PREFGEO: GENERALIZING PREFERENCE LEARNING ACROSS CONTEXTS VIA GEOMETRIC REWARD FACTORIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Preference-based reinforcement learning is a popular paradigm for aligning AI models with human intent, but generalizing it to a new context, whether an unseen task or a new user, typically requires substantial additional feedback. In this work, we introduce PrefGeo, a geometric reward factorization framework for efficient preference learning across contexts. Unlike prior methods that fit a shared reward basis directly from preference data, PrefGeo recovers it from independently estimated context-specific rewards via regularized minimum-volume unmixing in a preference-induced Fisher geometry, capturing reusable preference structure. A new context is then adapted by updating only low-dimensional mixture weights. We evaluate PrefGeo on cross-task generalization in robotic manipulation, with both conventional policies and vision-language-action models, and on cross-user personalization in large language models and assistive human-robot interaction. PrefGeo consistently outperforms baselines in preference alignment and downstream task performance, especially when target feedback is scarce. Real-robot experiments and an assistive-feeding user study confirm its effectiveness in the real world. Project website: `https://prefgeo.github.io`.

1 INTRODUCTION

Preference-based reinforcement learning (PbRL) is a powerful paradigm for aligning AI models with human intent (Christiano et al., 2017; Lee et al., 2021b; Bıyık et al., 2024). However, scaling it across contexts remains challenging: PbRL typically learns context-specific reward models, e.g., per-task in robotic manipulation (Wang et al., 2025a) or per-user in large language model (LLM) personalization (Poddar et al., 2024). Consequently, generalizing to each new context (Figure 1a) necessitates collecting significant amount of additional preference data, which is impractical.

To mitigate this issue, one research direction replaces human annotations with synthetic feedback from LLMs (Wang et al., 2024; 2025c), which is scalable but is unreliable and expensive (Zheng et al., 2023). Another major direction aims to transfer preference knowledge across contexts by exploiting shared preference structures. For example, Hejna III & Sadigh (2023) explore meta-learning (Finn et al., 2017) in PbRL, PEARL (Liu et al., 2024b) propagates pseudo-preference labels via source-to-target optimal transport, and VPL (Poddar et al., 2024) learns latent user-conditioned models for pluralistic alignment. However, these approaches often require costly auxiliary training, such as learning a meta-preference reward or user prediction models, which limits their efficiency.

More recently, preference reward factorization (PRF) has emerged as a promising solution (Bose et al., 2025). PRF models context-specific preference rewards as mixtures of a shared low-dimensional reward basis, reducing adaptation to estimating mixture coefficients (Figure 1b). Existing approaches (Shenfeld et al., 2025; Bose et al., 2025; Lee et al., 2026a), however, often fail to recover a stable and reusable basis for two reasons (Figure 1c): i) *Factorization ambiguity:* the basis and coefficients are learned by maximizing preference likelihood on source preference data, coupling reward estimation with basis recovery. While enabling accurate source preference reconstructions, it does not provide an explicit criterion for identifying a compact basis that captures reusable preference structure for transfer. ii) *Geometry bias:* differences in reward parameters are typically measured in a Euclidean geometry, which treats all directions in reward-parameter space

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

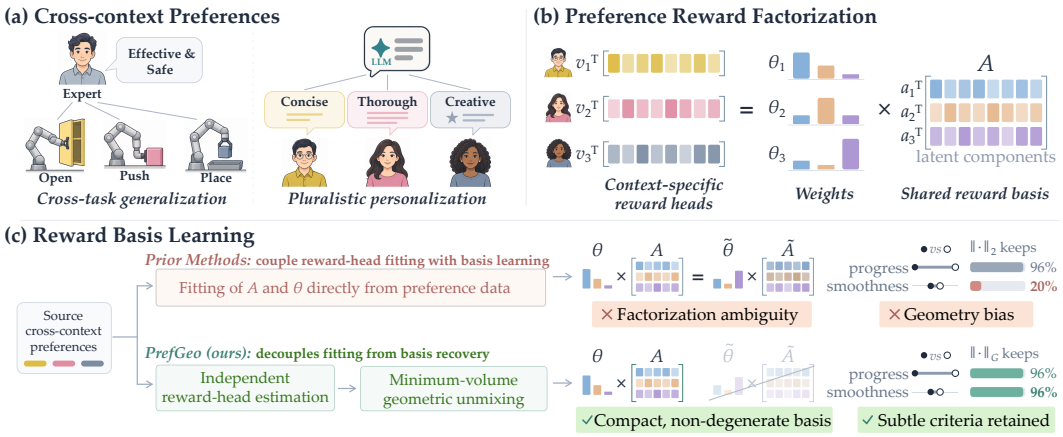


Figure 1: (a) We consider cross-task generalization and pluralistic personalization in cross-context preference learning. (b) PRF models each context-specific reward head as a mixture of a shared reward basis. (c) Prior PRF methods fit the factorization directly to preference data and suffer from *factorization ambiguity* and *geometry bias* (illustrative values obtained from Appendix A.3). In contrast, PrefGeo decouples reward-head estimation from basis recovery via geometric unmixing, yielding a reward basis for more stable and robust transfer.

equally, yet preference comparisons reveal them unevenly (Sadigh et al., 2017). Criteria along which compared behaviors differ only subtly may be suppressed by regularization or distorted by estimation noise (Hastie et al., 2009). Thus, basis recovery is shaped by how strongly the data express each criterion rather than by the preferences themselves (Ellis et al., 2024) (details in Appendix A).

To address these challenges, we introduce PrefGeo, a framework for efficient cross-context preference learning via geometric reward factorization. Rather than learning the reward basis (RB) vectors and their mixtures end-to-end from preference data, PrefGeo decouples context-specific reward estimation from RB recovery and casts the latter as a geometric unmixing problem: independently estimated source rewards are fixed points in a shared reward-parameter space, and the shared reward components are learned as simplex vertices whose convex combinations approximate these points.

PrefGeo operates at three stages as shown in Figure 2: i) *Predict:* we independently estimate a reward head for each source context. The source preference data also induces a common Fisher information metric (Amari & Nagaoka, 2000), which measures reward differences through their effect on preference predictions and regularizes the heads. ii) *Unmix:* under this metric, the fixed reward heads are decomposed into a reusable reward basis via a regularized minimum-volume criterion, which resolves factorization ambiguity by explicitly selecting the most compact, non-degenerate RB among those that reconstruct the source rewards, while the preference-induced geometry reduces geometry bias. iii) *Remix:* with the basis frozen, we adapt to a new context by initializing the mixture coefficients at the source mean and updating them from few-shot target feedback. Our key contributions include:

- We introduce PrefGeo, a geometric reward factorization framework that enables efficient few-shot preference learning in both cross-task generalization and cross-user personalization.
- We recast PRF as a decoupled, geometric unmixing problem, solved by a regularized minimum-volume criterion in preference-induced Fisher information geometry. This reduces factorization ambiguity and bias, yielding a reward basis that captures latent cross-context preference structure for more stable and robust transfer.
- We validate PrefGeo on six benchmarks spanning robotic manipulation with classic neural network policies and vision-language-action models (VLA), and LLM personalization, together with a real-world user study on assistive feeding and real-robot VLA manipulation. PrefGeo consistently outperforms baselines in improving task performance in unseen scenarios and preference alignment for unseen users, particularly when feedback in the target context is limited.

2 Preliminaries and Related Work

Preference Learning. PbRL learns a reward model aligned with human preference from pairwise feedback (Lee et al., 2021a). Let x denote an output of an AI model, such as a robot trajectory or an

LLM response. Given (x^A, x^B), the Bradley-Terry (BT) model (Bradley & Terry, 1952) defines

$$p_\psi(x^A, x^B) := P_\psi(x^A \succ x^B) = \sigma\big(R_\psi(x^A) - R_\psi(x^B)\big) \tag{1}$$

where σ is the logistic function and ψ is the reward model parameters. Given a dataset $\mathcal{D} = \{(x_i^A, x_i^B, y_i)\}_{i=1}^N$, with $y_i = 1$ indicating a preference for x_i^A, $y_i = 0$ for x_i^B, and $y_i = 0.5$ for equal preference, the reward model $R_\psi(x)$ is trained by minimizing the BT negative log-likelihood:

$$\mathcal{L}^{\mathrm{BT}}_{\psi;\mathcal{D}} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log p_{\psi,i} + (1 - y_i)\log(1 - p_{\psi,i})\right] \tag{2}$$

where $p_{\psi,i} := p_\psi(x_i^A, x_i^B)$. The learned reward model is used for downstream policy optimization.

Reward Factorization for Cross-Context Preference Transfer. Reward models in PbRL are typically learned for individual contexts, limiting their generalization to new tasks or users (Lee et al., 2021b). Preferences, however, often share a common underlying structure across contexts, such as motion smoothness and approach progress across manipulation tasks (Barreto et al., 2026). This motivates preference-transfer methods that reuse source knowledge to accelerate adaptation. PRF offers an efficient way through low-rank factorization of linear reward heads v over a common nonlinear behavior representation $z(x) \in \mathbb{R}^D$ (Bose et al., 2025; Shenfeld et al., 2025; Lee et al., 2026a). A context-specific preference reward $R_c(x)$ is modeled as

$$R_c(x) = v_c^\top z(x), \qquad v_c^\top = \theta_c^\top A, \tag{3}$$

where $A = [a_1, \ldots, a_K]^\top$ contains $K \ll D$ shared reward components a, each potentially capturing a distinct preference criterion, and θ_c specifies their context-specific mixture weights. Both v_c and a_k lie in a common D-dimensional reward-parameter space, and PRF assumes that preferences vary across contexts only within the K-dimensional subspace spanned by A. A new context therefore has K rather than D free parameters, so once A is learned, adaptation reduces to estimating θ_c from a handful of comparisons, without fitting a meta reward head (Hejna III & Sadigh, 2023) or a separate user-inference model (Poddar et al., 2024).

Factorization Ambiguity. Effective transfer requires a reward basis that captures reusable cross-context preference structure. Existing methods, however, typically learn A and θ_c directly from preference data via the BT preference loss, which constrains them only through reward heads v_c. Distinct factorizations satisfying $A^\top\theta_c = \widetilde{A}^\top\widetilde{\theta}_c$ for source contexts therefore yield identical source predictions while providing different reward bases for transfer (Donoho & Stodden, 2003). PrefGeo mitigates this ambiguity by separating context-specific reward estimation from basis recovery and explicitly favoring a compact, non-degenerate basis through minimum-volume geometric unmixing, inspired by identifiability results in blind unmixing (Lin et al., 2015).

Geometry Bias. Another challenge in PRF is that comparison data constrain different reward directions unequally. If source trajectory pairs differ in task progress but have nearly identical smoothness scores, the comparisons carry little information about the smoothness weight, even when smoothness matters just as much to the underlying preference. Euclidean criteria then suppress that weight far more than the progress weight, and Euclidean distances between heads misjudge how much contexts actually differ along it. This imbalance stems from what is compared rather than from the feature distribution, so normalizing or decorrelating the features does not remove it (Ellis et al., 2024). Existing PRF methods nonetheless rely on Euclidean criteria, such as L2 regularization and distances between weight vectors, which treat all directions as equally informed.

PrefGeo instead measures reward-parameter changes through their effect on preference predictions, using the Fisher information of the BT model (Fisher, 1922; Amari & Nagaoka, 2000). With $\Delta z_m = z(x_m^A) - z(x_m^B)$ the representation difference of a source comparison, a head v predicts $\sigma(v^\top\Delta z_m)$, and its Fisher information $\mathcal{I}(v)$ weights each $\Delta z_m \Delta z_m^\top$ by $p_m(1 - p_m)$, which depends on v. Evaluating it at the neutral head $v = 0$, where every probability is $1/2$, gives a single metric $G = \mathcal{I}(0)$ shared by all contexts. For a change $\delta v = v - v'$ between two reward heads,

$$\|\delta v\|_G^2 = \delta v^\top G\, \delta v = \frac{1}{4M}\sum_{m=1}^{M}\left(\delta v^\top \Delta z_m\right)^2, \tag{4}$$

where M is the number of source comparisons. It measures how much δv changes the comparison logits $v^\top\Delta z_m$, so directions that comparisons rarely probe receive little weight (Appendix B). We use G for head regularization, basis reconstruction, and the adaptation prior.

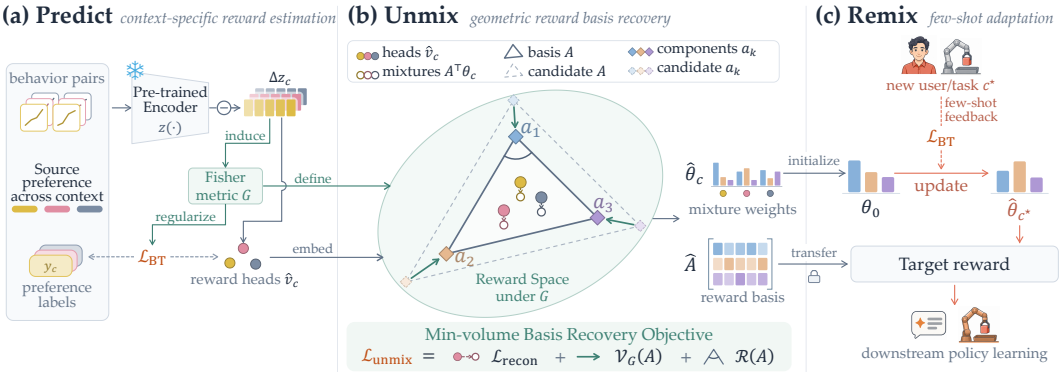


Figure 2: **Overview of PrefGeo.** (a) Predict: behaviors are encoded by a frozen pre-trained encoder, and a reward head $\hat{v}_c$ is fitted independently for each source context from its preference labels. The comparison features Δz_c induce a Fisher metric G, which regularizes the fitting and defines the geometry of the reward space. (b) Unmix: the heads are fixed points in this space, and a shared basis $\hat{A}$ is recovered by shrinking a candidate simplex around them under $\mathcal{L}_{\text{unmix}}$ (Eq. 9); vertices are the components a_k, and each head's position gives its mixture weights $\hat{\theta}_c$. (c) Remix: for a new context, the basis is frozen and only the weights are updated from few-shot feedback, initialized at the source mean.

3 Methodology

We consider cross-context preference learning, where a context c is either a task or a user: in cross-task generalization, a fixed expert's preferences are transferred across tasks; in cross-user personalization, different users hold pluralistic preferences within a fixed task. We assume access to several source contexts whose preferences share some underlying evaluation factors, such as those across a family of robotic manipulation tasks or a population of language model users. Each source context $c \in \mathcal{C}_{\text{src}}$ provides a preference dataset $\mathcal{D}_c = \{(x^A_{c,i}, x^B_{c,i}, y_{c,i})\}_{i=1}^{n_c}$. Our goal is to efficiently adapt to an unseen target context $c^\star \notin \mathcal{C}_{\text{src}}$, producing behavior aligned with its preferences using little target feedback. As shown in Figure 2, PrefGeo achieves this through the geometric *Predict-Unmix-Remix* procedure detailed below.

3.1 Predict: Context-Specific Reward Estimation

Preferences depend on behavioral characteristics, such as how a robot moves or how a response is expressed (Sadigh et al., 2017; Barreto et al., 2026). We therefore first encode each behavior x into a shared representation $z(x) \in \mathbb{R}^D$ with a pre-trained encoder that is kept frozen, allowing rewards to depend nonlinearly on the raw input while remaining linear in a common feature space. PrefGeo does not require a specialized encoder; we follow established practices for behavior feature extraction. For conventional robotic control, we use an MLP trajectory encoder trained with a reconstruction loss (Co-Reyes et al., 2018; Fabisch & Kirchner, 2021). For LLMs and VLAs, we follow Bose et al. (2025) to extract embeddings before the output layer of a pretrained, general reward model (Liu et al., 2024a; Lee et al., 2026b) (more details are in Appendix C).

Given $z(x)$, each comparison is summarized by its feature difference $\Delta z = z(x^A) - z(x^B)$. The differences pooled over all source contexts induce the common Fisher metric G in Eq. 4, which is computed once and shared by all subsequent steps. We then independently estimate a reward head v_c for each source context, defining $R_c(x) = v_c^\top z(x)$. Each head is fitted using its own preference dataset $\mathcal{D}_c$ by minimizing the BT loss in Eq. 2, regularized under G:

$$\hat{v}_c = \arg\min_{v \in \mathbb{R}^D} \mathcal{L}^{\text{BT}}_{v;\mathcal{D}_c} + \lambda_v \|v\|^2_G, \tag{5}$$

where $\|v\|^2_G$ is Eq. 4 with $\delta v = v - 0$, i.e., the G-distance of v from the neutral head, and λ_v controls the regularization strength.

3.2 Unmix: Geometric Reward Basis Recovery

Given source reward heads $\{\hat{v}_c\}$, we recover a shared reward basis A and mixture weights $\{\theta_c\}$ that explain their cross-context structure. We formulate this as geometric unmixing in reward-parameter

space under the common Fisher metric G: each source head is a fixed point, and K components $a_1, \ldots, a_K$ (the rows of A) are unknown vertices. Each head is approximated by a convex mixture: $\hat{v}_c \approx A^\top \theta_c$. Thus, the reconstructed rewards lie in the convex hull of the component vectors.

Reconstruction alone, however, cannot distinguish among multiple simplices that explain the same source rewards. We therefore minimize simplex volume alongside reconstruction error to favor a compact basis. To measure volume, we collect the simplex edges relative to the final vertex:

$$E(A) = \begin{bmatrix} (a_1 - a_K)^\top \\ \vdots \\ (a_{K-1} - a_K)^\top \end{bmatrix} \in \mathbb{R}^{(K-1)\times D}. \tag{6}$$

The matrix $E(A)GE(A)^\top$ contains their Fisher inner products, and its determinant is proportional to the squared simplex volume. We use the regularized log-volume:

$$\mathcal{V}_G(A) = \log\det\big(E(A)GE(A)^\top + \epsilon_{\text{vol}} I_{K-1}\big), \tag{7}$$

where $\epsilon_{\text{vol}} > 0$ ensures numerical stability. Under soft reconstruction constraints, however, volume can also be reduced by collapsing vertices or making simplex edges nearly linearly dependent. To discourage such degeneracies, we introduce two non-degeneracy penalties:

$$\mathcal{R}(A) = \underbrace{\lambda_{\text{scale}} \sum_{k=1}^{K} (\rho_k - 1)^2}_{\text{Scale anchoring}} + \underbrace{\lambda_{\text{coh}} \sum_{k\neq l} \kappa_{kl}^2}_{\text{Coherence penalty}}, \tag{8}$$

where $\rho_k = \|a_k\|_G^2$ is the mean squared comparison logit of component a_k on the source comparisons, and $\kappa_{kl} = a_k^\top G a_l / \sqrt{\rho_k \rho_l}$ is the cosine similarity between components a_k and a_l under G. Scale anchoring prevents volume from shrinking by uniformly contracting all components, and the coherence penalty discourages two components from encoding the same criterion. Finally, the full unmixing objective is defined as:

$$\begin{aligned} \mathcal{L}_{\text{unmix}} = \min_{A,\{\theta_c\}} \quad & \underbrace{\frac{1}{|\mathcal{C}_{\text{src}}|} \sum_{c\in\mathcal{C}_{\text{src}}} \|\hat{v}_c - A^\top \theta_c\|_G^2}_{\mathcal{L}_{\text{Reconstruction}}} + \underbrace{\lambda_{\text{vol}} \mathcal{V}_G(A)}_{\text{Minimum volume}} + \underbrace{\mathcal{R}(A)}_{\text{Non-degeneracy}} \\ \text{s.t.} \quad & \theta_c \geq 0, \qquad \mathbf{1}^\top \theta_c = 1 \quad \forall c \in \mathcal{C}_{\text{src}}. \end{aligned} \tag{9}$$

3.3 REMIX: FEW-SHOT ADAPTATION

With the recovered reward basis $\hat{A}$ frozen, adaptation to a new context requires estimating only its mixture coefficients (Figure 2c). We initialize them with the average source mixture:

$$\theta_0 = \frac{1}{|\mathcal{C}_{\text{src}}|} \sum_{c\in\mathcal{C}_{\text{src}}} \hat{\theta}_c. \tag{10}$$

This initialization gives greater weight to components emphasized across source contexts, providing a population-informed starting point for adaptation; it already yields a zero-shot reward $\theta_0^\top \hat{A} z(x)$. Given a small set of target comparisons $\mathcal{D}_{c^\star}$, we then estimate the target mixture coefficients by

$$\hat{\theta}_{c^\star} = \arg\min_{\theta\in\Delta^{K-1}} \mathcal{L}^{\text{BT}}_{\hat{A}^\top\theta;\mathcal{D}_{c^\star}} + \lambda_{\text{adapt}} \big\|\hat{A}^\top(\theta - \theta_0)\big\|_G^2, \tag{11}$$

where the regularizer limits departures from the source prior when target feedback is scarce. Measuring these departures in reward space under the same metric G, rather than in coefficient space, makes the prior independent of how the basis is parameterized: any two bases with the same span induce the same penalty on the resulting reward.

4 EXPERIMENTS AND ANALYSIS

4.1 SETUP

We evaluate PrefGeo on cross-task generalization and pluralistic personalization in classic robot manipulation, VLAs, assistive human-robot interaction, and LLMs. Across all settings, the reward basis is learned from source contexts only, and an unseen target context is adapted with a handful of preference feedback. Details of all environments and settings are provided in Appendix D.

Cross-task generalization. On Meta-World (Yu et al., 2020), each of five target tasks (Coffee Button, Drawer Open, Handle Pull, Sweep Into, Window Close) is paired with four source tasks sampled under an out-of-distribution (OOD) restriction so that neither the target object nor the target action appears in the sources, and the target policy is trained from scratch with the adapted reward. On LIBERO-Goal (Liu et al., 2023), one of the ten tasks is held out as the target while the other nine serve as sources, and a pre-trained $\pi_{0.5}$-base (Black et al., 2025) is fine-tuned with LoRA (Hu et al., 2022) using the adapted reward. Each source task provides 300 labels from a scripted teacher (Lee et al., 2021b) and the target task 20. We report the success rate.

Pluralistic personalization. On the Assistive Gym feeding task (Erickson et al., 2020), we follow Wang et al. (2025a) to simulate users who value different aspects of feeding (e.g., cautious versus efficient delivery), with 40 source users providing 100 labels each and 60 OOD target users providing 20 labels each, whose preference weights lie outside the range spanned by the source users. For LLMs, we follow the unseen-user protocol of LoRe (Bose et al., 2025) on Personal-LLM (Zollo et al., 2025) (16 source users, target feedback $b = 10$), PRISM (Kirk et al., 2024) (20, $b = 3$), and Reddit TL;DR (Völske et al., 2017; Stiennon et al., 2020) (20, $b = 10$). These small source populations reflect practical scenarios where large-scale annotations are unavailable. We report preference accuracy (Acc) (Poddar et al., 2024; Lambert et al., 2025) and Bradley-Terry negative log-likelihood (NLL) (Poddar et al., 2024) on held-out comparisons of unseen users. Because accuracy alone may not reflect downstream selection quality (Wen et al., 2025), we also report the best-of-N win rate (WR) (Gao et al., 2023) on all benchmarks except TL;DR, whose comparisons do not support a protocol comparable across methods (Appendix D.3).

We compare PrefGeo against four groups of baselines:

- *Preference reward factorization.* LoRe (Bose et al., 2025) and PReF (Shenfeld et al., 2025) fit A and θ_c end-to-end from source preference data, differing in initialization and constraints.
- *Other preference transfer.* MAML (Hejna III & Sadigh, 2023) meta-learns a reward model across source contexts; PEARL (Liu et al., 2024b) transfers source labels to the target via optimal transport; VPL (Poddar et al., 2024) learns a latent user-conditioned reward model.
- *Foundation-model feedback.* A natural alternative to transfer is to replace target human feedback with synthetic feedback from a frontier model, which has been shown to be effective in providing feedback for policy adaptation. We include PrefCLM (Wang et al., 2025b) instantiated with GPT-6 (OpenAI, 2026) on Meta-World.
- *Oracles.* PrefGT performs PbRL with ground-truth preference labels from scripted teachers using PEBBLE (Lee et al., 2021a), while SAC (Haarnoja et al., 2018) trains the target policy directly on the ground-truth task reward. Both access privileged reward information and serve as oracle references for PbRL and RL, respectively.

We also evaluate the following ablations of PrefGeo:

- *w/o Fisher*: replaces G with a trace-matched isotropic (Euclidean) metric in head regularization, reconstruction, and volume, keeping the heads independently estimated.
- *w/o Decoupling*: learns A and $\{\theta_c\}$ end-to-end from source preferences with the BT loss, retaining the volume and non-degeneracy terms; this isolates decoupling from the choice of objective.
- *w/o Volume*: sets $\lambda_{\text{vol}} = 0$, retaining reconstruction and non-degeneracy.
- *w/o Non-deg*: sets $\mathcal{R}(A) = 0$, retaining reconstruction and volume minimization.

All applicable baselines and ablations share the same behavior representation $z(x)$ and receive identical source and target comparisons, with target comparisons sampled uniformly at random or fixed by the LoRe protocol. Implementation details are provided in Appendix D.5.

4.2 Does PrefGeo Generalize to Unseen Tasks and Users?

PrefGeo outperforms existing transfer methods in both cross-task generalization and personalization. On cross-task generalization in Meta-World (Figure 3), PrefCLM with zero-shot GPT-6 feedback outperforms every existing transfer method, showing that with few target labels, reusing source preferences through these methods is less effective than querying a frontier model. PrefGeo is the only transfer method that surpasses it, achieving the best success rate among preference-based methods on all five tasks; it nearly matches SAC trained on the ground-truth reward and outperforms PrefGT, which uses ground-truth labels with $500\times$ more target feedback, on four of five tasks. On

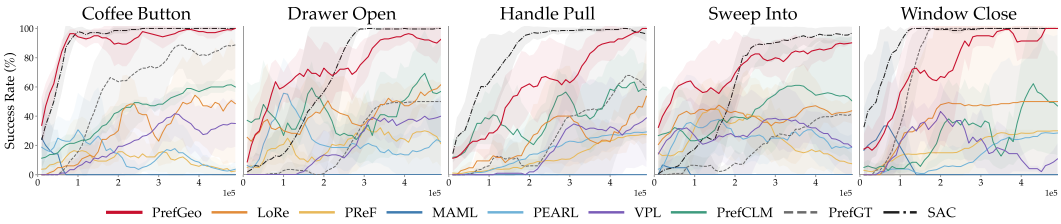


Figure 3: **Results of cross-task generalization** on unseen Meta-World tasks in terms of success rate, over 5 random seeds. Results on LIBERO-Goal are in Figure 13 of Appendix E.

Table 1: **Results of pluralistic personalization** on unseen users across different benchmarks. Mean ± std over 5 runs, best highlighted. Win rate (WR) is not reported on TL;DR (Appendix D.3).

	PersonalLLM			PRISM			TL;DR		Assistive Feeding		
Method	Acc% ↑	NLL ↓	WR% ↑	Acc% ↑	NLL ↓	WR% ↑	Acc% ↑	NLL ↓	Acc% ↑	NLL ↓	WR% ↑
LoRe	92.26 ±1.10	0.36 ±0.171	76.40 ±1.05	57.99 ±1.10	1.67 ±0.102	59.89 ±3.85	**65.38** ±1.97	1.45 ±0.081	70.75 ±1.28	0.69 ±0.004	69.30 ±3.03
PReF	91.64 ±1.03	0.29 ±0.042	75.18 ±1.12	55.09 ±3.31	1.93 ±0.478	56.84 ±4.72	56.18 ±1.82	1.46 ±0.868	56.60 ±2.71	0.71 ±0.001	65.70 ±1.76
MAML	89.00 ±1.76	1.11 ±0.291	70.14 ±0.88	**58.11** ±0.50	2.31 ±0.214	58.26 ±3.14	61.51 ±1.44	1.01 ±0.221	68.00 ±2.45	0.68 ±0.112	65.24 ±2.87
VPL	89.52 ±0.52	0.64 ±0.262	73.49 ±0.96	57.82 ±1.10	0.66 ±0.012	58.74 ±3.61	63.34 ±2.13	1.37 ±0.080	73.74 ±2.30	0.66 ±0.004	68.08 ±1.89
PrefGeo	**92.28** ±0.40	**0.17** ±0.010	**79.24** ±0.46	57.92 ±0.97	**0.62** ±0.009	**64.12** ±3.84	65.14 ±1.55	**0.87** ±0.004	**75.32** ±1.56	**0.63** ±0.009	**83.14** ±2.85

LIBERO-Goal (Figure 13), it raises the average success rate of $\pi_{0.5}$ from 51.6% to 74.7% (LoRe: 55.7%) and improves on all ten tasks, including a task (T8) where the base policy and LoRe fail.

The same holds for pluralistic personalization for unseen users (Table 1): PrefGeo achieves the lowest NLL on all four benchmarks and the highest WR on all three where it applies, while some baselines match or exceed its accuracy. Accuracy only checks whether the sign of each reward difference is correct, but an adapted reward is used to optimize or select behavior, which also depends on how large those differences are. NLL captures this by penalizing confident errors, which would push a policy strongly in the wrong direction (Guo et al., 2017), and WR directly measures whether the reward picks the preferred response among many candidates. Similar accuracy might therefore hide large differences in downstream quality: on PRISM, all methods lie within about 3 points of accuracy, yet PrefGeo's WR exceeds the best baseline by about 4 points, in line with prior findings on reward-model evaluation (Wen et al., 2025; Frick et al., 2025).

Baselines fall short for different reasons. MAML, PEARL, and VPL rely on auxiliary models, namely a meta-learned reward (Hejna III & Sadigh, 2023), pseudo-labels propagated through source-target optimal transport (Liu et al., 2024b), and a user-inference network (Poddar et al., 2024), which typically require many source contexts with thousands of labels each. With the few source contexts and limited feedback of our realistic setting, MAML fails to learn, and PEARL and VPL lag far behind. LoRe and PReF fit the factorization directly to preference data, leaving the basis underdetermined and measured in Euclidean geometry. They improve only modestly on Meta-World, and LoRe even falls below the base policy on several LIBERO tasks. PrefGeo avoids both limitations through decoupled geometric unmixing, whose effect on the recovered basis we analyze next.

4.3 Ablations and Feedback Efficiency

Figure 4 ablates PrefGeo's components and varies the amount of target and source feedback. Removing either the Fisher geometry or decoupling causes the largest drop: the success rate on Drawer Open plateaus at about half of PrefGeo's, and WR on Assistive Feeding falls by about 10 points, confirming that both geometry bias and factorization ambiguity must be addressed. The two regularizers matter less but serve distinct roles: without the volume term, learning is slower and plateaus lower as the basis is no longer compact, and without the non-degeneracy term, performance rises and then collapses on Drawer Open as volume minimization is free to shrink or merge components.

The recovered basis also makes PrefGeo feedback-efficient. It attains the lowest NLL at every target budget on Assistive Feeding, and its advantage is largest when target feedback is scarcest, where baselines must estimate a full reward from a few labels while PrefGeo only reweights the basis. On PersonalLLM, PrefGeo's NLL minimization saturates with only 8 source users, whereas the baselines need hundreds to approach the same performance.

We also provide additional ablation results in Appendix E.2: PrefGeo is robust to the basis dimension K once it covers the underlying criteria, and training LoRe on G-whitened features does not close

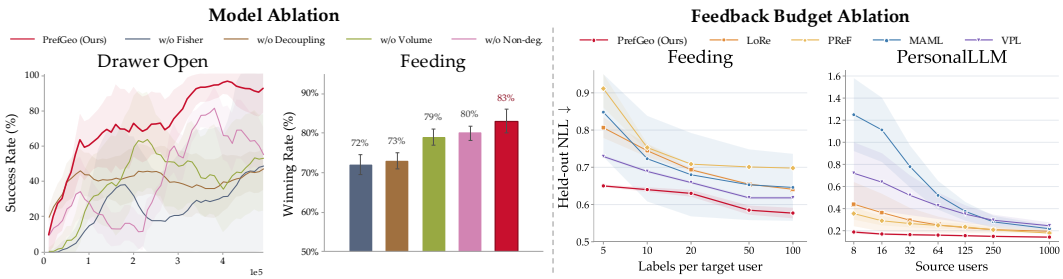


Figure 4: **Ablations.** Left: removing each component of PrefGeo, on Drawer Open (success rate) and Assistive Feeding (WR). Right: held-out NLL under varying target feedback on Assistive Feeding, and varying source users on PersonalLLM. More ablation results are in Appendix E.2.

the gap to PrefGeo, showing that the Fisher geometry cannot be replaced by feature preprocessing alone and that decoupled unmixing brings benefits beyond the geometry.

4.4 Does PrefGeo Recover a Meaningful and Reusable Reward Basis?

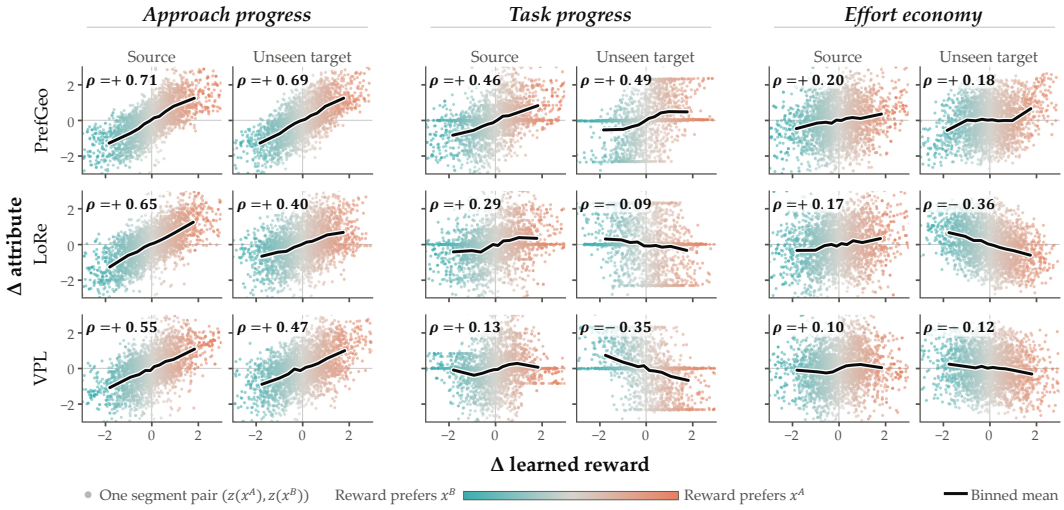


Figure 5: **Attribute alignment of learned rewards** on source tasks and the unseen target task Handle Pull on Meta-World. Each point is a segment pair $(z(x^A), z(x^B))$, colored by which segment the learned reward prefers. The horizontal axis is the learned reward difference and the vertical axis the difference in one semantic attribute, both standardized; the dark curve is the binned mean and ρ the Spearman correlation (Wissler, 1905).

We first probe what the learned rewards value by relating reward differences to three semantic behavioral attributes: approach progress, task progress, and effort economy (definitions in Appendix D.6). As shown in Figure 5, on source tasks, all methods prefer segments that show more of each attribute, and PrefGeo aligns most strongly with all three. On the unseen target task, however, only PrefGeo keeps this alignment; LoRe and VPL weaken or even reverse it, for example, favoring less efficient motion. Their rewards thus fit the source tasks but do not carry what users value to a new task, which is why training with them can degrade policy performance.

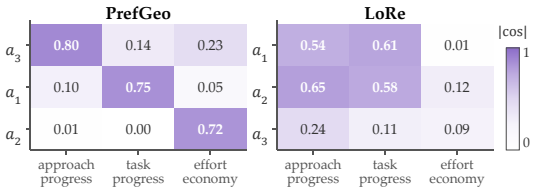


Figure 6: **Recovered reward components** vs. semantic attributes.

We further examine each component of the recovered basis by measuring its alignment with the three attributes (Figure 6), computed on 1200 segments from the four Meta-World source tasks with $K = 3$. The objective never observes these attributes, yet each PrefGeo component aligns with a single one: the volume term favors a

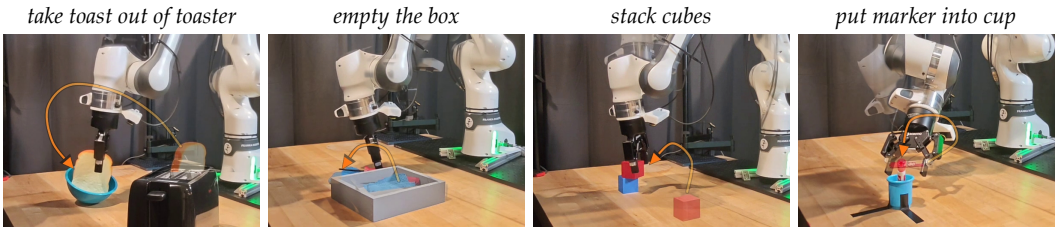


Figure 7: **Real-world VLA manipulation.** Success rates of $\pi_{0.5}$-DROID fine-tuned with PrefGeo on four held-out tasks using 20 target labels each; zero-shot base results are in Table 7.

compact basis, and the coherence penalty in $\mathcal{R}(A)$ discourages two components from encoding the same criterion, so the recovered components tend to separate distinct criteria, making the basis meaningful. Since they span the preference-relevant directions, a new task only needs to reweight them, making the basis reusable. LoRe's basis instead mixes approach and task progress across two components and contains no direction for effort economy; no reweighting can then express a preference for efficient motion, which explains its reversal on the target task in Figure 5. The Fisher geometry further contributes by retaining criteria that comparisons reveal only subtly, such as effort economy, which Euclidean criteria suppress.

4.5 Real-World Experiments

We first deploy PrefGeo on a real Franka arm, starting from $\pi_{0.5}$-DROID (Black et al., 2025; Khazatsky et al., 2024), on four manipulation tasks (Figure 7). Each task is held out as the target while the others serve as sources, with 100 preference pairs per source task and 20 on the target, constructed by pairing demonstrations with base-policy rollouts. With only 20 target labels, PrefGeo raises the average success rate from 22.5% to 67.5% and improves on all four tasks over the zero-shot base policy (Appendix F.1).

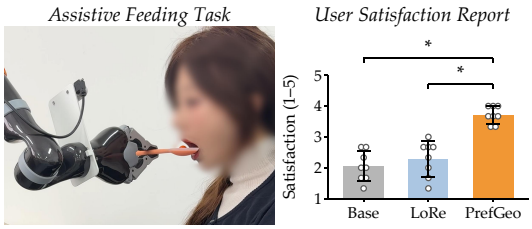


Figure 8: **User study.** Satisfaction ratings reported from 8 participants, each with 3 runs; $^*p < 0.05$.

We then conduct a user study on physical robot-assisted feeding with 8 participants and a Kinova Jaco arm (Figure 8). The reward basis is recovered in simulation from 40 simulated source users, and each participant provides only 10 preference labels on trajectory pairs, which are used to adapt the reward and fine-tune a base policy. Participants then interact with the Base, LoRe, and PrefGeo policies in a blind, randomized order. We found our participants to rate PrefGeo significantly higher in satisfaction than both Base and LoRe (Appendix F.2).

5 Conclusion and Future Work

We introduced PrefGeo, a geometric reward factorization framework that recovers a reusable reward basis from independently estimated context-specific rewards via minimum-volume unmixing in a preference-induced Fisher geometry. This allows adapting to a new context to reduce to reweighting the basis from a handful of feedback. Across robotic manipulation and assistive HRI in simulation and the real world, as well as LLM personalization, PrefGeo improves performance on unseen tasks and alignment with unseen users, especially when target feedback is scarce.

PrefGeo also has limitations that suggest exciting directions for future work. PrefGeo assumes that target contexts share the source preference criteria, so a genuinely novel criterion cannot be expressed; detecting such targets and expanding the basis online is a natural extension. It also treats contexts along a single dimension, whereas real deployments often vary across both tasks and users, which motivates joint factorizations, e.g., via tensor decomposition (Rendle & Schmidt-Thieme, 2010). More discussions are in Appendix G.

Ethics Statement

Our real-world user study involved 8 adult participants in a robot-assisted feeding task. The study was approved by the review board of our institution, and all participants gave informed consent before participating. The robot policy was trained in simulation to reach a 98% success rate before any interaction, an emergency stop was within reach of the investigator throughout, and participants' faces are blurred in all figures. The LLM benchmarks use publicly released datasets collected with participant consent (PRISM) or public data (Reddit TL;DR, PersonalLLM), and no personally identifiable information is used. Personalized reward models could in principle be used to reinforce harmful individual preferences; PrefGeo only reweights a basis learned from source contexts and does not introduce preferences beyond those present in its training data.

Reproducibility Statement

The full method is specified in Section 3.1-3.3, with the Fisher metric, its rank-deficiency handling, and implementation constants in Appendix B. Behavior representations are described in Appendix C; environments, task splits, preference generation, and simulated users in Appendices D.1 and D.2; metrics in Appendix D.3; budgets, hyperparameters, and the selection protocol in Appendix D.4; and baseline implementations in Appendix D.5. All results are averaged over five seeds or random user splits. Code will be released upon acceptance.

AI Use Statement

We used large language models to assist with editing the manuscript text and drafting figure layouts. All methods, experimental designs, and conclusions were developed by the authors, and all results were produced by running the code and verified by the authors.

References

Shun-ichi Amari and Hiroshi Nagaoka. *Methods of Information Geometry*, volume 191 of *Translations of Mathematical Monographs*. American Mathematical Society, 2000.

André Barreto, Vincent Dumoulin, Yiran Mao, Mark Rowland, Nicolas Perez-Nieves, Bobak Shahriari, Yann Dauphin, Doina Precup, and Hugo Larochelle. Capturing individual human preferences with reward features. *Advances in Neural Information Processing Systems*, 38:82119–82154, 2026.

Erdem Bıyık, Dylan P. Losey, Malayandi Palan, Nicholas C. Landolfi, Gleb Shevchuk, and Dorsa Sadigh. Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences. *The International Journal of Robotics Research (IJRR)*, 41 (1):45–67, 2022.

Erdem Bıyık, Nicolas Huynh, Mykel J. Kochenderfer, and Dorsa Sadigh. Active preference-based Gaussian process regression for reward learning and optimization. *The International Journal of Robotics Research (IJRR)*, 43(5):665–684, 2024.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: A vision-language-action model with open-world generalization. In *Proceedings of the Conference on Robot Learning (CoRL)*, volume 305, pp. 17–40. PMLR, 2025.

Avinandan Bose, Zhihan Xiong, Yuejie Chi, Simon Shaolei Du, Lin Xiao, and Maryam Fazel. Lore: Personalizing LLMs via low-rank reward modeling. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=bYu4DOqRY8.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3–4):324–345, 1952.

Daniel Brown, Wonjoon Goo, Prabhat Nagarajan, and Scott Niekum. Extrapolating beyond suboptimal demonstrations via inverse reinforcement learning from observations. In *International Conference on Machine Learning (ICML)*, pp. 783–792. PMLR, 2019.

Paul F. Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

John Co-Reyes, YuXuan Liu, Abhishek Gupta, Benjamin Eysenbach, Pieter Abbeel, and Sergey Levine. Self-Consistent Trajectory Autoencoder: Hierarchical reinforcement learning with trajectory embeddings. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, volume 80, pp. 1009–1018. PMLR, 2018. URL https://proceedings.mlr.press/v80/co-reyes18a.html.

David Donoho and Victoria Stodden. When does non-negative matrix factorization give a correct decomposition into parts? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2003.

Evan Ellis, Gaurav R. Ghosal, Stuart J. Russell, Anca Dragan, and Erdem Bıyık. A generalized acquisition function for preference-based reward learning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2814–2821, 2024.

Zackory Erickson, Vamsee Gangaram, Ariel Kapusta, C. Karen Liu, and Charles C. Kemp. Assistive gym: A physics simulation framework for assistive robotics. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10169–10176, 2020.

Alexander Fabisch and Frank Kirchner. Sample-efficient policy search with a trajectory autoencoder. In *NeurIPS Workshop on Robot Learning*, 2021. URL https://www.robot-learning.ml/2021/files/A1.pdf.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning (ICML)*, pp. 1126–1135. PMLR, 2017.

Ronald Aylmer Fisher. On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 222(594–604):309–368, 1922. doi: 10.1098/rsta.1922.0009.

Evan Frick, Tianle Li, Connor Chen, Wei-Lin Chiang, Anastasios N. Angelopoulos, Jiantao Jiao, Banghua Zhu, Joseph E. Gonzalez, and Ion Stoica. How to evaluate reward models for RLHF. In *International Conference on Learning Representations (ICLR)*, 2025.

Leo Gao, John Schulman, and Jacob Hilton. Scaling laws for reward model overoptimization. In *International Conference on Machine Learning (ICML)*, pp. 10835–10866. PMLR, 2023.

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

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International Conference on Machine Learning (ICML)*, pp. 1861–1870. PMLR, 2018.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, New York, NY, USA, 2nd edition, 2009. doi: 10.1007/978-0-387-84858-7.

Donald Joseph Hejna III and Dorsa Sadigh. Few-shot preference learning for human-in-the-loop RL. In *Proceedings of the Conference on Robot Learning (CoRL)*, pp. 2014–2025. PMLR, 2023.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, Peter David Fagan, Joey Hejna, Masha Itkina, Marion Lepert, Yecheng Jason Ma, Patrick Tree Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Beltran Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag R Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Rich Walke, Blake Wulfe, Ted Xiao, Jonathan Heewon Yang, Arefeh Yavary, Tony Z. Zhao, Christopher Agia, Rohan Baijal, Mateo Guaman Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Paul Foster, Jensen Gao, David Antonio Herrera, Minho Heo, Kyle Hsu, Jiaheng Hu, Donovon Jackson, Charlotte Le, Yunshuang Li, Xinyu Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Khuong Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew E. Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph J Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael C. Yip, Yuke Zhu, Thomas Kollar, Sergey Levine, and Chelsea Finn. DROID: A large-scale in-the-wild robot manipulation dataset. In *RSS 2024 Workshop: Data Generation for Robotics*, 2024. URL https://openreview.net/forum?id=Ml2pTYLNLi.

Hannah Rose Kirk, Alexander Whitefield, Paul Röttger, Andrew Bean, Katerina Margatina, Juan Ciro, Rafael Mosquera, Max Bartolo, Adina Williams, He He, Bertie Vidgen, and Scott A. Hale. The PRISM alignment dataset: What participatory, representative and individualised human feedback reveals about the subjective and multicultural alignment of large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Nathan Lambert, Valentina Pyatkin, Jacob Morrison, LJ Miranda, Bill Yuchen Lin, Khyathi Chandu, Nouha Dziri, Sachin Kumar, Tom Zick, Yejin Choi, Noah A. Smith, and Hannaneh Hajishirzi. RewardBench: Evaluating reward models for language modeling. In *Findings of the Association for Computational Linguistics (NAACL)*, 2025.

Gyuseok Lee, Wonbin Kweon, Zhenrui Yue, SeongKu Kang, Jiawei Han, and Dong Wang. Uncertainty-aware variational reward factorization via probabilistic preference bases for LLM personalization. *Conference on Language Modeling (COLM)*, 2026a. URL https://openreview.net/forum?id=D6HT5rLm6F.

Kimin Lee, Laura Smith, and Pieter Abbeel. PEBBLE: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. In *International Conference on Machine Learning (ICML)*, 2021a.

Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-pref: Benchmarking preference-based reinforcement learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*, 2021b. URL https://openreview.net/forum?id=ps95-mkHF_.

Tony Lee, Andrew Wagenmaker, Karl Pertsch, Percy Liang, Sergey Levine, and Chelsea Finn. RoboReward: General-purpose vision-language reward models for robotics, 2026b. URL https://arxiv.org/abs/2601.00675.

Chia-Hsiang Lin, Wing-Kin Ma, Wei-Chiang Li, Chong-Yung Chi, and ArulMurugan Ambikapathi. Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case. *IEEE Transactions on Geoscience and Remote Sensing*, 53(10):5530–5546, 2015.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, qiang liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL https://openreview.net/forum?id=xzEtNSuDJk.

Chris Yuhao Liu, Liang Zeng, Jiacai Liu, Rui Yan, Jujie He, Chaojie Wang, Shuicheng Yan, Yang Liu, and Yahui Zhou. Skywork-Reward: Bag of tricks for reward modeling in LLMs, 2024a. URL https://arxiv.org/abs/2410.18451.

Runze Liu, Yali Du, Fengshuo Bai, Jiafei Lyu, and Xiu Li. PEARL: Zero-shot cross-task preference alignment and robust reward learning for robotic manipulation. In *International Conference on Machine Learning (ICML)*, pp. 30946–30964. PMLR, 2024b.

Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, et al. Mediapipe: A framework for building perception pipelines. *arXiv preprint arXiv:1906.08172*, 2019.

OpenAI. GPT-6 Astra. https://developers.openai.com/api/docs/models/gpt-6-astra, 2026. API model documentation. Accessed September 21, 2026.

Sriyash Poddar, Yanming Wan, Hamish Ivison, Abhishek Gupta, and Natasha Jaques. Personalizing reinforcement learning from human feedback with variational preference learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 37:52516–52544, 2024.

Steffen Rendle and Lars Schmidt-Thieme. Pairwise interaction tensor factorization for personalized tag recommendation. In *Proceedings of the third ACM international conference on Web search and data mining*, pp. 81–90, 2010.

Dorsa Sadigh, Anca Dragan, Shankar Sastry, and Sanjit Seshia. Active preference-based learning of reward functions. In *Proceedings of Robotics: Science and Systems (RSS)*, 2017. doi: 10.15607/RSS.2017.XIII.053.

Idan Shenfeld, Felix Faltings, Pulkit Agrawal, and Aldo Pacchiano. Language model personalization via reward factorization. *Conference on Language Modeling (COLM)*, 2025. URL https://openreview.net/forum?id=E7Tu5yjqXw.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 3008–3021, 2020.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. TL;DR: Mining Reddit to learn automatic summarization. In Lu Wang, Jackie Chi Kit Cheung, Giuseppe Carenini, and Fei Liu (eds.), *Proceedings of the Workshop on New Frontiers in Summarization*, pp. 59–63, Copenhagen, Denmark, 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4508. URL https://aclanthology.org/W17-4508/.

Ruiqi Wang, Dezhong Zhao, Dayoon Suh, Ziqin Yuan, Guohua Chen, and Byung-Cheol Min. Personalization in human-robot interaction through preference-based action representation learning. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7377–7384, 2025a. doi: 10.1109/ICRA55743.2025.11128756.

Ruiqi Wang, Dezhong Zhao, Ziqin Yuan, Ike Obi, and Byung-Cheol Min. Prefclm: Enhancing preference-based reinforcement learning with crowdsourced large language models. *IEEE Robotics and Automation Letters (RA-L)*, 2025b.

Ruiqi Wang, Dezhong Zhao, Ziqin Yuan, Tianyu Shao, Guohua Chen, Dominic Kao, Sungeun Hong, and Byung-Cheol Min. PRIMT: Preference-based reinforcement learning with multimodal feedback and trajectory synthesis from foundation models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems (NeurIPS)*, 2025c.

Yufei Wang, Zhanyi Sun, Jesse Zhang, Zhou Xian, Erdem Bıyık, David Held, and Zackory Erickson. RL-VLM-F: Reinforcement learning from vision language foundation model feedback. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 51484–51501, 2024.

Xueru Wen, Jie Lou, Yaojie Lu, Hongyu Lin, Xing Yu, Xinyu Lu, Ben He, Xianpei Han, Debing Zhang, and Le Sun. Rethinking reward model evaluation: Are we barking up the wrong tree? In *International Conference on Learning Representations (ICLR)*, 2025.

Clark Wissler. The spearman correlation formula. *Science*, 22(558):309–311, 1905.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-World: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Proceedings of the Conference on Robot Learning (CoRL)*, pp. 1094–1100. PMLR, 2020.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023. URL `https://openreview.net/forum?id=uccHPGDlao`.

Thomas P. Zollo, Andrew Wei Tung Siah, Naimeng Ye, Ang Li, and Hongseok Namkoong. PersonalLLM: Tailoring LLMs to individual preferences. In *International Conference on Learning Representations (ICLR)*, 2025.

APPENDIX

TABLE OF CONTENTS

A Additional Analysis of Preference Reward Factorization 15

- A.1 Preference Reward Factorization
- A.2 Factorization Ambiguity
- A.3 Geometry Bias

B Preference-Induced Fisher Geometry 18

C Behavior Representation 19

D Experimental Details 20

- D.1 Environments and Tasks
- D.2 Preference Feedback
- D.3 Metrics
- D.4 Protocol and Hyperparameters
- D.5 Baseline Details
- D.6 Semantic Behavioral Attributes

E Extra Benchmark Results 25

- E.1 Results on LIBERO-Goal
- E.2 Extra Ablation Study

F Real-World Experiments 28

- F.1 Real-World VLA Manipulation
- F.2 User Study on the Feeding Task

G Limitations and Future Works 29

A Additional Analysis of Preference Reward Factorization

This appendix illustrates PRF and its two challenges with an assisted-feeding example. Throughout, we attach interpretable meanings (smoothness, approach comfort, efficiency) to reward components or feature coordinates purely for exposition: the learned representation need not be interpretable, and training observes only preference labels, never component functions or weights.

A.1 Preference Reward Factorization

Let x denote a feeding trajectory and c a user. A shared encoder maps x to a behavior representation $z(x) \in \mathbb{R}^D$, so that rewards can depend nonlinearly on raw behavior while all users' reward heads live in a common feature space. Given this representation, PRF models

$$R_c(x) = v_c^\top z(x) = \theta_c^\top A z(x) = \sum_{k=1}^{K} \theta_{c,k}\, a_k^\top z(x), \tag{12}$$

where $A = [a_1, \ldots, a_K]^\top \in \mathbb{R}^{K \times D}$ is the shared reward basis and $\theta_c \in \mathbb{R}^K$ holds the user-specific mixture coefficients. Row a_k scores one reward component, $\theta_{c,k}$ sets how much user c values it, and together they give the reward head $v_c = A^\top \theta_c$.

Suppose $K = 3$ components score smoothness, approach comfort, and efficiency on comparable scales. A cautious user with $\theta_c = (0.4, 0.5, 0.1)^\top$ emphasizes smooth motion and a gentle final approach, whereas an efficiency-oriented user with $\theta_c = (0.15, 0.15, 0.7)^\top$ favors prompt delivery. For a new user, the learned basis $\hat{A}$ is frozen and only θ is estimated from a few trajectory comparisons; choosing a gentler approach over a faster delivery, for instance, is evidence about that user's comfort-efficiency trade-off.

A.2 Factorization Ambiguity

Direct factorization learns the basis and the source coefficients jointly,

$$\min_{A, \{\theta_c\}} \sum_{c \in \mathcal{C}_{\text{src}}} \mathcal{L}^{\text{BT}}_{A^\top \theta_c; \mathcal{D}_c}. \tag{13}$$

LoRe (Bose et al., 2025) constrains the coefficients to the simplex; PReF (Shenfeld et al., 2025) instead uses SVD initialization and an L2 penalty. In either case, the preference-fitting term sees A and θ_c only through the composed heads $v_c = A^\top \theta_c$, so any two factorizations that produce the same heads are indistinguishable on source data. Figure 9 and the example below show that such factorizations can nonetheless adapt differently, in a way that cannot be absorbed by the choice of regularization strength.

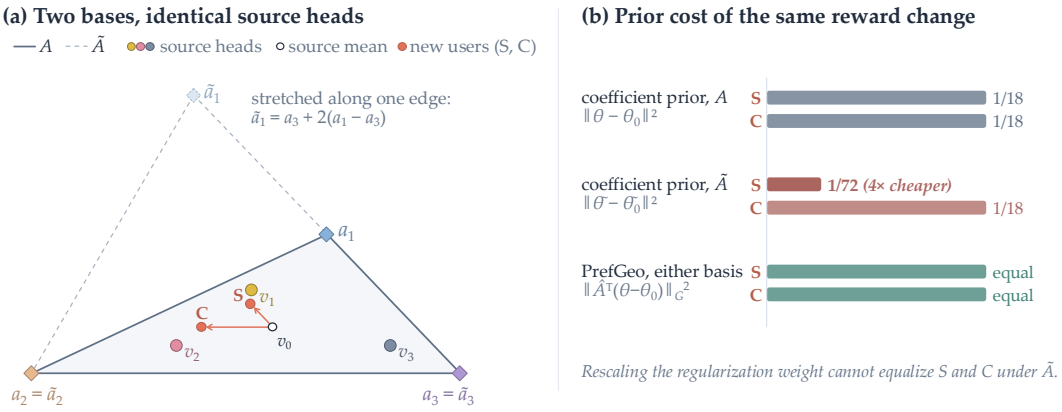


Figure 9: **Factorization ambiguity.** (a) $\tilde{A}$ stretches A along one edge yet yields the same source heads. (b) A coefficient prior makes departures toward smoothness (S) $4\times$ cheaper than toward comfort (C) under $\tilde{A}$ but not under A; PrefGeo's reward-space prior is identical under either basis.

Take $D = K = 3$, let the coordinates of $z(x)$ measure smoothness, approach comfort, and efficiency, and consider the bases

$$A = \begin{bmatrix} 0.6 & 0.3 & 0.1 \\ 0.3 & 0.6 & 0.1 \\ 0.2 & 0.2 & 0.6 \end{bmatrix}, \qquad \tilde{A} = \begin{bmatrix} 1.0 & 0.4 & -0.4 \\ 0.3 & 0.6 & 0.1 \\ 0.2 & 0.2 & 0.6 \end{bmatrix}. \tag{14}$$

Every component of either basis mixes all three attributes. The two bases share a_2 and a_3, but the first row of $\tilde{A}$ weights smoothness more heavily and penalizes efficiency, a "smooth but slow" component. Geometrically, $\tilde{A}$ stretches the triangle spanned by the rows of A along a single edge, $\tilde{a}_1 = a_3 + 2(a_1 - a_3)$, leaving the other two vertices unchanged.

The source coefficients below compensate for this stretch and yield identical heads under both bases:

c	$\theta_c^\top$	$\tilde{\theta}_c^\top$	$v_c^\top = \theta_c^\top A = \tilde{\theta}_c^\top \tilde{A}$
1	$(0.6, 0.3, 0.1)$	$(0.3, 0.3, 0.4)$	$(0.47, 0.38, 0.15)$
2	$(0.2, 0.6, 0.2)$	$(0.1, 0.6, 0.3)$	$(0.34, 0.46, 0.20)$
3	$(0.2, 0.1, 0.7)$	$(0.1, 0.1, 0.8)$	$(0.29, 0.26, 0.45)$

(15)

In general, $\tilde{\theta} = (\theta_1/2,\ \theta_2,\ \theta_3 + \theta_1/2)$. All coefficients are nonnegative and sum to one, so both factorizations satisfy the simplex constraint, assign the same reward to every trajectory, and attain the same source BT likelihood.

Suppose adaptation initializes at, and regularizes toward, the mean source coefficients, $\theta_0 = (1/3, 1/3, 1/3)$ and $\tilde{\theta}_0 = (1/6, 1/3, 1/2)$, which induce the same initial head. Consider two new users, one who values smoothness more than the source population and one who values comfort more:

$$\begin{aligned} \theta_{\mathrm{S}} &= (1/2,\ 1/3,\ 1/6), \qquad & \tilde{\theta}_{\mathrm{S}} &= (1/4,\ 1/3,\ 5/12), \\ \theta_{\mathrm{C}} &= (1/3,\ 1/2,\ 1/6), \qquad & \tilde{\theta}_{\mathrm{C}} &= (1/6,\ 1/2,\ 1/3). \end{aligned} \tag{16}$$

Under A, both users lie at the same distance from the prior, $\|\theta_{\mathrm{S}} - \theta_0\|_2^2 = \|\theta_{\mathrm{C}} - \theta_0\|_2^2 = 1/18$. Under $\tilde{A}$, the comfort user is equally far, $\|\tilde{\theta}_{\mathrm{C}} - \tilde{\theta}_0\|_2^2 = 1/18$, but the smoothness user is four times closer, $\|\tilde{\theta}_{\mathrm{S}} - \tilde{\theta}_0\|_2^2 = 1/72$. The expanded basis therefore makes departures toward smoothness four times cheaper than departures toward comfort, whereas the original basis treats them equally.

Since rescaling λ_{adapt} changes the cost of all departures by the same factor, it cannot remove this difference: the two factorizations encode different *relative* priors over how a new user may differ from the population. When target feedback is scarce and a few labels could be explained by either an increased preference for smoothness or for comfort, the regularized estimate under $\tilde{A}$ attributes them more to smoothness, while the estimate under A does not, even though both bases fit the source data identically and start from the same head. More generally, any reparameterization of the simplex that preserves the source heads maps the prior $\|\tilde{\theta} - \tilde{\theta}_0\|_2^2$ to a quadratic form in θ that is typically anisotropic, so the choice of basis determines which directions of adaptation are favored.

The example shows that source-prediction equivalence does not pin down regularized adaptation behavior when the prior is placed on the coefficients. PrefGeo removes this dependence by measuring departures in reward space, $\|\hat{A}^\top(\theta - \theta_0)\|_G^2$ (Eq. 11), which is identical for all bases with the same span. The choice of basis then affects adaptation only through the feasible set: since θ is restricted to the simplex, target rewards are confined to the convex hull of the components. A compact basis keeps this hull close to the source population, which reduces the variance of the adapted reward when target feedback is scarce, whereas an unnecessarily large basis admits rewards far from anything observed in the source contexts. This is the sense in which minimum-volume recovery favors transfer, while still allowing every component to mix behavioral attributes.

A.3 Geometry Bias

The second challenge concerns how reward heads are regularized and compared. The Euclidean distance

$$d_E^2(v, v') = \|v - v'\|_2^2 = \sum_{j=1}^{D} (v_j - v'_j)^2 \tag{17}$$

treats all directions of reward-parameter space equally, whereas preference comparisons reveal them unevenly. Coordinate-based criteria of this kind appear in PReF's L2 coefficient regulariza-

tion (Shenfeld et al., 2025) and in VRF's Euclidean-grounded latent distribution matching (Lee et al., 2026a); LoRe (Bose et al., 2025) optimizes the BT likelihood directly and uses no explicit head distance. Despite their different objectives, none of these methods equips basis recovery with a metric induced by the comparisons themselves. A related mismatch between parameter-space and functional importance has been observed in active preference learning (Ellis et al., 2024). The example below (Figure 10) shows how this mismatch first distorts the estimated heads and then the distances from which a basis is recovered.

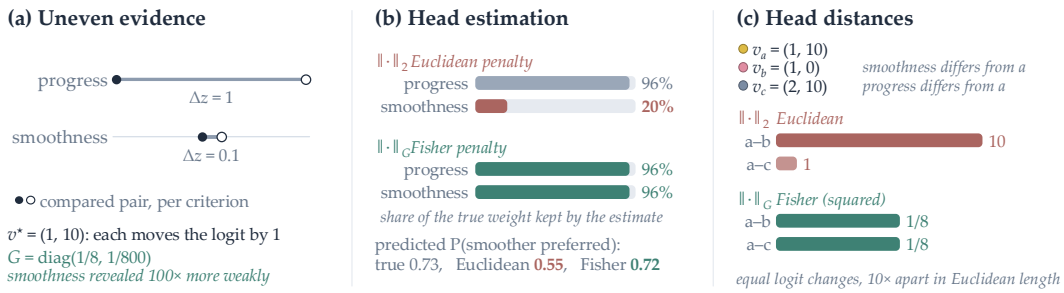


Figure 10: **Geometry bias.** (a) Comparisons reveal smoothness 100× more weakly than progress, though both matter equally. (b) A Euclidean penalty suppresses the weakly revealed criterion, while the Fisher penalty shrinks both equally. (c) Euclidean distance exaggerates the smoothness difference 10×; Fisher distance treats both as equal.

Let the first two coordinates of z encode task progress and smoothness. Half of the source comparisons differ clearly in progress, and the other half differ only subtly in smoothness:

$$\Delta z \in \left\{(1,\,0)^\top,\ (0,\,0.1)^\top\right\}, \qquad v^\star = (1,\,10)^\top. \tag{18}$$

The user with head $v^\star$ cares about both criteria equally: either kind of difference moves the comparison logit by $v^{\star\top}\Delta z = 1$. The comparison-induced metric of Appendix B is

$$G = \frac{1}{4M}\sum_m \Delta z_m \Delta z_m^\top = \operatorname{diag}\left(\tfrac{1}{8},\ \tfrac{1}{800}\right), \tag{19}$$

and since every comparison has logit 1, the Fisher information at $v^\star$ is exactly proportional to it, $\mathcal{I}(v^\star) = cG$ with $c = 4\sigma(1)(1-\sigma(1)) \approx 0.79$. Smoothness is thus revealed a hundred times more weakly than progress, although it matters just as much.

Consider first the estimation of a single head from n comparisons. To second order around $v^\star$, the BT loss has curvature $n\,\mathcal{I}(v^\star) = ncG$, so a regularized estimate shrinks $v^\star$ along each coordinate by

$$\underbrace{\frac{ncG_{jj}}{ncG_{jj}+2\lambda}}_{\text{Euclidean penalty } \lambda\|v\|_2^2} \quad \text{versus} \quad \underbrace{\frac{nc}{nc+2\lambda_G}}_{\text{Fisher penalty } \lambda_G\|v\|_G^2}. \tag{20}$$

The Euclidean factor depends on G_{jj} and therefore penalizes weakly revealed directions the most; the Fisher factor is the same for every direction. With $2\lambda/(nc) = 1/200$, the Euclidean estimate is $\hat{v} = (0.96,\,2.0)^\top$: progress is almost untouched, but the smoothness weight loses 80% of its value, and the predicted preference on a smoothness comparison drops from $\sigma(1) \approx 0.73$ to $\sigma(0.2) \approx 0.55$, close to indifference. Choosing λ_G to give the same shrinkage on progress, the Fisher estimate is $\hat{v} = (0.96,\,9.6)^\top$, which retains both criteria and predicts $\sigma(0.96) \approx 0.72$. The Euclidean penalty has not removed noise; it has removed a criterion that the user holds but the data express only subtly.

The same bias affects the distances from which a basis is recovered. Take three users,

$$v_a = (1,\,10)^\top, \qquad v_b = (1,\,0)^\top, \qquad v_c = (2,\,10)^\top, \tag{21}$$

so that a and b differ only in how much they value smoothness, and a and c only in how much they value progress. Functionally the two differences are equally large: each changes the logit of the corresponding comparisons by exactly 1. Yet

$$\|v_a - v_b\|_2 = 10, \quad \|v_a - v_c\|_2 = 1, \qquad \text{whereas} \qquad \|v_a - v_b\|_G^2 = \|v_a - v_c\|_G^2 = \tfrac{1}{8}. \tag{22}$$

In Euclidean geometry the smoothness difference appears ten times larger than the progress difference on the true heads, and about twice as large on the Euclidean-regularized estimates above (2.0 versus 0.96). Reconstruction error and simplex volume computed with such distances are dominated

by whichever criterion happens to have the larger coordinates, so the recovered basis is shaped by how strongly the data and the representation express each criterion rather than by how users differ in their preferences. In the Fisher geometry the two differences are equidistant, as they should be.

Feature preprocessing alone does not resolve this. Standardizing or whitening the features removes the dependence of Euclidean criteria on feature scale and correlation, and since rewards are linear in z, such preprocessing preserves the model. But it is driven by the feature distribution rather than by what is compared. Let $\Sigma_z = \mathrm{Cov}(z)$. If comparison pairs were drawn independently from the behavior distribution, then $\mathbb{E}[\Delta z \Delta z^\top] = 2\Sigma_z$, so $\|\delta v\|_G^2 = \frac{1}{2}\,\delta v^\top \Sigma_z \delta v$ and the Fisher geometry coincides, up to a constant, with Euclidean geometry after whitening by Σ_z. In practice, pairs are structurally matched: two responses to the same prompt, or two trajectories of the same task and scene. Writing $z = u + w$, where u is shared within a pair and w varies within it, gives $\Sigma_z = \Sigma_u + \Sigma_w$ but $\mathbb{E}[\Delta z \Delta z^\top] = 2\Sigma_w$. In the example, if smoothness varies widely across scenes while compared trajectories differ in it only subtly, standardizing by the overall spread of the smoothness feature makes its within-pair differences even smaller and the imbalance worse. With a generic pre-trained representation, Σ_u (variation across prompts or scenes) can dominate Σ_z, and only a normalization by the second moment of the comparison differences, which is what G provides, restores the balance. Even then, G must be used as a metric rather than as a feature transformation: whitening the features with G itself degrades LoRe (Table 5).

PrefGeo therefore uses the comparison-induced metric $G = \mathcal{I}(0)$ (Appendix B) for head regularization, basis reconstruction, volume, and the adaptation prior. It measures parameter differences by their effect on predicted preferences, so criteria that comparisons reveal only weakly are neither suppressed during estimation nor misjudged during basis recovery; invariance to linear feature reparameterization follows as a by-product. When a direction is not revealed by any source comparison at all, no geometry can recover it, and G assigns it zero length; in practice, a small ridge term keeps the reward flat along such directions.

B Preference-Induced Fisher Geometry

Fisher information geometry (Fisher, 1922; Amari & Nagaoka, 2000) measures a parameter change by its effect on the model's predictive distribution, and underlies natural-gradient and policy optimization methods. For a conditional model $p_\psi(y \mid q)$, the Fisher information matrix is

$$F(\psi) = \mathbb{E}_q\, \mathbb{E}_{y \sim p_\psi(\cdot|q)} \left[\nabla_\psi \log p_\psi(y \mid q)\, \nabla_\psi \log p_\psi(y \mid q)^\top \right], \tag{23}$$

and, under standard regularity conditions, a small perturbation δ satisfies

$$\mathbb{E}_q\, D_{\mathrm{KL}}(p_\psi(\cdot \mid q) \,\|\, p_{\psi+\delta}(\cdot \mid q)) = \tfrac{1}{2}\delta^\top F(\psi)\delta + o(\|\delta\|_2^2), \tag{24}$$

so the quadratic form $\delta^\top F(\psi)\delta$ assigns greater length to directions that change predictions more.

Specialization to the BT model. For a comparison $q_m = (x_m^A, x_m^B)$ with $\Delta z_m = z(x_m^A) - z(x_m^B)$, a reward head v predicts $p_m(v) = \sigma(v^\top \Delta z_m)$, where $v^\top \Delta z_m$ is the comparison logit, i.e., the log-odds of preferring x_m^A. The score of a binary label y is $\nabla_v \log p_v(y \mid q_m) = (y - p_m(v))\Delta z_m$, and taking its expected outer product and averaging over the M pooled source comparisons gives

$$\mathcal{I}(v) = \frac{1}{M} \sum_{m=1}^{M} p_m(v)\big(1 - p_m(v)\big)\, \Delta z_m \Delta z_m^\top. \tag{25}$$

The directions of this metric are set by which feature differences the comparisons contain, and its weights by how confidently v predicts them. $\mathcal{I}(v)$ is a finite sum of outer products and is therefore always well defined and positive semidefinite.

A common reference geometry. Evaluating $\mathcal{I}(v)$ at each context's fitted head would give every context a different geometry, in part because contexts differ in prediction confidence. To compare all heads in one geometry, PrefGeo evaluates the Fisher information at the neutral head $v = 0$:

$$G = \mathcal{I}(0) = \frac{1}{4M} \sum_{m=1}^{M} \Delta z_m \Delta z_m^\top. \tag{26}$$

At $v = 0$ every comparison has probability $1/2$ and hence the same weight $1/4$, so G depends only on the source comparison features, not on any labels or fitted heads, and is computed once and

shared by all subsequent steps. The induced squared distance between two heads,

$$\|v - v'\|_G^2 = \frac{1}{4M} \sum_{m=1}^{M} \left[(v - v')^\top \Delta z_m \right]^2, \tag{27}$$

is one quarter of their mean squared difference in source comparison logits. Its KL interpretation in Eq. 24 holds only locally around $v = 0$; elsewhere we use G as a fixed reference geometry rather than as the exact local Fisher metric.

Illustration. In the example of Appendix A.3, Eq. 27 gives the distances in Eq. 22 directly: $v_a - v_b = (0, 10)^\top$ changes the logit of every smoothness comparison by 1 and of no progress comparison, and $v_a - v_c = (-1, 0)^\top$ does the reverse, so both have squared length $\frac{1}{4} \cdot \frac{1}{2} \cdot 1^2 = \frac{1}{8}$ despite Euclidean lengths of 10 and 1. The small weight $G_{22} = 1/800$ on the smoothness coordinate reflects that compared behaviors differ little along it, not that smoothness matters little to the user; it is exactly what makes a large smoothness weight and a small progress weight commensurable.

Relation to feature whitening. The metric is invariant to linear feature reparameterization: for any invertible S, the map $z \mapsto Sz$, $v \mapsto S^{-\top} v$ leaves every comparison logit, and hence Eq. 27, unchanged. Because G is label-independent, the same geometry can equivalently be supplied to any method by whitening the features with G, after which all G-norms become Euclidean norms. What distinguishes it from standard whitening is the statistic used: G normalizes by the second moment of the comparison differences rather than by the covariance of the features (Appendix A.3). We evaluate LoRe on G-whitened features in Table 5 to separate the contribution of this geometry from that of decoupled unmixing.

Rank deficiency and implementation. Since G is a sum of M rank-one terms, its rank is at most $\min(M, D)$. With high-dimensional representations, such as the reward-model features used for LLMs and VLAs, G is singular: directions in its null space change no source comparison, so they would be constrained neither by the BT loss nor by any G-based term. We therefore use $G + \epsilon I$ with $\epsilon = 10^{-3} \operatorname{tr}(G)/D$, which leaves the geometry essentially unchanged on the span of the comparison differences and places a small Euclidean penalty on the remaining directions, making every step well posed and keeping unprobed directions near zero rather than arbitrary. We also drop the constant factor $1/4$, i.e., we use $G = 4\mathcal{I}(0)$ in practice, which only rescales the coefficients λ and the level at which ρ_k is anchored in Eq. 8.

Uses of G in PrefGeo. The same metric is used throughout: the head regularizer $\|v\|_G^2$ in Eq. 5 is the distance of a head from the neutral head, the reconstruction and volume terms in Eq. 9 measure how well and how compactly the basis explains the heads, the non-degeneracy penalties in Eq. 8 compare components through their comparison logits, and the adaptation prior in Eq. 11 measures departures of the target reward from the source prior. Every stage thus weights reward directions by how strongly the source comparisons reveal them, while the preference labels still determine which behaviors each context favors.

C Behavior Representation

PRF requires a representation $z(x) \in \mathbb{R}^D$ that is shared by all contexts and fixed during preference learning, so that every reward head and every basis component lives in the same parameter space. We therefore obtain z from encoders that are trained without any preference labels and are kept frozen afterwards. This keeps the source heads directly comparable, prevents preference information from any context, including target contexts, from leaking into the features, and leaves all context-specific learning in the linear heads, which is what makes few-shot adaptation possible.

For classic robotic control, a behavior x is a trajectory segment of 100 state-action steps. No general-purpose pre-trained model exists for such low-dimensional inputs, so we follow trajectory-embedding practice (Co-Reyes et al., 2018; Fabisch & Kirchner, 2021) and train an MLP autoencoder with a reconstruction loss on unlabeled segments pooled from the source contexts; $z(x)$ is the D-dimensional bottleneck. Reconstruction retains how the robot moves without committing to any notion of which movements are preferred.

For LLMs and VLAs, a behavior is a prompt-response pair or an instruction-rollout pair. Following LoRe (Bose et al., 2025), we pass it through a pre-trained general-purpose reward model, Skywork-Reward (Liu et al., 2024a) for language and RoboReward (Lee et al., 2026b) for VLA rollouts, and take the hidden representation that feeds its scalar output layer as $z(x)$. These models are trained on broad preference data unrelated to our contexts, so their penultimate features already capture generic quality attributes, and fitting linear heads on such features is standard in reward-model personalization (Bose et al., 2025; Barreto et al., 2026).

Table 2: Behavior representations used in each domain.

Domain	Behavior x	Encoder (frozen)
Meta-World, Assistive Gym	trajectory segment	MLP autoencoder (reconstruction)
PersonalLLM, PRISM, Reddit TL;DR	prompt-response pair	Skywork-Reward, penultimate layer
LIBERO	instruction-rollout pair	RoboReward, penultimate layer

PrefGeo itself is agnostic to how z is obtained: it only assumes a fixed, shared representation in which rewards are linear. The comparison-induced metric G is computed from the feature differences of whichever encoder is used, so the geometry adapts to the representation without encoder-specific tuning, and a stronger encoder can be substituted without changing the method.

D Experimental Details

D.1 Environments and Tasks

Meta-World. Meta-World (Yu et al., 2020) is a suite of 50 table-top manipulation tasks with a simulated Sawyer arm, 39-dimensional proprioceptive and object-state observations, and 4-dimensional end-effector control. Each task has a hand-designed shaped reward and a binary success criterion. We use the five target tasks in Table 3 and Figure 11. For each target, four source tasks are drawn at random from the remaining tasks subject to the OOD restriction that neither the target object (e.g., drawer, window) nor the target action (e.g., open, close, pull) appears in any source task. All tasks use the Meta-World v2 versions with randomized object and goal positions. The target policy is trained from scratch with the adapted reward using SAC, following the protocol of Lee et al. (2021b).

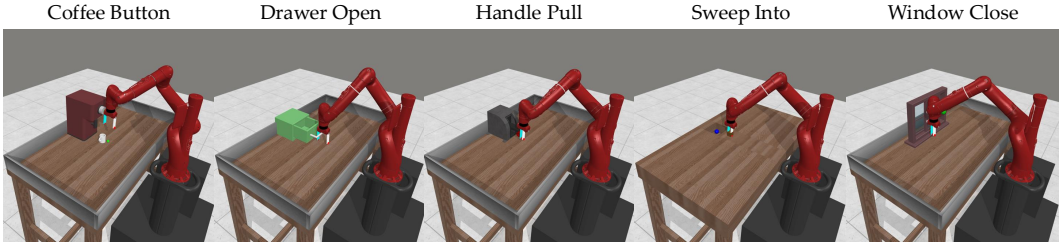


Figure 11: Meta-World target tasks.

Target task	Description
Coffee Button	Press the button on a coffee machine
Drawer Open	Grasp the handle and pull a drawer open
Handle Pull	Pull a handle upward
Sweep Into	Sweep a puck into a hole on the table
Window Close	Push a sliding window closed

Table 3: Meta-World target tasks.

LIBERO-Goal. LIBERO (Liu et al., 2023) is a lifelong robot-learning benchmark built on robosuite with a Franka arm, image observations, and language instructions. We use the LIBERO-Goal

suite (Figure 12), whose ten tasks share a single kitchen scene and object layout but differ in the instructed goal, so that transfer must rely on preference structure rather than on scene differences. In the order of T1–T10 in Figure 13, the tasks are: open the middle drawer of the cabinet, put the bowl on the stove, put the wine bottle on top of the cabinet, open the top drawer and put the bowl inside, put the bowl on top of the cabinet, push the plate to the front of the stove, put the cream cheese in the bowl, turn on the stove, put the bowl on the plate, and put the wine bottle on the rack. We follow a leave-one-out protocol in which each task is held out once as the target with the other nine as sources, and report results on all ten held-out tasks. The base policy is $\pi_{0.5}$-base (Black et al., 2025), fine-tuned with LoRA (rank 16 on the 2B vision-language backbone and 32 on the 300M action expert, applied to both the attention and feed-forward projections) on the target task using the adapted reward via rank-based hard filtering followed by supervised fine-tuning: each iteration scores that iteration's rollouts with the adapted reward, retains the top quartile, and fine-tunes the policy on the retained subset by behaviour cloning, so no policy gradient or value function is involved for three iterations, with 40 rollouts per task.

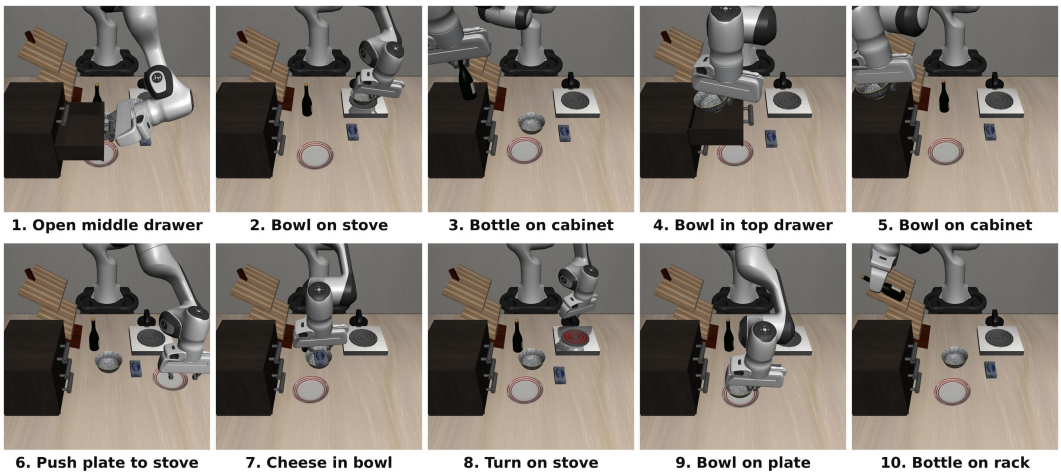


Figure 12: LIBERO-Goal suite.

Assistive Gym. Assistive Gym (Erickson et al., 2020) simulates physical human-robot interaction with a human model and an assistive robot. We use the feeding task, in which a Jaco arm holding a spoon with food must bring it to the mouth of a seated person, who moves slightly during the interaction. Observations include the end-effector pose, the spoon-to-mouth vector, and the human head pose; the environment reports the amount of food delivered and spilled and the contact forces applied to the person. The task is fixed across users, who differ only in how they value different aspects of the feeding motion (Appendix D.2).

LLM personalization benchmarks. PersonalLLM (Zollo et al., 2025) pairs 10,402 open-ended prompts (9,402 train, 1,000 test) with responses from eight LLMs, each scored by ten reward models. Users are simulated as mixtures of these reward models (Appendix D.2). PRISM (Kirk et al., 2024) contains 8,011 conversations from 1,500 participants in 75 countries, who rate several model responses per turn on value-laden and subjective prompts. We keep participants with at least six conversations and pair the chosen response of each turn with every rejected response, giving 39,365 comparisons from 25,806 turns. Reddit TL;DR (Völske et al., 2017; Stiennon et al., 2020) contains human comparisons between summaries of Reddit posts; following LoRe, we use annotator IDs as contexts and keep the 40 annotators who appear in both the train and validation splits.

For all three, we follow the unseen-user protocol of LoRe (Bose et al., 2025): users, not comparisons, are split into disjoint seen (source) and unseen (target) sets, and each unseen user provides a few-shot set for adaptation and a disjoint held-out set for evaluation. Unlike LoRe, which uses hundreds of source users, we restrict the source population to 16 users on PersonalLLM and 20 on PRISM and TL;DR, and adapt each unseen user with $b = 10$, 3, and 10 comparisons, respectively. On PersonalLLM, source users are fit on 47 comparisons each and every unseen user is evaluated on all 1,000 test prompts; on PRISM, each user's conversations are halved, one half for fitting or adaptation

and the other for evaluation; on TL;DR, source users are fit on up to 150 comparisons each and unseen users are evaluated on their full validation split. Every configuration is repeated over five random user splits.

D.2 Preference Feedback

Scripted teacher for cross-task generalization. Following B-Pref (Lee et al., 2021b), preference labels are generated by a scripted teacher. Let $J(\xi)$ denote the evaluation score of a behavior ξ, a fixed-length segment on Meta-World or a complete rollout on LIBERO. The teacher assigns

$$P[\xi^A \succ \xi^B] = \sigma\big(\beta\,[J(\xi^A) - J(\xi^B)]\big), \tag{28}$$

and we use the oracle limit $\beta \to \infty$, which deterministically prefers the behavior with the higher score, without injected label noise. On Meta-World, J is the undiscounted sum of the environment's shaped reward over the segment.

On LIBERO, we collect 4,000 complete rollouts across the ten tasks using two frozen $\pi_{0.5}$ checkpoints, with a 300-step cap at 20 Hz; these checkpoints are used only to generate behavior data, and fine-tuning always starts from $\pi_{0.5}$-base. To obtain diverse behavior quality, the collection varies the execution horizon over $\{1, 2, 4, 8\}$, the action scale over $\{0.7, 1.0, 1.3\}$, and the action noise over $\{0, 0.05, 0.10\}$. Since LIBERO provides only a binary success signal, we set $J(\tau) = U(\tau)$, a scripted utility combining task success with four motion-quality scores: precision (final goal error and gripper oscillations), safety (unsafe contacts, clearance to non-target objects, and post-grasp drops), smoothness (integrated jerk), and speed-efficiency (completion time, path length, and path efficiency). Each score is oriented so that larger is better and standardized within each task to have zero median and an interquartile range of one, and

$$U(\tau) = 3.0\,S(\tau) + 1.0\,f_{\text{prec}}(\tau) + 0.6\,f_{\text{safe}}(\tau) + 0.3\,f_{\text{smooth}}(\tau) + 0.3\,f_{\text{speed}}(\tau), \tag{29}$$

where $S(\tau) \in \{0, 1\}$ is the environment's success flag. The weights are fixed across all ten tasks.

Simulated users for assistive feeding. Following Wang et al. (2025a), each user c is represented by a weight vector $w_c \in \Delta^{J-1}$ over $J = 6$ aspects of the feeding motion, collected in $f(\tau) \in \mathbb{R}^J$: spoon-to-mouth distance, end-effector speed, total contact force on the person, tool force above 10 N, food spilled onto the person, and food entry speed at the mouth. Each aspect is accumulated as an episode-level cost and standardized over the trajectory pool. The user's ground-truth reward is $r_c(\tau) = -w_c^\top f(\tau)$, from which the oracle teacher above generates preference labels. We use 40 source users with 100 labels each, and 60 target users with 20 labels each.

User weights are sampled as $w_c \propto (\pi_c^\top T) \odot \exp(0.8\,\epsilon_c)$, where $\pi_c \sim \text{Dirichlet}(0.5 \cdot \mathbf{1}_3)$ mixes the three user types in T from Wang et al. (2025a) and $\epsilon_c \sim \mathcal{N}(0, I_J)$. For target users, before the multiplicative perturbation, the mixed profile is shifted toward end-effector speed, the aspect weighted least across these types: $\pi_c^\top T \leftarrow 0.6\,\pi_c^\top T + 0.4\,e_{\text{speed}}$, where e_{speed} is the corresponding unit vector. This shift changes the population mean while preserving its support, with all weights remaining strictly positive.

The behavior representation $z(x)$ is obtained from an encoder trained without preference labels and then frozen. Its inputs are limited to the robot's observations, namely spoon position and velocity, distance to the mouth, sensed contact force, and spoon orientation; the encoder has no access to $f(\tau)$ or w_c, and the food state underlying the last two aspects is unobserved.

Simulated and real users for LLMs. On PersonalLLM, each user is a convex combination $w_c \in \Delta^9$ of the ten released reward models, with utility $u_c(x, y) = w_c^\top s(x, y)$, where $s(x, y) \in \mathbb{R}^{10}$ collects the ten reward-model scores of response y to prompt x (Zollo et al., 2025). Following LoRe, each prompt yields one comparison per user, in which the response with the highest u_c among the eight is preferred to the one with the lowest; the same rule builds source training comparisons, target few-shot sets, and test comparisons. We draw $w_c \sim \text{Dirichlet}(\alpha \mathbf{1}_{10})$ with $\alpha = 0.01$, which concentrates each user on one or a few reward models, so that users differ sharply. On PRISM and TL;DR, users are real annotators and their recorded comparisons are used directly.

D.3 METRICS

Preference prediction. Let $\mathcal{D}_{c^\star}^{\text{test}} = \{(x_i^A, x_i^B, y_i)\}_{i=1}^n$ denote the held-out comparisons of an unseen context $c^\star$, where $y_i \in \{0, 1\}$ indicates whether x_i^A is preferred, and let $\hat{d}_i = \hat{R}_{c^\star}(x_i^A) - \hat{R}_{c^\star}(x_i^B)$ and $\hat{p}_i = \sigma(\hat{d}_i)$. We report

$$\text{Acc}_{c^\star} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\big[(2y_i - 1)\hat{d}_i > 0\big], \tag{30}$$

$$\text{NLL}_{c^\star} = -\frac{1}{n} \sum_{i=1}^{n} \Big[y_i \log \hat{p}_i + (1 - y_i) \log(1 - \hat{p}_i) \Big]. \tag{31}$$

Both are evaluated on strict preference pairs, excluding tied reference ratings or utilities, and equal predicted rewards count as errors in Acc. Accuracy measures pairwise ranking agreement, whereas NLL evaluates the predicted probabilities and penalizes confidently incorrect predictions.

Offline response selection. On PersonalLLM, PRISM, and Assistive Feeding, we evaluate best-of-N selection from fixed candidate pools. For each evaluation prompt or state $x \in \mathcal{X}_{c^\star}$ with pool $\mathcal{Y}_x$, the adapted reward selects $y_x^\star = \arg\max_{y \in \mathcal{Y}_x} \hat{R}_{c^\star}(x, y)$. A reference candidate $y_{\text{ref},x}$ is drawn uniformly from the same pool, independently of any model's selection, and is shared across methods, as are the pools and evaluation prompts. With $r_{c^\star}$ the reference utility, rating, or reward and $\Delta_x = r_{c^\star}(x, y_x^\star) - r_{c^\star}(x, y_{\text{ref},x})$,

$$\text{WR}_{c^\star} = \frac{1}{|\mathcal{X}_{c^\star}|} \sum_{x \in \mathcal{X}_{c^\star}} \Big(\mathbb{I}[\Delta_x > 0] + \tfrac{1}{2}\, \mathbb{I}[\Delta_x = 0] \Big). \tag{32}$$

Ties receive half credit, including when the selected and reference candidates coincide, so a uniform random selector has an expected WR of 0.5; with N candidates and a unique best one, an oracle selector has an expected WR of $1 - \frac{1}{2N}$. On PersonalLLM, each pool consists of the $N = 8$ responses to a test prompt and $r_{c^\star} = u_{c^\star}$; on PRISM, pools are held-out opening turns with $N = 4$ rated responses and $r_{c^\star}$ is the participant's rating; on Assistive Feeding, each pool consists of $N = 10$ rollouts of the base policy from the same initial state and $r_{c^\star}$ is the simulated user's reward. The corresponding oracle bounds are 0.9375, 0.875, and 0.95, so WR is not directly comparable across benchmarks. WR is averaged over users with at least one eligible pool and evaluates offline selection without generating new candidates or updating a policy.

We do not report WR on TL;DR: its recorded comparisons provide neither a utility for every candidate summary nor a label between an arbitrary selected summary and the reference, and scoring only the pairs that happen to be labeled would make the evaluation set depend on each method's selections.

Task success and aggregation. On robotic benchmarks, the success rate is the fraction of evaluation episodes in which the environment's task-specific success criterion is met. Per-context metrics are averaged over target contexts within each run, and we report the mean and standard deviation over 5 seeds.

D.4 PROTOCOL AND HYPERPARAMETERS

Table 4 summarizes the contexts and feedback budgets. All methods receive identical target comparisons: on the robotic simulation benchmarks, they are sampled uniformly at random from the target behavior pool; on the LLM benchmarks, they are the few-shot sets fixed by the LoRe protocol.

All hyperparameters are set using source data only; no hyperparameter or model variant is chosen with target labels or target test metrics. The volume weight $\lambda_{\text{vol}} = 0.05$ and the non-degeneracy weights $\lambda_{\text{scale}} = \lambda_{\text{coh}} = 0.02$ are fixed in advance and shared by all benchmarks, and the basis dimension is fixed per domain ($K = 3$ on Meta-World, $K = 4$ on LIBERO-Goal, $K = 3$ on the LLM benchmarks, and $K = 4$ on Assistive Feeding; see the sensitivity analysis in Appendix E.2). The head regularization weight λ_v is selected by the held-out BT NLL of the source heads on a behavior-disjoint split of the source comparisons. The adaptation weight λ_{adapt} is selected separately for each label budget B by leave-one-source-out: every source task or user in turn is treated as the target, adapted on exactly B of its own comparisons, and scored on its remaining compar-

Table 4: Contexts and preference budgets per benchmark.

Benchmark	Source contexts	Target contexts	Labels / source	Labels / target
Meta-World	4 tasks	5 tasks (one at a time)	300	20
LIBERO-Goal	9 tasks	1 task (leave-one-out)	300	20
Assistive Gym	40 users	60 users	100	20
PersonalLLM	16 users	1000 users	47	10
PRISM	20 users	643 users	30	3
Reddit TL;DR	20 users	20 users	150	10

isons. All grids are logarithmic and are extended whenever the selected value lies on the boundary. Baselines are tuned on the same source data with the same selection budget.

D.5 Baseline Details

All reward-learning baselines operate on the same frozen behavior representation $z(x)$ as PrefGeo (Appendix C), receive identical source and target comparisons, and share downstream policy learning, so that the way preference knowledge is transferred is the only difference between methods. Baseline hyperparameters were tuned on held-out source contexts with a budget comparable to PrefGeo's.

LoRe (Bose et al., 2025). LoRe learns a reward basis $A \in \mathbb{R}^{K \times D}$ and simplex-constrained mixture weights θ_c by maximizing the BT likelihood of all source comparisons jointly, and adapts to a new context by fitting $\theta_{c^\star}$ on its few-shot comparisons with A frozen. We use the authors' implementation with K equal to PrefGeo's and random initialization of A as in the original paper. Because A and θ_c are constrained only through the composed heads, the factorization is not identified (Section 1).

PReF (Shenfeld et al., 2025). PReF shares LoRe's model but initializes A from an SVD of the matrix of source preference statistics and replaces the simplex constraint with an L2 penalty on θ_c. Its uncertainty-based active query strategy for target users is orthogonal to basis recovery and is not used, as all methods receive identical target comparisons.

MAML (Hejna III & Sadigh, 2023). Following the few-shot PbRL framework of Hejna III & Sadigh (2023), a reward head on $z(x)$ is meta-trained with MAML (Finn et al., 2017) across source contexts so that a few gradient steps on a new context's comparisons yield a good reward, and is then fine-tuned on the target few-shot set. Unlike factorization methods, the adapted reward has D free parameters, which makes it the most flexible but also the most data-hungry transfer baseline.

PEARL (Liu et al., 2024b). PEARL transfers preference labels rather than reward parameters: it computes an optimal-transport plan between source and target behavior distributions in z-space and propagates source labels to target pairs as pseudo-labels, on which a target reward head is trained together with the target few-shot set. It requires target behaviors at transfer time and was designed for low-dimensional control trajectories, so we evaluate it on Meta-World.

VPL (Poddar et al., 2024). VPL learns a reward conditioned on a latent user embedding, inferred by a separate encoder from a set of the user's comparisons, and is trained end-to-end on source users. At test time, the encoder infers the latent from the target few-shot comparisons without gradient updates. For cross-task transfer, we treat tasks as users, which is the natural reading of a context in our framework.

PrefCLM (Wang et al., 2025b). PrefCLM replaces human preference labels with labels produced by a foundation model that is shown pairs of behaviors and asked which better achieves the task. We use its zero-shot variant, in which the model receives only a task description and the two trajectories, and instantiate it with GPT-6 Astra (OpenAI, 2026). This baseline represents the strongest alternative to transfer: if a frontier model can label the target task at no human cost, transfer methods must justify themselves against it. The zero-shot variant also makes the comparison clean, between prior

human preferences on other tasks and a frontier model's prior, with no human target labels beyond PrefGeo's 20. PrefCLM is evaluated on Meta-World only. It cannot be applied to personalization, since a foundation model has no information about an individual unseen user's preferences without the very feedback whose scarcity motivates few-shot personalization. On LIBERO, there is no established pipeline for turning foundation-model preference labels into VLA fine-tuning signals, and building one would amount to proposing a new method. Meta-World is also where this baseline is strongest, as the target tasks are simple and visually unambiguous, so restricting it there is conservative for PrefGeo.

PrefGT. PrefGT applies the standard single-context PbRL pipeline, PEBBLE (Lee et al., 2021a), directly to the target task with ground-truth labels from the oracle teacher, using 10,000 target comparisons, i.e., $500\times$ PrefGeo's target budget. It uses no source data and no factorization, and serves as a reference for what abundant target feedback achieves.

SAC (Haarnoja et al., 2018). SAC trains the policy directly on the ground-truth shaped reward with the same algorithm, network, and number of environment steps used for policy learning in all PbRL methods. It upper-bounds the downstream learner and separates reward-learning error from policy-learning error.

D.6 Semantic Behavioral Attributes

To interpret the learned rewards and basis (Section 4.4), we define three semantic attributes of a Meta-World segment $x = (s_1, a_1, \ldots, s_H, a_H)$, computed from simulator states that are never observed by the encoder or any learning method. Let p_t^{ee}, p_t^{obj}, and p^{goal} denote the end-effector, task-relevant object, and goal positions.

- *Approach progress*, how much the gripper moves toward the object: $f_{\mathrm{app}}(x) = \|p_1^{\mathrm{ee}} - p_1^{\mathrm{obj}}\|_2 - \|p_H^{\mathrm{ee}} - p_H^{\mathrm{obj}}\|_2$.
- *Task progress*, how much the object moves toward its goal: $f_{\mathrm{task}}(x) = \|p_1^{\mathrm{obj}} - p^{\mathrm{goal}}\|_2 - \|p_H^{\mathrm{obj}} - p^{\mathrm{goal}}\|_2$.
- *Effort economy*, the negative control effort: $f_{\mathrm{eff}}(x) = -\frac{1}{H}\sum_{t=1}^{H} \|a_t\|_2^2$.

For a segment pair (x^A, x^B), the attribute difference $\Delta f = f(x^A) - f(x^B)$ is standardized per task before plotting in Figure 5. For Figure 6, the direction of each attribute in z-space is the weight vector of a ridge regression from $z(x)$ to $f(x)$, fitted on 300 segments from the four source tasks, and each cell reports its absolute cosine similarity with a component a_k. Rows are ordered to place the best-matching pairs on the diagonal.

E Extra Benchmark Results

E.1 Results on LIBERO-Goal

Figure 13 reports per-task results, where T1-T10 follow the task order listed in Appendix D.1. PrefGeo improves over the base policy on all ten tasks, raising the average success rate from 51.6% to 74.7%, while LoRe reaches only 55.7%. The largest gains occur on tasks where the base policy is weak, most notably T8, on which both the base policy and LoRe fail entirely while PrefGeo reaches about 58%. LoRe instead falls below the base policy on T1, T5, and T6, indicating that a reward adapted on a poorly recovered basis can steer fine-tuning in the wrong direction, whereas PrefGeo's reward consistently provides a useful signal.

E.2 Extra Ablation Study

Additional Results. Figure 14 extends the component ablation to all five Meta-World target tasks. The trend matches Section 4.3: removing any component lowers the success rate on every task, and removing the Fisher geometry or decoupling generally causes the largest drop. Figure 15 varies the target feedback budget on TL;DR. PrefGeo attains the lowest NLL at every budget, and its

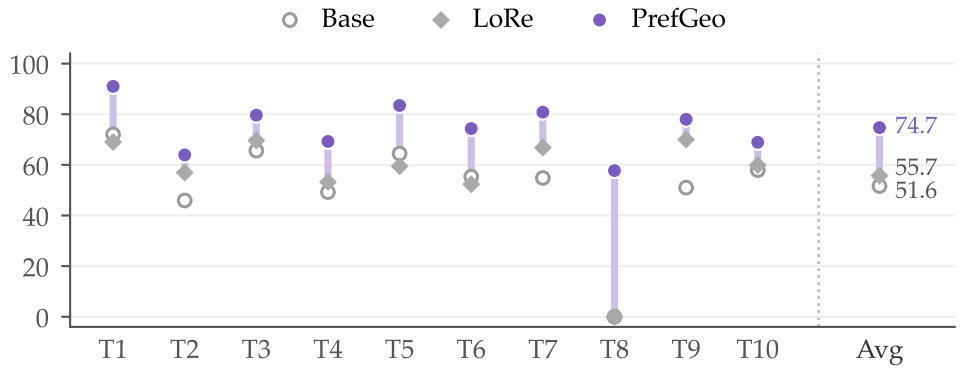


Figure 13: **Cross-task generalization on LIBERO-Goal.** Success rate of $\pi_{0.5}$-base before fine-tuning (Base) and after fine-tuning with rewards adapted by LoRe and PrefGeo, with each task held out as the target and the other nine as sources, using 20 target labels. The shaded bar marks PrefGeo's gain over Base; averages over the ten tasks are shown on the right.

advantage is largest when labels are scarcest, where a handful of labels even raises the NLL of LoRe and MAML.

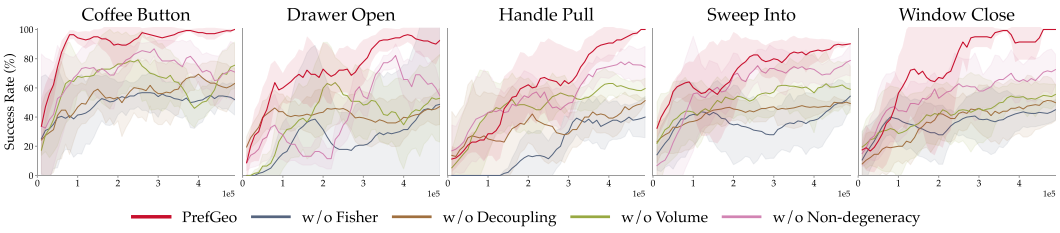


Figure 14: **Component ablation on Meta-World.** Success rate on the five target tasks when removing each component of PrefGeo.

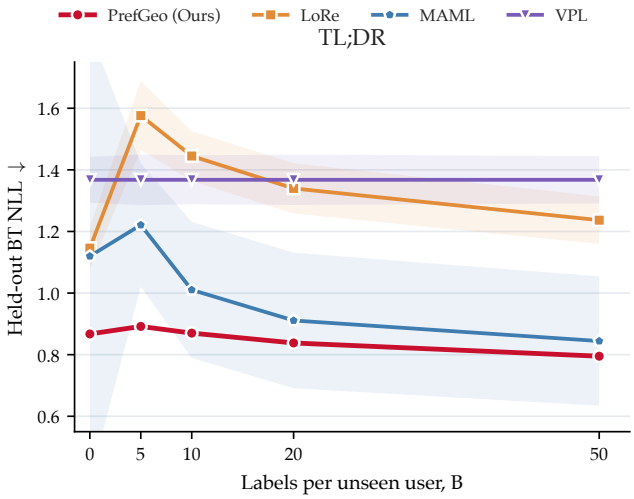


Figure 15: **Target feedback budget on TL;DR.** Held-out NLL on unseen users as the number of target labels varies.

Robustness to K. We vary the basis dimension $K \in \{1, 3, 5, 8\}$ on PersonalLLM and PRISM and $K \in \{2, 3, 8\}$ on two Meta-World tasks, Drawer Open and Sweep Into, keeping all other hyperparameters fixed (Figure 16). With $K = 1$, the basis reduces to a single reward shared by

all contexts, so the target weights cannot adapt; this setting yields the highest held-out NLL on both LLM benchmarks (PersonalLLM: 0.205; PRISM: 0.685), confirming that transfer requires more than one reward direction. Increasing K to 3 gives the largest improvement, and larger K changes the NLL by at most 0.003 on PersonalLLM and 0.007 on PRISM. On Meta-World, the success rate varies by about one point across $K \in \{2, 3, 8\}$ on both tasks. Once K covers the underlying criteria, the volume term and the coherence penalty keep additional components from expanding the simplex or duplicating existing ones, so the extra capacity is left largely unused rather than overfit. PrefGeo therefore does not require K to be tuned precisely, and a moderate over-estimate is safe.

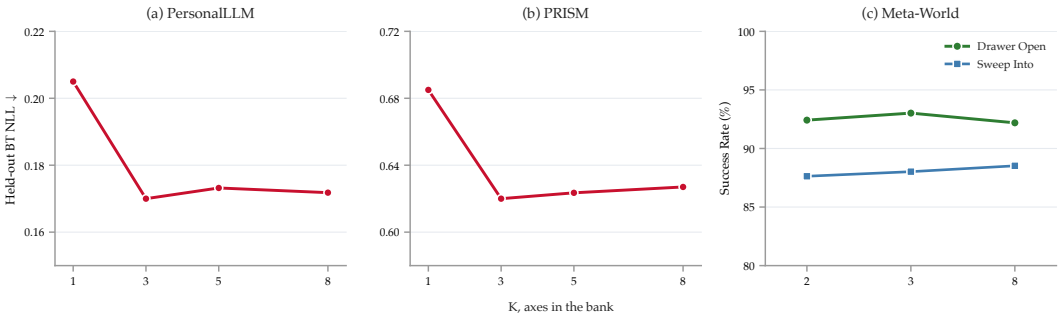


Figure 16: **Sensitivity to the basis dimension K.** (a, b) Held-out NLL on unseen users for PersonalLLM and PRISM, where $K = 1$ corresponds to a single shared reward. (c) Target success rate on two Meta-World tasks. All other hyperparameters are fixed.

Fisher geometry versus feature whitening. Since the comparison-induced metric G depends only on source comparison features and not on labels, one may ask whether its benefit can be obtained by simply preprocessing the features. With $G = U\Lambda U^\top$, whitening $z \mapsto \Lambda^{-1/2}U^\top z$ turns every G-norm into a Euclidean norm, so we train LoRe on these whitened features while keeping the source and target comparisons, the basis dimension, the hyperparameter selection, and the evaluation unchanged. If whitening brought LoRe to the level of PrefGeo, the Fisher geometry would reduce to a feature transformation that any method could adopt; otherwise, the geometry must act through the way PrefGeo uses it. As shown in Table 5, whitening does not help but degrades LoRe: accuracy drops from 92.26% to 89.27% and WR from 76.40% to 71.23% on PersonalLLM, accuracy drops from 70.75% to 65.23% and WR from 69.30% to 62.44% on Assistive Feeding, and NLL increases on both, while PrefGeo remains ahead of both LoRe variants on every metric.

Table 5: Fisher geometry versus feature whitening on PersonalLLM and Assistive Feeding. Mean ± std over 5 seeds. †LoRe trained on features whitened with the comparison-induced metric G.

	PersonalLLM			Assistive Feeding		
Method	Acc% ↑	WR% ↑	NLL ↓	Acc% ↑	WR% ↑	NLL ↓
LoRe	92.26 ±1.10	76.40 ±1.05	0.36 ±0.171	70.75 ±1.28	69.30 ±3.03	0.69 ±0.004
LoRe + G-whitening†	89.27 ±2.31	71.23 ±1.01	0.44 ±0.082	65.23 ±4.01	62.44 ±2.51	0.73 ±0.011
PrefGeo	**92.28** ±0.40	**79.24** ±0.46	**0.17** ±0.010	**75.32** ±1.56	**83.14** ±2.85	**0.63** ±0.009

The BT likelihood depends on a head only through the comparison logits $v^\top \Delta z$, which are invariant to linear reparameterizations of z, so whitening leaves LoRe's source objective intact; the difference arises in directions that source comparisons barely probe. Whitening rescales each direction by the inverse square root of its eigenvalue under G and thus amplifies weakly probed directions, along which target comparisons can still differ, so the adapted reward extrapolates where the source data provide almost no evidence. PrefGeo instead uses $G+\epsilon I$ as a metric for regularization and unmixing: unprobed directions receive no gradient from the data and are held near zero by the ridge term, keeping the reward flat along them (Appendix B). The Fisher geometry is therefore not a feature transformation that existing methods can simply adopt, and since PrefGeo outperforms LoRe with and without whitening, the remaining gap reflects the contribution of decoupling head estimation from minimum-volume basis recovery.

F Real-World Experiments

F.1 Real-World VLA Manipulation

Platform. All real-world VLA experiments follow the standard DROID setup (Khazatsky et al., 2024): a 7-DoF Franka Emika Panda arm with a Robotiq 2F-85 gripper, an external ZED 2 stereo camera, and a wrist-mounted ZED Mini camera. The policy receives the two external views, the wrist view, the proprioceptive state, and a language instruction, and outputs EEF target position action chunks at 15 Hz, matching the DROID interface. We start from the public $\pi_{0.5}$-DROID checkpoint (Black et al., 2025).

Tasks. We use four table-top tasks, listed in Table 6 with their language instructions and success criteria. Object positions are randomized within a 40×80 cm region at the start of each episode.

Table 6: Real-world VLA tasks.

Task	Instruction	Success criterion
Toast	"Put the slice of bread from the toaster to the bowl"	Toast placed in the bowl
Box	"Remove the screwdriver and the cloth from the basket and put them on the table"	Both objects removed from the basket and placed on the table
Cubes	"Stack the red block on top of the blue block"	Red block resting on the blue block
Marker	"Pick up the red marker and put it into the blue cup"	Marker inside the cup

Preference data. No human preference labels are collected on the real robot. Instead, following the common practice of treating demonstrations as preferred over non-expert behavior (Brown et al., 2019; Bıyık et al., 2022), each preference pair consists of a segment from a successful demonstration and a segment from a rollout of the zero-shot base policy on the same task, with the demonstration segment labeled as preferred. Demonstrations are collected by teleoperation on our setup, and rollouts are collected with the base policy from the same initial configurations. Segments are encoded by the same frozen reward-model encoder as on LIBERO (Appendix C).

Protocol. We follow the leave-one-out protocol of LIBERO-Goal: each task is held out as the target while the other three serve as sources. Each source task provides 100 preference pairs and the target task 20. PrefGeo recovers the basis from the source heads, adapts the mixture weights on the 20 target pairs, and the base policy is fine-tuned on the target task with the adapted reward using the same procedure and LoRA configuration as on LIBERO. LoRe is run with the same source and target pairs, basis dimension, and fine-tuning procedure.

Evaluation. Each policy is evaluated for 10 trials per task with randomized initial object positions, and a trial counts as a success if the criterion in Table 6 is met within 900 steps. The zero-shot base policy is evaluated under the same conditions. Success rates are reported in Table 7.

Table 7: Real-world success rates, reported as successes over trials. LoRe and PrefGeo use the same source and target preference pairs and the same fine-tuning procedure.

	Toast	Box	Cubes	Marker
$\pi_{0.5}$-DROID (zero-shot)	2/10 (20%)	4/10 (40%)	2/10 (20%)	1/10 (10%)
+ LoRe (20 target labels)	1/10 (10%)	5/10 (50%)	2/10 (20%)	3/10 (30%)
+ PrefGeo (20 target labels)	**7/10 (70%)**	**9/10 (90%)**	**6/10 (60%)**	**5/10 (50%)**

F.2 User Study on the Feeding Task

To evaluate whether PrefGeo personalizes robot behavior to real users from a handful of feedback, we conducted a user study on physical robot-assisted feeding.

Setup. We use the feeding task of Assistive Gym (Erickson et al., 2020), in which a robot arm delivers food on a spoon or a fork to the mouth of a seated person without spilling. The real-world platform (Figure 8, left) consists of a Kinova Jaco assistive arm and an Intel RealSense D435 camera, with face-landmark detection (Lugaresi et al., 2019) tracking the participant's head and mouth and YOLO tracking the spoon or fork. The simulation was configured to match the real setup, including the chair position, human body shape, and arm mounting, to reduce the sim-to-real gap. An emergency stop was within reach of the investigator throughout the study.

Source preferences and base policy. The reward basis is recovered in simulation from 40 simulated source users (Appendix D.2), each providing 100 preference labels. To ensure safe and functional behavior before any personalization, a base policy was trained in simulation with the ground-truth task reward, reaching a success rate of 98%, and serves both as the starting point for fine-tuning and as the *Base* condition.

Participants and procedure. We recruited 8 participants (2 female, 6 male; age 25.6 ± 2.15); all gave informed consent. The study was approved by the review board of our institution. After an introduction to the task, each participant provided preference labels on 10 pairs of feeding trajectories rolled out by the base policy on the physical robot, choosing in each pair the delivery they preferred. These 10 labels were used to adapt the mixture weights of PrefGeo and of LoRe on the recovered basis of each method, and the base policy was then fine-tuned with each adapted reward in simulation before deployment. Each participant then interacted with the three resulting policies, Base, LoRe, and PrefGeo, three times each in a randomized order, without being told which policy was active.

Measures and analysis. After the interactions, participants rated their satisfaction with each policy on a 5-point Likert scale (1: strongly disagree, 5: strongly agree), and ratings were averaged over the three interactions per policy. Since every participant rated all three policies, we used a Friedman test followed by post-hoc Wilcoxon signed-rank tests with Holm correction.

Results. Satisfaction differed significantly across policies (Friedman $\chi^2(2) = 13.07$, $p = 0.001$). PrefGeo received the highest ratings and was rated significantly higher than both Base and LoRe (Holm-corrected $p < 0.05$; Figure 8, right), while LoRe improved only marginally over the base policy. This is consistent with the simulation results: with 10 labels, reweighting a basis that does not capture the relevant criteria cannot express a participant's individual preferences, whereas PrefGeo's basis allows the few labels to steer the policy toward them.

G Limitations and Future Works

Targets outside the source basis. PrefGeo assumes that a target context shares the preference criteria underlying the source contexts, so that its reward can be expressed by reweighting the recovered basis. When a target involves a genuinely novel criterion, for example, a new safety constraint absent from all source tasks, no reweighting can express it. Such cases can be detected from the reconstruction residual of the target head under G, which remains large when the target lies outside the span of the basis. A potential solution is to complement the basis with a low-dimensional residual head that is fitted only when this residual is large, or to add a new component online as target feedback accumulates, while keeping the existing components fixed to preserve transfer.

Single context dimension. We model contexts along one dimension, either tasks for a fixed expert or users for a fixed task. In practice, preferences often vary along both, as when a user's priorities differ across tasks. A natural extension is a tensor factorization in which mixture weights depend jointly on the user and the task, for example $\theta_{u,t} = f(\phi_u, \psi_t)$, allowing structure to be shared across users and tasks simultaneously and enabling generalization to unseen user-task combinations.

Cross-embodiment behavior representation. Rewards are linear in a frozen representation $z(x)$, so criteria that $z(x)$ does not encode cannot be captured by any reweighting. Although PrefGeo is agnostic to the choice of encoder and benefits directly from stronger ones, the representation itself is not adapted to preferences. This is most restrictive across embodiments: when source and target robots differ substantially in their state and action spaces, a single encoder cannot map their

behaviors into a common space, and the recovered basis no longer transfers directly. Potential solutions include fine-tuning the encoder on source preferences before unmixing, or alternating between encoder updates and basis recovery, with care to keep the representation shared across contexts. For cross-embodiment transfer, behaviors can be encoded in an embodiment-agnostic form, such as end-effector or object trajectories, keypoints, or video embeddings from vision-language models, or embodiment-specific encoders can be aligned to a shared latent space, for example, with contrastive objectives on paired or task-matched behaviors, so that the basis is recovered and reused in that common space.