceptually, if each demonstration corresponds with observing the entire stochastic policy of the demonstrator, e.g., $\tilde{\pi} : \mathcal{S} \to \Delta_{\mathcal{A}}$, then $\mathbf{f}(\tilde{\xi})$ can be simply replaced with $\mathbf{f}(\tilde{\pi}) \triangleq \mathbb{E}_{\xi\sim\tilde{\pi}}[\mathbf{f}(\xi)]$ in our subdominance definitions (5)-(8) and then demonstration policies $\tilde{\pi}$ can be used throughout the remainder of our formulation. Similarly, the optimal trajectory $\xi^*(\mathbf{w})$ is replaced by the optimal policy $\pi^*(\mathbf{w})$.

When trajectory samples $\tilde{\xi} \sim \tilde{\pi} \times \mathcal{T}$ are instead available, expected feature counts can be estimated from repeated samples from the same demonstrator: $\mathbf{f}(\tilde{\pi}) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{f}(\tilde{\xi})$. Developing methods if N is small (or a single trajectory) for each task is a topic worthy of future investigation.

C. Additional Experimental Results /Analyses

C.1. Cleaning Raw Demonstration Data

We provide examples of the impact of our two data cleaning procedures in this section. Figure 14 show the differences in position over time between raw and repetition-cleaned data on a single task. Removing initial state repetitions from demonstration trajectories reduces position-based features, such as $\sum_t x_t^2$, while not changing velocity-based features. Note that this procedure is imperfect: small initial movements may be present before the demonstrator responds to the revealed target. This phenomenon appears to be present for some demonstrations in Figure 5 and Figure 14.

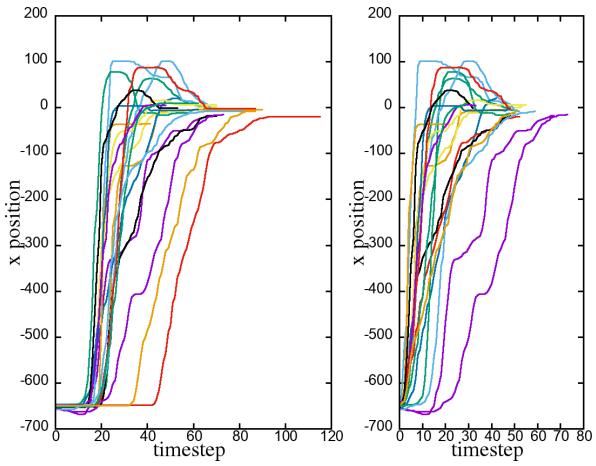

Figure 14. Demonstrations for a single task with initial and final repeated states included (left) and removed (right).

Figure 15 shows the differences in x and y position for data without and with single-timestep "jitters" removed. For this particular task, three demonstration trajectories have single timestep anomalous positions corresponding to the corner of the display. Removing these jitters significantly decreases both position-based features and velocity-based features.

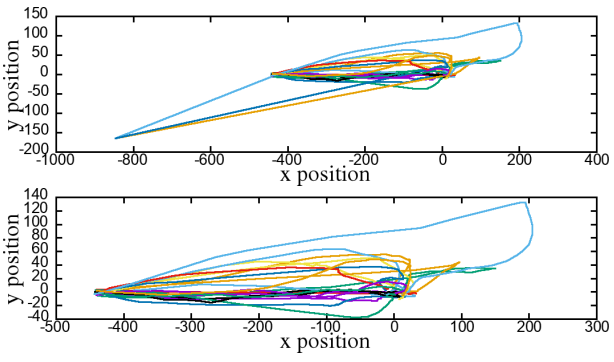

Figure 15. Demonstrations for a single task with single timestep "jitter" included (top) and removed (bottom).

We note that demonstrations cleaned using either or both

procedures Pareto dominate the corresponding raw demonstration trajectories in the salient cost features of interest. This shows that the noise being cleaned by these techniques do not increase subdominance, but would significantly impact methods based on suboptimality minimization.

C.2. Margin Comparisons for Absolute and Relative Subdominance Minimization

We show the differences between absolute subdominance and relative subdominance using a long-distance and a short-distance pointing task in Figure 16.

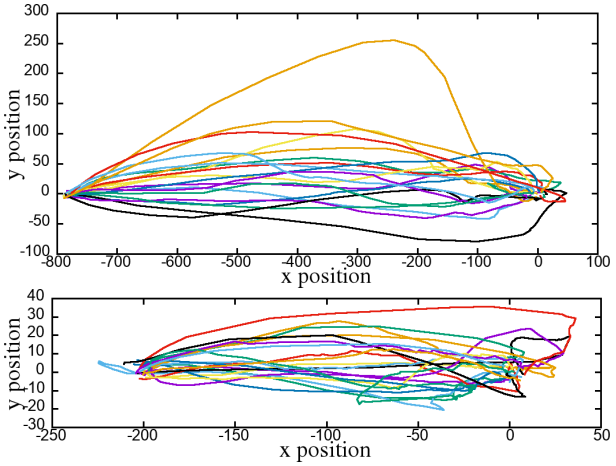

Figure 16. Twenty cursor trajectories for a long-distance (top) and a short-distance (bottom) pointing task.

The differences in margin boundaries for these two tasks are shown in Figure 17. These boundaries are extremely sensitive to the pointing task distance when using the absolute subdominance (left figures). Specifically, the long-distance pointing task has tight margin boundaries (relative to the demonstrations) and a small number of support vectors, while the short-distance pointing task has overly wide margin boundaries and many more support vectors. Though there are fewer of them, the support vectors for the long-distance task have an overly strong influence on the learned cost weights compared to short-distance support vectors.

The margin boundaries are much more similar between long-distance and short-distance pointing tasks using the relative subdominance (right figures). As a result, support vectors for both the short-distance and the long-distance pointing tasks have similar influence on the optimization of the learned cost weights.

Though we primarily emphasize the sensitivity of suboptimality minimization methods to lower quality demonstrations in this paper, those methods also tend to employ absolute differences in their suboptimality definitions and thus suffer from a sensitivity to demonstrations from tasks with inherently larger absolute costs.

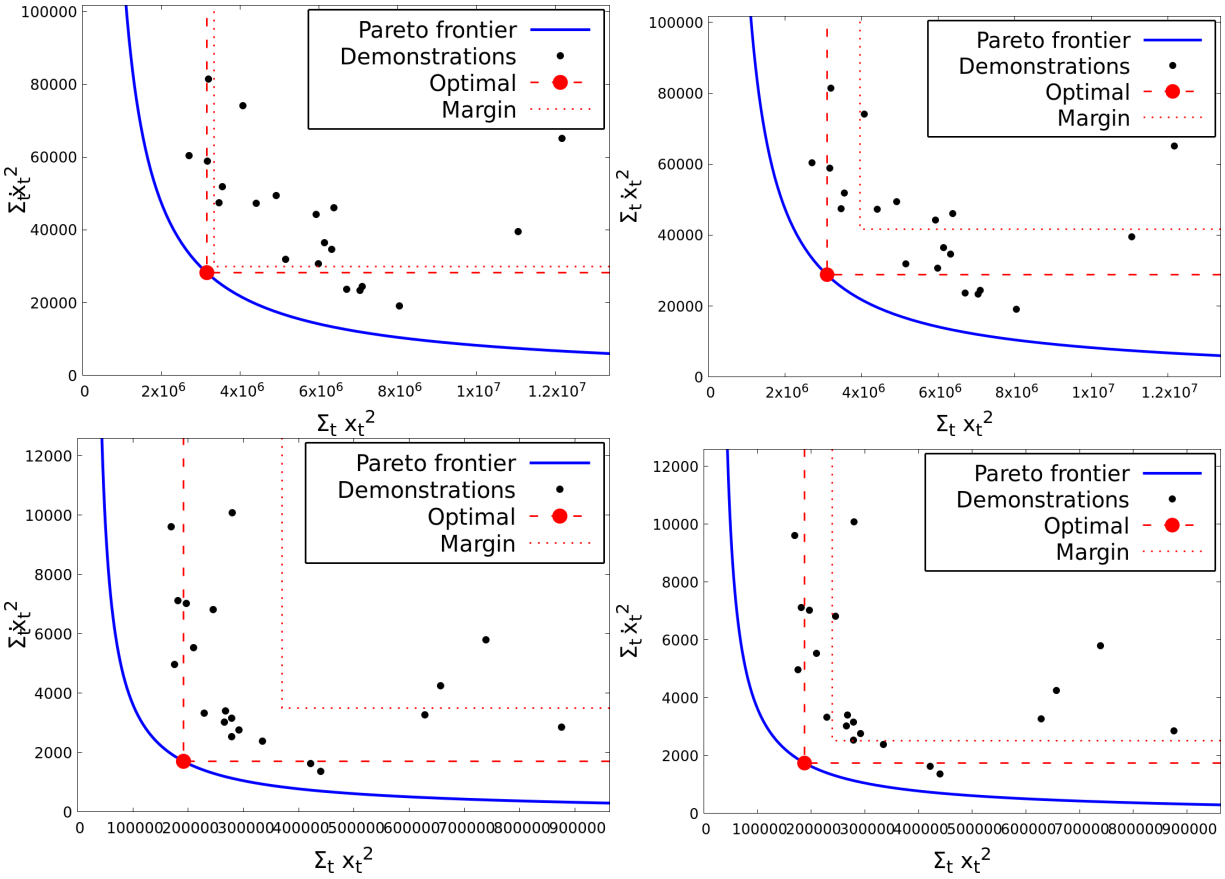

Figure 17. Pareto dominance and margin boundaries for absolute subdominance (left) and relative subdominance (right) on the long-distance (top) and short-distance (bottom) pointing tasks.